

CICS Transaction Server for z/OS



# Ressourcenreferenz

*Version 5 Release 5*



CICS Transaction Server for z/OS



# Ressourcenreferenz

*Version 5 Release 5*

**Hinweis**

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die Informationen unter „Bemerkungen“ auf Seite 525 gelesen werden.

# Inhaltsverzeichnis

## Informationen zu dieser PDF-Datei. . . vii

## Kapitel 1. Ressourcendefinitionsattribute 1

ATOMSERVICE-Ressourcendefinitionen . . . . .	1
ATOMSERVICE-Attribute . . . . .	1
BUNDLE-Ressourcen . . . . .	4
BUNDLE-Attribute . . . . .	5
CONNECTION-Ressourcen . . . . .	6
CONNECTION-Attribute . . . . .	10
DB2CONN-Ressourcen . . . . .	24
DB2CONN-Attribute . . . . .	24
DB2ENTRY-Ressourcen . . . . .	36
DB2ENTRY-Attribute . . . . .	37
DB2TRAN-Ressourcen . . . . .	42
DB2TRAN-Attribute . . . . .	42
DOCTEMPLATE-Ressourcen . . . . .	44
DOCTEMPLATE-Attribute . . . . .	45
ENQMODEL-Ressourcen . . . . .	48
ENQMODEL-Attribute . . . . .	50
FEPI-Ressourcen . . . . .	51
FILE-Ressourcen . . . . .	51
Ferne Dateien . . . . .	53
Coupling-Facility-Datentabellen . . . . .	54
Gemeinsam genutzte Datentabellen . . . . .	55
FILE-Attribute . . . . .	55
IPCONN-Ressourcen . . . . .	77
IPCONN-Attribute . . . . .	77
JOURNALMODEL-Ressourcen . . . . .	89
JOURNALMODEL-Attribute . . . . .	89
JOURNALMODEL-Standardattribute . . . . .	95
Beispiele . . . . .	95
JVMSERVER-Ressourcen . . . . .	97
JVMSERVER-Attribute . . . . .	97
LIBRARY-Ressourcen . . . . .	100
LIBRARY-Attribute . . . . .	101
LSRPOOL-Ressourcen . . . . .	105
LSRPOOL-Attribute . . . . .	106
MAPSET-Ressourcen . . . . .	111
MAPSET-Attribute . . . . .	112
MQCONN-Ressourcen . . . . .	114
MQCONN-Attribute . . . . .	115
MQMONITOR-Ressourcen . . . . .	118
MQMONITOR-Attribute . . . . .	120
PARTITIONSET-Ressourcen . . . . .	123
PARTITIONSET-Attribute . . . . .	124
PARTNER-Ressourcen . . . . .	126
PARTNER-Attribute . . . . .	127
PIPELINE-Ressourcen . . . . .	129
PIPELINE-Attribute . . . . .	130
PROCESSTYPE-Ressourcen . . . . .	133
PROCESSTYPE-Attribute . . . . .	134
PROFILE-Ressourcen . . . . .	136
PROFILE-Attribute . . . . .	137
PROGRAM-Ressourcen . . . . .	144
PROGRAM-Attribute . . . . .	145

SESSIONS-Ressourcen . . . . .	158
SESSIONS-Attribute . . . . .	159
TCIPSERVICE-Ressourcen . . . . .	172
TCIPSERVICE-Attribute . . . . .	173
TCIPSERVICE: In Wechselbeziehung zueinan- der stehende Attribute . . . . .	186
TDQUEUE-Ressourcen . . . . .	188
Ressourcendefinition mit Doppelfunktion für transiente Daten . . . . .	188
TDQUEUE-Attribute . . . . .	189
Erforderliche TDQUEUE-Definitionen . . . . .	204
Pseudowarteschlangen mit transienten Daten verwenden . . . . .	210
TERMINAL-Ressourcen . . . . .	211
Terminals zum Drucken . . . . .	211
Pipeline-Terminal-Definitionen . . . . .	214
Einheiten mit LDC-Listen . . . . .	215
APPC-Einzelsitzungsterminal (LUTYPE6.2) . . . . .	216
Terminals für Transaktionsrouting . . . . .	216
TERMINAL-Attribute . . . . .	222
TRANCLASS-Ressourcen . . . . .	235
TRANCLASS-Attribute . . . . .	235
TRANSACTION-Ressourcen . . . . .	237
TRANSACTION-Attribute . . . . .	239
TSMODEL-Ressourcen . . . . .	259
TSMODEL-Attribute . . . . .	259
TYPETERM-Ressourcen . . . . .	264
Standardwerte für TYPETERM-Attribute . . . . .	267
Unterstützte Einheiten . . . . .	274
TYPETERM-Attribute . . . . .	279
URIMAP-Ressourcen . . . . .	310
URIMAP: Zugehörige Ressourcen . . . . .	314
URIMAP-Attribute . . . . .	314
WEBSERVICE-Ressourcen . . . . .	332
WEBSERVICE-Attribute . . . . .	334

## Kapitel 2. CEDA-Befehle für Ressourcenmanagementtransaktionen . . . . 337

Befehl CEDA ADD . . . . .	337
Befehl CEDA ALTER . . . . .	338
Befehl CEDA APPEND . . . . .	341
Befehl CEDA CHECK . . . . .	342
Befehl CEDA COPY . . . . .	343
Befehl CEDA DEFINE . . . . .	346
Befehl CEDA DELETE . . . . .	348
Befehl CEDA DISPLAY . . . . .	350
Befehl CEDA DISPLAY GROUP . . . . .	352
Befehl CEDA DISPLAY LIST . . . . .	352
Befehl CEDA EXPAND . . . . .	353
Befehl CEDA EXPAND GROUP . . . . .	354
Befehl CEDA EXPAND LIST . . . . .	355
Befehl CEDA INSTALL . . . . .	355
Befehl CEDA LOCK . . . . .	358
Befehl CEDA MOVE . . . . .	359
Befehl CEDA REMOVE . . . . .	362

Befehl CEDA RENAME . . . . .	363
Befehl CEDA UNLOCK . . . . .	365
Befehl CEDA USERDEFINE . . . . .	367
Befehl CEDA VIEW . . . . .	370

### Kapitel 3. DFHCSDUP-Befehle des Dienstprogramms zur Ressourcenverwaltung . . . . . 373

DFHCSDUP-Befehl ADD . . . . .	373
DFHCSDUP-Befehl ALTER . . . . .	374
DFHCSDUP-Befehl APPEND . . . . .	376
DFHCSDUP-Befehl COPY . . . . .	377
DFHCSDUP-Befehl DEFINE . . . . .	380
DFHCSDUP-Befehl DELETE . . . . .	382
DFHCSDUP-Befehl EXTRACT . . . . .	385
DFHCSDUP-Befehl INITIALIZE . . . . .	386
DFHCSDUP-Befehl LIST . . . . .	387
DFHCSDUP-Befehl PROCESS . . . . .	389
DFHCSDUP-Befehl REMOVE . . . . .	390
DFHCSDUP-Befehl SCAN . . . . .	390
DFHCSDUP-Befehl SERVICE . . . . .	393
DFHCSDUP-Befehl UPGRADE . . . . .	393
DFHCSDUP-Befehl USERDEFINE . . . . .	394
DFHCSDUP-Befehl VERIFY . . . . .	396

### Kapitel 4. DFHCSDUP: Syntax- und Befehlsregeln . . . . . 399

### Kapitel 5. Ressourcendefinitionsmakros . . . . . 401

CLT: Befehlslistentabelle . . . . .	401
Programmabschnitt: DFHCLT TYPE=INITIAL	402
Alternative Systeme angeben: DFHCLT	
TYPE=LISTSTART . . . . .	403
Übernahmebefehle angeben: DFHCLT	
TYPE=COMMAND . . . . .	403
Nachrichten an Bediener: DFHCLT TYPE=WTO	404
Befehlsliste schließen: DFHCLT TYPE=LISTEND	405
PDIR: DL/I-Verzeichnis . . . . .	405
Programmabschnitt: DFHDLPB TYPE=INITIAL	405
Programmspezifikationsblöcke: DFHDLPB	
TYPE=ENTRY . . . . .	405
FCT: Dateisteuertabelle . . . . .	406
Programmabschnitt: DFHFCT TYPE=INITIAL	407
Lokale Dateien - DFHFCT TYPE=FILE . . .	407
DFHFCT-Beispiel . . . . .	416
MCT: Überwachungssteuertabelle . . . . .	416
Programmabschnitt: DFHMCT TYPE=INITIAL	417
Ereignisüberwachungspunkte des Benutzers:	
DFHMCT TYPE=EMP . . . . .	422
Steuerdatenaufzeichnung: DFHMCT	
TYPE=RECORD . . . . .	427
DFHMCT-Beispiele . . . . .	441
PLT: Programmlistentabelle . . . . .	442
Programmabschnitt: DFHPLT TYPE=INITIAL	443
Einträge in Programmlistentabelle: DFHPLT	
TYPE=ENTRY . . . . .	444
DFHPLT-Beispiel . . . . .	445
RST: Wiederherstellbare Servicetabelle . . . .	446

Programmabschnitt: DFHRST TYPE=INITIAL	447
Wiederherstellbare Serviceelemente: DFHRST	
TYPE=RSE . . . . .	447
DBCTL-Subsysteme: DFHRST TYPE=SUBSYS	448
DFHRST-Beispiel . . . . .	448
SRT: Systemwiederherstellungstabelle . . . .	449
Programmabschnitt: DFHSRT TYPE=INITIAL	449
Abbruchcodes: DFHSRT TYPE=SYSTEM USER	449
DFHSRT-Beispiel . . . . .	451
TCT: Terminalsteuertabelle . . . . .	451
DFHTCT-Makrotypen . . . . .	452
DFHTCT-Codes für logische Einheiten: z/OS	
Communications Server (nicht 3270). . . .	456
Sequenzielle Einheiten . . . . .	465
Ferne Terminals für Transaktionsrouting .	470
DFHTCT: CICS-Terminalliste . . . . .	474
TLT: Terminallistentabelle . . . . .	475
Programmabschnitt: DFHTLT TYPE=INITIAL	476
Einträge in Terminallistentabelle: DFHTLT	
TYPE=ENTRY . . . . .	476
DFHTLT-Beispiel . . . . .	477
TST: Tabelle für temporären Speicher . . . .	478
Programmabschnitt: DFHTST TYPE=INITIAL	480
Wiederherstellbarer temporärer Speicher:	
DFHTST TYPE=RECOVERY . . . . .	481
Lokaler temporärer Speicher: DFHTST	
TYPE=LOCAL . . . . .	483
Ferner temporärer Speicher: DFHTST	
TYPE=REMOTE . . . . .	484
Sicherheitsprüfung für temporären Speicher:	
DFHTST TYPE=SECURITY . . . . .	485
Gemeinsame Datennutzung für temporären	
Speicher: DFHTST TYPE=SHARED . . . .	486
DFHTST-Beispiel . . . . .	487
XLT: Transaktionslistentabelle . . . . .	488
Programmabschnitt: DFHXLT TYPE=INITIAL	489
Einträge in Transaktionslistentabelle: DFHXLT	
TYPE=ENTRY . . . . .	489
DFHXLT-Beispiel . . . . .	490

### Kapitel 6. Zur Wahrung der Kompatibilität beibehaltene veraltete Attribute. . 493

### Kapitel 7. Von CICS bereitgestellte Ressourcendefinitionen, Gruppen und Listen. . . . . 495

DFHLIST-Definitionen . . . . .	495
Von CICS bereitgestellte, nicht in DFHLIST enthal-	
tene Gruppen . . . . .	506
Von CICS bereitgestellte Kompatibilitätsgruppen	506
Gruppen von Beispielanwendungsprogrammen	507
TYPETERM-Definitionen in Gruppe DFHTYPE .	511
TERMINAL-Modelldefinitionen in Gruppe DFH-	
TERM . . . . .	516
PROFILE-Definitionen in Gruppe DFHEP . . .	519
PROFILE-Definitionen in Gruppe DFHISC . . .	519
PROFILE-Definitionen in Gruppe DFHSTAND .	520
Modelldefinitionen in Gruppe DFHPGAIP . . .	522

<b>Bemerkungen . . . . .</b>	<b>525</b>
<b>Index . . . . .</b>	<b>531</b>





---

## Informationen zu dieser PDF-Datei

Diese PDF-Datei dient als Referenz für die Attribute von CICS-Ressourcen. Informationen zur Vorgehensweise bei der Definition dieser Ressourcen finden Sie in der PDF-Datei mit dem Namen *Configuring CICS TS for z/OS*. Für CICS TS-Versionen vor Version 5.4 waren die Informationen in dieser PDF-Datei im *Ressourcendefinitionshandbuch* enthalten.

Detaillierte Informationen zu den Begriffen und zu der im vorliegenden Buch verwendeten Notation finden Sie unter *Conventions and terminology used in the CICS documentation* im IBM Knowledge Center.

### Datum dieser PDF-Datei

Diese PDF-Datei wurde erstellt am 14. Dezember 2018.



---

## Kapitel 1. Ressourcendefinitionsattribute

Die meisten Ressourcen können mithilfe von CICS RDO oder CICSplex SM Business Application Services (BAS) definiert werden.

---

### ATOMSERVICE-Ressourcendefinitionen

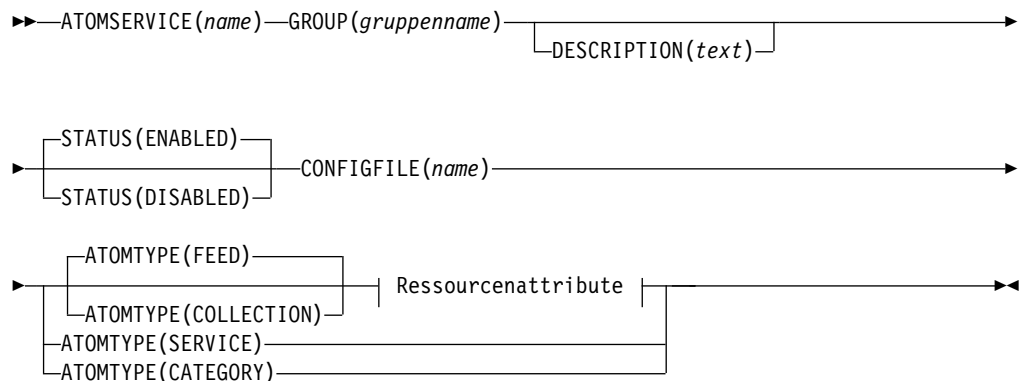
Eine ATOMSERVICE-Ressource definiert einen Atom-Service, einen Atom-Feed, eine Atom-Erfassung oder ein Atom-Kategoriedokument und identifiziert die Ressourcen, die verwendet werden, um die Daten für den Feed bereitzustellen.

Informationen zur Installation von ATOMSERVICE-Ressourcendefinitionen mit CEMT und CEDA finden Sie in *Installing ATOMSERVICE resource definitions*. Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in *Working with BAS ATOMSERVICE resource definitions*.

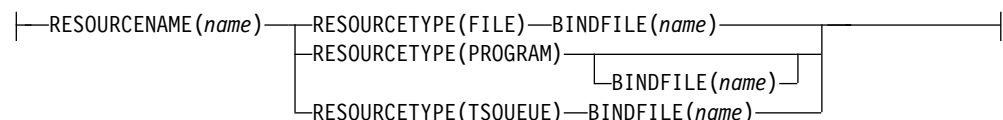
### ATOMSERVICE-Attribute

Beschreibt die Syntax und die Attribute der ATOMSERVICE-Ressource.

#### Syntax



#### Ressourcenattribute:



#### Attribute

##### ATOMSERVICE(name)

Gibt den aus acht Zeichen bestehenden Namen dieser Ressourcendefinition an.

##### Zulässige Zeichen:

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

**ATOMTYPE({FEED|SERVICE|COLLECTION|CATEGORY})**

Gibt den Typ des Atom-Dokuments an, das für diese ATOMSERVICE-Definition zurückgegeben wird.

**CATEGORY**

Ein Atom-Kategoriedokument, in dem die Kategorien für die Einträge einer Erfassung aufgelistet werden. Sie können ein Kategoriedokument einrichten, wenn Sie die gleichen Kategorien verwenden wollen, um mehrere Erfassungen zu definieren.

**COLLECTION**

Ein Atom-Erfassungsdokument, das eine Gruppe von Atom-Einträgen enthält, die von Web-Clients mit den HTTP-Anforderungen POST, PUT und DELETE bearbeitet werden können, und die mithilfe der HTTP-Anforderung GET abgerufen werden kann. Die Atom-Konfigurationsdatei für eine Erfassung muss mit dem Stammelement beginnen (<cics:atomservice type="collection">).

**FEED** Ein Atom-Feed-Dokument, in dem die Metadaten eines Feeds beschrieben werden und das die Atom-Einträge enthält, die die Daten für den Feed bereitstellen. Ein Atom-Feed kann mithilfe der HTTP-Anforderung GET abgerufen werden, jedoch nicht von einem Web-Client bearbeitet werden. Die Atom-Konfigurationsdatei für einen Atom-Feed muss mit dem Stammelement (<cics:atomservice type="feed">) beginnen.

**SERVICE**

Ein Atom-Servicedokument, das Informationen zu den bearbeitbaren Erfassungen bereitstellt, die auf einem Server zur Verfügung stehen.

**BINDFILE**(*name*)

Gibt den vollständig qualifizierten (absoluten) oder relativen Namen einer XML-Bindung an, die in z/OS UNIX System Services gespeichert ist. Dieses Attribut wird für ein Atom-Servicedokument oder -Kategoriedokument nicht verwendet. Sie erstellen eine XML-Bindung mithilfe des CICS-XML-Assistentenprogramms DFHLS2SC.

Für die Ressourcentypen FILE und TSQUEUE ist die XML-Bindung erforderlich. Sie gibt die Datenstrukturen an, die von der in RESOURCENAME angegebenen Ressource verwendet werden, die die Daten für das Atom-Dokument bereitstellt.

Für den Ressourcentyp PROGRAM ist eine XML-Bindung optional. Sie können sie mithilfe der Ressource erstellen, auf die das Programm zugreift, um die Daten für die Atom-Einträge (nicht das Programm selbst) abzurufen. Sie müssen eine XML-Bindung für den Ressourcentyp PROGRAM angeben, wenn Sie die Parameter für die Ressourcenbehandlung im Container DFHATOMPARGS verwenden, um Informationen von der Atom-Konfigurationsdatei an das Programm zu übergeben. Andernfalls geben Sie keine XML-Bindung an.

Der Name der XML-Bindung kann als absoluter Pfad einschließlich aller Verzeichnisse mit einem Schrägstrich als erstem Zeichen angegeben werden. Beispiel: /u/atom/atomictest.xsdbind. Alternativ hierzu kann er auch als Pfad relativ zum Ausgangsverzeichnis (HOME) der Benutzer-ID für die CICS-Region angegeben werden. Beispiel: atom/atomictest.xsdbind (ohne führenden Schrägstrich). Es können bis zu 255 Zeichen verwendet werden.

**Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

**CONFIGFILE**(*name*)

Gibt den vollständig qualifizierten (absoluten) oder relativen Namen einer Atom-Konfigurationsdatei an, die in z/OS UNIX System Services gespeichert ist. Eine Atom-Konfigurationsdatei enthält XML-Elemente, in denen Metadaten und Feldnamen für das Atom-Dokument angegeben sind, das für diese Ressourcendefinition zurückgegeben wird. Detaillierte Informationen hierzu finden Sie in *Creating an Atom configuration file*.

Der Name kann als absoluter Pfad einschließlich aller Verzeichnisse mit einem Schrägstrich als erstem Zeichen angegeben werden. Beispiel: /u/atom/myfeed.xml. Alternativ hierzu kann er auch als Pfad relativ zum Ausgangsverzeichnis (HOME) der Benutzer-ID für die CICS-Region angegeben werden. Beispiel: atom/myfeed.xml (ohne führenden Schrägstrich). Es können bis zu 255 Zeichen verwendet werden.

**Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

**DESCRIPTION**(*text*)

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

**GROUP**(*gruppenname*)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

**RESOURCENAME**(*name*)

Gibt den aus 1 - 16 Zeichen bestehenden Namen der CICS-Ressource an, die die Daten für diesen Atom-Feed oder die Erfassung enthält. Wenn die Daten für Ihre Atom-Einträge in einer Ressource gespeichert werden, auf die über eine Serviceroutine zugegriffen wird, dann geben Sie den Namen der Service-routine hier an. Dieses Attribut wird nicht für einen Atom-Service oder ein entsprechendes Kategoriedokument verwendet.

**Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

**RESOURCETYPE**(**{FILE|PROGRAM|TSQUEUE}**)

Gibt den Typ der CICS-Ressource an, die die Daten für diesen Atom-Feed oder

diese Atom-Erfassung enthält. Dieses Attribut wird nicht für einen Atom-Service oder ein Atom-Kategoriedokument verwendet. Für dieses Attribut ist der Standardwert TSQUEUE definiert.

**FILE** Eine CICS-Datei. In einem bestimmten Datensatz in der Datei werden die Daten für einen einzelnen Atom-Eintrag bereitgestellt. Eine Datei, in der Atom-Einträge gespeichert sind, muss über einen eindeutigen Schlüssel für die Datensätze verfügen und Sie können keine alternative Indexdatei verwenden, die mit dem Attribut NONUNIQUEKEY definiert wurde. Sie können einen beliebigen Typ einer VSAM-Datei verwenden, um Atom-Einträge zu speichern. Hierbei ist allerdings zu beachten, dass ESDS-Dateien (ESDS = Entry-Sequenced Data Set; Datei in Zugangsfolge) sich nicht für einen Feed eignen, der als bearbeitbare Erfassung eingerichtet werden soll, weil in einer ESDS-Datei keine Datensätze gelöscht werden können. Sie können keine BDAM-Datei verwenden.

**PROGRAM**

Eine Serviceroutine, bei der es sich um ein CICS-Anwendungsprogramm handelt, das zur Bereitstellung von Inhalten für Atom-Einträge geschrieben wurde.

**TSQUEUE**

Eine Warteschlange für temporären Speicher. In einem bestimmten Datensatz in der Warteschlange für temporären Speicher werden die Daten für einen einzelnen Atom-Eintrag bereitgestellt.

**STATUS({ENABLED|DISABLED})**

Gibt an, ob das Atom-Dokument, das in dieser Ressourcendefinition angegeben ist, verfügbar ist oder nicht.

---

## BUNDLE-Ressourcen

Eine BUNDLE-Ressource definiert ein CICS-Bundle, bei dem es sich um eine Bereitstellungseinheit für eine Anwendung handelt. Bei einem Bundle handelt es sich um eine Zusammenstellung aus CICS-Ressourcen, -Artefakten und -Referenzen sowie aus einem Manifest, das Sie in einer CICS-Region bereitstellen können, um eine vollständige Anwendung oder eine Anwendungskomponente darzustellen.

Beim Manifest handelt es sich um eine Datei, in der die Inhalte des Bundles einschließlich der Angaben zu den in der CICS-Region zu erstellenden Ressourcen und zur Position der unterstützenden Artefakte beschrieben werden. Außerdem finden Sie dort die für die erfolgreiche Ausführung der Anwendung erforderlichen Voraussetzungen sowie die Services, die die Anwendung für andere Anwendungen bereitstellen kann.

Ein Bundle wird unter z/OS UNIX bereitgestellt und besteht aus einer Verzeichnisstruktur mit Artefakten. Die BUNDLE-Ressource definiert, wo das Bundle unter z/OS UNIX bereitgestellt wird und welchen Status es aufweist. Wenn Sie eine BUNDLE-Ressource aktivieren, dann liest CICS das Manifest und führt dynamisch die Erstellung der Anwendungsressourcen durch, die im Manifest definiert sind. Andere Ressourcen, die als Voraussetzungen für die Anwendung definiert wurden, müssen in der CICS-Region vorhanden sein, damit die BUNDLE-Ressource erfolgreich aktiviert werden kann.

Wenn Sie CICS-Bundles verwenden, orientiert sich der Lebenszyklus der installierten Ressourcen am Lebenszyklus des Bundles. Wenn Sie ein CICS-Bundle verfü-

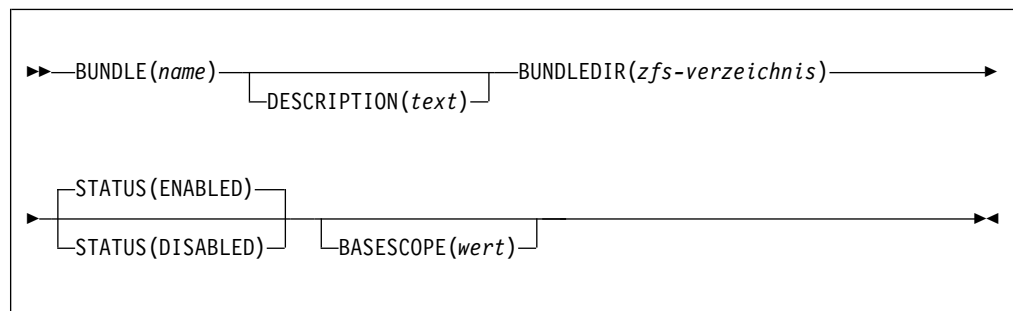
bar machen oder seine Verfügbarkeit zurücknehmen, wenn Sie es aktivieren oder inaktivieren oder es löschen, dann wird dieselbe Aktion auf alle Ressourcen angewendet, die vom CICS-Bundle dynamisch erstellt wurden. Informationen zu den Auswirkungen der Ressourcenpaketierung in CICS-Bundles finden Sie in Characteristics of bundled resources.

**Anmerkung:** Wenn Sie Ihr Bundle mit CICS Explorer erstellt und bereitgestellt haben, dann sollten Sie sicherstellen, dass der Quellcode des Bundles korrekt verwaltet wird. Das Bundle kann anhand der exportierten Daten in zFS nicht wiederhergestellt werden, sodass ein Fehler auf Ihrer Workstation zu Datenverlusten führen würde. Sie können die Exportfunktion von CICS Explorer verwenden, um das Bundle zu exportieren und im System für die Quellcodeverwaltung einzuchecken.

Informationen zum Installieren von BUNDLE-Ressourcendefinitionen finden Sie in Defining CICS bundles. Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with BAS BUNDLE resource definitions.

## BUNDLE-Attribute

Beschreibt die Syntax und die Attribute der BUNDLE-Ressource.



### BASESCOPE(wert)

Gibt die aus 1 - 255 Zeichen bestehende Zeichenfolge an, mit der der Bereich für ein Bundle definiert wird. Verwenden Sie möglichst einen URI (Uniform Resource Identifier). Verwenden Sie das Attribut BASESCOPE, wenn Sie ähnliche Bundles zu Gruppen zusammenfassen oder wenn Sie ein CICS-Bundle einer bestimmten Anwendung zuordnen wollen, das auf einer Plattform ausgeführt wird. Der URI für eine Anwendung weist das folgende Format auf:

cicsapplication://Plattform/Anwendungs-ID/Hauptversion/Nebenversion/Mikroversion

*Plattform* stellt hierbei den Namen der Plattform dar, auf der die Anwendung ausgeführt wird, *Anwendungs-ID* die ID des Anwendungsbundles, auf die die Haupt-, Neben- und Mikroversion der Anwendung folgen.

Standardmäßig erfolgt keine Angabe, sodass alle BUNDLE-Ressourcen in demselben Bereich installiert werden. Weitere Informationen hierzu finden Sie in Scoping of bundles.

#### Zulässige Zeichen:

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

### BUNDLE(name)

Gibt den aus einem bis acht Zeichen bestehenden Namen für das Bundle an.

**Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Sie sollten keine Namen verwenden, die mit der Zeichenfolge DFH beginnen, weil diese Zeichen zur Verwendung durch CICS reserviert sind.

CICS kann sie löschen oder verwerfen und durch CICS-Ressourcen ersetzen, wenn Sie ein Upgrade durchführen oder neue Funktionen oder Fixes bereitstellen.

**BUNDLEDIR**(zfs-verzeichnis)

Gibt den aus 1 - 255 Zeichen bestehenden vollständig qualifizierten Namen des Stammverzeichnisses für das Bundle unter z/OS UNIX an. Das erste Zeichen muss ein Schrägstrich (/) und das letzte Zeichen muss ebenfalls ein Schrägstrich (/) sein.

Der angegebene Wert muss ein gültiger Name einer UNIX-Datei sein:

- Er darf keine eingebetteten Leerzeichen enthalten.
- Er darf keine aufeinanderfolgenden Vorkommen des Zeichens / enthalten.
- Die Groß-/Kleinschreibung muss beachtet werden.

**Zulässige Zeichen:**

A-Z a-z 0-9 . / \_ # @ -

**DESCRIPTION**(text)

In diesem Feld können Sie eine Beschreibung für die Ressource angeben, die Sie definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Es gelten keine Einschränkungen für die Zeichen, die Sie verwenden können. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für jede linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Textcode zwei Hochkommas angeben.

**STATUS**(ENABLED | DISABLED)

Gibt den Anfangsstatus der BUNDLE-Ressource bei der Installation an.

**ENABLED**

Die BUNDLE-Komponente steht für die Verwendung zur Verfügung. CICS überprüft, ob alle Voraussetzungen für das Bundle in der Region vorhanden sind, und versucht, alle erforderlichen Ressourcen zu installieren und zu aktivieren, die im Bundle definiert sind.

**DISABLED**

Die BUNDLE-Komponente steht nicht für die Verwendung zur Verfügung. CICS überprüft die Voraussetzungen nicht und installiert die Ressourcen, die im Bundle definiert sind, aktiviert diese jedoch nicht.

---

## CONNECTION-Ressourcen

Eine CONNECTION-Ressource definiert ein fernes System, mit dem Ihr CICS-System über ISC (*Intersystem Communication; systemübergreifende Kommunikation*) oder über MRO (*Multiregion Operation; Mehrregionenbetrieb*) kommuniziert.

ISC verwendet das APPC- oder LUTYPE6.1-Kommunikationsprotokoll. MRO verwendet die IRC-, XM- oder die XCF/MRO-Zugriffsmethode.

Weitere Informationen zu diesem Thema finden Sie in IPCONN resources. Genau wie eine CONNECTION-Komponente definiert auch eine IPCONN-Komponente



eine Kommunikationsverbindung zu einem fernen System, in diesem Fall verwendet die Verbindung jedoch das TCP/IP-Protokoll.

Wenn Sie eine CONNECTION-Komponente definieren, dann geben Sie die Informationen an, die zur Identifikation des Systems und zum Festlegen seiner Basisattribute erforderlich sind. Sie geben in der SESSIONS-Definition Einzelheiten zu den Sitzungen an, die Sie zur Kommunikation mit dem System verwenden. CICS verwendet den CONNECTION-Namen zur Identifikation des anderen Systems, nachdem die Definition installiert wurde. Bei anderen CICS-Systemen, die via MRO verbunden sind, stimmt dieser Name normalerweise mit dem Namen überein, der in dem anderen CICS-System im Systeminitialisierungsparameter SYSIDNT angegeben ist. Bei anderen Systemen, die via ISC verbunden sind, basiert dieser Name normalerweise auf einem Akronym, das die Position des Systems oder die Organisation angibt, die Eigentümer des Systems ist (z. B. USA1 oder IBMC).

Der in REMOTESYSTEM angegebene Name in einer TRANSACTION-Definition oder in einer TERMINAL-Definition kann über den CONNECTION-Namen auf eine CONNECTION-Definition (oder über den IPCONN-Namen auf eine IPCONN-Definition) verweisen. Diese Attribute werden für das Transaktionsrouting verwendet.

Der in einer PROGRAM-Definition für REMOTESYSTEM angegebene Name kann über den CONNECTION-Namen auf eine CONNECTION-Definition (oder über den IPCONN-Namen auf eine IPCONN-Definition) verweisen. Dieses Attribut wird für die Verbindung zu verteilten Programmen verwendet.

In der CONNECTION-Definition werden **keine** zugeordneten SESSIONS-Komponenten aufgeführt.

Bevor Sie mit der Erstellung von Definitionen für Ressourcen für die übergreifende Kommunikation beginnen, sollten Sie sich mit den Informationen in Ressourcen für übergreifende Kommunikation definieren vertraut machen. Dort finden Sie zahlreiche nützliche Beispiele für die Attribute, die Sie für die unterschiedlichen Typen von Verbindungen und Sitzungen angeben müssen.

Besondere Hinweise zu unterschiedlichen Verbindungstypen:

#### **MRO-Verbindungen und -Sitzungen**

Sie definieren eine MRO-Verbindung mithilfe einer CONNECTION-Definition und die zugeordneten parallelen Sitzungen mithilfe einer SESSIONS-Definition.

#### **ACCESSMETHOD**

Geben Sie in der CONNECTION-Definition für diese Option IRC (für Kommunikation zwischen Regionen) oder XM (für speicherübergreifende Services) an. IRC wird verwendet, um die Verbindungen zu öffnen und zu schließen.

#### **PROTOCOL**

Geben Sie in der SESSIONS-Definition für PROTOCOL die Einstellung LU61 an. Geben Sie in der CONNECTION-Definition keinen Wert für PROTOCOL an.

#### **SENDPFX, SENDCOUNT, RECEIVEPFX, RECEIVECOUNT**

In einer SESSIONS-Definition geben Sie eine Anzahl für die Sendesitzungen und eine Anzahl für die Empfangssitzungen an. Die Werte, die Sie in diesen Attributen angeben, werden verwendet, um die Namen

der TCT-Einträge zu ermitteln, die bei der Installation der Definition erstellt werden (siehe hierzu Installing CONNECTION resource definitions).

### **APPC-Verbindungen und -Parallelsitzungen**

Für APPC werden die Sitzungen in Modussets zusammengefasst. Sie definieren jedes Modusset mit einer SESSIONS-Definition, sodass Sie über die gleiche Anzahl von SESSIONS-Definitionen und erforderlichen Modussets verfügen. Sie können die Verbindung als CONNECTION-Definition definieren. Die folgenden Attribute sind von Bedeutung:

#### **ACCESSMETHOD**

Geben Sie in der CONNECTION-Definition für dieses Attribut die Einstellung VTAM an.

#### **MAXIMUM**

Verwenden Sie diese Option, um die Anzahl der Sitzungen im Modusset zu steuern.

#### **MODENAME**

Geben Sie in der SESSIONS-Definition für jedes Modusset den Namen des Modussets mithilfe von MODENAME an. Hierbei handelt es sich um den Namen, unter dem das Modusset in CICS bei der Installation der Definition im aktiven System angelegt wird.

#### **PROTOCOL**

Geben Sie sowohl für CONNECTION- als auch für SESSIONS-Definitionen als Protokoll APPC an.

### **APPC-Einzelsitzungsterminal (LUTYPE6.2)**

Sie können ein APPC-Terminal als CONNECTION-SESSIONS-Paar oder als TERMINAL-TYPETERM-Paar definieren. Die TERMINAL-TYPETERM-Methode wird in „APPC-Einzelsitzungsterminal (LUTYPE6.2)“ auf Seite 216 beschrieben. Wenn Sie die CONNECTION-SESSIONS-Methode verwenden wollen, dann sind die folgenden Attribute von Bedeutung:

#### **ACCESSMETHOD**

Geben Sie in der CONNECTION-Definition für dieses Attribut die Einstellung VTAM an.

#### **MAXIMUM**

Für ein Einzelsitzungsterminal haben die Angaben 1,0 und 1,1 die gleichen Auswirkungen. (Weitere Informationen hierzu finden Sie in „CONNECTION-Attribute“ auf Seite 10.)

#### **MODENAME**

Geben Sie in der SESSIONS-Definition einen Wert für MODENAME an. Dies ist der Name, den CICS verwendet, um die Sitzung bei der Installation der Definition auf dem aktiven System zu identifizieren.

#### **PROTOCOL**

Geben Sie sowohl für CONNECTION- als auch für SESSIONS-Definitionen als Protokoll APPC an.

#### **SINGLESESS**

YES gibt an, dass die CONNECTION-Definition sich auf ein Einzelsitzungsterminal bezieht.

### **LUTYPE6.1-Verbindungen und -Sitzungen**

LUTYPE6.1-Verbindungen und -Sitzungen können auf die folgenden beiden Arten definiert werden:

- In einer CONNECTION- und einer SESSIONS-Definition.

- In einer CONNECTION-Definition und einer Reihe von SESSIONS-Definitionen (eine pro Sitzung erforderlich).

Wenn Ihre Sitzungen alle über die **identischen** Attribute verfügen sollen, dann definieren Sie jede Verbindung in einer CONNECTION-Definition und alle zugehörigen Sitzungen in einer SESSIONS-Definition.

#### **ACCESSMETHOD**

Geben Sie in der CONNECTION-Definition für dieses Attribut die Einstellung VTAM an.

#### **PROTOCOL**

Geben Sie in der SESSIONS-Definition und der CONNECTION-Definition hier LU61 an.

#### **RECEIVECOUNT, RECEIVEPFX, SENDCOUNT, SENDPFX**

Diese Attribute werden für MRO-Verbindungen und -Sitzungen verwendet.

Wenn für Ihre Sitzungen **unterschiedliche** Attribute definiert werden sollen, dann müssen Sie eine separate SESSIONS-Definition für jede Sitzung erstellen. Mit Ausnahme von NETNAMEQ ist diese Methode identisch mit der Methode für CICS-IMS-Sitzungen, die hier beschrieben wird.

**Anmerkung:** Für CICS-CICS-ISC-Verbindungen und -Sitzungen sollten Sie anstelle von LUTYPE6.1 APPC verwenden.

### **LUTYPE6.1-CICS-IMS-Verbindungen und -Sitzungen**

Für IMS muss jede Sitzung in einer separaten SESSIONS-Definition definiert werden, weil jede Sitzung über eine unterschiedliche NETNAMEQ-Angabe verfügen muss.

Sie definieren die Verbindung als eine CONNECTION-Definition und erstellen eine Anzahl von SESSIONS-Definitionen, wobei für jede SEND-Sitzung und RECEIVE-Sitzung jeweils eine Definition erforderlich ist.

#### **ACCESSMETHOD**

Geben Sie in der CONNECTION-Definition für dieses Attribut die Einstellung VTAM an.

#### **NETNAMEQ**

Dies ist der Name, der vom fernen IMS-System zur Identifikation der Sitzung verwendet wird.

#### **PROTOCOL**

Geben Sie sowohl für CONNECTION- als auch für SESSIONS-Definitionen als Protokoll LU61 an.

#### **SESSNAME**

Dies ist der Name, den CICS verwendet, um die Sitzung bei der Installation der Definition auf dem aktiven System zu identifizieren.

#### **RECEIVECOUNT**

#### **SENDCOUNT**

Verwenden Sie diese Attribute, um anzugeben, ob eine Sitzung eine SEND-Sitzung oder eine RECEIVE-Sitzung ist.

Bei einer RECEIVE-Sitzung handelt es sich um eine Sitzung, in der das lokale CICS-System das Primärsystem und der Konfliktverlierer ist. Sie wird angegeben, indem RECEIVECOUNT(1) angegeben und für SENDCOUNT der Standardwert (keine Angabe) übernommen wird. (SENDPFX oder RECEIVEPFX muss nicht angegeben werden.)

Bei einer SEND-Sitzung handelt es sich um eine Sitzung, in der das lokale CICS-System das Sekundärsystem und der Konfliktgewinner ist. Sie wird angegeben, indem SENDCOUNT(1) definiert und für RECEIVEDCOUNT der Standardwert (keine Angabe) übernommen wird.

### **INDIRECT-Verbindungen**

Eine INDIRECT-Verbindung stellt ein fernes System dar, für das Sie keine direkte Verbindung zum lokalen System definiert haben. Stattdessen kommunizieren die beiden Systeme über mindestens ein Transitsystem miteinander. Sie können diese Methode für das Transaktionsrouting verwenden. Das (indirekt verbundene) ferne System stellt immer die Terminalverwaltungsregion (TOR = Terminal-Ownning Region) dar. Das lokale System ist immer die Anwendungsverwaltungsregion (AOR = Application-Ownning Region) oder eine Transitregion innerhalb des Pfads für das Transaktionsrouting.

Indirekte Verbindungen werden nur dann benötigt, wenn Sie andere Terminals als z/OS Communications Server-Terminals für das Transaktionsrouting über Transitsysteme verwenden. Optional können Sie sie mit z/OS Communications Server-Terminals verwenden, wobei mehrere Pfade für das Transaktionsrouting möglich sind, um den bevorzugten Pfad zur Terminalverwaltungsregion anzugeben. Informationen zu den Gründen, die das Definieren von indirekten Verbindungen sinnvoll machen, und zu den Ressourcen, die für das Transaktionsrouting erforderlich sind, finden Sie in Defining indirect links for transaction routing.

Auf dem lokalen System benötigen Sie normale CONNECTION- und SESSIONS-Definitionen für die Transitsysteme, zu denen eine direkte Verbindung besteht. Für ACCESSMETHOD muss IRC oder XM mit der Angabe PROTOCOL(LU61) oder VTAM mit der Angabe PROTOCOL(APPC) festgelegt werden.

Für die INDIRECT-Verbindung (auch als indirekte Verbindung oder indirektes System bezeichnet) benötigen Sie auf dem lokalen System nur eine CONNECTION-Definition. Eine SESSIONS-Definition ist hingegen nicht erforderlich. Es werden die Sitzungen des Transitsystems verwendet. Die folgenden Attribute der CONNECTION-Definition sind von Bedeutung:

#### **ACCESSMETHOD**

Geben Sie hier INDIRECT an.

#### **INDSYS**

Geben Sie die CONNECTION-Definition für die MRO- oder APPC-Verbindung an, die den Ausgangspunkt eines Pfads zum Terminalverwaltungssystem darstellt.

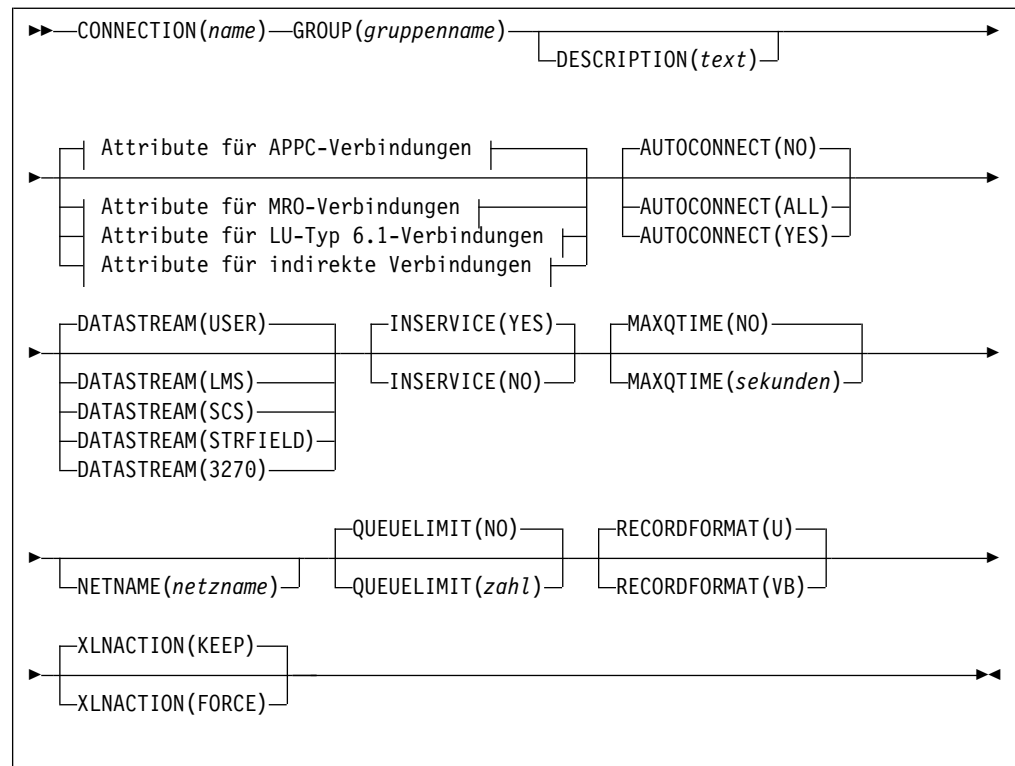
#### **NETNAME**

Geben Sie den Wert für APPLID für das Terminalverwaltungssystem an.

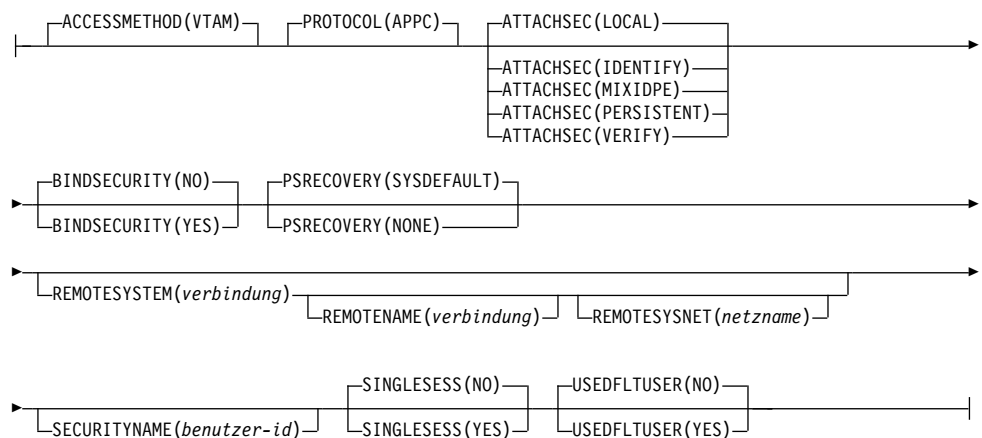
Informationen zur Installation von CONNECTION-Ressourcendefinitionen mit CEMT und CEDA finden Sie in Installing CONNECTION resource definitions. Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with BAS ISC/MRO connection resource definitions.

## **CONNECTION-Attribute**

Beschreibt die Syntax und die Attribute der CONNECTION-Ressource.

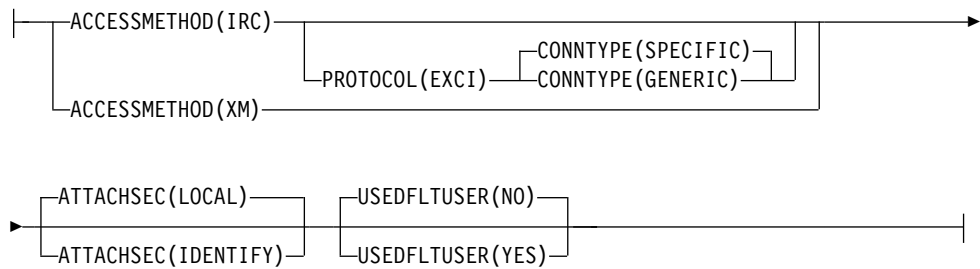


### Attribute für APPC-Verbindungen:

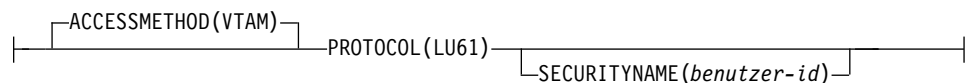


**Anmerkung:** VTAM wird jetzt als z/OS Communications Server bezeichnet.

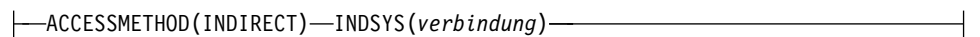
### Attribute für MRO-Verbindungen:



### Attribute für LU-Typ 6.1-Verbindungen:



### Attribute für indirekte Verbindungen:



### ACCESSMETHOD({VTAM|INDIRECT|IRC|XM})

Gibt die Zugriffsmethode an, die für diese Verbindung verwendet werden soll.

#### VTAM

Die Kommunikation zwischen der lokalen CICS-Region und dem in dieser Verbindungsdefinition definierten System erfolgt über z/OS Communications Server. Sie können z/OS Communications Server Intersystem Communication (ISC) für Systeme verwenden, die sich in unterschiedlichen MVS-Images oder in unterschiedlichen Adressräumen innerhalb desselben MVS-Images befinden.

#### INDIRECT

Die Kommunikation zwischen dem lokalen CICS-System und dem in dieser Verbindungsdefinition definierten System erfolgt über das System, das im Operanden INDSYS angegeben ist.

#### IRC

Die Kommunikation zwischen der lokalen CICS-Region und der in dieser Verbindungsdefinition definierten Region erfolgt über das IRC-Programm (IRC = Interregion Communication; Kommunikation zwischen Regionen) mit dem Namen DFHIRP, wobei der SVC-Modus (im Gegensatz zum speicherübergreifenden Modus (XM)) von DFHIRP verwendet wird.

**Anmerkung:** Diese Verwendung des Begriffs IRC ist spezifischer als seine allgemeine Verwendung.

Sie können IRC für den Mehrregionenbetrieb (MRO = Multiregion

Operation) bei Regionen verwenden, die sich in demselben MVS-Image oder in unterschiedlichen MVS-Images in einem Sysplex befinden.

**XM** Für die MRO-Kommunikation zwischen der lokalen CICS-Region und der in der zugehörigen CONNECTION-Definition definierten Region werden die speicherübergreifenden MVS-Services verwendet. Die einleitende Verbindung erfolgt über das IRC-Programm DFHIRP, wobei (anstelle des SVC-Modus) der XM-Modus von DFHIRP verwendet wird. Sie können XM für den Mehrregionenbetrieb bei Regionen verwenden, die sich in demselben MVS-Image oder in unterschiedlichen MVS-Images in einem Sysplex befinden.

**Anmerkung:** Der CICS-Supervisoraufruf des Typs 3 (SVC = Supervisor Call) wird für XM weiterhin benötigt, weil DFHIRP beim Öffnen der Verbindung verwendet wird. Weitere Informationen zu SVCs finden Sie in Installing the CICS SVCs in Installing.

Die speicherübergreifenden MVS-Services werden nur verwendet, wenn als Zugriffsmethode (ACCESSMETHOD) am anderen Ende der Verbindung ebenfalls XM definiert wurde.

Wenn die MRO-Partner sich in unterschiedlichen MVS-Images in einem Sysplex befinden und in CONNECTION entweder IRC oder XM angegeben ist, dann verwendet CICS automatisch XCF als Zugriffsmethode und ignoriert die Angabe von IRC oder XM.

**Anmerkung:** XCF kann nicht explizit definiert werden. Wenn Sie XCF verwenden möchten, dann müssen Sie entweder IRC oder XM angeben. Weitere Informationen zu XCF finden Sie in Cross-system multiregion operation (XCF/MRO).

#### **ATTACHSEC({LOCAL|IDENTIFY|VERIFY|PERSISTENT|MIXIDPE})**

Gibt die erforderliche Stufe der Benutzersicherheit an, die für die Herstellung der Verbindung erforderlich ist.

##### **IDENTIFY**

Eingehende Verbindungsanforderungen müssen eine Angabe zur Benutzer-ID enthalten. Geben Sie IDENTIFY ein, wenn das System, das die Verbindung herstellt, über einen Sicherheitsmanager verfügt. Dies ist z. B. bei einem anderen CICS-System der Fall.

##### **LOCAL**

Unter CICS muss der Client keine Benutzer-ID oder ein Kennwort angeben. Alle Anforderungen werden unter der Benutzer-ID ausgeführt, die im Attribut SECURITYNAME angegeben wurde. Wenn im Attribut PROTOCOL in der CONNECTION-Definition die Einstellung 'LU6.1' festgelegt ist, dann müssen Sie LOCAL angeben.

##### **MIXIDPE**

Eingehende Verbindungsanforderungen können entweder den Sicherheitstyp IDENTIFY oder PERSISTENT oder beide Sicherheitstypen verwenden. Der verwendete Sicherheitstyp hängt von der eingehenden Verbindungsanforderung ab.

##### **PERSISTENT**

Bei eingehenden Verbindungsanforderungen muss die erste Verbindungsanforderung eine Angabe zur Benutzer-ID und ein Benutzerkennwort enthalten. Für nachfolgende Verbindungsanforderungen ist

lediglich die Benutzer-ID erforderlich. Diese sollte nur zwischen einer programmierbaren Workstation (z. B. einem IBM® Personal Computer) und CICS verwendet werden.

#### **VERIFY**

Eingehende Verbindungsanforderungen müssen eine Angabe zur Benutzer-ID und ein Benutzerkennwort enthalten. Geben Sie VERIFY ein, wenn das System, das die Verbindung herstellt, keinen Sicherheitsmanager hat und daher nicht vertrauenswürdig ist. Geben Sie für die CICS-zu-CICS-Kommunikation nicht VERIFY an, weil CICS keine Kennwörter versendet.

#### **AUTOCONNECT({NO|YES|ALL})**

Für Systeme, die ACCESSMETHOD(VTAM) verwenden, können Sie mit AUTOCONNECT(YES) oder (ALL) angeben, dass Sitzungen eingerichtet werden sollen (dies bedeutet, dass eine BIND-Aktion ausgeführt werden soll). Solche Sitzungen werden während der CICS-Initialisierung oder dann eingerichtet, wenn Sie den Befehl CEMT oder EXEC CICS SET VTAM OPEN verwenden, um die Kommunikation mit z/OS Communications Server zu starten. Wenn die Verbindung zu diesem Zeitpunkt nicht hergestellt werden kann, weil das ferne System nicht verfügbar ist, dann müssen Sie die Verbindung anschließend anfordern, indem Sie den Befehl CEMT oder EXEC CICS SET CONNECTION(sysid) INSERVICE ACQUIRED verwenden. Dies gilt nicht, wenn das ferne System in der Zwischenzeit verfügbar wird und selbst die Kommunikation einleitet.

Für APPC-Verbindungen, für die SINGLESESS(NO) angegeben wurde, versucht CICS, beim Systemstart eine Bindung für die Sitzungen des LU Services Manager in der Modusgruppe SNASVCMG durchzuführen.

Für Verbindungsdefinitionen, für die SINGLESESS(YES) angegeben wurde, wird der Operand AUTOCONNECT ignoriert. Verwenden Sie stattdessen den Operanden AUTOCONNECT aus der Sitzungsdefinition.

**ALL** In dieser Definition ist ALL gleichbedeutend mit YES, Sie können jedoch ALL angeben, um die Konsistenz mit der Sitzungsdefinition zu wahren.

AUTOCONNECT(ALL) sollte für Verbindungen zu anderen CICS-Systemen nicht angegeben werden, weil dies zu einer bind-race-Situation führen kann.

**NO** CICS versucht nicht, bei der Verbindungsherstellung Sitzungsbindungen durchzuführen.

**YES** CICS versucht bei der Verbindungsherstellung nur Bindungen für Sitzungen herzustellen, bei denen es sich um Konfliktgewinner handelt.

Die Option AUTOCONNECT ist für LU6.1-Verbindungsdefinitionen nicht anwendbar. Für LU6.1-Verbindungen müssen Sie AUTOCONNECT(YES) in der SESSIONS-Definition angeben, wenn die Verbindung während der Initialisierung oder bei der CEDA-Installation eingerichtet werden soll. Geben Sie AUTOCONNECT(NO) in der SESSIONS-Definition an, wenn die Verbindung nicht während der Initialisierung oder bei der CEDA-Installation eingerichtet werden soll.

#### **BINDPASSWORD**

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt. Weitere Informationen hierzu finden Sie in Obsolete attributes.



**BINDSECURITY({NO|YES}) (nur APPC)**

Gibt an, ob für die Sicherheit beim Binden ein ESM (External Security Manager; externer Sicherheitsmanager) verwendet werden soll.

**NO** Es ist keine externe Bindungssicherheit erforderlich.

**YES** Wenn die Sicherheitsfunktion aktiv ist und der Systeminitialisierungsparameter XAPPC auf YES gesetzt ist, dann versucht CICS, den Sitzungsschlüssel aus RACF zu extrahieren, um die Bindungssicherheit zu erreichen. Wenn kein RACF-Profil verfügbar ist, dann schlägt die Bindung fehl.

**CONNECTION(*name*)**

Gibt den Namen dieser Verbindungsdefinition an. Der Name kann bis zu vier Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Dies ist der Name, der in Datei-, Terminal-, Transaktions- und Programmdefinitionen in REMOTESYSTEM angegeben wird. Es sollten keine Terminal- und Verbindungsdefinitionen mit demselben Namen vorhanden sein.

**CONNTYPE({SPECIFIC|GENERIC})**

Für EXCI-Verbindungen (EXCI = External CICS Interface; externe CICS-Schnittstelle) gibt dieses Attribut die Art der Verbindung an.

**GENERIC**

Die Verbindung wird für die Kommunikation zwischen einem Nicht-CICS-Clientprogramm und dem CICS-System genutzt und ist generisch. Eine generische Verbindung stellt eine MRO-Verbindung mit einer Anzahl von Sitzungen dar, die von mehreren EXCI-Benutzern gemeinsam genutzt werden. Für eine generische Verbindung können Sie das Attribut NETNAME nicht angeben.

**SPECIFIC**

Die Verbindung wird für die Kommunikation zwischen einem Nicht-CICS-Clientprogramm und der CICS-Region genutzt und ist eine spezielle Verbindung. Eine spezielle Verbindung stellt eine MRO-Verbindung mit mindestens einer Sitzung dar, die einem einzigen Benutzer in einem Clientprogramm zugeordnet ist. Für eine spezielle Verbindung ist die Angabe von NETNAME zwingend erforderlich.

**DATASTREAM({USER|3270|SCS|STRFIELD|LMS})**

Gibt den Typ des Datenstroms an.

**LMS** Der Datenstrom ist ein LMS-Datenstrom (LMS = Logical Message Services), der aus FMH4- und FMH8-Elementen besteht, die in der LUTYPE6.1-Architektur definiert sind.

**SCS** Der Datenstrom ist ein SCS-Datenstrom (SCS = SNA Character String; SNA-Zeichenfolge), der in der LUTYPE6.1-Architektur definiert ist.

**STRFIELD**

Der Datenstrom ist ein Datenstrom für ein strukturiertes Feld, der in der LUTYPE6.1-Architektur definiert ist.

**USER** Verwenden Sie für DATASTREAM den Standardwert USER, wenn der

Datenstrom benutzerdefiniert ist. Wenn die Kommunikation zwischen mehreren CICS-Systemen durchgeführt wird, dann verwenden Sie für DATASTREAM immer den Standardwert USER.

**3270** Bei dem Datenstrom handelt es sich um einen 3270-Datenstrom, der in der LUTYPE6.1-Architektur definiert ist.

**DESCRIPTION**(*text*)

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

**GROUP**(*gruppenname*)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

**INDSYS**(*verbindung*)

Gibt den Namen einer anderen CONNECTION-Definition an, in der ein Transitsystem definiert ist, das zur Übermittlung der Kommunikation zwischen diesem System und dem fernen System verwendet wird. Der Name kann bis zu vier Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Sie können ein Transitsystem nur dann angeben, wenn Sie ACCESSMETHOD(INDIRECT) festlegen.

**INSERVICE**(**YES**|**NO**)

Gibt den Status der Verbindung an, die definiert wird.

**NO** Über die Verbindung können weder Nachrichten empfangen noch Eingaben übertragen werden.

**YES** Über die Verbindung können Transaktionen eingeleitet werden und Nachrichten automatisch gesendet werden.

**MAXQTIME**(**NO**|*sekunden*)

Gibt eine Zeitsteuerung für die Wartezeit für Zuordnungsanforderungen in der Warteschlange an, die auf freie Sitzungen in einer Verbindung warten, die möglicherweise nicht mehr reagiert. Die maximal zulässige Zeitdauer in der Warteschlange wird nur dann verwendet, wenn in QUEUELIMIT ein Warte-

schlangengrenzwert angegeben wurde. In diesem Fall wird das Zeitlimit nur dann angewendet, wenn die Länge der Warteschlange den Warteschlangengrenzwert erreicht hat.

**NO** CICS verwaltet die Warteschlange für Zuordnungsanforderungen, die auf eine freie Sitzung warten. Für die Zeitdauer, für die Anforderungen in der Warteschlange verbleiben dürfen, wird kein Grenzwert festgelegt. Allerdings können für einzelne Anforderungen die DTIMOUT-Mechanismen gelten. In diesem Fall wird in der Parameterliste XZIQUE (im Feld UEPMXQT) der Wert X'FFFF' übergeben.

#### *sekunden*

Die ungefähre Obergrenze für die Zeitdauer, die Zuordnungsanforderungen in der Warteschlange für eine Verbindung eingereiht bleiben dürfen, die möglicherweise nicht mehr reagiert. Die Anzahl gibt einen Wert in Sekunden im Bereich zwischen 0 und 9999 an.

CICS verwendet das Attribut für die maximale Zeitdauer in der Warteschlange, um eine Warteschlange mit wartenden Zuordnungsanforderungen zu steuern. Wenn die Anzahl der in die Warteschlange eingestellten Zuordnungsanforderungen das Warteschlangenlimit (QUEUELIMIT) erreicht und dann für die Verbindung eine neue Zuordnungsanforderungen empfangen wird, dann wird die Warteschlange bereinigt, wenn die Verarbeitungsgeschwindigkeit der Warteschlange erkennen lässt, dass die durchschnittliche Verarbeitungszeitdauer der neuen Zuordnung den Wert für die maximale Warteschlangenzeitdauer übersteigt. In diesem Fall wird die Nachricht DFHZC2300 ausgegeben. Wenn die Warteschlange bereinigt wird, dann gibt das System für die in der Warteschlange befindlichen Zuordnungsanforderungen einen Fehler vom Typ SYSIDERR aus.

Es werden keine weiteren Anforderungen in die Warteschlange eingestellt, bis die Verbindung erfolgreich eine Sitzung freigeben konnte. Zu diesem Zeitpunkt gibt CICS die Meldung DFHZC2301 aus und nimmt die normale Warteschlangensteuerung wieder auf.

Die Einstellung von Zuordnungsanforderungen in die Warteschlange kann auch über das globale Benutzerexitprogramm XZIQUE gesteuert werden. Hierbei können Sie die von CICS bereitgestellten Statistikdaten verwenden, in denen der Status der Verbindung angegeben ist. Sie können diese Statistikdaten zusammen mit den von Ihnen angegebenen Werten für das Warteschlangenlimit und die maximale Warteschlangenzeitdauer verwenden, um differenzierte Entscheidungen zu den Warteschlangen zu treffen.

Der Wert für MAXQTIME wird in der XZIQUE-Parameterliste an das globale Benutzerexitprogramm XZIQUE übergeben, wenn der Exit aktiviert wird. Programmierinformationen zum Schreiben des globalen Benutzerexitprogramms XZIQUE finden Sie in XZIQUE exit for managing MRO and APPC intersystem queues.

Sie können auch die Option NOQUEUE|NOSUSPEND im Befehl ALLOCATE angeben, um zu verhindern, dass eine explizite Anforderung in die Warteschlange eingestellt wird. Programmierinformationen zu diesen API-Optionen finden Sie in CICS command summary.

#### **NETNAME**(*netzname*)

Gibt den Netznamen an, der das ferne System identifiziert. Der Name kann bis zu acht Zeichen lang sein. Der Name wird auf Basis der Regeln der Assemblersprache gebildet. Er muss mit einem alphabetischen Zeichen beginnen.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

In NETNAME wird die Anwendungs-ID (APPLID) des fernen Systems oder der fernen Region angegeben, es sei denn, Sie definieren eine LUTYPE6.1- oder APPC-Verbindung zu einer generischen z/OS Communications Server-Ressourcengruppe.

- Wenn Sie eine LUTYPE6.1-Verbindung zu einer generischen Ressource definieren, dann muss in NETNAME der generische Ressourcename und nicht die Anwendungs-ID (APPLID) eines der Gruppenmitglieder angegeben werden.
- Wenn Sie eine APPC-Verbindung zu einer generischen Ressource definieren, dann kann in NETNAME entweder der generische Ressourcename der Gruppe oder die Anwendungs-ID (Mitgliedsname) eines der Gruppenmitglieder angegeben werden. Wenn Sie allerdings einen Mitgliedsnamen angeben, und wenn dieses CICS-System nicht selbst Mitglied einer generischen CICS-Ressource ist, dann muss die Verbindung immer von diesem CICS-System angefordert werden („dieses CICS-System“ bedeutet hierbei die CICS-Region, in der die Verbindungsdefinition installiert wurde).

Für z/OS Communications Server stellt APPLID die Bezeichnung der Anweisung VBUILD TYPE=APPL für das ferne VTAM-System dar.

Wenn Sie für NETNAME keinen Wert angeben, dann wird standardmäßig der CONNECTION-Name verwendet.

Für doppelte NETNAME-Angaben gelten verschiedene Regeln. Folgende Elemente sind **nicht zulässig**:

- Zwei oder mehrere APPC-Verbindungen mit der gleichen Angabe für NETNAME.
- Eine APPC-Verbindung und eine LUTYPE6.1-Verbindung mit derselben Angabe für NETNAME.
- Zwei oder mehrere IRC-Verbindungen mit der gleichen Angabe für NETNAME.
- Zwei oder mehrere ferne APPC-Verbindungen mit der gleichen Angabe für NETNAME.
- Eine ferne APPC-Verbindung mit derselben Angabe für NETNAME, die auch für eine andere Verbindung oder ein lokales Terminal festgelegt wurde.

Folgende Elemente sind **zulässig**:

- Eine IRC-Verbindung und eine LUTYPE6.1-Verbindung mit derselben Angabe für NETNAME.
- Eine IRC-Verbindung und eine APPC-Verbindung mit derselben Angabe für NETNAME.
- Zwei oder mehrere LUTYPE6.1-Verbindungen mit der gleichen Angabe für NETNAME.
- Eine beliebige Verbindung mit der gleichen Angabe für NETNAME, die auch für ein fernes Terminal angegeben wurde.

**Für Verbindungen, die die LU-Alias-Funktion von z/OS Communications Server verwenden, gilt Folgendes:**

- **APPC-Synchronisationsebene 1:** Wenn die CICS-Region die dynamische LU-Alias-Funktion von z/OS Communications Server unterstützt (wenn also in der Anweisung APPL der CICS-Region LUAPFX=xx angegeben ist), dann wird angenommen, dass sich NETNAME in demselben Netz wie die CICS-Region befindet. Wenn dies nicht der Fall ist, dann muss die Ressource über eine lokale CDRSC-Definition für z/OS Communications Server mit der Angabe LUALIAS=netzname verfügen. Hierbei muss netzname mit der Angabe für NETNAME übereinstimmen, die in dieser CONNECTION-Definition festgelegt ist. Bei APPC-Verbindungen der Synchronisationsebene 1 handelt es sich generell um Workstations.

Beachten Sie hierbei, dass bestimmte Ressourcen der Synchronisationsebene 1 möglicherweise in die Synchronisationsebene 2 überführt werden, und zwar abhängig von ihrer Verbindung zu CICS. Beispiel: Wenn TXSeries *kein* PPC-Gateway verwendet, dann weist die Verbindung die Synchronisationsebene 1 auf. Wird ein PPC-Gateway verwendet, dann weist die Verbindung die Synchronisationsebene 2 auf.

- **APPC-Synchronisationsebene 2 und LUTYPE6.1:** Dieser Netznamen (NETNAME) wird als eindeutig angenommen. CICS gleicht ihn mit dem Netznamen ab, der in der Anweisung APPL von z/OS Communications Server definiert wurde. Diese Verbindungen bestehen im Allgemeinen zwischen CICS-Systemen. Es kann sich jedoch auch um TXSeries-Verbindungen handeln, die über ein PPC-Gateway hergestellt werden.

#### Regeln zu NETNAME und APPLID:

- - Wenn eine installierte CONNECTION-Definition denselben Namen wie eine installierte IPCONN-Definition aufweist, muss die Anwendungs-ID (APPLID) der IPCONN-Definition mit dem Netznamen (NETNAME) der CONNECTION-Definition übereinstimmen. Andernfalls wird abhängig von der jeweiligen Situation eine der folgenden Nachrichten ausgegeben:
  - DFHIS3009 - Wenn der Fehler während der automatischen IPCONN-Installation festgestellt wird.
  - DFHAM4913 - Wenn der Fehler während der IPCONN-Installation festgestellt wird.
  - DFHZC6312 - Wenn der Fehler während der CONNECTION-Installation oder der automatischen Installation festgestellt wird.
- Die IPCONN-Definition hat Vorrang gegenüber der CONNECTION-Definition. Dies bedeutet, dass CICS bei Vorhandensein einer IPCONN- und einer CONNECTION-Definition mit demselben Namen die IPCONN-Verbindung verwendet.
- Eine CONNECTION- und eine IPCONN-Komponente mit den gleichen Werten für NETNAME und APPLID müssen nicht zwingend den gleichen Namen haben.

Hierdurch ist es möglich, für die Kommunikation über TCP/IP eine andere System-ID zu verwenden, anstatt die CICS-Standardeinstellung zu benutzen und alle unterstützten Funktionen über die IPCONN-Verbindung weiterzuleiten, sofern diese existiert.

**PROTOCOL({APPC|LU61|EXCI|leer})**

Gibt den Typ des Protokolls an, das für die Verbindung verwendet werden soll.

#### APPC (LUTYPE6.2-Protokoll)

Advanced Program-to-Program Communication oder APPC-Protokoll.

Dies ist der Standardwert für ACCESSMETHOD(VTAM). Geben Sie diesen Wert für die CICS-CICS-ISC an.

- leer** Der Mehrregionenbetrieb (MRO = Multiregion Operation) zwischen CICS-Regionen. Für MRO darf für PROTOCOL keine Angabe definiert werden und in der SESSIONS-Definition müssen Sie für PROTOCOL die Einstellung 'LU6.1' angeben.
- EXCI** Die externe CICS-Schnittstelle. Definieren Sie diese Option, um anzugeben, dass diese Verbindung durch ein Nicht-CICS-Clientprogramm über die externe CICS-Schnittstelle (EXCI = External CICS Interface) verwendet wird.
- LU61** LUTYPE6.1-Protokoll. Geben Sie diese Option für die CICS-CICS-ISC oder die CICS-IMS-ISC, nicht jedoch für MRO an.

#### **PSRECOVERY({SYSDEFAULT|NONE})**

In einer CICS-Region, die mit der Unterstützung für persistente Sitzungen ausgeführt wird, gibt diese Option an, ob und wie LU6.2-Sitzungen beim Neustart des Systems innerhalb des Verzögerungsintervalls für persistente Sitzungen wiederhergestellt werden.

##### **NONE**

Die Bindung aller Sitzungen wird aufgehoben und sie erhalten die Markierung 'out-of-service' (außer Betrieb). Es wird keine CNOS-Wiederherstellung durchgeführt.

##### **SYSDEFAULT**

Wenn ein CICS-System, auf dem ein Fehler aufgetreten ist, innerhalb des Verzögerungsintervalls für persistente Sitzungen neu gestartet wird, dann werden die folgenden Aktionen ausgeführt:

- Die Modusgruppen der Benutzer werden mit dem Wert für SESSIONS RECOVOPTION wiederhergestellt.
- Die Modusgruppe SNASVCMG wird wiederhergestellt.
- Die Verbindung wird wieder in den Status ACQUIRED überführt und der zuletzt vorhandene variable CNOS-Status wird zurückgegeben.

#### **QUEUELIMIT({NO|zahl})**

Gibt die maximale Anzahl von Zuordnungsanforderungen an, die von CICS in die Warteschlange eingestellt werden soll, während auf freie Sitzungen gewartet wird:

**NO** Die Anzahl der Zuordnungsanforderungen, die von CICS in die Warteschlange eingestellt werden kann, während auf eine freie Sitzung gewartet wird, ist unbegrenzt. In diesem Fall wird in der Parameterliste XZIQUE (im Feld UEPQUELM) der Wert X'FFFF' übergeben.

*zahl* Die maximal zulässige Anzahl von Zuordnungsanforderungen (im Bereich zwischen 0 und 9999), die von CICS in die Warteschlange der Verbindung eingestellt werden können, während auf eine freie Sitzung gewartet wird. Wenn die Anzahl der in die Warteschlange eingestellten Zuordnungsanforderungen den definierten Grenzwert erreicht, dann wird für nachfolgende Zuordnungsanforderungen der Fehler SYSDERR ausgegeben, bis die Anzahl der Zuordnungsanforderungen in der Warteschlange wieder unter den Grenzwert fällt.

Dieses Warteschlangenlimit wird in der XZIQUE-Parameterliste an das globale Benutzerexitprogramm XZIQUE übergeben, wenn der Exit aktiviert wird.

Die Einstellung von Zuordnungsanforderungen in die Warteschlange kann auch über das Attribut MAXQTIME und über das globale Benutzerexitprogramm XZIQUE gesteuert werden. Weitere Informationen zur Steuerung von Warteschlangen finden Sie im Abschnitt zum Attribut MAXQTIME.

**Anmerkung:** Die Neuvereinbarung der Bindung (BIND) wird nicht ausgelöst, auch wenn nicht benutzte sekundäre Sitzungen vorhanden sind. Wird nicht der Befehl CEMT SET MODE verwendet, um die Neuvereinbarung zu erzwingen, wird das Warteschlangenlimit verwendet, sobald alle primären Sitzungen belegt sind.

#### **RECORDFORMAT({U|VB})**

Gibt den Typ der SNA-Kette an.

**U** Verwenden Sie für RECORDFORMAT den Standardwert U, wenn es sich bei der SNA-Kette um einen einzigen, nicht geblockten Datenstrom handelt. In der SNA-Kette können private Blockalgorithmen vorhanden sein. Verwenden Sie für RECORDFORMAT den Standardwert U, wenn die Kommunikation zwischen mehreren CICS-Systemen stattfindet.

**VB** Die SNA-Kette wird auf Basis des VLVB-Standards gemäß der Definition in der LUTYPE6.1-Architektur formatiert.

#### **REMOTENAME(*verbindung*)**

Gibt den Namen an, unter dem die APPC-Verbindung für das Transaktionsrouting auf dem System oder in der Region bekannt ist, das bzw. die die Verbindung verwaltet. Der Name kann bis zu vier Zeichen lang sein.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Beim fernen System oder der fernen Region kann es sich um eine APPC-Einheit (siehe hierzu APPC-Einheiten für Transaktionsrouting) handeln.

#### **REMOTESYSNET(*netzname*)**

Gibt den Netznamen (APPLID) des Systems an, das die Verbindung verwaltet. Der Name kann bis zu acht Zeichen lang sein. Er wird auf Basis der Regeln der Assemblersprache gebildet und muss mit einem alphabetischen Zeichen beginnen.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Das Attribut REMOTESYSNET kann für das Transaktionsrouting zu fernen APPC-Systemen oder -Einheiten verwendet werden. Es besteht keine direkte Verbindung zwischen der Region, in der diese Definition installiert ist, und dem System, das die Verbindung zu der fernen Einheit verwaltet. REMOTESYSNET muss nicht angegeben werden, wenn folgende Bedingungen zutreffen:

- Sie definieren eine lokale Verbindung (d. h. REMOTESYSNET ist nicht angegeben oder gibt die System-ID des lokalen Systems an).

- REMOTESYSTEM benennt eine direkte Verbindung zu dem System, das die Verbindung verwaltet. Hierbei ist der folgende Sonderfall zu beachten: Wenn die Verbindungsverwaltungsregion Mitglied einer Gruppe für generische z/OS Communications Server-Ressourcen ist und die direkte Verbindung zu ihr eine APPC-Verbindung ist, dann muss möglicherweise REMOTESYSNET angegeben werden. REMOTESYSNET wird in diesem Fall benötigt, wenn der in der CONNECTION-Definition angegebene NETNAME-Wert für die direkte Verbindung mit dem generischen Ressourcennamen (und nicht mit der Anwendungs-ID) der Verbindungsverwaltungsregion übereinstimmt.

#### **REMOTESYSTEM(*verbindung*)**

Gibt den Namen an, der die Verbindung für die übergreifende Kommunikation mit dem System identifiziert, das die Verbindung verwaltet. Der Name kann bis zu vier Zeichen lang sein.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Dies ist der CONNECTION-Name in der Verbindungsdefinition der Verbindung für die übergreifende Kommunikation.

REMOTESYSTEM wird für das Transaktionsrouting zu fernen APPC-Systemen oder -Einheiten verwendet. Erfolgt keine Angabe oder wird die System-ID des lokalen Systems angegeben, dann handelt es sich um eine lokale Verbindung dieses Systems. Wird der Name eines anderen Systems angegeben, dann handelt es sich um eine ferne Verbindung. Aus diesem Grund können Sie dieselbe Definition für die Verbindung in lokalen und auch fernen Systemen verwenden.

Sind zwischen diesem CICS-System und der Region, die Eigner (der Verbindung zu) der Einheit ist, Transitsysteme vorhanden, dann muss in REMOTESYSTEM die erste Verbindung im Pfad zu der Region angegeben werden, die Eigner der Einheit ist. Sind mehrere mögliche Pfade vorhanden, dann muss die erste Verbindung im bevorzugten Pfad angegeben werden.

#### **SECURITYNAME(*benutzer-id*)**

Nur bei APPC- und LU6.1-Verbindungen ist dies der Sicherheitsname des fernen Systems.

In einem CICS-System mit initialisierter Sicherheitsfunktion (SEC=YES) wird der Sicherheitsname verwendet, um die Berechtigung des fernen Systems einzurichten.

**Anmerkung:** Wenn in der SESSIONS-Definition, die der Verbindungsdefinition zugeordnet ist, die Option USERID angegeben wird, dann überschreibt ihr Wert die Benutzer-ID, die im Attribut SECURITYNAME angegeben ist. Dieser Wert wird dann für die Verbindungssicherheit verwendet.

Der Sicherheitsname (oder der Wert für USERID in der Sitzungsdefinition) muss eine gültige RACF-Benutzer-ID auf Ihrem System darstellen. Der Zugriff auf geschützte Ressourcen auf Ihrem System basiert auf dem RACF-Benutzerprofil und seiner Gruppenzugehörigkeit.

#### **SINGLESESS({NO|YES})**

Gibt an, ob die Definition für ein APPC-Terminal mit einer APPC-Verbindung für Einzelsitzungen zu CICS erstellt wurde.



**NO** Die Definition wurde nicht für einen APPC-Verbindung für Einzelsitzungen zu CICS erstellt.

**YES** Die Definition wird für ein APPC-Terminal mit einer APPC-Verbindung für Einzelsitzungen zu CICS erstellt.

Das Attribut MODENAME der SESSIONS-Definition kann verwendet werden, um einen Modusnamen für das Einzelsitzungsmodusset bereitzustellen.

Ein APPC-Einzelsitzungsterminal kann auch als TERMINAL-TYPETERM-Definition definiert werden. Sowohl die TERMINAL-TYPETERM-Definition als auch die CONNECTION-Definition können automatisch installiert werden. Wenn Sie die Verwendung der automatischen Installation in Betracht ziehen, dann erhalten Sie weiterführende Informationen in Autoinstalling z/OS Communications Server terminals.

**USEDFTUSER({NO|YES})) (nur APPC und MRO)**

Gibt die Aktion an, die ausgeführt werden soll, wenn eine eingehende FMH5-Anforderung nicht die Sicherheitsinformationen enthält, die vom Attribut ATTACHSEC impliziert werden.

**NO** Die Verbindungsanforderung wird zurückgewiesen und das System gibt eine Nachricht zu einem fehlerhaften Protokoll aus.

**YES** Die Verbindungsanforderung wird akzeptiert und der Transaktion wird die *Standard-Benutzer-ID* zugeordnet.

Weitere Informationen hierzu finden Sie in Implementing LU6.2 security und Implementing MRO security.

**XLNACTION({KEEP|FORCE})) (nur APPC und MRO)**

XLNACTION gibt die Aktion an, die ausgeführt werden soll, wenn ein neuer Protokollname vom Partnersystem empfangen wird. Der Empfang eines neuen Protokollnamens gibt an, dass der Partner seine Wiederherstellungsinformationen gelöscht hat.

**Anmerkung:** Im Mehrregionenbetrieb (MRO = Multiregion Operation) werden hier Verbindungen abgedeckt, bei denen für ACCESSMETHOD entweder IRC oder XM angegeben wurde.

**FORCE**

Die vordefinierten Entscheidungen für unbestätigte Arbeitseinheiten (gemäß Definition in den Attributen für unbestätigte Elemente der Transaktionsdefinition) werden implementiert, bevor neue Arbeiten mit dem neuen Protokollnamen gestartet werden. CICS löscht auch alle Informationen, die für möglicherweise erforderliche Auflösungen von Arbeitseinheiten aufbewahrt wurden, die auf dem Partnersystem nicht bestätigt wurden.

**Achtung:** Die Datenintegrität wird bei Verwendung dieser Option möglicherweise beeinträchtigt.

**KEEP** Die Wiederherstellungsinformationen werden aufbewahrt und es wird keine Aktion für unbestätigte Arbeitseinheiten ausgeführt.

**Für IRC** setzt die Verbindung die Verarbeitung bei den neuen Arbeitseinheiten fort. Unbestätigte Arbeitseinheiten können über die CEMT- oder SPI-Schnittstelle aufgelöst werden.

**Bei APPC** kann die Verbindung keine neuen Arbeiten ausführen, für die Protokolle der Synchronisationsebene 2 erforderlich sind, und zwar so lange nicht, bis alle ausstehenden wiederherstellbaren Arbeiten mit

dem Partner (d. h. unbestätigte Arbeitseinheiten oder relevante Informationen für die Arbeitseinheiten, die auf dem Partnersystem unter dem alten Protokollnamen nicht bestätigt wurden) über die CEMT- oder SPI-Schnittstelle abgeschlossen wurden.

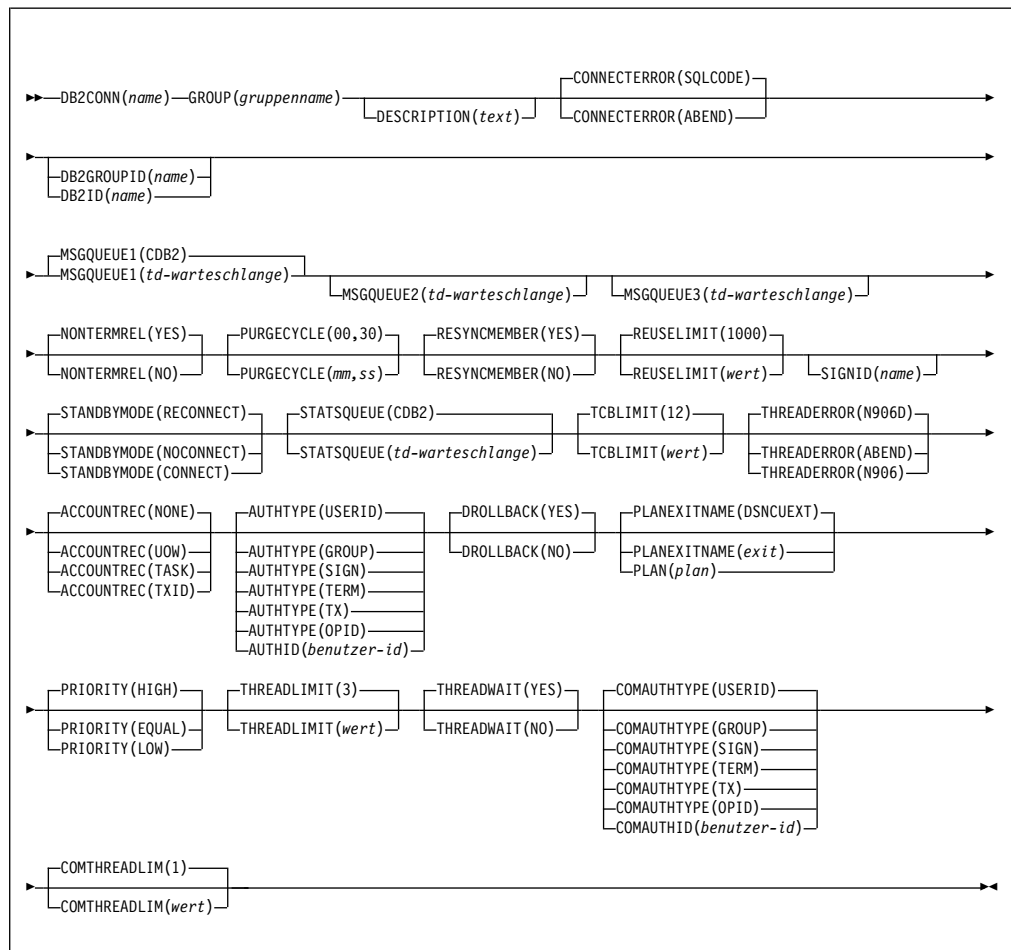
## DB2CONN-Ressourcen

Eine DB2CONN-Definition dient zur Definition der Attribute der Verbindung zwischen CICS und Db2 und der Verbindung zwischen den *Pool-Threads* und den *Befehlsthreads*, die in der Verbindung benutzt werden.

Informationen zur Installation von DB2CONN-Ressourcendefinitionen mit CEMT und CEDA finden Sie in Installing DB2CONN resource definitions. Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with BAS Db2 connection resource definitions.

## DB2CONN-Attribute

Beschreibt die Syntax und die Attribute der DB2CONN-Ressource.



## Allgemeine Attribute

Beschreibt die allgemeinen Attribute einer DB2CONN-Ressource.

**DB2CONN(name)**

Der Name, mit dem eine Db2-Verbindungsdefinition identifiziert werden kann. Der Name kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

**DESCRIPTION(text)**

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

**GROUP(gruppenname)**

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

**Verbindungsattribute**

Beschreibt die Attribute der DB2CONN-Ressource, die sich auf die Verbindung beziehen.

**CONNECTERROR({SQLCODE|ABEND})**

Gibt an, wie die Information, dass CICS über keine Verbindung zu Db2 verfügt, weil die Anschlussfunktion sich im Bereitschaftsmodus befindet, an die Anwendung zurückgemeldet wird, die eine SQL-Anforderung abgesetzt hat.

**ABEND**

Die Anwendung wird mit dem Abbruchcode AEY9 abnormal beendet.

**SQLCODE**

Die Anwendung empfängt den SQL-Code -923. Der SQL-Code (SQL-CODE) kann nicht angegeben werden, wenn für STANDBYMODE die Einstellung NOCONNECT festgelegt wurde.

**DB2GROUPID(name)**

Gibt die (bis zu vier Zeichen umfassende) Gruppen-ID einer Gruppe mit gemeinsamer Datennutzung der Db2-Subsysteme an. Die Gruppenanschlussfunktion stellt eine Verbindung zwischen CICS und einem beliebigen aktiven Mitglied dieser Gruppe mit gemeinsamer Datennutzung her. Gleichen Sie die Gruppen-ID mit dem Gruppenanschlussnamen ab, der in Db2 definiert wurde. Bei DB2 Version 10 kann die Gruppen-ID ein Untergruppenanschlussname sein, der in Db2 definiert wurde und der eine Teilmenge der Gruppe mit gemeinsamer Datennutzung angibt. Wenn für das Attribut DB2GROUPID keine

Angabe gemacht wird, dann wird die Gruppenanschlussfunktion nicht verwendet. Die Optionen DB2GROUPID und DB2ID dürfen nicht gemeinsam angegeben werden. Folgende Prioritätenregelung ist zu befolgen:

1. Durch die Angabe von DB2GROUPID wird der Wert für DB2ID, der bereits in der DB2CONN-Definition festgelegt wurde, mit Leerzeichen überschrieben.
2. Wenn Sie versuchen, die Optionen DB2GROUPID und DB2ID gemeinsam in derselben CEDA-Anzeige anzugeben, dann wird der Wert für DB2ID verwendet.
3. Wenn DB2ID für ein einzelnes Subsystem in einem Befehl CEMT oder **EXEC CICS SET DB2CONN** oder in einem Befehl DSNB STRT angegeben wird, dann überschreibt dieser Wert für DB2ID alle DB2GROUPID-Attribute, die in der installierten DB2CONN-Definition festgelegt wurden. Der Wert für DB2GROUPID wird in der installierten DB2CONN-Definition mit Leerzeichen überschrieben und muss erneut (mithilfe des Befehls CEDA oder **SET DB2CONN**) festgelegt werden, um die Gruppenanschlussfunktion verwenden zu können.

#### **DB2ID**(name)

Gibt den Namen des Db2-Subsystems an, zu dem die CICS-Db2-Anschlussfunktion eine Verbindung herstellen will. Standardmäßig ist dieses Feld leer. Wenn Sie die Gruppenanschlussfunktion benutzen wollen, dann geben Sie in der DB2CONN-Definition anstelle von DB2ID einen Wert für DB2GROUPID an. Der Wert für DB2ID, der in der installierten DB2CONN-Definition festgelegt wurde, kann mit einer Db2-Subsystem-ID überschrieben werden, die in einem Befehl DSNB STRT oder in DB2ID in einem Befehl SET DB2CONN angegeben wurde. Wenn für DB2ID in der installierten DB2CONN-Definition weiterhin kein Wert angegeben wird und auch DB2GROUPID leer bleibt, dann können Sie im Systeminitialisierungsparameter INITPARM eine Db2-Subsystem-ID angeben. Wird mit keiner dieser Methoden eine Db2-Subsystem-ID angegeben, und wird auch kein Wert für DB2GROUPID festgelegt, dann wird die DB2ID-Standardeinstellung (Leerzeichen) durch DSN ersetzt, wenn der Verbindungsaufbau versucht wird. Demzufolge lautet die Hierarchie für die Ermittlung des Db2-Subsystems wie folgt:

1. Verwenden Sie die Subsystem-ID, wenn sie in einem Befehl DSNB STRT angegeben ist.
2. Verwenden Sie den Wert für DB2ID in der installierten DB2CONN-Definition, wenn dieser nicht leer ist.
3. Verwenden Sie den Wert für DB2GROUPID in der installierten DB2CONN-Definition für den Gruppenanschluss, wenn dieser nicht leer ist.
4. Verwenden Sie die Subsystem-ID, wenn diese in INITPARM angegeben wurde, falls die Werte für DB2ID und DB2GROUPID in der zuletzt installierten DB2CONN-Definition leer sind (oder zu einem späteren Zeitpunkt auf Leerzeichen gesetzt wurden). Bei jedem Systemstart wird immer INITPARM verwendet, wenn die zuletzt installierte DB2CONN-Definition keine Angabe für DB2ID und DB2GROUPID enthalten hat. Dies gilt auch dann, wenn die Werte für DB2ID oder DB2GROUPID später mit dem Befehl SET geändert wurden.
5. Verwenden Sie eine Standard-Subsystem-ID für DSN.

DB2GROUPID und DB2ID dürfen nicht zusammen angegeben werden. Wenn Sie sie dennoch in derselben CEDA-Anzeige gemeinsam angeben, dann wird der Wert für DB2ID verwendet. Wenn ein Wert für DB2GROUPID in einem Befehl CEMT oder **EXEC CICS SET DB2CONN** angegeben wird, dann überschreibt

dieser Wert alle Angaben für DB2ID, die in der installierten DB2CONN-Definition festgelegt wurden, und der Wert für DB2ID wird mit Leerzeichen überschrieben.

**MSGQUEUE1**(**{CDB2|td-warteschlange}**)

Gibt das erste Ziel für transiente Daten an, an das nicht erwartete Nachrichten der CICS-Db2-Anschlussfunktion gesendet werden. Dieses erste Ziel darf nicht leer sein.

**MSGQUEUE2**(**td-warteschlange**)

Gibt das zweite Ziel für transiente Daten an, an das nicht erwartete Nachrichten der CICS-Db2-Anschlussfunktion gesendet werden.

**MSGQUEUE3**(**td-warteschlange**)

Gibt das dritte Ziel für transiente Daten an, an das nicht erwartete Nachrichten der CICS-Db2-Anschlussfunktion gesendet werden.

**NONTERMREL**(**{YES|NO}**)

Gibt an, ob eine Nicht-Terminal-Transaktion Threads für die Wiederverwendung bei temporären Synchronisationspunkten freigibt.

**NO** Nicht-Terminal-Transaktionen geben keine Threads für die Wiederverwendung an temporären Synchronisationspunkten frei.

**YES** Nicht-Terminal-Transaktionen geben Threads für die Wiederverwendung an temporären Synchronisationspunkten frei.

**PURGE CYCLE**(**{00|mm}, {30|ss}**)

Gibt die Dauer (in Minuten und Sekunden) des Bereinigungszyklus für geschützte Threads an. Die Dauer des Bereinigungszyklus liegt im Bereich zwischen 5 Sekunden und 59 Minuten 59 Sekunden. Wenn Sie keinen Wert für PURGE CYCLE angeben, dann werden vom System standardmäßig 30 Sekunden festgelegt; PURGE CYCLE= 00, 30.

Ein geschützter Thread wird nicht sofort beendet, wenn er freigegeben wird. Er wird erst nach zwei abgeschlossenen Bereinigungszyklen beendet, wenn er in der Zwischenzeit nicht wiederverwendet wurde. Aus diesem Grund wird bei einem Bereinigungszyklus von 30 Sekunden ein geschützter Thread 30 - 60 Sekunden nach seiner Freigabe bereinigt. Der erste Bereinigungszyklus nach dem Start der Anschlussfunktion umfasst immer fünf Minuten. Anschließend werden die Bereinigungszykluswerte angewendet. Ein nicht geschützter Thread wird beendet, wenn er (am Synchronisationspunkt oder am Ende der Task) freigegeben wird, wenn keine anderen Transaktionen auf diesem DB2ENTRY auf einen Thread warten. Nur Threads, die zu einem DB2ENTRY gehören, können geschützt werden. Pool-Threads und Befehlsthreads können nicht geschützt werden.

**RESYNCMEMBER**(**{YES|NO}**)

Wenn Sie die Gruppenanschlussfunktion verwenden, dann benutzen Sie das Attribut RESYNCMEMBER, um die Strategie auszuwählen, die von CICS angewendet werden soll, wenn ausstehende Arbeitseinheiten für das letzte Mitglied einer Db2-Gruppe mit gemeinsamer Datennutzung gehalten werden, mit dem CICS verbunden war.

**YES** Werden ausstehende Arbeitseinheiten gehalten, dann müssen Sie eine Resynchronisation mit dem letzten Mitglied einer Db2-Gruppe mit gemeinsamer Datennutzung durchführen, mit dem CICS verbunden war. CICS ignoriert die Gruppenanschlussfunktion und die CICS-Db2-Anschlussfunktion wartet, bis eine erneute Verbindung zu diesem letzten Mitglied der Db2-Gruppe mit gemeinsamer Datennutzung hergestellt werden kann, um unbestätigte Arbeitseinheiten aufzulösen. Arbeitsein-

heiten, die als unbestätigt verzögert eingestuft werden, werden nicht in diesen Prozess eingebunden, weil CICS selbst nicht in der Lage ist, diese Arbeitseinheiten zum aktuellen Zeitpunkt aufzulösen. Die Resynchronisation für diese Arbeitseinheiten wird durchgeführt, nachdem CICS die Resynchronisation mit dem fernen Koordinator abgeschlossen hat.

**NO** Gibt an, dass keine Resynchronisation erforderlich ist. CICS führt einen Versuch zur erneuten Herstellung einer Verbindung zum letzten verbundenen Mitglied der Db2-Gruppe mit gemeinsamer Datennutzung durch. Verläuft dieser Versuch erfolgreich, dann können die unbestätigten Arbeitseinheiten (mit Ausnahme der Arbeitseinheiten, die als unbestätigt verzögert eingestuft werden) aufgelöst werden. Schlägt der Versuch fehl, dann verwendet CICS die Gruppenanschlussfunktion, um eine Verbindung zu einem beliebigen aktiven Mitglied der Db2-Gruppe mit gemeinsamer Datennutzung herzustellen. Des Weiteren gibt das System eine Warnung (DFHDB2064) aus, in der Sie darüber informiert werden, dass nicht aufgelöste unbestätigte Arbeitseinheiten beim letzten Mitglied der Gruppe vorhanden sein können, mit dem CICS verbunden war.

#### **REUSELIMIT**(*wert*)

Gibt einen Wert im Bereich zwischen 0 und 10000 an, der für die maximal zulässige Anzahl der Wiederverwendungen eines Threads steht, bevor dieser beendet wird. Der Standardwert ist 1000. Der Wert 0 bedeutet, dass kein Grenzwert für die Anzahl der Wiederverwendungen des Threads definiert wurde. Dies war in CICS-Versionen vor CICS TS 4.2 der Fall. Allerdings kann es durch CICS-Db2-Threads mit langer Laufzeit, die ständig wiederverwendet werden, zu einer Zunahme der Ressourcen in Db2 kommen, die zu Speicherproblemen und abnormalen Beendigungen sowie zu Ausfällen der Db2-Subsysteme führen können.

Der Grenzwert für die Wiederverwendung gilt für ungeschützte Threads im Pool und in DB2ENTRY sowie für geschützte DB2ENTRY-Threads. Ein ungeschützter Thread wird wiederverwendet, wenn zu dem Zeitpunkt, zu dem er von einer Transaktion freigegeben wird, eine neue Transaktion wartet. Ein geschützter Thread wird wiederverwendet, wenn eine neue Transaktion einen Thread benötigt, während der Thread gegen eine Beendigung geschützt ist. In beiden Fällen kann der Thread, nachdem der Grenzwert für die Wiederverwendung erreicht wurde, nicht mehr von Transaktionen verwendet werden. Wenn die Transaktion, die den Thread momentan verwendet, den Thread freigibt, dann beendet CICS den Thread und erstellt ihn neu, um so Db2-Ressourcen freizugeben, bevor festgestellt wird, ob neue Arbeiten für den Thread anstehen oder ob der Thread geschützt werden soll.

Bei Verwendung des Standardwerts 1000 sind der Threadspeicher und der EDM-Poolspeicher unterhalb der 2-GB-Grenze gegen eine Überbelegung geschützt, wenn Sie die Db2-Bindungsoption RELEASE(DEALLOCATE) verwenden, ohne dass es hierdurch zu einer Beeinträchtigung der Leistung kommt. Wenn allerdings die Db2-Überwachungs- und -Statistikfunktion eine übermäßige Belegung des Db2-Threadspeichers und/oder des EDM-Poolspeichers erkennt, dann kann dieser Grenzwert reduziert werden. Wenn die CICS-Db2-Statistikfunktion zeigt, dass die Pool- oder Einstiegsthreads häufig den Grenzwert für die Wiederverwendung erreichen und wenn genügend virtueller Speicher und Realspeicher verfügbar ist, um den Db2-Threadspeicher zu erhöhen, dann kann dieser Grenzwert hingegen erhöht werden.

Das Festlegen eines niedrigen Werts für das Wiederverwendungslimit hat Einfluss auf die Leistung und führt zu einer Zunahme der Prozessoraktivität und zu einem geringeren Durchsatz. Es gibt jedoch Situationen, in denen Sie möglicherweise einen niedrigen Wert festlegen möchten. Dies kann beispielsweise dann sinnvoll sein, wenn Sie die Möglichkeit einer Änderung der Db2-Bindungsoptionen von RELEASE(COMMIT) in RELEASE(DEALLOCATE) für einen Plan oder ein Paket erwägen. In diesem Fall kann vorübergehend ein niedriger Wert verwendet werden, um das Szenario zu testen.

#### **SIGNID**(*name*)

Gibt die Berechtigungs-ID an, die von der CICS-Db2-Anschlussfunktion verwendet wird, wenn die Anmeldung bei Db2 für Pool- oder DB2ENTRY-Threads mit der Angabe AUTHTYPE(SIGN) erfolgt. Der Standardwert sind Leerzeichen, die bei der Installation der DB2CONN-Definition durch die Anwendungs-ID des CICS-Systems ersetzt werden. Die von Ihnen angegebene ID kann bis zu acht Zeichen lang sein.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Wenn Sie eine Benutzer-ID im Attribut SIGNID angeben, dann führt CICS eine Ersatzbenutzerprüfung für die Benutzer-ID aus, mit der die Installation durchgeführt wird. In ähnlicher Weise wird für die Benutzer-ID der CICS-Region eine Ersatzbenutzerprüfung während der Gruppenlisteninstallation bei einem Kalt- oder Erststart von CICS durchgeführt.

Wenn die ID, die Sie angeben, mit der Benutzer-ID der CICS-Region übereinstimmt und wenn Sie für jeden Befehls-, Pool- oder Einstiegsthread die Angabe AUTHTYPE(SIGN) festlegen, dann wird das RACF-Umgebungsselement für die Zugriffssteuerung (ACEE) für die Benutzer-ID der CICS-Region an Db2 übergeben.

#### **STANDBYMODE**( { **RECONNECT** | **CONNECT** | **NOCONNECT** } )

Gibt die Aktion an, die von der CICS-Db2-Anschlussfunktion ausgeführt werden soll, wenn Db2 nicht aktiv ist, während versucht wird, eine Verbindung zwischen CICS und Db2 herzustellen.

##### **CONNECT**

Gibt an, dass die CICS-Db2-Anschlussfunktion im Bereitschaftsmodus darauf warten soll, dass Db2 aktiviert wird. Wenn die Verbindung hergestellt ist und in Db2 zu einem späteren Zeitpunkt ein Fehler auftritt, dann wird die Verarbeitung der CICS-Db2-Anschlussfunktion beendet.

##### **NOCONNECT**

Gibt an, dass die CICS-Db2-Anschlussfunktion beendet werden soll.

##### **RECONNECT**

Gibt an, dass die CICS-Db2-Anschlussfunktion in den Bereitschaftsmodus versetzt werden und auf Db2 warten soll. Wenn in Db2 später nach der Herstellung der Verbindung ein Fehler auftritt, dann wird die CICS-Db2-Anschlussfunktion wieder in den Bereitschaftsmodus versetzt und CICS stellt erneut eine Verbindung zu Db2 her, nachdem Db2 wiederhergestellt wurde.

### **STATSQUEUE({CDB2|*td-warteschlange*})**

Gibt das Ziel für transiente Daten für die Statistiken der CICS-Db2-Anschlussfunktion an, die beim Systemabschluss der CICS-Db2-Anschlussfunktion erstellt werden.

### **TCBLIMIT({12|*wert*})**

Gibt die maximal zulässige Anzahl von Tasksteuerblöcken (TCBs = Task Control Blocks) an, die zur Verarbeitung von Db2-Anforderungen verwendet werden können. Der Standardwert ist 12. Die minimale Anzahl ist 4 und die maximale Anzahl 2000. CICS verwendet offene TCBs im L8- und L9-Modus, um Db2-Anforderungen zu verarbeiten. Das Attribut TCBLIMIT der DB2CONN-Definition reguliert die Anzahl der offenen TCBs, die für den Zugriff auf Db2 verwendet werden können. Dabei handelt es sich um die Anzahl der TCBs, die sich bei Db2 identifizieren und eine Verbindung in Db2 erstellen können.

Weitere Informationen zu den TCB-Grenzwerten und den Modi für offene TCBs finden Sie in Open TCB management.

Der Wert für TCBLIMIT steuert die Gesamtzahl der Threads für die CICS-Region. Aus diesem Grund entspricht der empfohlene Wert für TCBLIMIT der Summe aller Threadbegrenzungswerte (d. h., der Summe aller THREADLIMIT-Attribute der Db2-Verbindung und Db2-Eintragsressourcendefinitionen plus dem Wert für COMTHREADLIMIT in der Db2-Verbindungsdefinition) bis zum Grenzwert 2000.

Wenn der automatisch von CICS gesetzte Grenzwert für die Anzahl der offenen TCBs im L8- und L9-Modus erreicht ist, sodass keine weiteren offenen TCBs mehr erstellt werden können, wird die Ausführung der Task mit HTYPE(DISPATCH) und HVALUE(OPEN\_TCB) ausgesetzt. CICS legt diesen Grenzwert auf Basis der Formel  $(2 * \text{MXT-Wert}) + 32$  mit dem MXT- oder MAX-TASKS-Grenzwert für die CICS-Region fest. Wird dieser Grenzwert nicht überschritten, sondern der Wert für TCBLIMIT, dann wird die Task mit HTYPE(CDB2CONN) ausgesetzt. In dieser Situation wird, obwohl CICS über einen offenen TCB verfügt, die maximal zulässige Anzahl von offenen TCBs für den Zugriff auf Db2 (gemäß Definition in TCBLIMIT) verwendet.

Wenn der numerische Wert für TCBLIMIT ermittelt wird, dann müssen Sie den Betrag berücksichtigen, der für den Parameter MAX USERS in der Db2-Installationsanzeige DSNTIPE angegeben wurde.

### **THREADERROR({N906D|N906|ABEND})**

Gibt die Verarbeitung an, die nach einem Fehler beim Erstellen eines Threads ausgeführt werden soll.

#### **ABEND**

Wenn der erste SQL-Fehler erkannt wird, dann erstellt CICS abhängig vom Typ des Fehlers einen Transaktionsspeicherauszug für den Abbruchcode AD2S, AD2T oder AD2U. Für den ersten Fehler wird die Transaktion nicht abgebrochen. Für einen zweiten oder nachfolgenden SQL-Fehler wird die Ausführung der Transaktion mit dem Abbruchcode AD2S, AD2T oder AD2U abgebrochen. Die Transaktion muss beendet und erneut initialisiert werden, bevor eine neue SQL-Anforderung abgesetzt werden kann.

#### **N906D**

Ein Transaktionsspeicherauszug muss erstellt werden und die DSNCSQL-Ressourcenmanagerschnittstelle (RMI = Resource Manager Interface), die der Transaktion zugeordnet ist, darf *nicht* inaktiviert werden. Die Transaktion empfängt den SQL-Code -906, wenn eine wei-



tere SQL-Anforderung abgesetzt wird, es sei denn, die Transaktion setzt die Anweisung SYNCPOINT ROLLBACK ab. Wird SYNCPOINT ohne die Option ROLLBACK angegeben, dann führt dies zu einem Abbruch mit dem Abbruchcode ASP3 oder ASP7. Der Transaktionsspeicherauszug zeichnet einen Abbruch mit dem Abbruchcode AD2S, AD2T oder AD2U auf.

- N906** Die DSNCSQL-RMI, die der Transaktion zugeordnet ist, wird *nicht* inaktiviert. Die Transaktion empfängt den SQL-Code -906, wenn eine weitere SQL-Anforderung abgesetzt wird, es sei denn, die Transaktion setzt die Anweisung SYNCPOINT ROLLBACK ab. Wird SYNCPOINT ohne die Option ROLLBACK angegeben, dann führt dies zu einem Abbruch mit dem Abbruchcode ASP3 oder ASP7.

## Pool-Thread-Attribute

Beschreibt die Attribute der DB2CONN-Ressource, die sich auf den Pool-Thread beziehen.

### ACCOUNTREC({NONE|TASK|TXID|UOW})

Gibt das Mindestvolumen für die Db2-Abrechnung an, die für Transaktionen benötigt wird, die Pool-Threads verwenden. Das angegebene Minimum kann wie im Folgenden beschrieben überschritten werden.

#### NONE

Für Transaktionen, die mit Pool-Threads arbeiten, sind keine Abrechnungsdatensätze erforderlich.

Db2 generiert mindestens einen Abrechnungsdatensatz für jeden Thread, nachdem der Thread beendet wurde. Bei Änderungen der Berechtigung werden zusätzliche Abrechnungsdatensätze generiert.

- TASK** Die CICS-Db2-Anschlussfunktion generiert mindestens einen Abrechnungsdatensatz für jede CICS-Task.

Eine Transaktion, die mehrere Arbeitseinheiten (UOWs = Units of Work) enthält und in der die Threads an Synchronisationspunkten freigegeben werden, kann für jede der UOWs einen anderen Thread verwenden. Dies hat zur Folge, dass für jede UOW ein Abrechnungsdatensatz erstellt werden kann.

- TXID** Die CICS-Db2-Anschlussfunktion führt zur Generierung eines Abrechnungsdatensatzes, sobald die Transaktions-ID, die den Thread verwendet, geändert wird.

Da Pool-Threads normalerweise von einer Reihe unterschiedlicher Transaktions-IDs verwendet werden, besteht eine erhöhte Wahrscheinlichkeit, dass eine Transaktion, die mehrere UOWs enthält und in der die Threads an Synchronisationspunkten freigegeben werden, für jede UOW einen unterschiedlichen Thread verwendet. In diesem Fall wird möglicherweise für jede UOW ein Abrechnungsdatensatz erstellt.

- UOW** Die CICS-Db2-Anschlussfunktion führt zur Generierung eines Abrechnungsdatensatzes für jede UOW. Dabei wird davon ausgegangen, dass der Thread am Ende der UOW freigegeben wird.

### AUTHID(benutzer-id)

Gibt die Benutzer-ID an, die bei Verwendung von Pool-Threads für die Sicherheitsprüfung benutzt wird.

Verwenden Sie AUTHID nicht, wenn Sie RACF für bestimmte oder alle Aspekte der Sicherheitsprüfung in Ihrem Db2-Adressraum verwenden. Benutzen Sie stattdessen AUTHTYPE mit der Option GROUP, SIGN oder USERID. Sie müssen AUTHTYPE verwenden, weil Threads, die AUTHID verwenden, das erforderliche RACF-Umgebungselement für die Zugriffssteuerung (ACEE = Access

Control Environment Element) nicht an Db2 übergeben. Die von Ihnen angegebene ID kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

**AUTHTYPE({USERID|OPID|GROUP|SIGN|TERM|TX})**

Gibt den Typ der ID an, der für Threads in dieser DB2CONN-Definition verwendet werden kann.

Wenn Sie RACF für bestimmte oder alle Aspekte der Sicherheitsprüfung in Ihrem Db2-Adressraum verwenden, dann sollten Sie die Option GROUP, SIGN oder USERID benutzen. Sie müssen eine dieser Optionen verwenden, weil nur Threads, die mit diesen Optionen definiert wurden, das erforderliche RACF-Umgebungselement für die Zugriffssteuerung (ACEE) an Db2 übergeben.

Wenn Sie allerdings die Option SIGN angeben, dann wird das ACEE nur dann an Db2 übergeben, wenn der für das Attribut SIGNID in der DB2CONN-Definition angegebene Wert mit der Benutzer-ID der CICS-Region übereinstimmt.

Das ACEE ist nicht erforderlich, wenn Sie nur die interne Sicherheitsfunktion von Db2 verwenden. In diesem Fall können Sie jede der verfügbaren Optionen verwenden.

**USERID**

Die Benutzer-ID, die der CICS-Transaktion zugeordnet ist, wird als Berechtigungs-ID verwendet. Wenn die Benutzer-ID weniger als acht Zeichen lang ist, wird sie rechts mit Leerzeichen aufgefüllt.

**Important:** Geben Sie nicht AUTHTYPE(USERID) an, wenn Sie den Db2-Beispiexit für die Anmeldung mit dem Namen DSN@SGN verwenden, da dies zum Fehler 'SQL -922' führen kann. Geben Sie stattdessen COMMAUTHTYPE(GROUP) an.

**OPID** Die Bedieneridentifikation, die der Benutzer-ID zugeordnet ist, die der CICS-Transaktion zugewiesen wurde, wird als Berechtigungs-ID verwendet. Die aus drei Zeichen bestehende Bedieneridentifikation wird auf der rechten Seite mit Leerzeichen aufgefüllt, um so die aus acht Zeichen bestehende Berechtigungs-ID zu bilden.

**GROUP**

Gibt die Benutzer-ID sowie den Namen der verbundenen Gruppe als Berechtigungs-ID an. In der folgenden Tabelle wird dargestellt, wie diese beiden Werte von Db2 interpretiert werden.

An Db2 übergebene IDs	Interpretation der Werte durch Db2
CICS-Benutzer-ID für die Anmeldung (USERID)	Stellt die primäre Db2-Berechtigungs-ID dar.

An Db2 übergebene IDs	Interpretation der Werte durch Db2
Name der verbundenen RACF-Gruppe	Wenn die RACF-Liste der Gruppenoptionen nicht aktiv ist, dann verwendet Db2 den Namen der verbundenen Gruppe, der von der CICS-Anschlussfunktion bereitgestellt wird, als sekundäre Db2-Berechtigungs-ID. Wenn die RACF-Liste der Gruppenoptionen aktiv ist, dann ignoriert Db2 den Namen der verbundenen Gruppe, der von der CICS-Anschlussfunktion bereitgestellt wird, der Wert wird jedoch in der Db2-Liste der sekundären Db2-Berechtigungs-IDs aufgeführt.

Um die Option GROUP benutzen zu können, müssen Sie in den Systeminitialisierungsparametern der Region die Einstellung SEC=YES angeben.

Wenn keine RACF-Gruppen-ID für diese Benutzer-ID (USERID) verfügbar ist, dann wird ein Feld mit acht Leerzeichen als Gruppen-ID an Db2 übergeben.

**SIGN** Gibt an, dass das Attribut SIGNID der Db2-Verbindungsdefinition als Ressourcenberechtigungs-ID verwendet wird.

**TERM** Gibt die Terminalidentifikation als Berechtigungs-ID an. Die aus vier Zeichen bestehende Terminalidentifikation wird auf der rechten Seite mit Leerzeichen aufgefüllt, um so die aus acht Zeichen bestehende Berechtigungs-ID zu bilden.

Wenn die Transaktion keinem Terminal zugeordnet wurde (z. B. bei Initialisierung mit dem Befehl START), dann geben Sie AUTHTYPE-(TERM) nicht an.

**TX** Gibt die Transaktionskennung als Berechtigungs-ID an. Die aus vier Zeichen bestehende Transaktionskennung wird auf der rechten Seite mit Leerzeichen aufgefüllt, um so die aus acht Zeichen bestehende Berechtigungs-ID zu bilden.

#### **DROLLBACK({YES|NO})**

Gibt an, ob die CICS-Db2-Anschlussfunktion die Anweisung SYNCPOINT ROLLBACK einleitet, wenn eine Transaktion, die in eine Deadlock-Auflösung eingebunden ist, ausgewählt wird.

**YES** Die Anschlussfunktion leitet eine Anweisung SYNCPOINT ROLLBACK ein, bevor die Steuerung wieder an die Anwendung zurückgegeben wird. An das Programm wird der SQL-Rückgabecode -911 zurückgegeben.

Geben Sie nicht YES an, wenn der Pool von Transaktionen verwendet wird, für die im Rahmen einer OTS-Transaktion Enterprise-Beans ausgeführt werden. Die Verwendung von SYNCPOINT ROLLBACK ist in OTS-Transaktionen nicht zulässig. Sie sollten das Definieren einer DB2ENTRY-Komponente in Betracht ziehen, für die DROLLBACK(NO) für die Verwendung durch Transaktionen angegeben ist, die Enterprise-Beans als Teil einer OTS-Transaktion ausführen.

**NO** Die Anschlussfunktion leitet keinen Rollback für eine Transaktion ein. An die Anwendung wird der SQL-Rückgabecode -913 zurückgegeben.

**PLAN(plan)**

Gibt den Namen des Plans an, der für alle Pool-Threads verwendet werden soll.

**PLANEXITNAME({DSNCUEXT|exit})**

Gibt den Namen des dynamischen Planexits an, der für Pool-Threads verwendet werden soll. Wenn Sie die Werte für PLAN und PLANEXITNAME ändern, während aktive Transaktionen für den Pool vorhanden sind, dann werden die neuen Werte angewendet, wenn die Transaktion den Thread zum nächsten Mal freigibt.

**PRIORITY({HIGH|EQUAL|LOW})**

Gibt die Priorität der TCBs (Task Control Blocks; Tasksteuerblöcke) des Pool-Threads im Verhältnis zum CICS-Haupttasksteuerblock (QR-TCB) an. Bei den TCBs des Threads handelt es sich um offene L8-TCBs.

**HIGH** Die TCBs des Threads weisen eine höhere Priorität als der CICS-QR-TCB auf.

**EQUAL**

Die TCBs des Threads weisen eine gleiche Priorität wie der CICS-QR-TCB auf.

**LOW** Die TCBs des Threads weisen eine niedrigere Priorität als der CICS-QR-TCB auf.

**THREADLIMIT({3|wert})**

Gibt die aktuelle Maximalanzahl von Pool-Threads an, die in der CICS-Db2-Anschlussfunktion aktiv sein dürfen, bevor Anforderungen gemäß den Angaben im Attribut THREADWAIT in den Wartestatus versetzt werden oder zurückgewiesen werden. Der Standardwert von THREADLIMIT (3) stellt gleichzeitig den Mindestwert dar, den Sie angeben können. Der Maximalwert darf nicht größer als der Wert sein, der für TCBLIMIT angegeben wurde.

**THREADWAIT({YES|NO})**

Gibt an, ob Transaktionen auf einen Pool-Thread warten oder abnormal beendet werden sollen, wenn die Anzahl der aktiven Pool-Threads die Threadbegrenzung erreicht.

Von der CICS-Db2-Anschlussfunktion wird der eindeutige Abbruchcode AD3T (Nachricht DFHDB2011) ausgegeben, wenn Sie THREADWAIT=NO angeben und die Anzahl der Pool-Threads überschritten wird.

**YES** Wenn alle Threads belegt sind, dann muss eine Transaktion warten, bis ein Thread verfügbar wird. Eine Transaktion kann so lange warten, wie in CICS als zulässig definiert ist. Normalerweise wird gewartet, bis ein Thread verfügbar wird.

**NO** Wenn alle Threads belegt sind, dann wird die Transaktion mit dem Code AD2T oder AD3T abnormal beendet.

**Befehlsthreadattribute**

Beschreibt die Attribute der DB2CONN-Ressource, die sich auf den Befehlsthread beziehen. Ein Befehlsthread wird verwendet, wenn die DSNC-Transaktion zum Senden eines Befehls an Db2 verwendet wird und wenn CICS einen Befehl zum Abbrechen eines Threads absetzt, der in Db2 im Rahmen der Verarbeitung einer CICS-Anforderung zum Bereinigen einer Task oder zum Erzwingen einer Bereinigung aktiv ist.

Im Folgenden sind die Beschreibungen der Befehlsthreadattribute für die Db2-Verbindungsdefinition aufgeführt:

### COMAUTHID(*benutzer-id*)

Gibt an, welche ID von der CICS-Db2-Anschlussfunktion bei Verwendung von Befehlsthreads für die Sicherheitsprüfung benutzt werden soll.

Wenn Sie RACF für bestimmte oder alle Aspekte der Sicherheitsprüfung in Ihrem Db2-Adressraum verwenden, dann sollten Sie anstelle von COMAUTHID das Attribut COMAUTHTYPE mit der Option USERID oder GROUP benutzen. Sie können außerdem COMMAUTHTYPE mit der Option SIGN verwenden, wenn das Attribut SIGNID in der DB2CONN-Definition mit der Benutzer-ID der CICS-Region übereinstimmt. Dies ist darauf zurückzuführen, dass Threads, die COMAUTHID verwenden, das erforderliche RACF-Umgebungselement für die Zugriffssteuerung (ACEE = Access Control Environment Element) nicht an Db2 übergeben. Das ACEE ist nicht erforderlich, wenn Sie nur die interne Sicherheitsfunktion von Db2 verwenden. In diesem Fall können Sie COMAUTHID verwenden. Die von Ihnen angegebene ID kann bis zu acht Zeichen lang sein.

#### Zulässige Zeichen:

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

### COMAUTHTYPE({USERID|OPID|GROUP|SIGN|TERM|TX})

Gibt den Typ der ID an, die bei Verwendung von Befehlsthreads für die Sicherheitsprüfung benutzt werden kann.

Wenn Sie RACF für bestimmte oder alle Aspekte der Sicherheitsprüfung in Ihrem Db2-Adressraum verwenden, dann sollten Sie die Option GROUP, SIGN oder USERID benutzen. Dies ist darauf zurückzuführen, dass nur Threads, die mit diesen Optionen definiert wurden, das erforderliche RACF-Umgebungselement für die Zugriffssteuerung (ACEE) an Db2 übergeben. Wenn Sie allerdings die Option SIGN angeben, dann wird das ACEE nur dann an Db2 übergeben, wenn der für das Attribut SIGNID in der DB2CONN-Definition angegebene Wert mit der Benutzer-ID der CICS-Region übereinstimmt.

#### USERID

Die aus einem bis acht Zeichen bestehende Benutzer-ID, die der CICS-Transaktion zugeordnet ist, wird als Berechtigungs-ID verwendet. Der Name kann bis zu acht Zeichen lang sein.

#### Zulässige Zeichen:

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

**Important:** Geben Sie nicht COMMAUTHTYPE(USERID) an, wenn Sie den Db2-Beispiexit für die Anmeldung mit dem Namen DSN@SGN verwenden, da dies zum Fehler 'SQL -922' führen kann. Geben Sie stattdessen COMMAUTHTYPE(GROUP) an.

**OPID** Die Operatoridentifikation, die der Benutzer-ID zugeordnet ist, die der CICS-Funktion für die Transaktionsanmeldung zugeordnet ist, wird als (mit drei Zeichen auf eine Länge von acht Zeichen aufgefüllte) Berechtigungs-ID verwendet.

## GROUP

Gibt die aus einem bis acht Zeichen bestehende Benutzer-ID (USERID) sowie den Namen der verbundenen Gruppe als Berechtigungs-ID an. In der folgenden Tabelle wird dargestellt, wie diese beiden Werte von Db2 interpretiert werden.

An Db2 übergebene IDs	Interpretation der Werte durch Db2
CICS-Benutzer-ID für die Anmeldung (USERID)	Stellt die primäre Db2-Berechtigungs-ID dar.
Name der verbundenen RACF-Gruppe	Wenn die RACF-Liste der Gruppenoptionen nicht aktiv ist, dann verwendet Db2 den Namen der verbundenen Gruppe, der von der CICS-Anschlussfunktion bereitgestellt wird, als sekundäre Db2-Berechtigungs-ID. Wenn die RACF-Liste der Gruppenoptionen aktiv ist, dann ignoriert Db2 den Namen der verbundenen Gruppe, der von der CICS-Anschlussfunktion bereitgestellt wird, der Wert wird jedoch in der Db2-Liste der sekundären Db2-Berechtigungs-IDs aufgeführt.

Zur Verwendung der Option CGROUP muss für das CICS-System in der CICS-Systeminitialisierungstabelle (SIT) die Einstellung SEC=YES angegeben sein.

Wenn keine RACF-Gruppen-ID für diese Benutzer-ID (USERID) verfügbar ist, dann wird ein Feld mit acht Leerzeichen als Gruppen-ID an Db2 übergeben.

**SIGN** Gibt an, dass das Attribut SIGNID von DB2CONN als Ressourcenberechtigungs-ID verwendet wird.

**TERM** Gibt die (aus vier Zeichen bestehende und auf acht Zeichen aufgefüllte) Terminalidentifikation als Berechtigungs-ID an. Eine Berechtigungs-ID kann auf diese Weise nicht abgerufen werden, wenn ein Terminal nicht mit der Transaktion verbunden ist.

Wenn eine Transaktion (mit einem CICS-Befehl) gestartet wird und wenn dieser Transaktion kein Terminal zugeordnet wurde, dann darf COMAUTHTYPE(TERM) nicht verwendet werden.

**TX** Gibt die (aus vier Zeichen bestehende und auf acht Zeichen aufgefüllte) Transaktionskennung als Berechtigungs-ID an.

### COMTHREADLIM({1|wert})

Die Zahl gibt die momentan maximal zulässige Anzahl von Befehlsthreads an, die in der CICS-Db2-Anschlussfunktion aktiv sein dürfen, bevor es zu einem Überlauf der Anforderungen in den Pool kommt.

---

## DB2ENTRY-Ressourcen

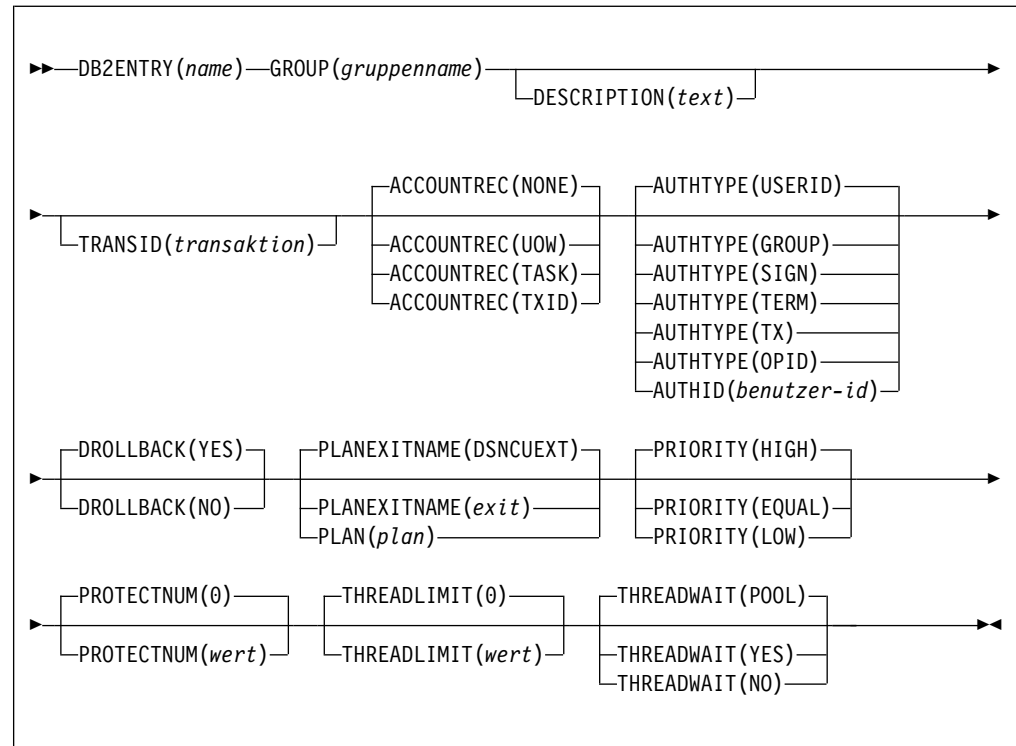
Eine DB2ENTRY-Ressource definiert die Attribute von *Einstiegsthreads*, die von der CICS-Db2-Anschlussfunktion verwendet werden.

Eine Transaktion oder Transaktionsgruppe kann der DB2ENTRY-Ressource zugeordnet werden. Eine Transaktionsgruppe kann mithilfe einzelner oder mehrerer Platzhalterzeichen (siehe Wildcard characters for transaction IDs) dargestellt werden. Darüber hinaus können einer DB2ENTRY-Ressource weitere Transaktionen zugeordnet werden, indem ein DB2TRAN-Element definiert wird.

Informationen zum Installieren von Db2-Eintragsressourcendefinitionen finden Sie in *Installing Db2 entry resource definitions*. Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in *Working with BAS Db2 entry resource definitions*.

## DB2ENTRY-Attribute

Beschreibt die Syntax und die Attribute der DB2ENTRY-Ressource.



### Allgemeine Attribute

Beschreibt die allgemeinen Attribute einer DB2ENTRY-Ressource.

#### DB2ENTRY(name)

Der aus einem bis acht Zeichen bestehende Name, der zur Identifikation einer Db2-Eintragsdefinition dient.

##### Zulässige Zeichen:

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

#### DESCRIPTION(text)

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

### **GROUP**(*gruppenname*)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

#### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

### **Attribute für Threadauswahl**

Beschreibt die Attribute der DB2ENTRY-Ressource, die sich auf die Threadauswahl beziehen.

#### **TRANSID**(*transaktion*)

Gibt die Transaktions-ID an, die dem Eintrag zugeordnet ist. Hier kann nur eine Transaktion angegeben werden. Allerdings kann durch Verwendung einzelner oder mehrerer Platzhalterzeichen in TRANSID (siehe „Platzhalterzeichen für Transaktions-IDs“ auf Seite 43) eine Transaktionsgruppe dargestellt werden. Für diese DB2ENTRY-Ressource können zusätzliche Transaktionen definiert werden. Hierzu muss ein DB2TRAN-Wert definiert werden, der auf diese DB2ENTRY-Ressource verweist. Die Angabe der Transaktions-ID (Transid) ist für die DB2ENTRY-Ressource optional. Alle Transaktionen können über eine DB2TRAN-Definition einer DB2ENTRY-Ressource zugeordnet werden. Wenn einer DB2ENTRY-Ressource nur eine Transaktion zugeordnet ist, ist es allerdings einfacher, die Angabe in DB2ENTRY durchzuführen.

**Anmerkung:** Wird hier eine Transaktions-ID angegeben, dann wird ein implizites DB2TRAN-Objekt erstellt, wenn die DB2ENTRY-Definition installiert wird. Das implizite DBTRAN-Objekt erhält einen Namen mit DFH, der mit der angegebenen Transaktions-ID verknüpft wird. Diese impliziten DB2TRANs erscheinen in CICS- und CICSplex SM-Ansichten von DB2TRAN-Ressourcen. Wenn eine DB2ENTRY-Ressource geändert und dabei eine andere Transaktions-ID angegeben wird, dann wird bei der DB2ENTRY-Neuinstallation das vorherige implizite DB2TRAN-Objekt gelöscht und ein neues implizites DB2TRAN-Objekt erstellt, das die neue Transaktions-ID darstellt, die der DB2ENTRY-Ressource zugeordnet ist.

### **Attribute für Threadoperationen**

Beschreibt die Attribute der DB2ENTRY-Ressource, die sich auf Threadoperationen beziehen.

#### **ACCOUNTREC**(**{NONE|TASK|TXID|UOW}**)

Gibt das Mindestvolumen für die Db2-Abrechnung an, die für Transaktionen benötigt wird, die diesen Db2-Eintrag verwenden. Das angegebene Minimum kann wie im Folgenden beschrieben überschritten werden.

##### **NONE**

Für Transaktionen, die Threads aus dieser DB2ENTRY-Ressource verwenden, sind keine Abrechnungsdatensätze erforderlich.

Allerdings generiert Db2 mindestens einen Abrechnungsdatensatz für jeden Thread, nachdem der Thread beendet wurde. Bei Änderungen der Berechtigung werden zusätzliche Datensätze generiert.



**TASK** Die CICS-Db2-Anschlussfunktion generiert mindestens einen Abrechnungsdatensatz für jede CICS-Task.

Eine Transaktion, die mehrere Arbeitseinheiten (UOWs = Units of Work) enthält und in der die Threads an Synchronisationspunkten freigegeben werden, kann für jede der UOWs einen anderen Thread verwenden. Dies hat zur Folge, dass für jede UOW ein Abrechnungsdatensatz erstellt werden kann.

**TXID** Die CICS-Db2-Anschlussfunktion führt zur Generierung eines Abrechnungsdatensatzes, sobald die Transaktions-ID, die den Thread verwendet, geändert wird.

Diese Option gilt für Db2-Eintragsdefinitionen, die von mehreren Transaktions-IDs verwendet werden. Da Threads normalerweise an Synchronisationspunkten freigegeben werden, verwendet eine Transaktion, die mehrere UOWs enthält, möglicherweise einen anderen Thread für jede UOW. Dies hat zur Folge, dass für jede UOW ein Abrechnungsdatensatz erstellt werden kann.

**UOW** Die CICS-Db2-Anschlussfunktion führt zur Generierung eines Abrechnungsdatensatzes für jede UOW. Dabei wird davon ausgegangen, dass der Thread am Ende der UOW freigegeben wird.

#### **AUTHID**(*benutzer-id*)

Gibt die Benutzer-ID an, die bei Verwendung dieser DB2ENTRY-Komponente für die Sicherheitsprüfung benutzt wird.

Verwenden Sie AUTHID nicht, wenn Sie RACF für bestimmte oder alle Aspekte der Sicherheitsprüfung in Ihrem Db2-Adressraum verwenden. Benutzen Sie stattdessen AUTHTYPE mit der Option GROUP, SIGN oder USERID. Sie müssen AUTHTYPE verwenden, weil Threads, die AUTHID benutzen, nicht das erforderliche RACF-Umgebungselement für die Zugriffssteuerung (ACEE = Access Control Environment Element) an Db2 übergeben. Die von Ihnen angegebene ID kann bis zu acht Zeichen lang sein.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

#### **AUTHTYPE**(**{USERID|OPID|GROUP|SIGN|TERM|TX}**)

Gibt den Typ der ID an, die bei Verwendung dieser DB2ENTRY-Komponente für die Sicherheitsprüfung benutzt werden kann.

Wenn Sie RACF für bestimmte oder alle Aspekte der Sicherheitsprüfung in Ihrem Db2-Adressraum verwenden, dann sollten Sie die Option GROUP, SIGN oder USERID benutzen. Sie müssen eine dieser Optionen verwenden, weil nur Threads, die mit diesen Optionen definiert wurden, das erforderliche RACF-Umgebungselement für die Zugriffssteuerung (ACEE) an Db2 übergeben. Wenn Sie allerdings die Option SIGN angeben, dann wird das ACEE nur dann an Db2 übergeben, wenn der für das Attribut SIGNID in der DB2CONN-Definition angegebene Wert mit der Benutzer-ID der CICS-Region übereinstimmt.

Das ACEE ist nicht erforderlich, wenn Sie nur die interne Sicherheitsfunktion von Db2 verwenden. In diesem Fall können Sie jede der verfügbaren Optionen verwenden.

### USERID

Die Benutzer-ID (USERID), die der CICS-Transaktion zugeordnet ist, wird als Berechtigungs-ID verwendet. Wenn die Benutzer-ID weniger als acht Zeichen lang ist, wird sie rechts mit Leerzeichen aufgefüllt.

Wird der Db2-Beispielexit für die Anmeldung mit dem Namen DSN3@SGN mit der Angabe AUTHTYPE(USERID) verwendet, dann sendet der Exit die Benutzer-ID als primäre Berechtigungs-ID an Db2 und den Namen der verbundenen Gruppe als sekundäre ID an Db2. Wenn der Beispielexit für die Anmeldung verwendet wird, dann stimmen die Werte für AUTHTYPE(USERID) und AUTHTYPE(GROUP) überein.

**OPID** Die Bedieneridentifikation, die der Benutzer-ID zugeordnet ist, die der CICS-Funktion für die Transaktionsanmeldung zugewiesen wurde, wird als Berechtigungs-ID verwendet. Die aus drei Zeichen bestehende Bedieneridentifikation wird auf der rechten Seite mit Leerzeichen aufgefüllt, um so die aus acht Zeichen bestehende Berechtigungs-ID zu bilden.

### **GROUP**

Gibt die aus einem bis acht Zeichen bestehende Benutzer-ID (USERID) sowie den Namen der verbundenen Gruppe als Berechtigungs-ID an. In der folgenden Tabelle wird dargestellt, wie diese beiden Werte von Db2 interpretiert werden.

An Db2 übergebene IDs	Interpretation der Werte durch Db2
CICS-Benutzer-ID für die Anmeldung (USERID)	Stellt die primäre Db2-Berechtigungs-ID dar.
Name der verbundenen RACF-Gruppe	Wenn die RACF-Liste der Gruppenoptionen nicht aktiv ist, dann verwendet Db2 den Namen der verbundenen Gruppe, der von der CICS-Anschlussfunktion bereitgestellt wird, als sekundäre Db2-Berechtigungs-ID. Wenn die RACF-Liste der Gruppenoptionen aktiv ist, dann ignoriert Db2 den Namen der verbundenen Gruppe, der von der CICS-Anschlussfunktion bereitgestellt wird, der Wert wird jedoch in der Db2-Liste der sekundären Db2-Berechtigungs-IDs aufgeführt.

Um die Option GROUP benutzen zu können, müssen Sie in den Systeminitialisierungsparametern der Region die Einstellung SEC=YES angeben.

Wenn keine RACF-Gruppen-ID für diese Benutzer-ID (USERID) verfügbar ist, dann wird ein Feld mit acht Leerzeichen als Gruppen-ID an Db2 übergeben.

**SIGN** Gibt an, dass das Attribut SIGNID von DB2CONN als Ressourcenberechtigungs-ID verwendet wird.

**TERM** Gibt die (aus vier Zeichen bestehende und auf acht Zeichen aufgefüllte) Terminalidentifikation als Berechtigungs-ID an. Eine Berechtigungs-ID kann auf diese Weise nicht abgerufen werden, wenn ein Terminal nicht mit der Transaktion verbunden ist.

Wenn eine Transaktion mit einem CICS-Befehl gestartet wird und wenn dieser Transaktion kein Terminal zugeordnet wurde, dann darf AUTHTYPE(TERM) nicht verwendet werden.

**TX** Gibt die Transaktionskennung als Berechtigungs-ID an. Die aus vier Zeichen bestehende Transaktionskennung wird auf der rechten Seite mit Leerzeichen aufgefüllt, um so die aus acht Zeichen bestehende Berechtigungs-ID zu bilden.

**DROLLBACK({YES|NO})**

Gibt an, ob die CICS-Db2-Anschlussfunktion die Anweisung SYNCPOINT ROLLBACK einleitet, wenn eine Transaktion als Opfer einer Deadlock-Auflösung ausgewählt wird.

**YES** Die Anschlussfunktion leitet eine Anweisung SYNCPOINT ROLLBACK ein, bevor die Steuerung wieder an die Anwendung zurückgegeben wird. An das Programm wird der SQL-Rückgabecode -911 zurückgegeben.

Geben Sie nicht YES an, wenn die DB2ENTRY-Komponente von Transaktionen verwendet wird, für die im Rahmen einer OTS-Transaktion Enterprise-Beans ausgeführt werden. Die Verwendung von SYNCPOINT ROLLBACK ist in OTS-Transaktionen nicht zulässig.

**NO** Die Anschlussfunktion leitet für diese Transaktion keinen Rollback ein. An die Anwendung wird der SQL-Rückgabecode -913 zurückgegeben.

**PLAN(plan)**

Gibt den Namen des Plans an, der für diesen Eintrag verwendet werden soll.

**PLANEXITNAME(DSNCUEXT|exit)**

Gibt den Namen des dynamischen Planexits an, der für diese Db2-Eintragsdefinition verwendet werden soll. Wenn Sie die Werte für PLAN und PLANEXITNAME ändern, während für die Db2-Eintragsdefinition aktive Transaktionen vorhanden sind, dann werden die neuen Werte angewendet, wenn die Transaktion den Thread zum nächsten Mal freigibt.

**PRIORITY({HIGH|EQUAL|LOW})**

Gibt die Priorität der TCBs (TCB = Task Control Block; Tasksteuerblock) des Threads für diese DB2ENTRY-Komponente im Verhältnis zum CICS-Haupttasksteuerblock (QR-TCB) an. Wenn CICS mit DB2 Version 6 oder einer höheren Version dieses Produkts verbunden ist, dann werden als TCBs des Threads offene L8-TCBs von CICS verwendet.

**HIGH** Die TCBs des Threads weisen eine höhere Priorität als der CICS-QR-TCB auf.

**EQUAL**

Die TCBs des Threads weisen eine gleiche Priorität wie der CICS-QR-TCB auf.

**LOW** Die TCBs des Threads weisen eine niedrigere Priorität als der CICS-QR-TCB auf.

**PROTECTNUM({0|wert})**

Gibt die maximale Anzahl der geschützten Threads an, die für diese Db2-Eintragsdefinition zulässig sind. Ein Thread, der von einer Transaktion freigegeben wird, wenn keine anderen Arbeiten in der Warteschlange vorhanden sind, kann *geschützt* werden. Dies bedeutet, dass er nicht sofort beendet wird. Ein geschützter Thread wird erst nach zwei vollständigen Bereinigungszyklen beendet, wenn er in der Zwischenzeit nicht erneut verwendet wurde. Wenn der Bereinigungszyklus beispielsweise 30 Sekunden beträgt, dann wird ein geschützter Thread 30 - 60 Sekunden nach seiner Freigabe beendet, wenn er nicht in der Zwischenzeit wiederverwendet wurde. Der erste Bereinigungszyklus nach dem Starten der CICS-Db2-Anschlussfunktion hat eine Länge von fünf

Minuten. Anschließend wird der PURGECYCLE-Wert angewendet. Threads werden nur geschützt, während sie inaktiv sind. Wenn eine Transaktion einen geschützten Thread wiederverwendet, dann wird der Thread aktiviert und die aktuelle Anzahl der geschützten Threads wird verringert.

#### **THREADLIMIT({0|wert})**

Gibt die maximale Anzahl der aktiven Threads für diese Db2-Eintragsdefinition an, die von der CICS-Db2-Anschlussfunktion zugelassen wird, bevor Anforderungen in den Wartestatus versetzt, abnormal beendet oder in den Pool umgeleitet werden.

#### **THREADWAIT({POOL|YES|NO})**

Gibt an, ob Transaktionen auf einen DB2ENTRY-Thread warten oder abnormal beendet werden sollen oder ob es zu einem Überlauf in den Pool kommen soll, wenn die Anzahl der aktiven DB2ENTRY-Threads den in THREADLIMIT angegebenen Wert erreicht.

**POOL** Wenn alle Threads belegt sind, wird die Transaktion umgeleitet, um den Thread-Pool zu verwenden. Ist auch der Pool vollständig belegt, wird die Transaktion, wenn Sie in der Db2-Verbindungsdefinition THREADWAIT(NO) angegeben haben, mit dem Abbruchcode AD3T abnormal beendet.

**NO** Wenn alle Threads belegt sind, dann wird die Transaktion mit dem Abbruchcode AD2P abnormal beendet.

**YES** Wenn alle Threads belegt sind, dann muss die Transaktion warten, bis ein Thread verfügbar wird.

---

## **DB2TRAN-Ressourcen**

Eine DB2TRAN-Ressource dient zur Definition einer Transaktion oder Transaktionsgruppe, die einer DB2ENTRY-Komponente zugeordnet ist. Diese Transaktionen können zusätzlich zu den Transaktionen verwendet werden, die in der DB2ENTRY-Komponente angegeben sind.

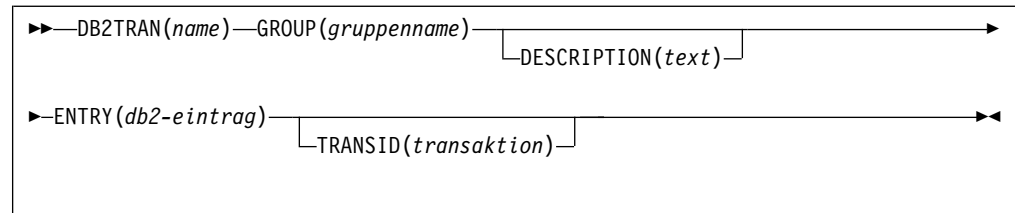
Für eine bestimmte Transaktion kann nur eine DB2TRAN-Definition installiert werden. Wird versucht, eine zweite DB2TRAN-Definition zu installieren, die sich explizit auf die gleiche Transaktions-ID bezieht, dann schlägt diese Aktion fehl.

Die DB2TRAN-Definition lässt zu, dass einer DB2ENTRY-Komponente eine unbegrenzte Anzahl von Transaktionen zugeordnet wird, darunter auch Namen, in denen Platzhalterzeichen verwendet werden. Sie können eine beliebige Anzahl von DB2TRAN-Elementen definieren, die einer einzelnen DB2ENTRY-Komponente zugeordnet werden sollen.

Informationen zum Installieren von Db2-Transaktionsressourcendefinitionen finden Sie in *Installing Db2 transaction resource definitions*. Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in *Working with BAS Db2 transaction resource definitions*.

## **DB2TRAN-Attribute**

Beschreibt die Syntax und die Attribute der DB2TRAN-Ressource.



### DB2TRAN(*name*)

Der aus einem bis acht Zeichen bestehende Name, der zur Identifikation dieser Db2-Transaktionsdefinition dient.

#### Zulässige Zeichen:

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

### DESCRIPTION(*text*)

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

### ENTRY(*db2-eintrag*)

Gibt den Namen der Db2-Eintragsdefinition an, auf die sich diese Db2-Transaktionsdefinition bezieht. Hierbei handelt es sich um die Db2-Eintragsdefinition, der diese zusätzliche Transaktion zugeordnet werden soll.

### GROUP(*gruppenname*)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

#### Zulässige Zeichen:

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

### TRANSID(*transaktion*)

Gibt die Transaktions-ID an, die dem Eintrag zugeordnet ist. Wenn keine Transaktions-ID (TRANSID) angegeben wird, dann werden standardmäßig die ersten vier Zeichen des Db2-Transaktionsdefinitionsnamens verwendet. Die Transaktions-ID kann Platzhalterzeichen (siehe „Platzhalterzeichen für Transaktions-IDs“) enthalten.

## Platzhalterzeichen für Transaktions-IDs

Beim Definieren einer DB2TRAN-Ressource können Sie eine generische Transaktions-ID angeben und dabei das Sternsymbol (\*) und das Plussymbol (+) als Platzhalterzeichen verwenden.

- Ein Stern (\*) kann einem Transaktionsnamen hinzugefügt oder alleine verwendet werden. Auf diese Weise kann jeder Name als Element mit generischen Komponenten definiert werden. Ein Transaktionsname, an dessen Ende ein Stern ange-

geben wird, steht für null bis drei nicht angegebene Zeichen in der Transaktions-ID. Beispiel: Die Transaktions-ID (TRansid) „T\*” steht für die Transaktionen „T”, „TA”, „TAB” und „TABE”.

- Ein Pluszeichen (+) ist an jeder beliebigen Position zulässig, um ein einzelnes Zeichen darzustellen.
- Ein Stern (\*) ohne weitere Angaben steht für eine beliebige Transaktion und kann als alternative Pooldefinition fungieren. Diese Angabe unterscheidet sich von einem Pool durch das zusätzliche Attribut, durch das festgelegt wird, dass für DB2ENTRY ein Überlauf in den Pool ausgeführt wird, wenn die Threadzuordnung den Grenzwert überschreitet.

Die Regeln für den Abgleich legen fest, dass die präziseste Übereinstimmung verwendet wird. Beispiel: Die Transaktion FRED verwendet DB2ENTRY(1) und gibt somit die generische Transaktions-ID „FRE\*” an, wohingegen DB2ENTRY(2) die generische Transaktions-ID „F\*” angibt. Des Weiteren ist ein Pluszeichen (+) präziser als ein Stern (\*). Beispiel: „FRE+” ist präziser als „FRE\*”.

Wird AUTHTYPE(TX) angegeben, dann wird die eigentliche TXID als primäre Berechtigungs-ID an Db2 übergeben und nicht der Name, in dem Platzhalterzeichen verwendet werden.

Beachten Sie hierbei, dass der Befehl INQUIRE DB2TRAN die einzelnen Transaktionen, die mit der generischen Transaktions-ID übereinstimmen, nicht identifizieren kann, wenn ein DB2TRAN-Element mit einer generischen Transaktions-ID definiert wird, die ein Platzhalterzeichen umfasst. Sie können z. B. den folgenden Befehl absetzen:

```
CEMT INQUIRE DB2TRAN(*) TRANSID(ABCD)
```

Daraufhin werden die Details des DB2TRAN-Elements angezeigt, dem die Transaktion ABCD zugeordnet ist. Wenn allerdings das DB2TRAN-Element mit der Transaktions-ID „ABC\*” definiert wird, dann kann der Befehl INQUIRE DB2TRAN das DB2TRAN-Element nicht mit der Transaktions-ID „ABCD” abgleichen und gibt eine Antwort zurück, in der Sie darüber informiert werden, dass das gesuchte Element nicht gefunden werden konnte.

---

## DOCTEMPLATE-Ressourcen

Eine DOCTEMPLATE-Ressource definiert die Attribute einer *Dokumentschablone*.

Eine Dokumentschablone ist eine Einheit von Informationen, die zum Erstellen eines Dokuments verwendet wird. Eine Dokumentschablone kann festen Text und Symbole enthalten, die Text darstellen, dessen Wert von einem Anwendungsprogramm bereitgestellt wird. Dokumentschablonen können von einer CICS-Anwendung erstellt oder aus einer externen Quelle abgerufen werden. Weitere Informationen hierzu finden Sie in CICS documents and document templates.

Die Schablone kann sich an einer der folgenden Positionen befinden:

- In einer partitionierten MVS-Datei (angegeben in den Attributen DDNAME und MEMBERNAME).
- In einer Warteschlange für temporären Speicher (angegeben im Attribut TS-QUEUE).
- In einer Warteschlange mit transienten Daten (angegeben im Attribut TD-QUEUE).
- In einem CICS-Programm (angegeben im Attribut PROGRAM).

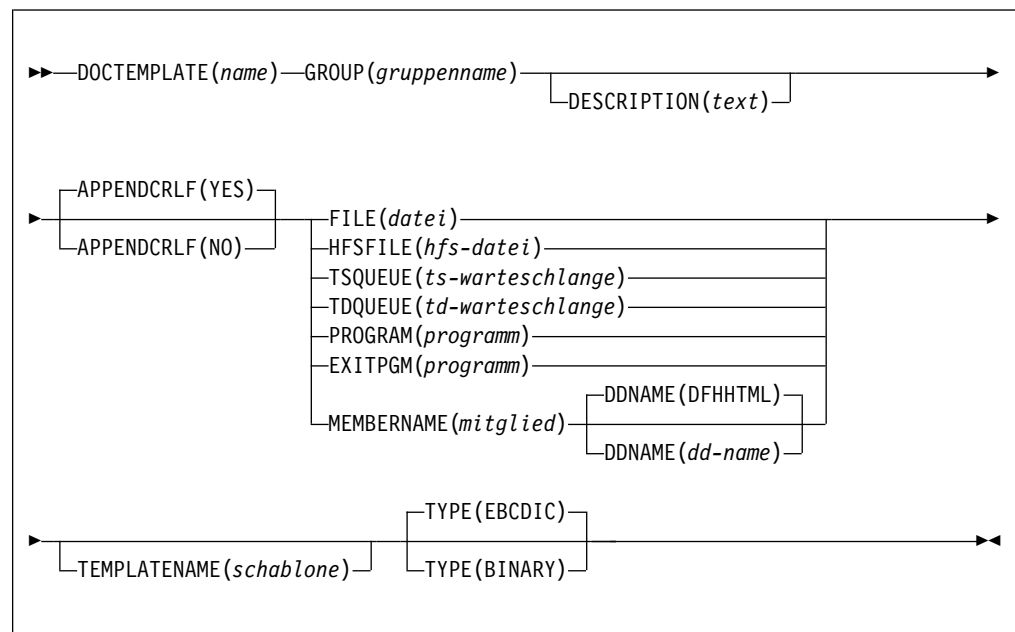
- In einer CICS-Datei (angegeben im Attribut FILE).
- In einer z/OS UNIX System Services-Datei (angegeben im Attribut HFSFILE).

Die Schablone kann auch von einem (im Attribut EXITPGM angegebenen) Exitprogramm zurückgegeben werden.

Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with BAS document template resource definitions.

## DOCTEMPLATE-Attribute

Beschreibt die Syntax und die Attribute der DOCTEMPLATE-Ressource.



### APPENDCRLF(YES|NO)

Gibt an, ob CICS in den logischen Datensätzen der Schablone nachfolgende Leerzeichen löschen und Rücklauf-/Zeilenvorschubzeichen hinzufügen soll.

### DDNAME(DFHHTML|dd-name)

Wenn sich die Schablone in einer partitionierten MVS-Datei (PDS = Partitioned Data Set) befindet, wird der DD-Name der PDS angegeben. Der Name kann bis zu acht Zeichen lang sein.

#### Zulässige Zeichen:

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Wenn Sie einen Wert für das Attribut MEMBERNAME, jedoch keinen Wert für DDNAME angeben, dann wird der Standardwert DFHHTML übernommen.

Wenn Sie dieses Attribut angeben, darf EXITPGM, FILE, HFSFILE, PROGRAM, TDQUEUE oder TSQUEUE nicht angegeben werden.

### DESCRIPTION(text)

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem

Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

**DOCTEMPLATE**(*name*)

Gibt den Namen dieser Dokumentschablonendefinition an. Der Name kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

**EXITPGM**(*programm*)

Gibt den Namen eines Exitprogramms an, das eine Schablone generiert. Der Name kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Wenn Sie dieses Attribut angeben, darf DDNAME, FILE, HFSFILE, MEMBER-NAME, PROGRAM, TDQUEUE oder TSQUEUE nicht angegeben werden.

**FILE**(*datei*)

Wenn sich die Schablone in einer CICS-Datei befindet, wird der Name der Datei angegeben. Der Name kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Wenn Sie dieses Attribut angeben, darf DDNAME, EXITPGM, HFSFILE, MEMBERNAME, PROGRAM, TDQUEUE oder TSQUEUE nicht angegeben werden.

**GROUP**(*gruppenname*)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

**HFSFILE**(*hfs-datei*)

Wenn sich die Schablone in einer z/OS UNIX System Services-Datei befindet, wird in dieser Option der vollständig qualifizierte (absolute) oder relative Name der z/OS UNIX-Datei angegeben. Der Name kann als absoluter Name



einschließlich aller Verzeichnisse und beginnend mit einem Schrägstrich (/) angegeben werden. Beispiel: /u/facts/images/bluefish.jpg. Alternativ hierzu kann er auch als Name relativ zum Ausgangsverzeichnis (HOME) der Benutzer-ID für die CICS-Region angegeben werden. Beispiel: facts/images/bluefish.jpg. Es können bis zu 255 Zeichen verwendet werden.

**Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Wenn Sie dieses Attribut angeben, darf DDNAME, EXITPGM, MEMBERNAME, PROGRAM, TDQUEUE oder TSQUEUE nicht angegeben werden.

**Anmerkung:** Die CICS-Region muss über die Berechtigungen für den Zugriff auf z/OS UNIX und außerdem über die Berechtigung für den Zugriff auf das z/OS UNIX-Verzeichnis, das die Datei enthält, sowie auf die Datei selbst verfügen. Weitere Informationen zur Vorgehensweise beim Erteilen dieser Berechtigungen finden Sie in Giving CICS regions access to z/OS UNIX directories and files.

**MEMBERNAME**(*mitglied*)

Wenn sich die Schablone in einer partitionierten MVS-Datei (PDS = Partitioned Data Set) befindet, wird der Name des Mitglieds angegeben, das die Schablone enthält. Der Name kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Wenn Sie dieses Attribut angeben, darf EXITPGM, FILE, HFSFILE, PROGRAM, TDQUEUE oder TSQUEUE nicht angegeben werden.

**PROGRAM**(*programm*)

Wenn sich die Schablone in einem CICS-Programm befindet, wird der Name des Programms angegeben. Der Name kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Wenn Sie dieses Attribut angeben, darf DDNAME, EXITPGM, FILE, HFSFILE, MEMBERNAME, TDQUEUE oder TSQUEUE nicht angegeben werden.

**TDQUEUE**(*td-warteschlange*)

Wenn sich die Schablone in einer Warteschlange mit transienten Daten befindet, wird der Name der Warteschlange angegeben. Der Name kann bis zu vier Zeichen lang sein.

**Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Wenn Sie dieses Attribut angeben, darf DDNAME, EXITPGM, FILE, HFSFILE, MEMBERNAME, PROGRAM oder TSQUEUE nicht angegeben werden.

### TEMPLATENAME(*schablone*)

Gibt den Namen an, unter dem die Schablone für Anwendungsprogramme bekannt ist, die sie benutzen. Der Name kann bis zu 48 Zeichen lang sein.

#### Zulässige Zeichen:

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Wenn Sie für dieses Attribut keinen Wert angeben, wird der Wert des Attributs DOCTEMPLATE verwendet, der auf der rechten Seite mit Leerzeichen erweitert wird.

### TSQUEUE(*ts-warteschlange*)

Wenn sich die Schablone in einer Warteschlange für temporären Speicher befindet, wird der Name der Warteschlange angegeben. Der Name kann bis zu 16 Zeichen lang sein.

#### Zulässige Zeichen:

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Wenn Sie dieses Attribut angeben, darf DDNAME, EXITPGM, FILE, HFSFILE, MEMBERNAME, PROGRAM oder TDQUEUE nicht angegeben werden.

### TYPE({EBCDIC|BINARY})

Gibt das Format des Inhalts der Schablone an.

#### BINARY

Wenn die Schablone aus der Schablonenbibliothek geladen wird, dann führt das System keine Syntaxanalyse des Inhalts der Schablone durch.

#### EBCDIC

Wenn die Schablone aus der Schablonenbibliothek geladen wird, werden die Inhalte als EBCDIC-Text analysiert.

---

## ENQMODEL-Ressourcen

In einer ENQMODEL-Definition wird eine benannte Ressource definiert, für die die Befehle **EXEC CICS ENQ** und **EXEC CICS DEQ** über einen sysplexweiten Bereich verfügen.

CICS verwendet ENQMODEL-Ressourcen mit z/OS Global Resource Serialization, um den sysplexweiten Schutz von Ressourcen bereitzustellen, die von mehreren Anwendungen benutzt werden. Lokale Enqueues (Einreihungen in die Warteschlange) in einer einzelnen CICS-Region werden im CICS-Adressraum verwaltet. Sysplexweite Enqueues, die mehrere CICS-Regionen betreffen, werden mit z/OS Global Resource Serialization verwaltet. Weitere Informationen zu Global Resource Serialization finden Sie in z/OS MVS Planning: Global Resource Serialization.

Werden die Befehle **EXEC CICS ENQ** und **EXEC CICS DEQ** für eine Ressource abgesetzt, dann überprüft CICS, ob eine übereinstimmende installierte ENQMODEL-Definition vorhanden ist. Das Attribut ENQSCOPE einer ENQMODEL-Ressource definiert die Gruppe der Regionen, die mit demselben Geltungsbereich für Enqueues arbeiten. Wenn für das Attribut ENQSCOPE kein Wert angegeben wird (Standardwert), dann behandelt CICS alle übereinstimmenden Enqueue- oder Dequeue-Anforderungen (Anforderungen zur Entfernung aus der Warteschlange) als lokale Anforderungen in Bezug zur absetzenden CICS-Region. Wenn für ENQSCOPE ein Wert angegeben wird, dann behandelt CICS die Enqueue- oder Dequeue-Anforderungen als sysplexweit und übergibt einen Warteschlangennamen

und einen Ressourcennamen an z/OS Global Resource Serialization, damit die Einreihung in die Warteschlange verwaltet werden kann.

z/OS Global Resource Serialization kann entweder in einer Ringkonfiguration oder einer Sternkonfiguration konfiguriert werden. Aus Gründen der Leistung wird eine Ringkonfiguration für Produktionsbereiche in Mehrsystemumgebungen nicht empfohlen. Sie können auch den CICS-Systeminitialisierungsparameter **NQRNL** verwenden, um anzugeben, ob z/OS Global Resource Serialization die RNL-Verarbeitung für CICS-Enqueue-Anforderungen verwenden soll. Dies kann sich auf den Geltungsbereich von Ressourcen auswirken. Weitere Informationen zur Konfiguration von z/OS Global Resource Serialization für CICS finden Sie in *Global CICS enqueue and dequeue: improving performance*.

Sie können Sysplex-Enqueues anzeigen, die mit den Funktionen von z/OS Global Resource Serialization arbeiten. Verwenden Sie dazu den Befehl **DISPLAY GRS**. Beispiel:

```
D GRS,RES=(DFHEqname|*,[ rname|,*])
```

Dabei gilt Folgendes:

**qname**

Gibt ein aus vier Zeichen bestehendes ENQSCOPE-Element an, das mit einer ENQMODEL-Ressource definiert wird.

**rname**

Gibt ein ENQNAME-Element an, das mit einer ENQMODEL-Ressource definiert wird.

Sysplexweite Enqueues werden nur für eine Ressource, nicht jedoch für Enqueues nach Adresse unterstützt. Diese Funktion wurde nicht für das Sperren wiederherstellbarer Ressourcen konzipiert.

Informationen zur Installation von Ressourcendefinitionen für Enqueue-Modelle mit CEMT und CEDA finden Sie in *Installing enqueue model resource definitions*. Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in *Working with BAS enqueue model resource definitions*.

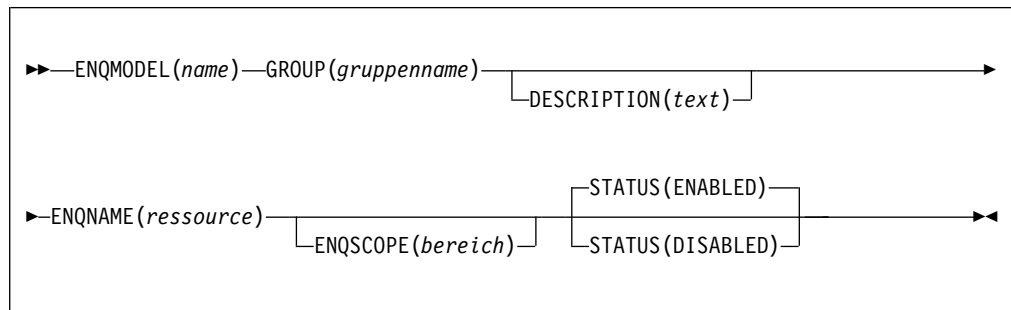
## Zugriff auf ENQMODEL-Ressourcen

Für die CICS-Regionen, die die sysplexweite Enqueue- oder Dequeue-Funktion verwenden müssen, müssen alle erforderlichen ENQMODEL-Ressourcen definiert und installiert worden sein. Um dies sicherzustellen, sollten Sie in CICS-Regionen eine CSD (CICS-Systemdefinition) gemeinsam nutzen. Für die Initialisierungsgruppenlisten sollten dieselben ENQMODEL-Gruppen eingebunden werden.

Anwendungen benutzen Sysplex-Enqueues, wenn Sie entsprechende ENQMODEL-Ressourcen definieren. An den Anwendungsprogrammen ist keine Änderung erforderlich. Für Anwendungen, in denen der Ressourcename dynamisch konfiguriert wird und daher nicht im Voraus bekannt ist, können Sie die Programmexits XNQEREQ und XNQEREQC der Enqueue-EXEC-Schnittstelle verwenden, um Zeichen am Anfang des Ressourcennamens anzugeben, die mit einer passenden ENQMODEL-Ressourcendefinition übereinstimmen. Weitere Informationen zu diesen Benutzerexits finden Sie in *Enqueue EXEC interface program exits XNQEREQ and XNQEREQC*.

## ENQMODEL-Attribute

Beschreibt die Syntax und die Attribute der ENQMODEL-Ressource.



### DESCRIPTION(text)

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

### ENQMODEL(name)

Gibt den Namen dieser ENQMODEL-Definition an. Der Name kann bis zu acht Zeichen lang sein.

#### Zulässige Zeichen:

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Dieser Name wird verwendet, um die ENQMODEL-Definition in der CSD-Datei zu identifizieren. Innerhalb des aktiven CICS-Systems wird er nicht verwendet.

### ENQNAME(ressource)

Gibt den aus 1 - 255 Zeichen bestehenden Ressourcennamen an.

#### Zulässige Zeichen:

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Sie können auch einen Stern (\*) als letztes Zeichen verwenden, um einen generischen Namen anzugeben.

### ENQSCOPE(bereich)

Gibt den aus vier Zeichen bestehenden optionalen Namen für den Bereich des Enqueue-Modells (Modell für Einreihung in die Warteschlange) an. Wenn dieser Name nicht angegeben wird oder wenn für den Namen Leerzeichen angegeben werden, verfügen übereinstimmende Enqueue-Modelle über einen lokalen Bereich.

### GROUP(gruppenname)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

**STATUS({ENABLED|DISABLED})**

Gibt an, ob das Enqueue-Modell im Status ENABLED oder DISABLED installiert werden soll. Der Standardwert ist ENABLED.

**ENABLED**

Übereinstimmende Enqueue-Anforderungen (Anforderungen zur Einreihung in die Warteschlange) werden auf normale Weise verarbeitet.

**DISABLED**

Übereinstimmende Enqueue-Anforderungen werden zurückgewiesen und die absetzende Task wird abnormal beendet. Übereinstimmende INSTALL CREATE- und DISCARD-Anforderungen werden verarbeitet.

---

## FEPI-Ressourcen

Knotenlistendefinitionen, Pooldefinitionen, Eigenschaftengruppendefinitionen und Ziellistendefinitionen werden für FEPI-Ressourcen verwendet.

In den FEPI-Knotenlistendefinitionen werden die physischen Merkmale und Verfahrensmerkmale von FEPI-Knoten beschrieben. Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with FEPI node list resource definitions.

In den FEPI-Pooldefinitionen werden die physischen Merkmale und Verfahrensmerkmale von FEPI-Pools beschrieben. Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with FEPI pool resource definitions.

In den FEPI-Eigenschaftengruppendefinitionen werden die physischen Merkmale und Verfahrensmerkmale von FEPI-Eigenschaftengruppen beschrieben. Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with FEPI property set resource definitions.

In den FEPI-Ziellistendefinitionen werden die physischen Merkmale und Verfahrensmerkmale von FEPI-Zielen beschrieben. Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with FEPI target list resource definitions.

---

## FILE-Ressourcen

Eine FILE-Ressource definiert die physischen Merkmale und die Verfahrensmerkmale einer Datei.

Die FILE-Definition umfasst die Attribute, die Informationen zu den Datensatzmerkmalen, den für die Datei zulässigen Operationstypen, den Wiederherstellungsattributen und den Operationen bereitstellen, die im Journal aufgezeichnet werden sollen. CICS-Dateien entsprechen normalerweise physischen Dateien, die in VSAM definiert werden müssen, bevor sie benutzt werden können. Mit CICS-Dateien können in Ihren Anwendungen die folgenden Aktionen ausgeführt werden:

- Direkter Zugriff auf die Datensätze in einer Datei.
- Zugriff auf Datensätze in einer Datentabelle, die aus der Datei geladen wurde.
- Zugriff auf Datensätze in einer Coupling-Facility-Datentabelle ohne Einbindung einer Datei (weil in der CFDT-Dateidefinition LOAD(NO) angegeben ist).

Die folgenden Ressourcen, die den CICS-Dateien zugeordnet sind, können mit RDO verwaltet werden:

- VSAM-Dateien (einschließlich Dateien, die auf von CICS verwaltete und benutzerverwaltete Datentabellen, Coupling-Facility-Datentabellen sowie auf Dateien verweisen, die auf VSAM-Dateien verweisen).
- Ferne VSAM-Dateien.
- Ferne BDAM-Dateien.
- VSAM-LSR-Pools (LSR = Local Shared Resources; gemeinsam genutzte lokale Ressourcen), die mit LSRPOOL-Ressourcendefinitionen definiert werden.

Damit eine Datei auf einem aktiven CICS-System benutzt werden kann, muss zuerst ihre Definition auf dem System installiert werden. Die CICS-Dateisteuerung verwendet die installierte Definition, um die Datei zu suchen, wenn auf sie zugegriffen werden muss. Außerdem wird die Definition verwendet, um die Anzahl der Tasks zu dokumentieren, die die Datei benutzen, und um die Verarbeitungsstatistiken zu erfassen und weitere Dateisteuerungsinformationen zu verwalten.

Informationen zur Installation von FILE-Ressourcendefinitionen mit CEMT und CEDA finden Sie in *Installing FILE resource definitions*. Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in *Working with BAS FILE resource definitions*.

## FILE-Ressourcen in CICS-Bundles

Sie können ein CICS-Bundle verwenden, um eine FILE-Ressourcendefinition zu erstellen, zu bearbeiten und zu installieren. Wenn Sie eine FILE-Ressource auf diese Weise erstellen, dann können Sie das CICS-Bundle zum Verwalten des Lebenszyklus dieser Ressource verwenden.

Folgende Dateitypen werden für die Definition in CICS-Bundles unterstützt:

- VSAM-Dateien (einschließlich Dateien, die auf von CICS verwaltete und benutzerverwaltete Datentabellen, Coupling-Facility-Datentabellen sowie auf Dateien verweisen, die auf VSAM-Dateien verweisen).
- Ferne VSAM-Dateien.
- Ferne BDAM-Dateien.

Der Anfangsstatus einer dynamisch generierten FILE-Ressource wird aus dem Anfangsstatus des Bundles abgeleitet, das die Ressource definiert. Demzufolge ist es nicht möglich, eine FILE-Ressource mit der STATUS-Angabe UNENABLED zu definieren, um das implizite Öffnen von Dateien durch Anwendungen zu verhindern.

Sie können eine FILE-Ressource abfragen, die von einem CICS-Bundle dynamisch generiert wird. Verwenden Sie hierzu den Befehl **EXEC CICS INQUIRE FILE** oder **CEMT INQUIRE FILE**.

Der Befehl **DISCARD** kann für eine FILE-Ressource, die von einem CICS-Bundle dynamisch generiert wird, nicht abgesetzt werden. Sie müssen das CICS-Bundle löschen. CICS wendet die Operation auf die JVMSERVER-Ressource an.

Zum Ändern der Attribute einer FILE-Ressource, die in einem CICS-Bundle definiert und installiert wurde, können Sie den Ressourceneditor in CICS Explorer verwenden, um die Definition im CICS-Bundle zu ändern und eine neue Version des CICS-Bundles oder der Anwendung zu installieren, über das bzw. die sie bereitgestellt wurde. Sie können den Befehl **SET FILE** verwenden, um die Attribute der dynamisch generierten Ressource zu ändern. Diese Änderungen werden jedoch nicht katalogisiert und können nach einem Warmstart von CICS nicht wiederhergestellt werden.

Um den Status einer FILE-Ressource zu ändern, die in einem CICS-Bundle definiert und installiert wurde, müssen Sie den Status des CICS-Bundles oder der Anwendung ändern, über das bzw. die sie bereitgestellt wurde. Wenn ein Problem mit dem Inaktivieren eines CICS-Bundles auftritt, mit dem eine FILE-Ressource definiert wird, dann können Sie den Befehl **EXEC CICS SET FILE** oder **CEMT SET FILE** für die dynamisch generierte Ressource verwenden, wenn diese Aktion erforderlich ist. Befolgen Sie die Fehlerbehebungsprozedur, die in Anwendungsfehler diagnostizieren erläutert wird, um das Problem zu diagnostizieren und geeignete Maßnahmen zu ergreifen.

Weitere Informationen zum Definieren von Ressourcen in CICS-Bundles finden Sie in Defining CICS bundles.

## Ferne Dateien

Wenn mehrere CICS-Systeme verbunden sind, dann können diese Systeme die Dateien der anderen beteiligten Systeme ebenfalls benutzen. Sämtliche Dateien müssen in CICS definiert werden. Dies betrifft auch die Dateien, die zu einem der anderen Systeme gehören. Dateien auf anderen Systemen werden als *fern* bezeichnet. Auf ferne Dateien wird über die CICS-Funktionsverlagerung von Dateisteuerungsanforderungen in die ferne Region zugegriffen, die Eigner der Datei ist.

Dies gilt jedoch nicht für Dateien, auf die im RLS-Modus zugegriffen wird, oder für Coupling-Facility-Datentabellen, die immer als lokale Dateien definiert sind.

Die Ressourcenattribute, die von CICS für ferne Dateien benötigt werden, gelten nicht speziell für die benutzte Zugriffsmethode. Aus diesem Grund können Sie sowohl ferne BDAM-Dateien als auch ferne VSAM-Dateien mithilfe von RDO definieren.

Wenn Sie ein fernes System in REMOTESYSTEM angeben, dann können Sie auch einen Wert für REMOTENAME angeben, bei dem es sich um den Namen der Datei handelt, die auf dem fernen System verwendet wird.

Wenn Sie in REMOTESYSTEM einen Namen angeben, der der Angabe für SYSIDNT der CICS-Region entspricht, in der die Dateidefinition installiert ist, dann installiert CICS die Definition als lokale Datei. Andernfalls installiert CICS die Definition als ferne Datei.

Wenn eine Dateidefinition als ferne Datei installiert wird, dann werden nur die folgenden Attribute verwendet:

- REMOTESYSTEM
- REMOTENAME
- RECORDSIZE
- KEYLENGTH

Die anderen Attribute werden verwendet, wenn dieselbe Dateidefinition als lokale Definition installiert wird. Weitere Informationen finden Sie in Defining remote resources for function shipping.

## Coupling-Facility-Datentabellen

Die Unterstützung für Coupling-Facility-Datentabellen bietet eine Methode zur gemeinsamen Nutzung von Dateidaten mithilfe der Dateisteuerung von CICS. Hierzu wird weder eine FOR (File-Ownning Region; Dateiverwaltungsregion) noch die VSAM-RLS-Unterstützung benötigt. Die Daten werden in einer Coupling-Facility-Listenstruktur in einer Tabelle gehalten, die zahlreiche Ähnlichkeiten mit einer gemeinsam genutzten, benutzerverwalteten Datentabelle aufweist.

Anders als benutzerverwaltete Datentabellen müssen die Coupling-Facility-Datentabellen nicht vorab aus einer Quellendatei geladen werden. Das Laden einer Coupling-Facility-Datentabelle wird mit den Attributen DSNAME und LOAD der Dateiressourcendefinition gesteuert, mit denen diese CFDTs (Coupling-Facility-Datentabellen) vollständig von den Anwendungsprogrammen mit den Daten gefüllt werden können, die sie verwenden. Dazu wird LOAD(NO) angegeben.

Die Vorgehensweise bei der Verwendung von LOAD(YES) und der DSNAME-Attribute ermöglicht Ihnen die Steuerung des Ladevorgangs einer CFDT auf verschiedene Arten. Beispiel:

1. Jede CICS-Region kann die Coupling-Facility-Datentabelle laden. Beim ersten Öffnen der Datei wird die CFDT geladen. Hierbei spielt es keine Rolle, welche CICS-Region die Öffnungsanforderung absetzt. Die Datei wird durch das ladende CICS-System im schreibgeschützten Modus geöffnet. In allen Dateidefinitionen für die Tabelle sind LOAD(YES) und der Dateiname (DSNAME) einer Quellendatei angegeben. Bei Verwendung dieses Ansatzes müssen Sie sicherstellen, dass in jeder Dateidefinition dieselbe Datei angegeben wird. Andernfalls wird die Datei, die in der ersten Definition angegeben ist, die geöffnet werden soll, in die CFDT geladen. CICS überprüft nicht, ob die DSNAME-Werte für alle Dateien, die auf dieselbe CFDT verweisen, identisch sind.
2. Für das Laden der Coupling-Facility-Datentabelle kann eine CICS-Region verantwortlich sein. Die ladende Region enthält eine Dateidefinition für die CFDT, in der LOAD(YES) und der DSNAME-Wert für die Datei angegeben sind. Die Öffnung durch das ladende CICS-System erfolgt im schreibgeschützten Modus. Weitere CICS-Regionen benötigen keinen Zugriff auf die Quellendatei, können die CFDT jedoch erst öffnen, nachdem sie von der ladenden Region geöffnet wurde. Die Dateidefinitionen für die CFDT in den nicht ladenden Regionen müssen ebenfalls die Angabe LOAD(YES) enthalten, der DSNAME-Wert ist jedoch nicht erforderlich.

Sie können den Zugriff auf eine Coupling-Facility-Datentabelle beschränken, bis der Ladevorgang unter Verwendung von zwei (oder mehr) Dateinamen abgeschlossen wurde, die auf dieselbe Coupling-Facility-Datentabelle verweisen. Gehen Sie wie folgt vor, um den Zugriff auf diese Weise zu steuern:

- Definieren Sie nur einen Dateinamen so, dass mit ihm die Datentabelle geladen werden kann. Geben Sie hierzu LOAD(YES) und DSNAME(*dateiname*) an. Erstellen Sie keine Verweise von Anwendungsprogrammen auf diesen Dateinamen.
- Definieren Sie weitere Dateien zur Verwendung durch Anwendungsprogramme und stellen Sie sicher, dass mit dieser Dateidefinition das Laden der Tabelle nicht eingeleitet werden kann. Geben Sie LOAD(YES) an, vergewissern



Sie sich jedoch, dass der Dateiname nicht im Attribut DSNAME angegeben wird und dass keine Datendefinitionsanweisung für die betreffenden Dateien angegeben wird.

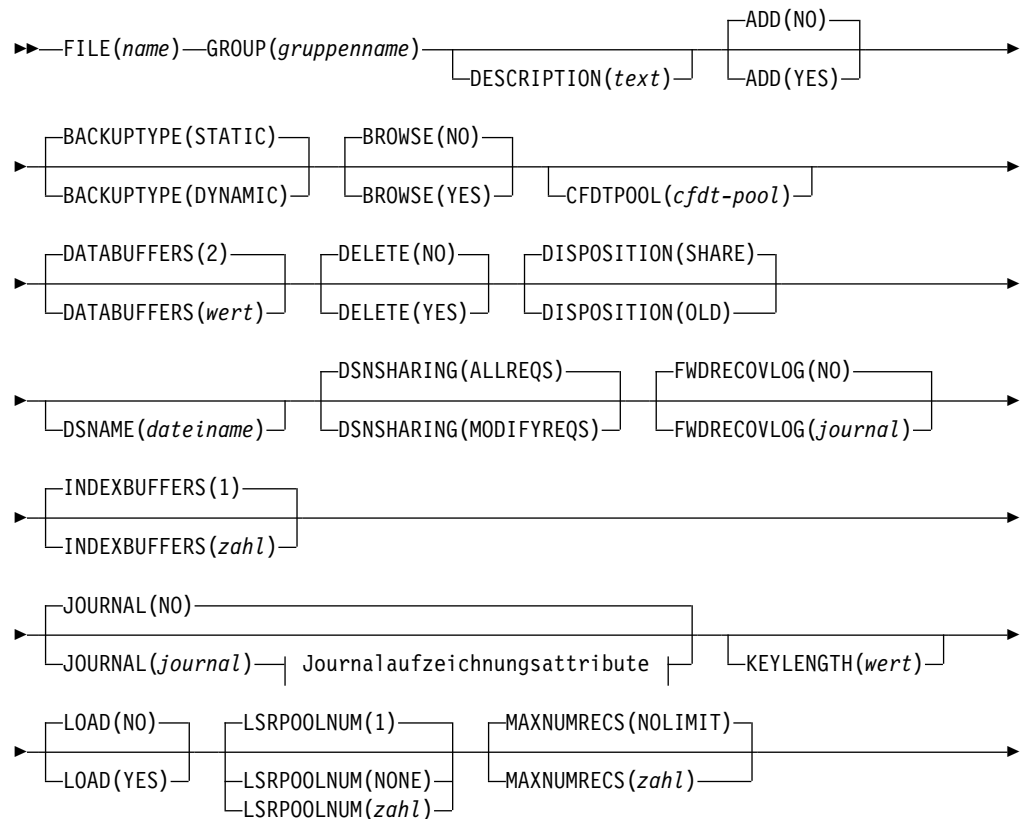
- Vergewissern Sie sich, dass die ladefähige Datei geöffnet wird, bevor Anwendungsprogramme möglicherweise Zugriff auf die Datentabelle benötigen. Definieren Sie z. B. den Namen der zum Laden der Tabelle verwendeten Datei mit OPENTIME(STARTUP). Auf diese Weise kann sichergestellt werden, dass die Datei in der Endphase der CICS-Initialisierung automatisch geöffnet wird.
  - Vergewissern Sie sich, dass Anwendungsprogramme auf die Datentabelle zugreifen, indem sie auf einen Dateinamen verweisen, der nicht zum Laden der Daten verwendet wird.
3. Sie können die beiden hier beschriebenen Ansätze auch kombinieren. Dabei können bestimmte CICS-Regionen die Tabelle laden, während sie für andere CICS-Regionen geladen werden muss.

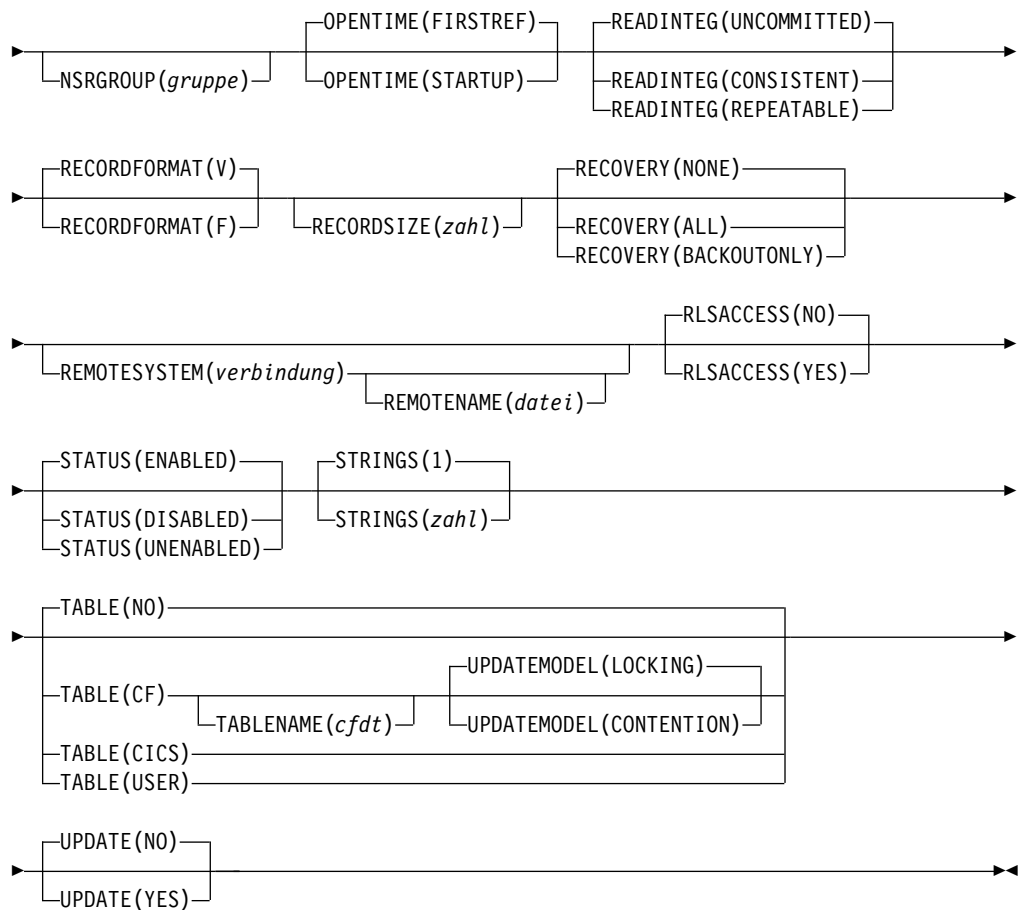
## Gemeinsam genutzte Datentabellen

Von CICS verwaltete Datentabellen und benutzerverwaltete Datentabellen werden zusammen als *gemeinsam genutzte Datentabellen* bezeichnet.

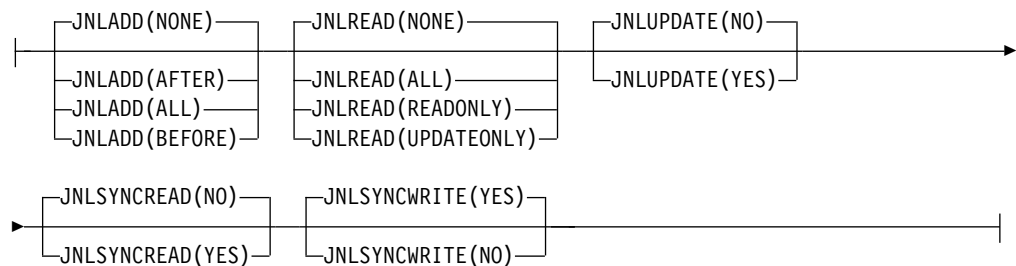
## FILE-Attribute

Beschreibt die Syntax und die Attribute der FILE-Ressource.





### Journalaufzeichnungsattribute:



### ADD({NO|YES})

Gibt an, ob Datensätze zu der Datei hinzugefügt werden können.

### BACKUPTYPE({STATIC|DYNAMIC})

CICS-VSAM-Dateien können als für die Sicherung auswählbar definiert werden, während sie zur Aktualisierung geöffnet sind. Dieses Attribut wird nicht für Dateien verwendet, die mit `RLSACCESS(YES)` definiert wurden, und auch nicht, wenn die Wiederherstellungsoptionen im ICF-Katalog definiert sind. Um in CICS die Verwendung dieses Attributs anstelle der Wiederherstellungsoptionen im Katalog zu erzwingen, müssen Sie für den Systeminitialisierungsparameter **NONRLSRECOV** die Option `FILEDEF` festlegen. Für Dateien, auf die im RLS-Modus zugegriffen wird, müssen Sie den Sicherungstyp in der Dateidefinition im ICF-Katalog angeben.

Dieses Attribut wird für Coupling-Facility-Datentabellen ignoriert. Wenn im ICF-Katalog für eine Quellendatei, die der Tabelle zugeordnet ist, Wiederherstellungsattribute definiert sind, werden diese ebenfalls ignoriert. Für eine CFDT (Coupling-Facility-Datentabelle) kann keine Sicherung von geöffneten Dateien (BWO = Backup-While-Open) ausgewählt werden.

Folgende Werte sind zulässig:

#### **DYNAMIC**

Geben Sie DYNAMIC zusammen mit der Einstellung ALL des Attributs RECOVERY an, um die Datei für die Sicherung auswählbar zu machen, während sie zur Aktualisierung geöffnet ist.

#### **STATIC**

Die Datei ist nicht für eine Sicherung auswählbar, während sie zur Aktualisierung geöffnet ist.

#### **BROWSE({NO|YES})**

Gibt an, ob Datensätze nacheinander aus der Datei abgerufen werden können.

#### **CFDTPOOL(*cfdt-pool*)**

Gibt den Namen des Datentabellenpools der Coupling-Facility an, der die Tabelle enthält, die durch diese Dateidefinition definiert wird. Dieses Attribut ist erforderlich, wenn Sie TABLE(CF) angeben.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Coupling-Facility-Datentabellen können (zu Abrechnungs-, Sicherheits- und Verwaltungszwecken usw.) in CFDT-Gruppen (Pools) aufgeteilt werden. Die Namen aller Coupling-Facility-Datentabellenpools müssen innerhalb des Sysplex eindeutig sein, können aber mit den Namen anderer Pooltypen wie z. B. TS-Pools für die gemeinsame Datennutzung, identisch sein.

Um eine Datei, die auf eine Coupling-Facility-Datentabelle verweist, öffnen zu können, muss ein CFDT-Server für den angegebenen Pool auf dem MVS-System ausgeführt werden, auf dem die Anforderung zum Öffnen abgesetzt wurde. Wenn der erforderliche Server nicht gestartet wurde, schlägt die Anforderung zum Öffnen der Datei fehl.

**Anmerkung:** Die Verwendung des Attributs CFDTPOOL ist nur für CFDTs sinnvoll. Sie können einen Poolnamen für eine Datei angeben, die nicht mit TABLE(CF) definiert ist, diese Angabe wird jedoch von CICS ignoriert. Wenn Sie dann die Dateidefinition so ändern, dass sie auf eine Coupling-Facility-Datentabelle verweist, tritt der in CFDTPOOL angegebene Name in Kraft.

#### **DATABUFFERS({2|wert})**

Gibt die Anzahl der Puffer an, die für Daten verwendet werden sollen. Verwenden Sie einen Wert im Bereich zwischen 2 (Standardwert) und 32767. Der Minimalwert, den Sie angeben können, ist um den Wert 1 höher als die Anzahl der Zeichenfolgen, die im Attribut STRINGS definiert wurde.

#### **DELETE({NO|YES})**

Gibt an, ob Datensätze aus der Datei gelöscht werden können.

#### **DESCRIPTION(*text*)**

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem

Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

**DISPOSITION({SHARE|OLD})**

Gibt die Disposition dieser Datei an.

**OLD** Entspricht dem Parameter DISP=OLD in JCL.

**SHARE**

Entspricht dem Parameter DISP=SHR in JCL.

**DSNAME(*dateiname*)**

Gibt den Namen der Datei (in dem für das Betriebssystem bekannten Format) an, der für diese Datei verwendet werden soll. Für DSNAME kann ein Wert mit einer Länge zwischen 1 und 44 Zeichen angegeben werden, der den Regeln für MVS-Dateinamen entspricht (siehe hierzu den Parameter DSNAME in der z/OS MVS-JCL-Referenz).

**Zulässige Zeichen:**

A-Z 0-9 \$ @ # . -

Wenn beim Öffnen einer Datei keine JCL-Anweisung für diese Datei vorhanden ist, dann wird vor dem Öffnen eine dynamische Zuordnung der Datei mit diesem DSNAME-Wert durchgeführt. Wenn die Dateidefinition auf eine Datentabelle (CICS, USER oder CF) verweist, dann muss der Wert für DSNAME mit dem entsprechenden Wert für eine VSAM-Basis-KSDS (Key-Sequenced Data Set; Datei in Schlüsselfolge) übereinstimmen. Ein Pfad oder eine alternative Indexdatei ist nicht zulässig.

Der DSNAME-Wert, der in einer Datendefinitionsanweisung für diese Datei im CICS-JCL-Code für den Start angegeben wurde, hat Vorrang gegenüber dem DSNAME-Wert, der in dieser Dateidefinition angegeben ist.

**Coupling-Facility-Datentabellen**

Wenn in der Dateidefinition LOAD(YES) angegeben ist und wenn diese noch nicht geöffnet wurde, dann gibt DSNAME den Namen der Quelldatei an, aus der die Tabelle geladen werden soll. Alternativ hierzu können Sie auch die Quelldatei in einer Datendefinitionsanweisung im CICS-JCL-Code für den Start angeben. Die angegebene Datei muss eine VSAM-Basis-KSDS sein.

Wenn der Quelldatei ein Pfad oder ein alternativer Index zugeordnet ist, dann werden die über die Datei durchgeführten Aktualisierungen weder in der Quelldatei noch in ihren zugeordneten alternativen Indizes wiedergegeben. Eine Coupling-Facility-Datentabelle ist vollkommen unabhängig von der Quelldatei, nachdem der Ladevorgang abgeschlossen wurde.

Wenn das Laden der Tabelle durch das Öffnen einer anderen Datei eingeleitet werden soll, die in einer anderen Dateidefinition angegeben wurde, dann geben Sie dieses Attribut nicht an. In diesem Fall müssen Sie außerdem sicherstellen, dass der Dateiname nicht in einer Datendefinitionsanweisung im CICS-JCL-Code angegeben ist. Das Öffnen der Datei schlägt so lange fehl, bis der CFDT-Ladevorgang eingeleitet wird. Weitere Informationen zum Laden einer Coupling-Facility-Datentabelle aus einer Datei finden in Coupling-Facility-Datentabellen.

Wenn LOAD(NO) angegeben wurde, wird dieses Attribut nicht benötigt und vom System ignoriert.

**DSNSHARING({ALLREQS|MODIFYREQS})**

Gibt an, ob für die VSAM-Datei die gemeinsame Nutzung von VSAM-Dateinamen verwendet wird. Folgende Werte sind zulässig:

**ALLREQS**

Die gemeinsame Nutzung von Dateinamen wird im ACB festgelegt, wenn die Datei geöffnet wird, und wird deshalb für alle Dateianforderungen verwendet.

**MODIFYREQS**

Die gemeinsame Nutzung von Dateinamen wird im ACB beim Öffnen der Datei nur dann festgelegt, wenn eine DELETE-, ADD- oder UPDATE-Operation für die Datei festgelegt wird.

**FILE(*name*)**

Gibt den Namen der Datei an. Der Name kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Der Name darf nicht mit einem numerischen Zeichen beginnen.

**FWDRECOVLOG({NO|*journal*})**

Gibt das Journal an, das dem Protokolldatenstrom der MVS-Systemprotokollfunktion entspricht, der für die aktualisierende Wiederherstellung benutzt wird.

Dieses Attribut wird für Coupling-Facility-Datentabellen ignoriert. Wenn im ICF-Katalog für eine Quellendatei, die der Tabelle zugeordnet ist, Wiederherstellungsattribute definiert sind, werden diese ebenfalls ignoriert. Für eine CFDT kann keine aktualisierende Wiederherstellung durchgeführt werden.

**NO** Für diese Datei ist keine Protokollierung für die aktualisierende Wiederherstellung erforderlich.

*journal* Die Nummer, die das Journal angibt, das von CICS als Protokoll für die aktualisierende Wiederherstellung verwendet werden soll. CICS-Journalnamen haben das Format DFHJ*nm*. Hierbei steht *nm* für einen Wert im Bereich zwischen 01 und 99. Die nach der aktualisierenden Wiederherstellung erstellten Images (Nachimages) werden in den MVS-Protokolldatenstrom geschrieben, dem der Journalname DFHJ*nm* zugeordnet ist.

**Anmerkung:** In CICS Transaction Server for z/OS stellt DFHJ01 nicht das Systemprotokoll dar.

Dieses Attribut wird von CICS nur verwendet, wenn die folgenden Bedingungen erfüllt sind:

- RECOVERY(ALL) wurde angegeben.
- RLSACCESS(NO) wurde angegeben.
- Im ICF-Katalog sind keine Wiederherstellungsattribute definiert.

Wenn Sie die Wiederherstellungsattribute für eine Datei im ICF-Katalogeintrag für die entsprechende Datei definieren, dann verwendet CICS immer die Wiederherstellungsattribute für den ICF-Katalog und ignoriert die Attribute in der

FILE-Ressource. Um in CICS die Verwendung der FILE-Ressourcenattribute anstelle der Wiederherstellungsoptionen im Katalog zu erzwingen, müssen Sie für den Systeminitialisierungsparameter **NONRLSRECOV** die Option **FILEDEF** festlegen. Sie können die im ICF-Katalog definierten Wiederherstellungsattribute ändern, indem Sie den Befehl **IDCAMS ALTER** verwenden. Dies wird nicht verhindert, während ACBs für eine Datei geöffnet sind. Wenn Sie jedoch die Wiederherstellungsattribute ändern, beachten Sie die möglichen Auswirkungen auf die Datenintegrität.

CICS erstellt eine Kopie der Wiederherstellungsattribute für eine Datei aus dem ICF-Katalog bei der ersten Öffnung für eine Aktualisierung in einer Abfolge von Anforderungen zum Öffnen für eine Datei. Dies bedeutet, dass eine einzelne CICS-Region nicht durch eine Aktualisierung der Wiederherstellungsattribute beeinflusst wird. Wird jedoch eine Datei im RLS-Modus geöffnet und werden die Attribute im ICF-Katalog geändert, dann kann eine zweite CICS-Region die gleiche Datei zur Aktualisierung öffnen und eine andere Attributgruppe kopieren. Hierdurch kann es zu einer Beeinträchtigung der Datenintegrität kommen.

Wenn Sie die Wiederherstellungsattribute ändern müssen, die im ICF-Katalog definiert wurden (z. B. zum Ändern des Namens des Protokolldatenstroms für die aktualisierende Wiederherstellung), dann müssen Sie die Datei in den Wartemodus versetzen, bevor Sie Änderungen durchführen. Dadurch wird sichergestellt, dass die Datei nicht im RLS-Modus verwendet werden kann, bis die Änderungen abgeschlossen sind und der Wartestatus der Datei wieder aufgehoben wird.

#### **GROUP**(*gruppenname*)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

#### **INDEXBUFFERS**(**{1|zahl}**)

Gibt die Anzahl der Puffer an, die für den Index verwendet werden sollen. Verwenden Sie einen Wert im Bereich zwischen 1 (Standardwert) und 32767. Der Minimalwert, den Sie angeben können, entspricht der Anzahl der Zeichenfolgen, die im Attribut **STRINGS** definiert wurde.

#### **JNLADD**(**{NONE|BEFORE|AFTER|ALL}**)

Gibt die Hinzufügeoperationen an, die in dem Journal aufgezeichnet werden sollen, das im Attribut **JOURNAL** angegeben ist. Folgende Werte sind zulässig:

##### **AFTER**

Fügt die Schreiboperation für die Dateisteuerung nach der VSAM-E/A-Operation zum Journal hinzu.

##### **ALL**

Fügt die Schreiboperation für die Dateisteuerung sowohl vor als auch nach Abschluss der VSAM-E/A-Operation zum Journal hinzu.

##### **BEFORE**

Fügt die Schreiboperation für die Dateisteuerung vor der VSAM-E/A-Operation zum Journal hinzu.

### NONE

Fügt keine Schreiboperationen zum Journal hinzu.

### **JNLREAD**(({NONE|UPDATEONLY|READONLY|ALL}))

Gibt die Leseoperationen an, die in dem Journal aufgezeichnet werden sollen, das im Attribut JOURNAL angegeben ist. Folgende Werte sind zulässig:

**ALL** Fügt alle Leseoperationen zum Journal hinzu.

### NONE

Fügt keine Leseoperationen zum Journal hinzu.

### **READONLY**

Fügt nur READ ONLY-Operationen (nicht READ UPDATE-Operationen) zum Journal hinzu.

### **UPDATEONLY**

Fügt nur READ UPDATE-Operationen (nicht READ ONLY-Operationen) zum Journal hinzu.

### **JNLSYNCREAD**(({NO|YES}))

Gibt an, ob die Datensätze für die automatische Journalführung, die für READ-Operationen in das in JOURNAL angegebene Journal geschrieben werden, synchron oder asynchron geschrieben werden sollen.

### **JNLSYNCWRITE**(({YES|NO}))

Gibt an, ob die Datensätze für die automatische Journalführung, die für WRITE-Operationen in das in JOURNAL angegebene Journal geschrieben werden, synchron oder asynchron geschrieben werden sollen.

### **JNLUPDATE**(({NO|YES}))

Gibt an, ob die REWRITE- und DELETE-Operationen in dem Journal aufgezeichnet werden sollen, das im Attribut JOURNAL angegeben wird.

### **JOURNAL**(({NO|*journal*}))

Gibt an, ob die automatische Journalführung für diese Datei verwendet werden soll. Die im Journal aufgezeichneten Daten haben das Format des VSAM-Datensatzes und werden für die vom Benutzer gesteuerte Journalführung verwendet.

Die im Journal aufzuzeichnenden Daten werden durch die Attribute JNLADD, JNLREAD, JNLSYNCREAD, JNLSYNCWRITE und JNLUPDATE identifiziert.

Bei einer von CICS verwalteten Datentabelle wird die Journalführung nur für Anforderungen ausgeführt, die zu VSAM-E/A-Anforderungen führen.

Bei einer benutzerverwalteten Datentabelle oder einer Coupling-Facility-Datentabelle wird für Dateisteuerungsoperationen keine Journalführung ausgeführt. Obwohl die automatische Journalführung für diese Tabellen nicht unterstützt wird, versucht CICS, wenn Sie eine Journalnummer angeben, den Protokolldatenstrom für das angegebene Journal zu öffnen, wenn die Datei geöffnet wird.

**Anmerkung:** Die automatische Journalführung arbeitet unabhängig von der Protokollierung im System und zu den Protokollen für die aktualisierende Wiederherstellung, die in den Attributen RECOVERY und FWDRECOVLOG angegeben werden.

Folgende Werte sind zulässig:

**NO** Für diese Datei wird keine automatische Journalführung ausgeführt.

**zahl** Die Nummer, die das Journal angibt, das von CICS als automatisches

Journal verwendet werden soll. CICS-Journalnamen haben das Format DFHJ $nm$ . Hierbei steht  $nm$  für einen Wert im Bereich zwischen 01 und 99.

**Anmerkung:** In CICS Transaction Server for z/OS stellt DFHJ01 nicht das Systemprotokoll dar.

#### **KEYLENGTH**(*wert*)

Gibt die Länge des logischen Schlüssels von Datensätzen in fernen Dateien in Byte und in Coupling-Facility-Datentabellen an, die mit LOAD(NO) angegeben wurden.

Wenn KEYLENGTH hier nicht definiert ist, muss die Option KEYLENGTH in den Dateisteuerbefehlen in den Anwendungsprogrammen angegeben werden, die auf diese Datei verweisen. Wenn KEYLENGTH hier nicht definiert ist und im Anwendungsprogramm nicht angegeben wurde und wenn der Schlüssel länger als vier Zeichen ist, dann ist der Standardwert 4.

#### **Ferne Dateien**

Der Bereich für die Schlüssellänge liegt zwischen 1 und 255.

#### **Coupling-Facility-Datentabellen**

Der Bereich für die Schlüssellänge liegt zwischen 1 und 16. Die Schlüssellänge ist nur erforderlich, wenn LOAD(NO) angegeben wurde.

Sie können optional eine Schlüssellänge für Coupling-Facility-Datentabellen angeben, die mit LOAD(YES) angegeben wurden. In diesem Fall müssen Sie Folgendes beachten:

- Die Schlüssellänge wird aus dem ICF-Katalogeintrag für die Datei abgerufen, aus der die Tabelle geladen wird. Wenn Sie eine Schlüssellänge angeben, muss die Schlüssellänge mit der für die Quelldatei angegebenen Schlüssellänge übereinstimmen. Andernfalls schlägt der Versuch, die Datei zu öffnen, mit einer Fehlermeldung fehl.
- Wenn CICS beim Öffnen der Datei feststellt, dass die CFDT bereits erstellt wurde und dass sich die Schlüssellänge von der beim Laden der Datei verwendeten Länge unterscheidet, schlägt das Öffnen fehl.

Wenn Sie eine Schlüssellänge für eine Datei angeben, die keine ferne Datei ist oder die nicht auf eine CFDT verweist, hat dies keine Auswirkungen, es sei denn, die Datei wird als ferne Datei oder als Referenz auf eine CFDT neu definiert. Wenn Sie eine Schlüssellänge angeben, dann müssen Sie beachten, dass der von einem Befehl INQUIRE FILE zurückgegebene Wert wie folgt lautet:

- Wenn die Datei geöffnet ist, dann gibt CICS den Wert zurück, der von VSAM abgerufen wurde. Dieser kann sich von dem Wert unterscheiden, der in der Dateidefinition angegeben wurde.
- Wenn die Datei geschlossen ist, dann gibt CICS den Wert zurück, der in der Dateidefinition angegeben wurde.

Der Wert für dieses Attribut muss im gesamten Sysplex in allen Dateidefinitionen, die auf dieselbe Coupling-Facility-Datentabelle verweisen, identisch sein.

#### **LOAD**( {NO|YES} )

Gibt an, ob die Coupling-Facility-Datentabelle aus einer Quelldatei geladen werden soll, wenn sie zum ersten Mal geöffnet wird.

**NO**     Gibt an, dass die Coupling-Facility-Datentabelle nicht aus einer Quel-



lendatei geladen werden muss. Sie kann von Anwendungsprogrammen in vollem Umfang genutzt werden, sobald sie geöffnet wurde. Die Tabelle wird von den Anwendungsprogrammen, die sie verwenden, geladen. Dies ist die Standardmethode für eine Coupling-Facility-Datentabelle.

**YES** Gibt an, dass die Coupling-Facility-Datentabelle aus einer Quellendatei geladen werden muss, bevor sie in vollem Umfang genutzt werden kann. Für die Anwendungsprogramme, die diese Coupling-Facility-Datentabelle benutzen, ist es erforderlich, dass die Tabelle die Datensätze aus einer Quellendatei enthält. Das Laden muss nicht abgeschlossen sein, bevor auf die Daten zugegriffen werden kann.

Dieses Attribut ist nur für Dateien sinnvoll, die mit dem Attribut TABLE(CF) definiert wurden. Sie können das Attribut LOAD für eine Datei angeben, die nicht mit TABLE(CF) definiert ist, diese Angabe wird jedoch von CICS ignoriert. (Von CICS verwaltete und benutzerverwaltete Tabellen werden immer automatisch aus einer Quellendatei geladen.) Wenn Sie dann die Dateidefinition so ändern, dass sie auf eine Coupling-Facility-Datentabelle verweist, tritt der im Attribut LOAD angegebene Wert in Kraft.

Vergewissern Sie sich, dass der Wert für dieses Attribut im gesamten Sysplex in allen Dateidefinitionen, die auf dieselbe Coupling-Facility-Datentabelle verweisen, identisch ist.

Weitere Informationen zur Verwendung dieses Attributs finden Sie in Coupling-Facility-Datentabellen.

#### **LSRPOOLID({1|2|3|4|5|6|7|8|NONE})**

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt.

Der für LSRPOOLID in vorhandenen Definitionen angegebene Wert wird an die neue Option LSRPOOLNUM übertragen.

#### **LSRPOOLNUM({1|zahl|NONE})**

Gibt die Identität des lokalen gemeinsam genutzten Ressourcenpools an. Der Standardwert für LSRPOOLNUM ist 1, es sei denn, es wurde ein Wert für das Attribut NSRGROUP angegeben. In diesem Fall lautet der Standardwert für LSRPOOLNUM NONE.

#### **NONE**

Gibt an, dass die Datei, die dieser Datei zugeordnet ist, nicht gemeinsam genutzte VSAM-Ressourcen (NSR = Nonshared Resources) verwendet.

Für eine gemeinsam genutzte CICS-Datentabelle (von CICS verwaltet oder benutzerverwaltet) kann NONE nicht angegeben werden, weil diese Typen von Datentabellen einen LSR-Pool verwenden müssen. Diese Einschränkung gilt jedoch nicht für eine Coupling-Facility-Datentabelle, für die Sie NONE angeben können.

Nicht gemeinsam genutzte VSAM-Ressourcen (NSR) werden für Transaktionen, die die Transaktionsisolation verwenden, nicht unterstützt. Geben Sie ISOLATE(NO) an, wenn Sie Transaktionen definieren, die auf VSAM-Dateien zugreifen, die NSR verwenden. Sie können die Dateianforderung auch über eine Funktionsverlagerung an eine ferne Region senden. Das Programm DFHMIRS, das die Anforderung ausführt, wird mit der EXECKEY-Angabe CICS definiert. Ein CICS-Schlüsselprogramm hat Lese- und Schreibzugriff auf den CICS-Schlüsselspeicher und den Benutzerschlüsselspeicher der eigenen Task und aller anderen Tasks. Dies gilt unabhängig davon, ob die Transaktionsisolation aktiv ist.

*zahl*

Gibt die Nummer des gemeinsam genutzten VSAM-Ressourcenpools an, der von der VSAM-Datei verwendet wird, die dieser Datei zugeordnet ist. Der Wert muss im Bereich zwischen 1 und 255 liegen. Die Datei ist zur Verwendung von lokalen gemeinsam genutzten VSAM-Ressourcen (LSR = Local Shared Resources) definiert. Definieren Sie die Puffer, Zeichenfolgen und anderen Ressourcen explizit in einer LSRPOOL-Ressourcendefinition, die dem zugeordneten LRPOOLNUM-Wert entspricht.

Wenn in der Dateidefinition RLSACCESS(YES) angegeben ist, wird der LSRPOOLNUM-Wert standardmäßig ignoriert, wenn CICS die Datei öffnet. Wenn Sie jedoch in einer Dateidefinition, in der ein LSR-Pool angegeben ist, die Einstellung RLSACCESS(NO) in RLSACCESS(YES) ändern, dann sollte der LSRPOOLNUM-Wert beibehalten werden. LSRPOOLNUM stellt sicher, dass die Datei korrekt auf einen LSR-Pool verweist, wenn vom RLS- in den LSR-Modus gewechselt wird.

#### **MAXNUMRECS ( {NOLIMIT | *zahl* } )**

Gibt die maximale Anzahl der Datensätze (Einträge) an, die in die Datentabelle aufgenommen werden sollen. Sie können dieses Attribut verwenden, um zu verhindern, dass eine nicht mehr steuerbare Transaktion folgende Ressourcen belegt:

- Den gesamten Speicher im Server-Pool, wenn die Tabelle eine Coupling-Facility-Datentabelle ist.
- Den gesamten Speicher im MVS-Datenspeicherbereich, wenn die Tabelle eine von CICS verwaltete oder benutzerverwaltete Tabelle ist.

Dieses Attribut ist nur für Dateien sinnvoll, bei denen für das Attribut TABLE entweder CICS, USER oder CF definiert wurde. Sie können MAXNUMRECS für eine Datei angeben, die mit TABLE(NO) definiert wurde. Dies hat jedoch keine Auswirkungen. Wenn Sie dann die Dateidefinition so ändern, dass sie auf eine Datentabelle verweist, tritt der im Attribut MAXNUMRECS angegebene Wert in Kraft.

#### **NOLIMIT**

Für die Anzahl der Datensätze, die in der Tabelle gespeichert werden können, gibt es keine benutzerdefinierte Begrenzung. In CICS besteht eine Begrenzung von 2.147.483.647, dem maximalen Vollwortwert.

*zahl*

Gibt die maximal zulässige Anzahl von Datensätzen in der Tabelle im Bereich von 1 bis 99999999 an. Wenn Sie einen Grenzwert für eine wiederherstellbare Coupling-Facility-Datentabelle festlegen, geben Sie einen Wert an, der etwa 5 bis 10 % höher als die maximale Anzahl von Datensätzen ist, die die Tabelle enthalten soll. Dies ermöglicht die Erstellung zusätzlicher Datensätze, die intern für die Verarbeitung wiederherstellbarer Anforderungen verwendet werden können. Die für diese interne Verarbeitung zulässige Spanne ist von der Nutzungsstufe der Coupling-Facility-Datentabelle und von der Art dieser Nutzung abhängig. Eine Auswirkung dieser internen Verarbeitung besteht darin, dass die NOSPACE-Bedingung mit dem RESP2-Wert 102 für WRITE- oder REWRITE-Anforderungen an eine wiederherstellbare Coupling-Facility-Datentabelle abgesetzt werden kann, die offensichtlich weniger Datensätze umfasst, als im MAXNUMRECS-Grenzwert angegeben wurde.

#### **NSRGROUP (*gruppe*)**

Gibt einen symbolischen Namen (mit maximal acht Zeichen) an, um Dateidefinitionen zu gruppieren, die sich auf dieselbe VSAM-Basisdatei beziehen. Der

Wert ist rein symbolisch und muss nicht auf eine bestimmte Dateidefinition verweisen. Es ist lediglich erforderlich, dass alle Dateidefinitionen, die gruppiert werden müssen, den gleichen Namen haben. Sie müssen dieses Attribut nicht angeben, um die korrekte Verarbeitung zu gewährleisten. Wenn Sie es jedoch nicht angeben, kann dies negative Auswirkungen auf die Leistung Ihres Systems haben.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Das Attribut NSRGROUP wirkt sich nur auf Dateien aus, die auf Dateien verweisen, die mit nicht gemeinsam genutzten VSAM-Ressourcen arbeiten. Das Attribut NSRGROUP darf für eine Datentabelle nicht angegeben werden. Es steht in Beziehung zu dem VSAM-Konzept gemeinsam genutzter Dateinamen, durch das VSAM veranlasst wird, eine einzelne Steuerblockstruktur für die Zeichenfolgen und Puffer zu erstellen, die von allen Dateien benötigt werden, die derselben Basisdatei zugeordnet sind.

Wenn das erste Mitglied einer solchen Gruppe von Dateien geöffnet wird, muss die Gesamtzahl der Zeichenfolgen, die für alle Dateieinträge in der Gruppe zugeordnet werden sollen, in VSAM (mit dem Wert für BSTRNO im Zugriffssteuerungsblock) angegeben werden. Die VSAM-Steuerblockstruktur wird dieses Mal unabhängig davon erstellt, ob die erste zu öffnende Datei einem Pfad oder einer Basisdatei zugeordnet ist. Der Wert von BSTRNO wird zu diesem Zeitpunkt berechnet, indem die STRINGS-Werte in allen Dateidefinitionen mit demselben Attribut NSRGROUP hinzugefügt werden. Nachdem die erste Datei in der Gruppe geöffnet wurde, wirken sich alle neu zu der Gruppe hinzugefügten Dateien nicht mehr auf die bereits erstellte VSAM-Steuerblockstruktur aus. Diese Vorgehensweise ändert sich nur dann, wenn alle Dateien, die für den Basiscluster geöffnet wurden, geschlossen und anschließend erneut geöffnet werden würden.

Die gemeinsame Nutzung von Dateinamen wird von CICS als Standardeinstellung für alle VSAM-Datei erzwungen. Die gemeinsame Nutzung von Dateinamen ist nicht aktiviert, wenn eine Datei zur schreibgeschützten Verarbeitung mit DSNSHARING=MODIFYREQS geöffnet wird. Eine Datei mit der Einstellung DSNSHARING=MODIFYREQS wird allerdings weiterhin bei der BSTRNO-Berechnung berücksichtigt.

Wenn eine Datei mit nicht gemeinsam genutzten VSAM-Ressourcen arbeitet und wenn Sie das Attribut NSRGROUP nicht angeben, dann wird die VSAM-Steuerblockstruktur möglicherweise mit einer unzureichenden Anzahl von Zeichenfolgen für die spätere Verarbeitung erstellt. In diesem Fall ruft VSAM die Funktion zum dynamischen Hinzufügen von Zeichenfolgen auf, um die zusätzlichen Steuerblöcke für die Zeichenfolgen bereitzustellen, da sie erforderlich sind. Dieser Mechanismus arbeitet jedoch nicht effizient und der zusätzlich benötigte Speicher wird erst wieder freigegeben, wenn das Ende der CICS-Ausführung erreicht ist.

Für Dateien, für die die Verwendung von gemeinsam genutzten lokalen VSAM-Ressourcen angegeben wurde (LSRPOOLNUM=n, wobei n für einen Wert zwischen 1 und 255 steht), hat die Angabe von NSRGROUP keine Auswirkungen.

Abb. 1 zeigt ein Beispiel für die Vorgehensweise beim Angeben der erforderlichen Dateisteuerungsdefinition für eine VSAM-Basisdatei und einen alternativen Indexpfad.

```

CEDA DEFINE FILE(VSAM10B) GROUP(XXXXXX)
      DSNAME(DTGCAT.VSAM10B)
      DISPOSITION(SHARE) ADD(YES)
      BROWSE(YES) DELETE(YES) READ(YES)
      UPDATE(NO) RECORDFORMAT(F)
      STRINGS(8) LSRPOOLNUM(NONE)
      RECOVERY(NONE) NSRGROUP(GROUP1)
      INDEXBUFFERS(8) DATABUFFERS(9)
CEDA DEFINE FILE(VSAM10P) GROUP(XXXXXX)
      DSNAME(DTGCAT.VSAM10P)
      LSRPOOLNUM(NONE) DISPOSITION(SHARE)
      STRINGS(5) NSRGROUP(GROUP1)
      BROWSE(YES) DELETE(NO) READ(YES)
      ADD(NO) UPDATE(NO) RECORDFORMAT(F)
      RECOVERY(NONE) INDEXBUFFERS(5)
      DATABUFFERS(6)

```

Abbildung 1. Definition einer VSAM-Basisdatei und eines alternativen Indexpfads.

#### **OPENTIME({FIRSTREF|STARTUP})**

Gibt an, wann die Datei geöffnet wird. Folgende Werte sind zulässig:

##### **FIRSTREF**

Die Datei bleibt so lange geschlossen, bis eine Anforderung zum Öffnen der Datei mit einer der folgenden Methoden abgesetzt wird:

- Mit einem Hauptterminalbefehl.
- Mit einem Befehl EXEC CICS SET FILE OPEN in einem Anwendungsprogramm.
- Mit einer impliziten Anweisung zum Öffnen.

##### **STARTUP**

Die Datei wird sofort nach der CICS-Initialisierung durch eine automatisch eingeleitete CICS-Transaktion (CSFU) geöffnet, es sei denn, der Status der Datei lautet UNENABLED, wenn die Datei geschlossen bleibt.

#### **READ({YES|NO})**

Gibt an, ob Datensätze in dieser Datei gelesen werden können.

#### **READINTEG({UNCOMMITTED|CONSISTENT|REPEATABLE})**

Gibt die Stufe der Leseintegrität an, die für Dateien erforderlich ist, die mit RL-SUCCESS(YES) definiert wurden. Die Leseintegrität gilt nicht für Dateien, die sich nicht im RLS-Zugriffsmodus befinden, gemeinsam genutzte CICS-Datentabellen oder Coupling-Facility-Datentabellen.

Mit READINTEG können Sie eine Standardstufe der Leseintegrität für eine Datei festlegen. Die Standardstufe der Leseintegrität wird von Programmen verwendet, in denen keine der API-Leseintegritätsoptionen UNCOMMITTED, CONSISTENT oder REPEATABLE in dem Befehl READ, READNEXT oder READPREV angegeben wird. Wenn ein Anwendungsprogramm allerdings eine dieser Angaben explizit verwendet, um die Leseintegrität anzugeben, dann überschreibt die Option API jeden Wert, der in diesem Attribut READINTEG angegeben ist.

**Anmerkung:** Sie können die Leseintegritätsoptionen nur in den CICS-API-Befehlen für die Dateisteuerung oder in CICS-Dateiressourcendefinitionen ange-

ben. Die Verwendung des funktional entsprechenden Parameters in der Datendefinitionsanweisung für mit CICS geöffnete Dateien ist nicht möglich.

Sie können CONSISTENT oder REPEATABLE in einer Dateiressourcendefinition angeben, um die Leseintegrität für Programme verfügbar zu machen, die geschrieben wurden, bevor diese Optionen in der API verfügbar waren. Auf diese Weise sind keine Änderungen an solchen Programmen notwendig. In diesem Fall müssen Sie beachten, dass die Durchsetzung konsistenter oder wiederholbarer Leseoperationen unerwartete Deadlocks zur Folge haben kann. In Programmen kann auch die Bedingung LOCKED auftreten.

### **CONSISTENT**

Der Datensatz wird mit konsistenter Leseintegrität gelesen. Wenn der Datensatz von einer anderen Transaktion geändert wird, wartet die READ-Anforderung, bis die Aktualisierung abgeschlossen ist, deren Ablaufsteuerung davon abhängt, ob die Datei wiederherstellbar oder nicht wiederherstellbar ist:

- Für eine wiederherstellbare Datei wird die READ-Anforderung abgeschlossen, nachdem die Aktualisierungstransaktion den nächsten Synchronisationspunkt oder Rollback beendet hat.
- Für eine nicht wiederherstellbare Datei wird die READ-Operation abgeschlossen, sobald die VSAM-Anforderung, die die Aktualisierung ausführt, abgeschlossen ist.

CONSISTENT ist nur dann zulässig, wenn Sie auch RLSACCESS(YES) angeben. Die Ressourcendefinition wird mit einem Fehler zurückgewiesen, wenn Sie CONSISTENT für eine Nicht-RLS-Datei angeben.

### **REPEATABLE**

Der Datensatz wird mit wiederholbarer Leseintegrität gelesen. Wenn der Datensatz von einer anderen Transaktion geändert wird, wartet die READ-Anforderung, bis die Aktualisierung abgeschlossen ist, deren Ablaufsteuerung davon abhängt, ob die Datei wiederherstellbar oder nicht wiederherstellbar ist:

- Für eine wiederherstellbare Datei wird die READ-Anforderung abgeschlossen, nachdem die Aktualisierungstransaktion den nächsten Synchronisationspunkt oder Rollback beendet hat.
- Für eine nicht wiederherstellbare Datei wird die READ-Operation abgeschlossen, sobald die VSAM-Anforderung, die die Aktualisierung ausführt, abgeschlossen ist.

Nach Abschluss der Leseoperation wird eine gemeinsam genutzte Sperre beibehalten, bis der nächste Synchronisationspunkt erreicht wird. Dadurch wird sichergestellt, dass keiner der Datensätze, die in einer Arbeitseinheit gelesen werden, geändert werden können, während die Task weitere Leseanforderungen absetzt. Fehlerantworten wie z. B. NOTFND sind möglicherweise nicht wiederholbar.

REPEATABLE ist nur dann zulässig, wenn Sie auch RLSACCESS(YES) angeben. Die Ressourcendefinition wird mit einem Fehler zurückgewiesen, wenn Sie REPEATABLE für eine Nicht-RLS-Datei angeben.

### **UNCOMMITTED**

Der Datensatz wird ohne Leseintegrität gelesen. CICS ruft den aktuellen Wert des Datensatzes ab, der in VSAM bekannt ist. Es wird kein Versuch unternommen, diese Leseanforderung mit einer gleichzeitigen Aktualisierungsaktivität für denselben Datensatz zu serialisieren. Der zurückgegebene Datensatz kann eine Version beinhalten, die von einer

anderen Transaktion aktualisiert wurde, aber noch nicht festgeschrieben wurde. Dieser Datensatz kann sich ändern, wenn die Aktualisierung anschließend zurückgesetzt wird.

**Anmerkung:**

1. UNCOMMITTED gibt dieselbe Stufe der Integrität an, die durch die Releases von CICS bereitgestellt wird, die das Attribut READ-INTEG nicht unterstützen.
2. Geben Sie für einen beliebigen Datentabellentyp UNCOMMITTED an. Alle anderen Werte als UNCOMMITTED sind zulässig, wenn RLSACCESS(YES) angegeben ist, sie werden jedoch ignoriert, wenn für die Datei auch TABLE(CF), TABLE(CICS) oder TABLE(USER) angegeben ist.

**RECORDFORMAT({V|F})**

Gibt das Format der Datensätze in der Datei an.

**F** Die Datensätze haben eine feste Länge. Geben Sie für VSAM-Dateien diese Option nur an, wenn die Servicedefinition der VSAM-Zugriffsmethode Datensätze mit fester Größe (d. h. mit einer durchschnittlichen Größe gleich der maximalen Größe) angibt und alle Datensätze in der Datei diese Größe haben.

F ist für benutzerverwaltete Datentabellen und Coupling-Facility-Datentabellen ungültig.

**V** Die Datensätze haben eine variable Länge. Alle benutzerverwalteten Datentabellen und Coupling-Facility-Datentabellen müssen mit variabler Länge angegeben werden. Andernfalls gibt CICS eine Fehlermeldung zurück, die besagt, dass RECORDFORMAT(F) zu einem Konflikt mit der Option TABLE(CF) oder TABLE(USER) führt und ignoriert wird.

**RECORDSIZE(zahl)**

Gibt die maximale Länge in Byte von Datensätzen in einer fernen Datei oder einer Coupling-Facility-Datentabelle an. Die angegebene Größe kann im Bereich zwischen 1 und 32767 liegen.

**Nur für Coupling-Facility-Datentabellen**

Dieser Wert ist erforderlich, wenn in der Dateidefinition für die Tabelle LOAD(NO) angegeben ist.

Sie können dieses Attribut auch angeben, wenn LOAD(YES) festgelegt ist (z. B. zur Vereinfachung eines Wechsels der Dateidefinition zwischen LOAD(NO) und LOAD(YES)). Wenn Sie LOAD(YES) angeben, muss der Wert für die Datensatzgröße jedoch mit dem entsprechenden Wert für die Quelldatei übereinstimmen. Andernfalls kann CICS die Tabelle nicht öffnen. Es gibt drei Bedingungen, die es CICS ermöglichen, aufgrund einer falschen Datensatzgröße bei Verwendung von LOAD(YES) einen Fehler festzustellen:

1. Vor dem Öffnen der Tabelle überprüft CICS, ob die in VSAM definierte Datensatzgröße der Datei, aus der die Coupling-Facility-Datentabelle geladen werden soll, identisch mit der Größenangabe (sofern vorhanden) in der Dateidefinition ist. Weichen die Angaben für die Datensatzgröße ab, gibt CICS die Fehlermeldung DFHFC7081 aus.
2. Die Datensatzgröße (sofern angegeben) in der Dateidefinition ist identisch mit der entsprechenden Angabe für die Datei in VSAM,

beim Öffnen der Tabelle stellt CICS jedoch fest, dass die Tabelle bereits mit Daten geladen wurde, die eine andere Datensatzgröße aufweisen. Dies ist möglicherweise darauf zurückzuführen, dass die Daten aus einer Datei als der in dieser Dateidefinition angegebenen Datei geladen wurden. In diesem Fall gibt CICS die Fehlermeldung DFHFC7082 zurück.

3. Die Dateidefinition für die zu öffnende Tabelle gibt eine Datensatzgröße an, jedoch keinen Dateinamen, da die Tabelle durch das Öffnen einer anderen Datei geladen werden soll. Wenn die Tabelle bereits erstellt wurde, schlägt das Öffnen einer Datei, die eine andere Datensatzgröße angibt, mit der Nachricht DFHFC7083 fehl.

Zur Vermeidung dieser Fehler sollten Sie sich vergewissern, dass der Wert für dieses Attribut im gesamten Sysplex in allen Dateidefinitionen, die auf dieselbe Coupling-Facility-Datentabelle verweisen, identisch ist. Andernfalls sollten Sie das Attribut für Dateien, für die LOAD(YES) festgelegt ist, nicht angeben.

Wenn Sie eine Datensatzgröße für eine Datei angeben, die keine ferne Datei ist oder die nicht auf eine CFDT verweist, hat dies keine Auswirkungen, es sei denn, die Datei wird als ferne Datei oder mit einem Verweis auf eine Coupling-Facility-Datentabelle neu definiert. Wenn Sie eine Datensatzgröße angeben, dann müssen Sie beachten, dass der von einem Befehl INQUIRE FILE zurückgegebene Wert wie folgt lautet:

- Wenn die Datei geöffnet ist, dann gibt CICS den Wert zurück, der von VSAM abgerufen wurde. Dieser kann sich von dem Wert unterscheiden, der in der Dateidefinition angegeben wurde.
- Wenn die Datei geschlossen ist, dann gibt CICS den Wert zurück, der in der Dateidefinition angegeben wurde.

**Anmerkung:** Bei Coupling-Facility-Datentabellen erzielen Sie einen erheblichen Leistungszuwachs, wenn Sie die Datensatzgröße auf maximal 63 Byte begrenzen können. Dies ist auf die Art und Weise der Speicherung von Datensätzen in der Coupling-Facility zurückzuführen.

#### **RECOVERY({NONE|BACKOUTONLY|ALL})**

Gibt den Typ der Wiederherstellung an, die für die Datei erforderlich ist.

Dieses Attribut wird nicht für Dateien verwendet, die mit RLSACCESS(YES) definiert wurden, und auch nicht, wenn die Wiederherstellungsoptionen im ICF-Katalog definiert sind. Wenn im ICF-Katalog LOG definiert ist, ignoriert CICS die Option RECOVERY und verwendet den LOG-Wert aus dem ICF-Katalog auch für Dateien, für die RLSACCESS(NO) definiert wurde. Wurde im ICF-Katalog LOG(ALL) angegeben, dann übernimmt CICS auch die Werte für LOGSTREAMID und BWO aus dem ICF-Katalog. Um in CICS die Verwendung dieses Attributs anstelle der Wiederherstellungsoptionen im Katalog zu erzwingen, müssen Sie für den Systeminitialisierungsparameter **NONRLSRECOV** die Option FILEDEF festlegen.

Für Dateien, auf die im RLS-Modus zugegriffen wird, müssen Sie die Wiederherstellungsparameter mit der Dateidefinition im ICF-Katalog angeben. Weitere Informationen hierzu finden Sie in Defining files as recoverable resources.

Bei Coupling-Facility-Datentabellen und benutzerverwalteten Tabellen, die mit einer Quellendatei definiert sind, werden alle Wiederherstellungsattribute im ICF-Katalog ignoriert. Die Wiederherstellungsattribute sind eine Eigenschaft der Datei und nicht des zugeordneten Datasets.

Für Coupling-Facility-Datentabellen muss der Wert für das Wiederherstellungsattribut im gesamten Sysplex in allen Dateidefinitionen, die auf dieselbe Coupling-Facility-Datentabelle verweisen, identisch sein.

**ALL** Eine Ausnahme bilden die Coupling-Facility-Datentabellen, die die eigene Wiederherstellung verwalten und die Services des Protokollmanagers oder des Wiederherstellungsmanagers nicht verwenden. Die vor der Wiederherstellung erstellten Images (Vorimages) werden im Systemprotokoll aufgezeichnet, die nach der Wiederherstellung erstellten Images (Nachimages) werden in dem Journal aufgezeichnet, das im Attribut FWDRECOVLOG angegeben ist.

Datensätze, die in FWDRECOVLOG geschrieben werden, sind unabhängig von automatischen Journalführungsoptionen, die möglicherweise festgelegt wurden.

RECOVERY=ALL bietet zusammen mit FWDRECOVLOG die Möglichkeit, die Anforderungen eines Dienstprogramms für die aktualisierende Wiederherstellung von den Anforderungen der automatischen Journalführung zu trennen. Zusätzliche Informationen, die über die automatische Journalführung nicht zur Verfügung stehen, werden im FWDRECOVLOG-Protokoll aufgezeichnet. Die Angabe RECOVERY=ALL in Kombination mit FWDRECOVLOG stellt die bevorzugte Methode zur Bereitstellung der Unterstützung für die aktualisierenden Wiederherstellung dar.

Bereits vorhandene Dienstprogramme für die aktualisierende Wiederherstellung, die mit den Makroeinstellungen JREQ=(WU,WN) und JID=FCT arbeiten, können mit diesen Einstellungen weiterhin benutzt werden. Die funktional entsprechenden RDO-Angaben dieser Einstellungen für die automatische Journalführung sind JNLADD=BEFORE, JNLUPDATE=YES und das Attribut JOURNAL.

Bei von CICS verwalteten Datentabellen werden die Datentabelle und ihre Quellendatei protokolliert, im Journal aufgezeichnet und zusammen wiederhergestellt.

Bei benutzerverwalteten Tabellen hat die Angabe von ALL dieselbe Wirkung wie die Angabe von BACKOUTONLY: Es wird nur ein dynamisches Zurücksetzen (Backout) bereitgestellt. Für benutzerverwaltete Tabellen wird die aktualisierende Wiederherstellung nicht unterstützt.

Für Coupling-Facility-Datentabellen können Sie ALL nicht angeben.

**Anmerkung:** Wird ALL für VSAM-ESDS-Dateien angegeben, kann CICS ADD-Operationen nicht zurücksetzen. Zur Beseitigung dieser Situation müssen Sie den Benutzerexit XFCLDEL so codieren, dass die Datei nicht aufgrund des Fehlers geschlossen wird.

#### **BACKOUTONLY**

Eine Ausnahme bilden die Coupling-Facility-Datentabellen, die die eigene Wiederherstellung verwalten und die Services des Protokollmanagers oder des Wiederherstellungsmanagers nicht verwenden. Die vor der Wiederherstellung erstellten Images (Vorimages) werden im Systemprotokoll aufgezeichnet.

Bei von CICS verwalteten Datentabellen gibt BACKOUTONLY an, dass die Datentabelle und die zugehörige Quellendatei wiederhergestellt werden können. Sie werden beide in einem Schritt aktualisiert und ggf. auch in einem Schritt wiederhergestellt.



Bei benutzerverwalteten Tabellen gibt BACKOUTONLY nur ein dynamisches Zurücksetzen an. Es werden keine Protokolldatensätze geschrieben. Aus diesem Grund gibt es bei einem Wiederanlauf nach einem Systemabsturz keine Wiederherstellung.

Für Coupling-Facility-Datentabellen ist BACKOUTONLY nur dann zulässig, wenn die Coupling-Facility-Datentabelle mit UPDATEMODEL(LOCKING) definiert wurde. Sie dürfen dieses Attribut für UPDATEMODEL(CONTENTION) nicht angeben. Die Angabe von BACKOUTONLY impliziert, dass eine Coupling-Facility-Datentabelle die Möglichkeit zur UOW-Wiederherstellung bietet. Dies bedeutet, dass in einer Arbeitseinheit an der CFDT durchgeführte Aktualisierungen zurückgesetzt werden, wenn die Arbeitseinheit fehlschlägt. Dies ist auch der Fall, wenn in CICS oder auf dem CFDT-Server ein Fehler auftritt, während die Arbeitseinheit gerade ausgeführt wird, oder wenn in MVS ein Fehler auftritt.

**Anmerkung:** Wird BACKOUTONLY für VSAM-ESDS-Dateien angegeben, kann CICS ADD-Operationen nicht zurücksetzen. Zur Beseitigung dieser Situation müssen Sie den Benutzerexit XFCLDEL so codieren, dass die Datei nicht aufgrund des Fehlers geschlossen wird.

#### NONE

Für diese Datei gibt es keine Wiederherstellungsprotokollierung.

#### **REMOTENAME**(*datei*)

Gibt an, ob die Datei sich auf einem fernen System befindet. Außerdem gibt diese Option den Namen an, unter dem die Datei im System oder in der Region bekannt ist, in dem bzw. der sie sich befindet. Der Name kann bis zu acht Zeichen lang sein. Wird REMOTENAME nicht angegeben, wird der im Attribut FILE festgelegte Name verwendet.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Wenn Sie einen fernen Namen angeben, verwendet CICSplex SM diesen Namen bei der Zuordnung der Datei zu einem zugehörigen System. Wenn Sie ein fernes System, aber keinen fernen Namen angeben, wird der lokale Name (also der Name dieser Dateidefinition) sowohl für das Ziel als auch auf den zugehörigen Systemen verwendet.

#### **REMOTESYSTEM**(*verbindung*)

Gibt den Namen der Verbindung zu dem fernen System an, auf dem sich die Datei befindet.

Für IPIC-Verbindungen gibt REMOTESYSTEM die ersten vier Zeichen des IPCONN-Namens in der IPCONN-Definition an, die zur Verfügung steht und angefordert werden kann.

Für MRO- und APPC-Verbindungen gibt REMOTESYSTEM den CONNECTION-Namen in der CONNECTION-Definition an. Wenn Sie REMOTESYSTEM angeben, können Sie auch einen Wert für REMOTENAME angeben, um den Namen der Datei auf dem fernen System festzulegen.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

**Anmerkung:** Wenn Sie eine Ressourcendefinition ändern und anstelle von RLSACCESS(NO) die Einstellung RLSACCESS(YES) angeben, müssen Sie den Namen des fernen Systems entfernen. Andernfalls setzt CICS die Funktionsverlagerung von Dateianforderungen fort.

**RESSECCNUM**

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt. Weitere Informationen hierzu finden Sie in *Obsolete attributes*.

**RLSACCESS({NO|YES})**

Gibt an, ob CICS die Datei im RLS-Modus öffnen soll.

**NO** Die Datei soll nicht im RLS-Modus geöffnet werden. Wenn Sie RLSACCESS(NO) angeben oder den Standardwert übernehmen, dann öffnet CICS die Datei abhängig von der Angabe im Attribut LSRPOOLNUM im LSR- oder NSR-Zugriffsmodus. Wenn Sie außerdem LSRPOOLNUM(NONE) angeben, dann lautet der Zugriffsmodus NSR. Geben Sie LSRPOOLNUM(*zahl*) an, dann wird der LSR-Zugriffsmodus verwendet.

**YES** Die Datei soll im RLS-Modus geöffnet werden. Wenn Sie RLSACCESS(YES) angeben, hat diese Angabe Vorrang vor dem Attribut LSRPOOLNUM, das beim Öffnen der Datei ignoriert wird.

Durch die Angabe von RLSACCESS(YES) werden die Auswirkungen bestimmter anderer Attribute geändert, die in der FILE-Definition definiert werden (siehe hierzu Tabelle 1).

*Tabelle 1. Auswirkungen von RLSACCESS(YES) auf andere FILE-Attribute*

Attribut	Auswirkung von RLSACCESS(YES)
PASSWORD	Wird ignoriert.
LSRPOOLNUM	Wird ignoriert.
DSNSHARING	Wird ignoriert.
STRINGS	Wird ignoriert. Dateien mit dem RLS-Zugriffsmodus weisen immer 1024 Zeichenfolgen auf.
REMOTESYSTEM REMOTENAME RECORDSIZE KEYLENGTH	Die Bedeutung dieser Attribute bleibt unverändert. Doppelt vorhandene FILE-Definitionen für lokale und ferne Systeme sind jedoch für Dateien im RLS-Zugriffsmodus von geringerem Wert. Bei RLS sind möglicherweise zahlreiche FORs (File-Ownning Regions; Dateiverwaltungsregionen) anstatt nur einer vorhanden und einer lokalen CICS-Region stehen mehrere FORs zur Auswahl zur Verfügung.
DATABUFFERS INDEXBUFFERS	Wird ignoriert.

Tabelle 1. Auswirkungen von RLSACCESS(YES) auf andere FILE-Attribute (Forts.)

Attribut	Auswirkung von RLSACCESS(YES)
TABLE	Die Angabe von TABLE(CICS) ist nicht zulässig. Wenn Sie diese Option angeben, dann wird eine Fehlermeldung ausgegeben. TABLE(USER) oder TABLE(CF) ist hingegen zulässig. Auf die Quelldatei (sofern vorhanden) wird im RLS-Modus zugegriffen, um die Datentabelle zu laden. Anschließend greifen Anforderungen direkt über die Datentabellenservices auf die Datentabelle zu.
RECOVERY	Wird ignoriert. Das Wiederherstellungsattribut wird für eine RLS-Datei aus dem ICF-Katalog abgerufen.
FWDRCOVLOG	Wird ignoriert. Der Name des Protokolldatenstroms für die aktualisierende Wiederherstellung wird für eine RLS-Datei aus dem ICF-Katalog abgerufen.
BACKUPTYPE	Wird ignoriert. Der Sicherungstyp für RLS wird anhand des Sicherungsdienstprogramms DFSMSdss ermittelt.

**Anmerkung:**

1. Wenn eine Datei im RLS-Modus geöffnet wird, dann werden alle für die Attribute PASSWORD, LSRPOOLNUM, DSNSHARING, STRINGS, DATA-BUFFERS und INDEXBUFFERS angegebenen Werte ignoriert (siehe hierzu Tabelle 1 auf Seite 72). Wenn Sie hingegen den Befehl CEMT oder EXEC CICS SET FILE verwenden, um den Wert für RLSACCESS von YES in NO zu ändern, dann werden diese Werte nicht weiter ignoriert und CICS verwendet sie, wenn die Datei geschlossen und anschließend im Nicht-RLS-Modus erneut geöffnet wird.
2. CICS verwendet immer den RLS-Zugriffsmodus aus der Dateiressourcendefinition. Sie können diese Einstellung nicht mit dem Parameter RLS=NRI oder RLS=CR in einer Datendefinitionsanweisung überschreiben.

**STATUS({ENABLED|DISABLED|UNENABLED})**

Gibt den Anfangsstatus der Datei nach einer CICS-Initialisierung mit START=COLD oder START=INITIAL an. Sie können den Status einer geschlossenen Datei mit der Hauptterminaltransaktion CEMT ändern. Der Status einer Datei (ENABLED, DISABLED oder UNENABLED) nach einem Neustart von CICS wird auf den Status beim vorherigen Systemabschluss zurückgesetzt.

**DISABLED**

Jede Anforderung für diese Datei, die von einem Anwendungsprogramm auf Befehlsebene stammt, führt dazu, dass die Bedingung DISABLED an das Programm übergeben wird.

**ENABLED**

Für diese Datei ist eine normale Verarbeitung zulässig.

**UNENABLED**

Diese Einstellung verhindert, dass die Datei von einer impliziten Öffnungsanweisung geöffnet wird, die von einem Anwendungsprogramm stammt. Wird versucht, auf die Datei zuzugreifen, dann gibt das System die Bedingung NOTOPEN aus. Im Gegensatz hierzu ändert eine explizite Anforderung zum Öffnen der Datei (z. B. ein CEMT- oder EXEC-CICS-Befehl SET FILE OPEN) den Status in ENABLED, bevor versucht wird, die Datei zu öffnen.

Dieses Attribut wird für FILE-Ressourcen ignoriert, die dynamisch von einem CICS-Bundle generiert werden. Der Anfangsstatus einer FILE-Ressource wird aus dem Anfangsstatus des Bundles abgeleitet, das die Ressource definiert.

### STRINGS({1|wert})

Gibt die Anzahl (im Bereich zwischen 1 und 255) gleichzeitiger Anforderungen an, die für die Datei verarbeitet werden können. Wenn die Anzahl der Anforderungen den angegebenen Wert erreicht, werden alle weiteren Anforderungen von CICS in die Warteschlange eingestellt, bis eine der aktiven Anforderungen beendet wird. Dies gilt sowohl für Dateien, die gemeinsam genutzte Ressourcen verwenden, als auch für Dateien, die keine gemeinsam genutzten Ressourcen verwenden. Wenn in der Dateidefinition RLSACCESS(YES) angegeben ist, dann sollten Sie beachten, dass der STRINGS-Wert ignoriert wird. Bei Verwendung des RLS-Zugriffsmodus erhalten Sie immer 1024 Zeichenfolgen.

Für Dateien, die lokale gemeinsam genutzte Ressourcen verwenden, wird diese Anzahl von VSAM nicht verwendet. Sie wird von CICS verwendet, und zwar nicht nur in der hier beschriebenen Weise, sondern auch zur Berechnung des Standardwerts in der Pufferpooldefinition.

#### Hinweise:

1. Wenn Sie einen STRINGS-Wert auswählen, beachten Sie, dass ein bestimmter Anteil (20 %) der angegebenen Anzahl von Zeichenfolgen von CICS für die Verwendung in schreibgeschützten Anforderungen reserviert wird.
2. Wenn Sie einen STRINGS-Wert für eine Datei in Zugangsfolge (ESDS = Entry-Sequenced Data Set) auswählen, sollten Sie Folgendes beachten:
  - Wenn eine ESDS als 'add-only'-Datei (d. h. ausschließlich im Schreibmodus) verwendet wird, dann müssen Sie die Zeichenfolgennummer 1 benutzen. Alle Zeichenfolgennummern, die größer als 1 sind, können sich erheblich auf die Leistung des Systems auswirken, weil es zu Konflikten bezüglich der exklusiven Steuerung kommen kann, die auftreten, wenn mehrere Tasks versuchen, gleichzeitig in die ESDS zu schreiben.
  - Wenn eine ESDS sowohl für Schreib- als auch für Leseoperationen verwendet wird, wobei 80 % der Aktivitäten auf Schreiboperationen entfallen, sollten Sie zwei Dateidefinitionen definieren, wobei eine der Dateien zum Schreiben und die andere zum Lesen verwendet wird.
3. Für benutzerverwaltete Datentabellen und Coupling-Facility-Datentabellen beschränkt der STRINGS-Wert *nicht* die Anzahl der gleichzeitigen Anforderungen für die Tabelle. Allerdings begrenzt der Wert die Anzahl der gleichzeitigen Anforderungen während des Ladens einer benutzerverwalteten Tabelle.
4. Bei mit CICS verwalteten Datentabellen beschränkt der STRINGS-Wert die Anzahl der gleichzeitigen Anforderungen zum Aktualisieren der Tabelle. Die Anzahl der gleichzeitigen Leseanforderungen wird nicht begrenzt.

### TABLE({NO|CICS|USER|CF})

Gibt den Typ der erforderlichen Datentabelle an.

**CF** Eine Coupling-Facility-Datentabelle (CFDT). Diese bleibt unabhängig von der zugehörigen Quellendatei. Änderungen an der Tabelle werden nicht in der entsprechenden Quellendatei (sofern vorhanden) nachvollzogen. Eine Quellendatei ist für eine CFDT optional und wird mit LOAD(YES) in der Dateidefinition angegeben.

Wenn Sie CF festlegen, dann geben Sie auch Folgendes an:

- CFDTPOOL zur Angabe des Namens des Coupling-Facility-Pools, in dem sich die Tabelle befindet.
- LOAD zur Angabe, ob die Tabelle aus einer Quellendatei geladen werden soll (oder ob der Standardwert NO verwendet werden soll).

- UPDATEMODEL zur Angabe, ob die Tabelle das CONTENTION- oder LOCKING-Aktualisierungsmodell verwenden soll (oder ob der Standardwert LOCKING verwendet werden soll).
- RECORDFORMAT als V (oder Standardwert V).
- MAXNUMRECS mit dem erforderlichen Wert.

Eine Coupling-Facility-Datentabelle erfordert einen Coupling-Facility-Datentabellenserver. Informationen zur Vorgehensweise beim Starten eines CFDT-Servers finden Sie in Server für Coupling-Facility-Datentabellen einrichten und ausführen.

**CICS** Eine von CICS verwaltete Datentabelle. Diese spiegelt automatisch alle Änderungen wider, die an der Tabelle in der zugehörigen Quellendatei vorgenommen wurden. Wenn Sie CICS festlegen, dann geben Sie auch Folgendes an:

- LSRPOOLNUM mit einem Wert zwischen 1 und 255.
- MAXNUMRECS mit dem erforderlichen Wert.

**NO** Die Datentabelle ist nicht erforderlich.

**USER** Eine benutzerverwaltete Tabelle. Diese bleibt unabhängig von der zugehörigen Quellendatei. Änderungen an der benutzerverwalteten Tabelle werden nicht in der entsprechenden Quellendatei nachvollzogen. Wenn Sie USER festlegen, dann geben Sie auch Folgendes an:

- LSRPOOLNUM mit einem Wert zwischen 1 und 255.
- RECORDFORMAT als VARIABLE (oder Standardwert VARIABLE).
- MAXNUMRECS mit dem erforderlichen Wert.

#### **TABLENAME**(*cfdt*)

Gibt den Namen der Coupling-Facility-Datentabelle an, auf die über diese Dateidefinition zugegriffen wird. Der Name kann bis zu acht Zeichen lang sein.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Wenn Sie dieses Attribut nicht angeben, wenn TABLE(CF) angegeben wurde, wird standardmäßig der für FILE angegebene Name verwendet. Damit CICS-Regionen eine Coupling-Facility-Datentabelle gemeinsam nutzen können, müssen die Dateidefinitionen, die in jeder Region installiert sind, denselben CFDT-POOL-Namen und TABLENAME-Wert (oder FILE-Namen, wenn TABLENAME nicht verwendet wird) angeben. Der Wert für TABLENAME muss nur innerhalb seines Pools eindeutig sein.

Beachten Sie, dass der Tabellename nicht nur eine Kennung für die Tabelle ist, sondern auch als Ressourcenname in Sicherheitsprüfungen verwendet wird.

Dieses Attribut ist nur für Dateien sinnvoll, die mit dem Attribut TABLE(CF) definiert wurden. Sie können einen Tabellennamen für eine Datei angeben, die nicht mit TABLE(CF) definiert ist, diese Angabe wird jedoch von CICS ignoriert. Wenn Sie dann die Dateidefinition so ändern, dass sie auf eine Coupling-Facility-Datentabelle verweist, tritt der im Attribut TABLENAME angegebene Wert in Kraft.

#### **UPDATE**(**{NO|YES}**)

Gibt an, ob Datensätze in dieser Datei aktualisiert werden können.

## UPDATEMODEL({LOCKING|CONTENTION})

Gibt den Typ des Aktualisierungsmodells an, das für eine Coupling-Facility-Datentabelle verwendet werden soll.

### LOCKING

Gibt an, dass die CFDT mit dem Sperrmodell aktualisiert wird. Dies bedeutet, dass Datensätze gesperrt werden, wenn sie zum Aktualisieren gelesen werden, sodass sie erst dann von anderen Arbeitseinheiten geändert werden können, wenn die Aktualisierungsanforderung abgeschlossen ist. Für wiederherstellbare Tabellen wird die Aktualisierungsanforderung am Synchronisationspunkt abgeschlossen. Für nicht wiederherstellbare Tabellen wird die Aktualisierungsanforderung abgeschlossen, wenn der Befehl REWRITE, DELETE oder UNLOCK ausgeführt wird. LOCKING ist der Standardwert für eine Datei, für die TABLE(CF) angegeben wurde. Mit dem LOCKING-Modell kann die CFDT wie folgt definiert werden:

- **Nicht wiederherstellbar** bedeutet, dass CFDT-Aktualisierungen nicht zurückgesetzt werden, wenn eine Arbeitseinheit fehlschlägt und dass die Sperren nur für die Dauer einer Anforderung gehalten werden. Sie geben an, dass eine CFDT nicht wiederherstellbar ist, indem Sie RECOVERY(NONE) festlegen.
- **Wiederherstellbar** oder UOW-wiederherstellbar bedeutet, dass in einer Arbeitseinheit an der CFDT durchgeführte Aktualisierungen zurückgesetzt werden, wenn die Arbeitseinheit fehlschlägt. Dies ist auch der Fall, wenn in CICS oder auf dem CFDT-Server ein Fehler auftritt, während die Arbeitseinheit gerade ausgeführt wird, oder wenn in MVS ein Fehler auftritt. Sie geben an, dass eine CFDT wiederherstellbar ist, indem Sie RECOVERY(BACKOUTONLY) festlegen.

Eine wiederherstellbare CFDT, die das Sperrmodell verwendet, ist wie eine wiederherstellbare Datei oder ein wiederherstellbares Dataset. Eine Ausnahme bildet hierbei lediglich die Tatsache, dass sie einen Fehler der Coupling-Facility, in der sie gespeichert ist, nicht übersteht. Für eine Coupling-Facility-Datentabelle kann keine aktualisierende Wiederherstellung durchgeführt werden.

### CONTENTION

Gibt an, dass die CFDT mit dem Konfliktmodell aktualisiert wird. Dies bedeutet, dass Datensätze nicht gesperrt werden, wenn sie zum Aktualisieren gelesen werden. Bei einer nachfolgenden REWRITE- oder DELETE-Anforderung wird ein Fehler zurückgegeben, wenn der Satz durch eine andere Task geändert oder gelöscht wurde, nachdem er für die Aktualisierung gelesen wurde. Die CFDT muss nicht wiederherstellbar sein (RECOVERY(NONE)). Dies bedeutet, dass Aktualisierungen nicht zurückgesetzt werden, wenn eine Arbeitseinheit fehlschlägt.

Der Wert für dieses Attribut muss im gesamten Sysplex in allen Dateidefinitionen, die auf dieselbe Coupling-Facility-Datentabelle verweisen, identisch sein.

Dieses Attribut ist nur für Dateien sinnvoll, die mit dem Attribut TABLE(CF) definiert wurden. Sie können das Aktualisierungsmodell für eine Datei angeben, die nicht mit TABLE(CF) definiert ist, diese Angabe wird jedoch von CICS ignoriert. Wenn Sie dann die Dateidefinition so ändern, dass sie auf eine Coupling-Facility-Datentabelle verweist, tritt der im Attribut UPDATEMODEL angegebene Wert in Kraft.

## IPCONN-Ressourcen

Eine IPCONN-Ressource definiert eine TCP/IP-Kommunikationsverbindung (TCP/IP = Transmission Control Protocol/Internet Protocol) zu einem fernen System. Diese Kommunikationsverbindung wird auch als IPIC-Verbindung bezeichnet.

Einige der Attribute für eingehende Elemente der IPIC-Verbindung werden in der TCPIPService-Definition angegeben, die in der Option TCPIPService der IP-CONN-Definition angegeben ist.

Der in einer PROGRAM-Definition für REMOTESYSTEM angegebene Name kann über den IPCONN-Namen auf eine IPCONN-Definition verweisen. Dieses Attribut wird für die Verbindung zu verteilten Programmen verwendet.

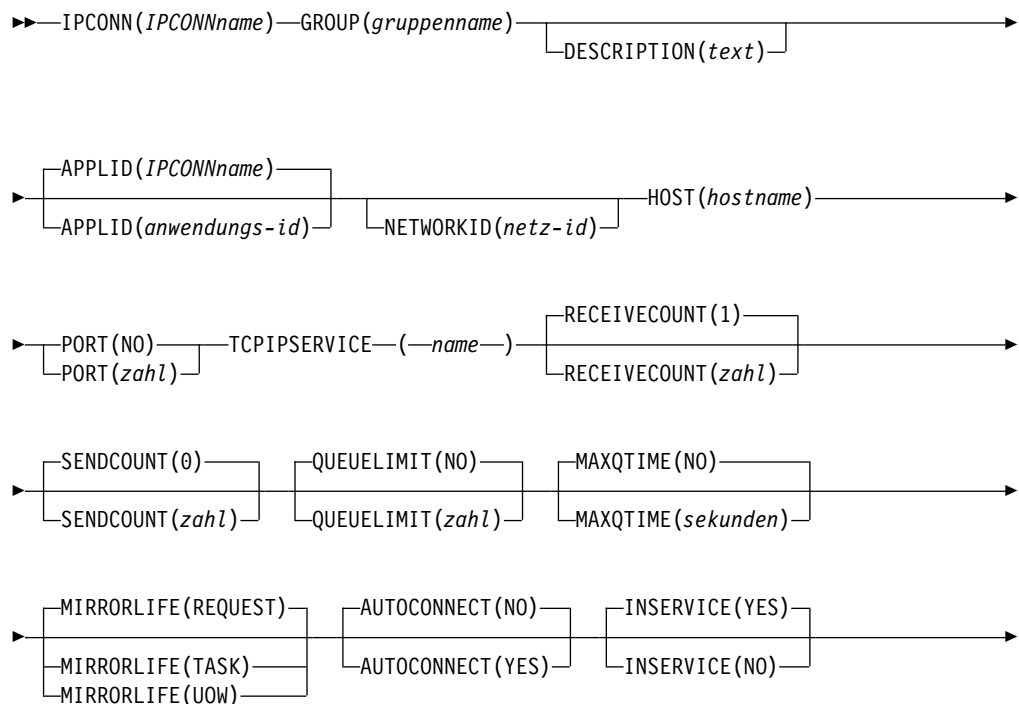
Anweisungen zum Definieren von IPCONN-Ressourcen finden Sie in [Defining IPIC connections](#).

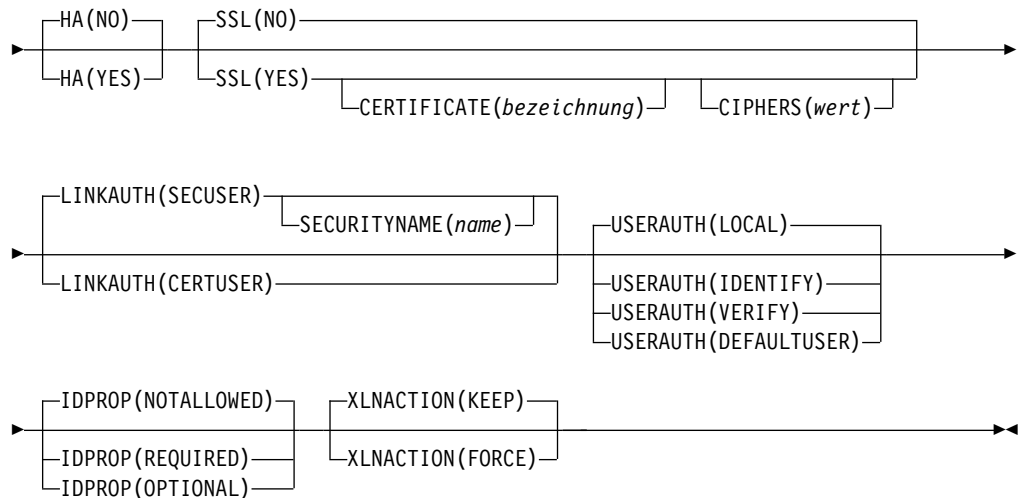
Informationen zur Installation von IPCONN-Ressourcendefinitionen mit CEMT und CEDA finden Sie in *Installing IPCONN resource definitions*. Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in *Working with BAS IPCONN resource definitions*.

## IPCONN-Attribute

Beschreibt die Syntax und die Attribute der IPCONN-Ressource.

## Syntax





## Attribute

**APPLID** ({ IPCONNname | *anwendungs-id* })

Gibt die Anwendungs-ID (APPLID) des fernen Systems an. Wenn es sich bei dem fernen System um eine CICS-Region handelt, dann wird die zugehörige Anwendungs-ID im Parameter APPLID definiert. Der Wert für *anwendungs-id* kann bis zu acht Zeichen lang sein und muss mit einem alphabetischen Zeichen beginnen.

### Zulässige Zeichen:

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Geben Sie für Verbindungen zur CICS-Region der erweiterten Wiederherstellungsfunktion (XRF = Extended Recovery Facility) die generische Anwendungs-ID (APPLID) der fernen Region an.

Wird für APPLID kein Wert angegeben, dann verwendet CICS den IPCONN-Namen.

Wird IPCONN zur Herstellung einer Verbindung zu einem HA-Cluster verwendet, dann muss dieses Feld den aus einem bis acht Zeichen bestehenden Wert enthalten, mit dem der Cluster identifiziert wird. Dieser Wert dient zur Unterscheidung dieser IPCONN von allen anderen installierten IPCONNn in der Region. Der Name einer Anwendungs-ID (APPLID), die auf eine bestimmte CICS-Region verweist, darf nicht mit dem Namen eines HA-Clusters übereinstimmen.

Für doppelte APPLIDs gelten die folgenden Regeln:

- Es ist nicht zulässig, zwei oder mehr IPCONN-Definitionen zu installieren, in denen die gleiche Anwendungs-ID (APPLID) *und* die gleiche Netz-ID (NETWORKID) angegeben ist. Die Kombination von APPLID und NETWORKID kann verwendet werden, um sicherzustellen, dass Systeme innerhalb eines Netzes eindeutig benannt werden. Weitere Informationen hierzu finden Sie in der Beschreibung zur Option NETWORKID.



- Sie *können* eine IPCONN-Definition installieren, in der die gleichen Werte für APPLID und NETNAME für eine installierte MRO-, APPC- oder LUTY-PE6.1-CONNECTION-Definition angegeben sind.
- Wenn eine installierte IPCONN-Definition denselben Namen wie eine installierte CONNECTION-Definition aufweist, muss die Anwendungs-ID (APPLID) der IPCONN-Definition mit dem Netznamen (NETNAME) der CONNECTION-Definition übereinstimmen. Andernfalls wird abhängig von der jeweiligen Situation eine der folgenden Nachrichten ausgegeben:
  - DFHIS3009 - Wenn der Fehler während der automatischen IPCONN-Installation festgestellt wird.
  - DFHAM4913 - Wenn der Fehler während der IPCONN-Installation festgestellt wird.
  - DFHZC6312 - Wenn der Fehler während der CONNECTION-Installation oder der automatischen Installation festgestellt wird.

Die IPCONN-Definition hat Vorrang gegenüber der CONNECTION-Definition. Dies bedeutet, dass CICS bei Vorhandensein einer IPCONN- und einer CONNECTION-Definition mit demselben Namen die IPCONN-Verbindung verwendet.

- Eine CONNECTION-Definition und eine IPCONN-Definition mit demselben Wert für NETNAME und APPLID müssen nicht denselben Namen haben, damit für die Kommunikation über TCP/IP eine andere System-ID (*system-id*) verwendet werden kann, anstatt die CICS-Standardeinstellung zu benutzen und alle unterstützten Funktionen über die IPCONN-Verbindung weiterzuleiten, sofern diese existiert.

Diese Regeln werden während der Installation geprüft.

#### **AUTOCONNECT({NO|YES})**

Gibt an, ob Sitzungen eingerichtet werden sollen, wenn die IPCONN-Definition installiert wird (z. B. während der CICS-Initialisierung, wenn Sie einen nachfolgenden Befehl CEDA INSTALL angeben oder den Befehl CEMT oder EXEC CICS SET TCPIP OPEN verwenden, um die Kommunikation mit TCP/IP zu starten). Wenn die Verbindung zu diesem Zeitpunkt nicht hergestellt werden kann, weil das ferne System nicht verfügbar ist, dann können Sie die Verbindung anschließend anfordern, indem Sie den Befehl CEMT oder EXEC CICS SET IPCONN(*name*) INSERVICE ACQUIRED verwenden. Dies gilt nicht, wenn das ferne System in der Zwischenzeit verfügbar wird und die Kommunikation einleitet.

**NO** CICS versucht nicht, Sitzungen einzurichten, wenn IPCONN installiert wird.

**YES** CICS versucht, Sitzungen einzurichten, wenn IPCONN installiert wird.

Zur Erreichung der Konnektivität bei der Installation der IPCONN-Definition sollten Sie die folgenden Bedingungen berücksichtigen:

1. Die TCPIPSERVICE-Definition, die in der Option TCPIPSERVICE dieser IPCONN-Definition angegeben ist, muss auch in dieser Region installiert werden und über die Angabe PROTOCOL(IPIC) verfügen.
2. Zusammengehörige IPCONN- und TCPIPSERVICE-Definitionen müssen in der fernen Region installiert werden:
  - a. Die Option HOST der IPCONN-Definition in der fernen Region muss diese Region angeben.
  - b. Die Option PORT der IPCONN-Definition in der fernen Region muss die gleiche Portnummer angeben, die in der Option

PORTNUMBER der lokalen TCPIPService-Definition angegeben ist, die in dieser IPCONN definiert wurde.

- c. Die TCPIPService-Definition in der fernen Region (angegeben in der IPCONN-Definition der fernen Region) muss die Angabe PROTOCOL(IPIC) enthalten und außerdem in der Option PORTNUMBER die gleiche Portnummer, die auch in der Option PORT dieser IPCONN angegeben ist.

Die Angabe AUTOCONNECT(YES) ist nicht zulässig, wenn PORT(NO) angegeben wurde.

#### **CERTIFICATE**(*bezeichnung*)

Gibt die Bezeichnung eines X.509-Zertifikats an, das während des SSL-Handshakes als Clientzertifikat verwendet werden soll, wenn die IPIC-Verbindung angefordert wird und wenn die TCPIPService-Ressource, die in der HOST- und PORT-Definition festgelegt ist, mit der Angabe SSL(CLIENTAUTH) definiert ist. Wird dieses Attribut nicht angegeben, dann wird das Standardzertifikat verwendet, das im Schlüsselring der Benutzer-ID der CICS-Region angegeben ist.

Zertifikatsbezeichnungen können bis zu 32 Byte lang sein.

Das Zertifikat muss in einem Schlüsselring in der Datenbank des externen Sicherheitsmanagers gespeichert werden. Weitere Informationen finden Sie in *Building a key ring manually*.

Wenn Sie dieses Attribut angeben, dann müssen Sie auch SSL(YES) festlegen.

#### **CIPHERS**(*wert*)

Das Attribut CIPHERS kann auf eine der beiden folgenden Arten angegeben werden:

- Eine aus maximal 56 Hexadezimalziffern bestehende Zeichenfolge, die als Liste mit bis zu 28 zweistelligen Cipher-Suite-Codes interpretiert wird.
- Der Name der Spezifikationsdatei für die SSL-Cipher-Suite, bei der es sich um eine z/OS UNIX-Datei im Unterverzeichnis security/ciphers des Verzeichnisses handelt, das im Systeminitialisierungsparameter **USSCONFIG** angegeben ist. Wenn **USSCONFIG** z. B. auf den Wert /var/cicsts/dfhconfig gesetzt wird und **CIPHERS** auf den Wert strongciphers.xml, dann lautet der vollständig qualifizierte Dateiname /var/cicsts/dfhconfig/security/ciphers/strongciphers.xml. Weitere Informationen hierzu finden Sie in *Creating an SSL cipher suite specification file*.

Wenn Sie die CEDA-Transaktion zum Definieren der Ressource verwenden, dann wird das Attribut von CICS automatisch mit einer Standardliste zulässiger Codes initialisiert. Damit CICS das Attribut initialisieren kann, muss der Systeminitialisierungsparameter KEYRING in der CICS-Region angegeben werden, in der CEDA ausgeführt wird. Wird KEYRING nicht angegeben, dann kann CICS das Attribut nicht initialisieren. Die Standardliste der Codes trägt die Bezeichnung 35363738392F303132330A1613100D15120F0C, es sei denn, der Systeminitialisierungsparameter **NISTSP800131A=CHECK** wird festgelegt. In diesem Fall lautet die Bezeichnung 35363738392F303132330A1613100D.

Sie können die Cipher-Codes anders sortieren oder aus der ursprünglichen Liste entfernen. Allerdings ist es nicht möglich, Cipher-Codes hinzuzufügen, die sich nicht in der Standardliste für die angegebene Verschlüsselungsstufe befinden. Zum Zurücksetzen des Werts auf die standardmäßige Codeliste müssen Sie alle Cipher-Suite-Codes löschen. Das Feld wird dann automatisch erneut mit der Standardliste gefüllt.

Weitere Informationen hierzu finden Sie in Cipher suites and cipher suite specification files.

#### **DESCRIPTION**(*text*)

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Es gelten keine Einschränkungen für die Zeichen, die Sie verwenden können. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für jede linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

#### **GROUP**(*gruppenname*)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

#### **HA**({**NO**|**YES**})

Gibt an, ob die IPCONN verwendet werden kann, um eine Verbindung zu einem Hochverfügbarkeitscluster herzustellen.

**NO** Dies ist der Standardwert. Die IPCONN kann nicht verwendet werden, um eine Verbindung zu einem Hochverfügbarkeitscluster herzustellen.

**YES** Die IPCONN muss eine Verbindung zu einer Region herstellen, die Teil eines Hochverfügbarkeitsclusters ist.

#### **HOST**(*hostname*)

Gibt den Hostnamen des fernen Systems oder seine IPv4- oder IPv6-Adresse an. Der Name kann bis zu 116 Zeichen lang sein. Sie können IPv4- und IPv6-Adressen in verschiedenen zulässigen Formaten angeben. Weitere Informationen zu den Adressformaten finden Sie in IP addresses.

Wenn Sie eine IPv6-Adresse (oder einen Hostnamen, der in eine IPv6-Adresse aufgelöst wird) angeben, dann müssen Sie sicherstellen, dass Sie in einer Dualmodusumgebung (IPv4 und IPv6) arbeiten und dass der Client oder der Server, mit dem Sie kommunizieren, ebenfalls in einer Dualmodusumgebung (IPv4 und IPv6) arbeitet. Weitere Informationen zu IPv6 finden Sie in Understanding IPv6 and CICS.

Das Attribut HOST darf nur alphanumerische Zeichen, Bindestriche (-), Doppelpunkte (:) oder Punkte (.) enthalten, obwohl die Verwendung von Doppelpunkten bei Angabe eines zeichenbasierten Hostnamens anstelle einer IP-Adresse nicht zulässig ist. CICS überprüft den Hostnamen während der Definition. Der Hostname kann in Großbuchstaben, Kleinbuchstaben oder in gemischter Groß-/Kleinschreibung eingegeben werden. Wenn jedoch anstelle einer IP-Adresse ein zeichenbasierter Hostname angegeben wird, wird dieser Hostname in der IPCONN-Definition in Kleinbuchstaben konvertiert.

HOST wird ignoriert, wenn für das Attribut SENDCOUNT der Wert null angegeben wurde. Bei HOST handelt es sich um ein erforderliches Attribut, wenn der Wert von SENDCOUNT größer als null ist.

**IDPROP({NOTALLOWED|OPTIONAL|REQUIRED})**

Gibt an, ob die verteilte Identität durch den Absender an das verbundene System übertragen wird. Die Angabe des Attributs IDPROP ist nur dann sinnvoll, wenn eine Verbindung zu einem Ziel außerhalb eines Sysplex hergestellt werden soll und wenn sie primär zur Verhinderung der Übertragung verteilter Identitäten zwischen Unternehmen dient. Bei Verbindungen zwischen Systemen innerhalb desselben Sysplex arbeitet die Verbindung in derselben Weise wie bei Angabe von IDPROP(OPTIONAL) und ignoriert alle anderen Einstellungen.

Verteilte Identitäten werden nur übertragen, wenn die verbundenen Systeme alle folgenden Kriterien erfüllen:

- Beide Systeme sind an der Weitergabe von Identitäten beteiligt. Weitere Informationen hierzu finden Sie in Support and requirements for identity propagation.
- Die Systeme befinden sich in demselben Sysplex oder werden über SSL verbunden.
- Für das empfangende System wurde in der IPCONN-Ressourcendefinition USERAUTH(IDENTIFY) angegeben.

**NOTALLOWED**

Eine Benutzer-ID, die der sendenden Transaktion zugeordnet ist, wird für Anforderungen gesendet, die diese Verbindung benutzen. NOTALLOWED ist der Standardwert.

**REQUIRED**

Für Anforderungen, die diese Verbindung verwenden, ist eine verteilte Identität erforderlich. Wurde REQUIRED angegeben, dann muss das empfangende System die Verwendung verteilter Identitäten unterstützen. Die Benutzer-ID, die der sendenden Transaktion zugeordnet ist, wird nicht gesendet.

**OPTIONAL**

Wenn eine verteilte Identität vorhanden ist, dann wird sie gesendet. Die Benutzer-ID, die der sendenden Transaktion zugeordnet ist, wird ebenfalls gesendet.

**INSERVICE({NO|YES})**

Gibt den Status der IPCONN-Ressource bei der Installation an.

**NO** Die Verbindung kann keine Nachrichten empfangen oder Ausgabedaten übertragen.

**YES** Die Verbindung steht für die Verwendung zur Verfügung.

**IPCONN(IPCONNname)**

Gibt den Namen dieser IPCONN-Definition an. Der Name kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Wenn diese IPCONN für eine DPL (Distributed Program Link; Verbindung zu verteilten Programmen) zwischen CICS TS 3.2 oder späteren Regionen, für das Transaktionsrouting zwischen CICS TS 4.1 oder späteren Regionen, für Anforderungen für die Dateisteuerung, für transiente Daten oder temporären Spei-

cher per Funktionsverlagerung zwischen CICS TS 4.2 oder späteren Regionen verwendet werden soll, die die IPIC-Konnektivität nutzen, dann muss der Name mit dem vier Zeichen umfassenden *lokalen Namen* (SYSID) übereinstimmen, unter dem das ferne System in CICS bekannt ist. Der Name muss mit vier nachfolgenden Leerzeichen aufgefüllt werden.

**Anmerkung:** Sie können den Namen (SYSID) der fernen Zielregion einer DPL-Anforderung wie folgt angeben:

- Mit der Option REMOTESYSTEM der installierten PROGRAM-Definition.
- Mit der Option SYSID des Befehls EXEC CICS LINK PROGRAM.
- Mit dem Programm für das dynamische Routing.

Der IPCONN-Name kann mit dem Namen einer installierten MRO- oder AP-PC-CONNECTION-Definition identisch sein.

#### **LINKAUTH({CERTUSER|SECUSER})**

Gibt an, wie die Benutzer-ID für die Verbindungssicherheit auf einem CICS-System mit initialisierter Sicherheitsfunktion (SEC=YES) eingerichtet wird.

##### **CERTUSER**

Die TCP/IP-Kommunikation mit dem Partnersystem muss für SSL konfiguriert werden und vom Partnersystem muss während des SSL-Handshakes ein Zertifikat empfangen werden.

Die IPCONN-Ressource muss auf eine TCPIPSERVICE-Ressource verweisen, die mit SSL(CLIENTAUTH) definiert ist.

Das empfangene Zertifikat muss im externen Sicherheitsmanager so definiert werden, dass es einer Benutzer-ID zugeordnet ist, die zum Einrichten der Verbindungssicherheit verwendet wird.

##### **SECUSER**

Gibt an, dass die Benutzer-ID, die im Attribut SECURITYNAME angegeben ist, zum Einrichten einer Verbindungssicherheit benutzt wird.

Der Standardwert ist LINKAUTH(SECUSER).

#### **MAXQTIME({NO|sekunden})**

Gibt die maximal zulässige Zeitdauer an, die in die Warteschlange eingestellte Zuordnungsanforderungen, die auf eine freie Sitzung für diese Verbindung warten, in der Warteschlange verbleiben können, bevor diese bereinigt wird.

##### **Anmerkung:**

- Die maximal zulässige Zeit in der Warteschlange wird nur benutzt, wenn in der Option QUEUELIMIT ein Grenzwert für die Länge der Warteschlange angegeben wird.
- Der Grenzwert für die Zeitdauer wird nur dann angewendet, wenn die Warteschlangenlänge den in QUEUELIMIT angegebenen Wert erreicht.

**NO** CICS verwaltet die Warteschlange für Zuordnungsanforderungen, die auf eine freie Sitzung warten. Für die Zeitdauer, für die Anforderungen in der Warteschlange verbleiben dürfen, wird kein Grenzwert festgelegt. Allerdings können für einzelne Anforderungen die DTIMOUT-Mechanismen gelten.

##### **sekunden**

Die näherungsweise berechnete, maximal zulässige Zeitdauer (in Sekunden), die Zuordnungsanforderungen, die auf eine freie Sitzung

warten, in der Warteschlange verbleiben können, wenn diese Verbindung nicht zu reagieren scheint. Der Wert für *sekunden* muss im Bereich zwischen 0 und 9999 liegen.

Wenn die Warteschlange der eingestellten Zuordnungsanforderungen die (im Attribut QUEUELIMIT angegebene) maximale Länge erreicht und dann für die Verbindung eine neue Zuordnungsanforderungen empfangen wird, dann wird die Warteschlange bereinigt und das System gibt die Nachricht DFHIS500 aus, wenn die Verarbeitungsgeschwindigkeit der Warteschlange erkennen lässt, dass die durchschnittliche Verarbeitungszeitdauer der neuen Zuordnung den Wert für die maximale Warteschlangenzeitdauer übersteigt. Wenn die Warteschlange bereinigt wird, dann gibt das System für die in der Warteschlange befindlichen Zuordnungsanforderungen einen Fehler des Typs SYSIDERR aus.

Es werden keine weiteren Anforderungen in die Warteschlange eingestellt, bis die Verbindung erfolgreich eine Sitzung freigeben konnte. Zu diesem Zeitpunkt gibt CICS die Nachricht DFHIS5001 aus und nimmt die normale Warteschlangensteuerung wieder auf.

#### **MIRRORLIFE ( {REQUEST|TASK|UOW} )**

Gibt die kürzeste Laufzeit der Spiegeltask für Anforderungen für die Dateisteuerung, für transiente Daten und für temporären Speicher per Funktionsverlagerung an, die von dieser Region empfangen werden. Normalerweise werden Spiegeltasks möglichst bald beendet, um die Anzahl der aktiven Tasks auf ein Minimum zu beschränken und um eine Sitzung nicht über einen längeren Zeitraum zu belegen. Für bestimmte Anwendungen ist es allerdings effizienter, sowohl die Spiegeltask als auch die Sitzung bis zum nächsten Synchronisationspunkt beizubehalten, obwohl diese Aufbewahrung aus Gründen der Datenintegrität nicht erforderlich ist. Eine Transaktion, die zahlreiche READ FILE-Anforderungen an ein fernes System absetzt, kann mit einer einzelnen Spiegeltask anstelle einer separaten Spiegeltask für jede Anforderung möglicherweise besser verarbeitet werden. Auf diese Weise können Sie den Aufwand für die Zuordnung von Sitzungen auf der Sendeseite und für das Herstellen einer Verbindung für die Spiegeltasks auf der Empfangsseite reduzieren. Der angegebene MIRRORLIFE-Wert wird in der Laufzeit der Spiegeltask erst dann wiedergegeben, wenn eine Anforderung für die Dateisteuerung, für transiente Daten oder temporären Speicher per Funktionsverlagerung verarbeitet wird.

#### **REQUEST**

Die Spiegeltask wird so bald wie möglich beendet. Für Arbeiten, die nicht wiederherstellbar sind und für die keine Spiegelung erforderlich ist, um so eine Sitzung in der Ressourcenverwaltungsregion (ROR = Resource-Owning Region) beizubehalten (z. B. eine READ FILE-Anforderung ohne Aktualisierung), wird diese Aktion nach der Verarbeitung der Anforderung durchgeführt. Für nicht wiederherstellbare Arbeiten, für die eine Spiegelung erforderlich ist, um eine Sitzung beizubehalten (z. B. eine Anforderung zum Durchsuchen von Dateien), wird diese Aktion ausgeführt, sobald die Notwendigkeit für die Aufrechterhaltung des Haltestatus entfällt (z. B. wenn ENDBR abgesetzt wird). Für wiederherstellbare Arbeiten bleibt die Spiegelung bis zum nächsten Synchronisationspunkt erhalten. Dies ist der Standardwert.

**TASK** Die Spiegeltask bleibt für die Anwendung verfügbar, die die ferne Anforderung absetzt, bis die Task der Anwendung beendet wird. Dieser Wert reduziert den Aufwand für die erneute Zuordnung einer Sitzung und die erneute Herstellung der Verbindung für die Spiegeltask für

eine nachfolgende Anforderung, die auf einen Benutzersynchronisationspunkt folgt. Geben Sie diesen Wert nicht an, wenn Tasks mit langer Laufzeit für Anforderungen für die Dateisteuerung, für transiente Daten oder temporären Speicher per Funktionsverlagerung verwendet werden, weil nach der Zuordnung der Sitzung zu einer bestimmten Task die Sitzung erst nach Abschluss der Task freigegeben wird, und zwar auch dann, wenn sie nicht verwendet wird. Die Sitzung steht nicht zur Verwendung durch andere Tasks zur Verfügung, bis sie freigegeben wird.

**UOW** Die Spiegeltransaktion bleibt für die Anwendung verfügbar, die die ferne Anforderung absetzt, bis der nächste Synchronisationspunkt ausgegeben wird. Dieser Wert führt zu einer Reduzierung des Aufwands für die erneute Einrichtung der Kommunikation mit der Spiegeltransaktion, wenn in der Anwendung weitere Anforderungen für die Dateisteuerung, für transiente Daten oder temporären Speicher per Funktionsverlagerung in dieser Arbeitseinheit vorhanden sind.

Dieser Parameter ist zulässig, wenn er in der IPCONN-Definition der Ressourcenverwaltungsregion angegeben wird. Geben Sie die Werte für TASK oder UOW mit diesem Parameter nur nach sorgfältiger Überlegung an. Dies gilt insbesondere dann, wenn DPL-Anforderungen mit SYNCONRETURN oder TRANSID verwendet werden.

#### **NETWORKID**(*netz-id*)

Gibt die Netz-ID des fernen Systems an. Die Netz-ID des fernen Systems entspricht entweder dem NETID-Wert von z/OS Communications Server oder (bei VTAM=NO-Systemen) dem Wert des Systeminitialisierungsparameters UOWNETQL.

Wenn NETWORKID nicht angegeben wird, dann geht CICS davon aus, dass das ferne System sich in demselben Netz wie das lokale System befindet. In diesem Fall verwendet CICS die Netz-ID (NETID) von z/OS Communications Server oder den Wert des Systeminitialisierungsparameters UOWNETQL dieses CICS-Systems. Hierbei handelt es sich um das CICS-System, auf dem diese Definition installiert ist.

Geben Sie NETWORKID an, wenn Sie eine Verbindung zu einem fernen System herstellen wollen, das sich in einem anderen Netz befindet und deshalb einen unterschiedlichen z/OS Communications Server-Wert für NETID oder UOWNETQL aufweist. In diesem Fall können mehrere ferne Systeme mit der gleichen Anwendungs-ID (APPLID) arbeiten. Obwohl die CICS-Anwendungs-IDs (APPLIDs) innerhalb eines Sysplex eindeutig sein müssen, kann es z. B. erforderlich sein, eine Verbindung zu einem System außerhalb des Sysplex oder in einem anderen Sysplex herzustellen. Die Kombination der Attribute APPLID und NETWORKID stellt sicher, dass auf das ferne System mit einem eindeutigen Namen verwiesen wird.

Der NETWORKID-Wert muss mit der Netz-ID des fernen Systems übereinstimmen.

Erfolgt hier keine Angabe, dann wird der NETWORKID-Wert abgeleitet, wenn die IPCONN-Ressource zum ersten Mal installiert wird. Er wird zwischen Warmstarts auch dann nicht geändert, wenn der lokale Wert für NETID sich ändert.

Der Name kann bis zu acht Zeichen lang sein und basiert auf den Regeln der Assemblersprache. Er muss mit einem alphabetischen Zeichen beginnen.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

**PORT(NO|zahl)**

Gibt die Dezimalzahl des Ports an, an dem die ferne Region empfangsbereit ist.

**NO** Diese IPCONN-Ressource wird nicht für abgehende Anforderungen verwendet (d. h., es sind keine Sendesitzungen vorhanden), wenn eine Verbindung von CICS Transaction Gateway hergestellt wird.

NO ist nicht zulässig für Verbindungen zwischen CICS- und CICS-IPCONN-Ressourcen.

Wird NO angegeben, dann wird für AUTOCONNECT die Einstellung NO erzwungen.

**zahl** Ein Wert im Bereich zwischen 1 und 65535. Die Portnummer wird mit dem HOST-Wert kombiniert, um das Ziel für abgehende Anforderungen auf dieser IPCONN-Ressource zu ermitteln.

**QUEUELIMIT({NO|zahl})**

Gibt die maximale Anzahl von Zuordnungsanforderungen an, die von CICS in die Warteschlange eingestellt werden soll, während auf freie Sitzungen gewartet wird:

**NO** Die Anzahl der Zuordnungsanforderungen, die von CICS in die Warteschlange eingestellt werden kann, während auf eine freie Sitzung gewartet wird, ist unbegrenzt.

**zahl** Die maximal zulässige Anzahl von Zuordnungsanforderungen (im Bereich zwischen 0 und 9999), die von CICS in die Warteschlange der Verbindung eingestellt werden kann, während auf eine freie Sitzung gewartet wird. Wenn die Anzahl der in die Warteschlange eingestellten Zuordnungsanforderungen den definierten Grenzwert erreicht, dann wird für nachfolgende Zuordnungsanforderungen der Fehler SYSIDERR ausgegeben, bis die Anzahl der Zuordnungsanforderungen in der Warteschlange wieder unter den Grenzwert fällt.

**RECEIVECOUNT({1|zahl})**

Gibt die Anzahl der Empfangssitzungen im Bereich zwischen 1 und 999 an. Hierbei handelt es sich um Sitzungen, die eingehende Anforderungen empfangen. Die Anzahl der verwendeten Empfangssitzungen ist auch von der Anzahl der Sendesitzungen abhängig, die auf dem fernen System definiert sind. Nach Herstellung der Verbindung werden diese Werte ausgetauscht und der niedrigere Wert wird verwendet.

**SECURITYNAME(name)**

Gibt den Sicherheitsnamen des fernen Systems an, der für die Verbindungssicherheit verwendet werden soll.

In einem CICS-System mit initialisierter Sicherheitsfunktion (SEC=YES), auf dem mit der Einstellung LINKAUTH(SECUSER) gearbeitet wird, wird der Sicherheitsname verwendet, um die Berechtigung des fernen Systems einzurichten.

Der Sicherheitsname muss eine gültige RACF-Benutzer-ID in dieser Region sein. Der Zugriff auf geschützte Ressourcen in dieser Region basiert auf dem



RACF-Benutzerprofil und seiner Gruppenzugehörigkeit. Wenn es sich bei dem während der IPCONN-Installation angegebenen Sicherheitsnamen nicht um eine gültige RACF-Benutzer-ID handelt, dann verwendet CICS für den Sicherheitsnamen die Standard-Benutzer-ID.

Der Standardwert ist LINKAUTH(SECUSER).

#### **SENDCOUNT({0|zahl})**

Gibt die Anzahl der Sendesitzungen im Bereich zwischen 0 und 999 an. Hierbei handelt es sich um Sitzungen, die abgehende Anforderungen senden. Die Anzahl der verwendeten Sendesitzungen ist auch von der Anzahl der Empfangssitzungen abhängig, die auf dem fernen System definiert sind. Nach Herstellung der Verbindung werden diese Werte ausgetauscht und der niedrigere Wert wird verwendet. Wird 0 angegeben, dann kann diese IPCONN nur eingehende Arbeiten verarbeiten. Das Senden von Anforderungen an das verbundene System ist hingegen nicht möglich.

Die Angabe von SENDCOUNT(0) ist für Verbindungen zwischen CICS- und CICS-IPCONN-Ressourcen nicht zulässig.

Wenn SENDCOUNT(0) angegeben wird, erzwingt dies die Verwendung von PORT(NO). Wird ein SENDCOUNT-Wert größer als null verwendet, dann muss für PORT ein numerischer Wert angegeben werden.

Der Versuch, eine IPCONN-Ressource anzufordern, für die SENDCOUNT(0) angegeben ist, schlägt fehl, weil kein Port definiert ist.

#### **SSL({NO|YES})**

Gibt an, ob zum Verschlüsseln der übertragenen Daten Secure Sockets Layer (SSL) verwendet wird. Beachten Sie hierbei, dass bei Verwendung von SSL für MAXSSLTCBs ein Wert angegeben werden sollte, der mindestens dem Doppelten der Anzahl der IPIC-Verbindungen entspricht, die SSL in der CICS-Region verwenden.

**NO** Secure Sockets Layer (SSL) wird nicht verwendet. Beim Anfordern der Verbindung werden keine Sicherheitsprüfungen durchgeführt. Für abgehende Nachrichten wird keine Verschlüsselung durchgeführt.

**YES** Wenn für den Systeminitialisierungsparameter SEC die Einstellung YES festgelegt ist, dann wird Secure Sockets Layer (SSL) verwendet. Wenn für die in den Attributen HOST und PORT angegebene TCPIPSERVICE-Ressource die Angabe SSL(CLIENTAUTH) definiert wurde, dann extrahiert CICS das Clientzertifikat, das im Attribut CERTIFICATE angegeben ist und verwendet dieses Zertifikat beim Anfordern der IPIC-Verbindung zum Partnersystem. Für alle Nachrichten, die über diese IPCONN-Ressource gesendet werden, wird die SSL-Verschlüsselung durchgeführt. Die Verschlüsselungsstufe ist von dem Wert abhängig, der in der Option CIPHERS angegeben wird.

#### **TCPIPSERVICE(name)**

Gibt den Namen einer TCPIPSERVICE-Definition an, die die Angabe PROTOCOL(IPIC) enthält, mit der die Attribute für die eingehende Verarbeitung für diese IPCONN-Ressource definiert werden.

#### **USERAUTH({LOCAL|IDENTIFY|VERIFY|DEFAULTUSER})**

Gibt an, wie die Benutzer-ID für die Benutzersicherheit bei der Verbindungsherstellung auf einem CICS-System mit initialisierter Sicherheitsfunktion (SEC=YES) eingerichtet wird.

##### **LOCAL**

CICS akzeptiert keine Benutzer-ID und kein Kennwort, die bzw. das

von einem Client stammt. Alle Anforderungen werden unter der Benutzer-ID der Verbindung oder der Standard-Benutzer-ID ausgeführt, wenn keine Benutzer-ID für die Verbindung vorhanden ist.

#### **IDENTIFY**

Eingehende Verbindungsanforderungen müssen eine Angabe zur Benutzer-ID enthalten. Geben Sie IDENTIFY ein, wenn das System, das die Verbindung herstellt, als vertrauenswürdig in Bezug auf die Vorauthentifizierung von Benutzern eingestuft wird. Dies ist z. B. dann der Fall, wenn es sich um ein anderes CICS- oder CICS TG-System handelt.

Die SSL-Clientauthentifizierung muss aktiv sein oder das System, das die Verbindung herstellt, muss sich im selben Sysplex befinden.

#### **VERIFY**

Eingehende Verbindungsanforderungen müssen eine Angabe zur Benutzer-ID und ein Kennwort enthalten. Geben Sie VERIFY an, wenn die Systeme, die eine Verbindung herstellen, nicht identifiziert werden können und nicht vertrauenswürdig sind.

#### **DEFAULTUSER**

CICS akzeptiert keine Benutzer-ID und kein Kennwort, die bzw. das vom Partnersystem stammt. Alle Anforderungen werden unter der Standard-Benutzer-ID ausgeführt.

#### **XLNACTON({KEEP|FORCE})**

Gibt die Aktion an, die ausgeführt werden soll, wenn ein neuer Protokollname vom Partnersystem empfangen wird. Der Empfang eines neuen Protokollnamens gibt an, dass der Partner seine Wiederherstellungsinformationen gelöscht hat.

**KEEP** Die Wiederherstellungsinformationen werden aufbewahrt und es werden keine vordefinierten Aktionen für unbestätigte Arbeitseinheiten ausgeführt.

Die Verbindung kann keine neuen Arbeiten ausführen, für die Protokolle der Synchronisationsebene 2 erforderlich sind, und zwar so lange nicht, bis alle ausstehenden wiederherstellbaren Arbeiten mit dem Partner (d. h. unbestätigte Arbeitseinheiten oder relevante Informationen für die Arbeitseinheiten, die auf dem Partnersystem unter dem alten Protokollnamen nicht bestätigt wurden) über die CEMT- oder SPI-Schnittstelle abgeschlossen wurden. Da die Verbindung für die CICS-zu-CICS-Kommunikation verwendet wird, für die immer Protokolle der Synchronisationsebene 2 verwendet werden, kann die Verbindung erst dann angefordert werden, wenn alle ausstehenden wiederherstellbaren Arbeiten mit dem Partner abgeschlossen sind.

#### **FORCE**

Bevor neue Arbeiten unter dem neuen Protokollnamen gestartet werden, werden die vordefinierten Entscheidungen für unbestätigte Arbeitseinheiten (UOWs) gemäß den Angaben in den Attributen für unbestätigte Elemente der TRANSACTION-Definition implementiert. CICS löscht auch alle Informationen, die für möglicherweise erforderliche Auflösungen von Arbeitseinheiten aufbewahrt wurden, die auf dem Partnersystem nicht bestätigt wurden.

**Achtung:** Die Datenintegrität wird bei Verwendung dieser Option möglicherweise beeinträchtigt.

Der Parameter XLNACTION wird für die IPCONN-Definitionen, die von CICS Transaction Gateway verwendet werden, ignoriert.

---

## JOURNALMODEL-Ressourcen

Eine JOURNALMODEL-Ressource definiert die Verbindung zwischen einem CICS-Journalnamen (oder einer entsprechenden Kennung) und den zugeordneten physischen Protokolldatenströmen, die von der MVS-Systemprotokollfunktion verwaltet werden, oder zwischen dem Journalnamen und dem SMF-Protokoll.

Obwohl Journalmodelle hauptsächlich für Benutzerjournale verwendet werden, können Sie sie auch für das Systemprotokoll und für die Protokolle zur aktualisierenden Wiederherstellung (nur Nicht-RLS-Protokolle) definieren. Für Protokolle für die aktualisierende Wiederherstellung sollten Sie jedoch alle Protokolldatenstromnamen für die aktualisierende Wiederherstellung im VSAM-Katalog definieren. Dies ist für VSAM-Dateien, die im RLS-Modus verarbeitet werden, zwingend erforderlich, für Dateien, die nicht im RLS-Modus verarbeitet werden, jedoch optional.

Anders als bei der Journalsteuertabelle müssen Sie nicht für jedes von CICS benutzte Journal ein Journalmodell definieren. Stattdessen definieren Sie einige generische Modelldefinitionen, in denen die Zuordnung zu den Protokolldatenstromnamen für die Mehrzahl Ihrer CICS-Journale beschrieben wird. Sie werden feststellen, dass Sie die Standardmodelle, die von CICS bereitgestellt werden, verwenden können und keine eigenen Modelle definieren müssen. Zusätzlich zu den generischen Modellen, können Sie die erforderlichen speziellen Modelle für die Fälle definieren, in denen eine spezielle Behandlung erforderlich ist (z. B. für die SMF-Protokollierung oder die Zusammenführung mit anderen Protokolldatenströmen).

Sie können die JOURNALMODEL-Definitionen jederzeit ändern, Journaleinträge, die von CICS bereits mit einer Modelldefinition erstellt wurden, enthalten die Änderungen jedoch nur, wenn Sie zuerst den vorhandenen Eintrag mit dem Befehl DISCARD JOURNALNAME() löschen.

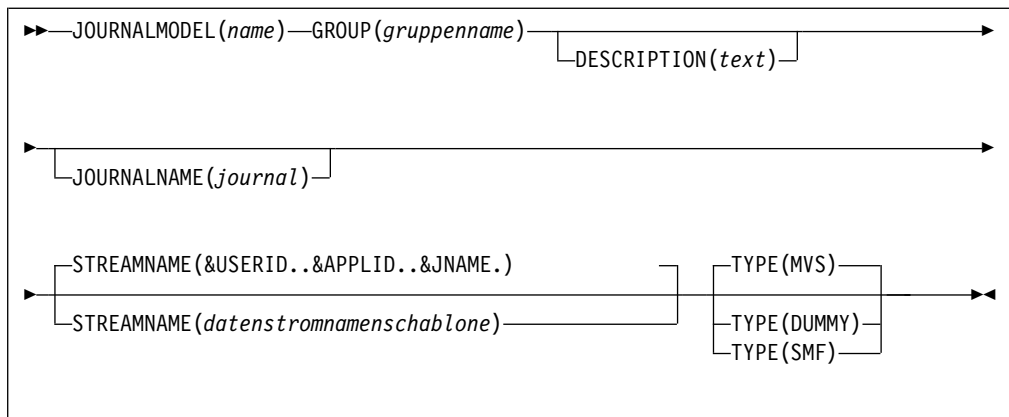
**Hinweis zur Kompatibilität:** Zur Sicherstellung der API-Kompatibilität mit Releases vor CICS Transaction Server for z/OS unterstützt CICS weiterhin numerische Journalkennungen im Bereich zwischen 01 und 99 für die folgenden Zwecke:

- Für die automatische Journalführung der Dateisteuerung gemäß der Angabe in den FILE-Ressourcen (oder für BDAM-Dateien in den DFHFCT-Makroeinträgen).
- Für die automatische Journalführung der Terminalsteuerung gemäß der Angabe in den PROFILE-Ressourcendefinitionen.
- Für die Protokollierung zur aktualisierenden Wiederherstellung gemäß der Angabe in den FILE-Ressourcen.
- Für die Benutzerjournalführung mithilfe von API-Journalbefehlen (z. B. beim Befehl EXEC CICS WRITE JOURNALNUM).

Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with BAS journal model resource definitions.

## JOURNALMODEL-Attribute

Beschreibt die Syntax und die Attribute der JOURNALMODEL-Ressource.



#### DESCRIPTION(text)

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

#### GROUP(gruppenname)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

##### Zulässige Zeichen:

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

#### JOURNALMODEL(name)

Gibt den Namen dieser JOURNALMODEL-Definition an.

Der Journalmodellname wird verwendet, um auf eine bestimmte JOURNALMODEL-Definition in der CSD-Datei zu verweisen. Er muss nicht dem CICS-Journalnamen entsprechen. Der JOURNALMODEL-Name wird allerdings auch als JOURNALNAME verwendet, wenn Sie das Attribut JOURNALNAME nicht angeben.

Der Name kann bis zu acht Zeichen lang sein.

##### Zulässige Zeichen:

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

#### JOURNALNAME(journal)

Gibt die Journalnamen an, für die diese Definition gilt. Wenn Sie das Attribut JOURNALNAME nicht angeben, wird der Name, den Sie im Attribut JOURNALMODEL angeben, als Journalname verwendet. Der Name kann entweder

der spezifische Name eines Journals oder ein generischer Name sein, obwohl ein generischer Name für Systemprotokollmodelle und Modelle für das Protokoll der Protokolle nicht sinnvoll ist.

Der Name kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Die von Ihnen definierten Namen können das folgende Format aufweisen:

**Für Systemprotokolle**

Um ein Journalmodell (JOURNALMODEL) für Systemprotokolle zu definieren, müssen Sie den Namen für den primären Systemprotokoll-datenstrom als DFHLOG und für den sekundären Protokoll-datenstrom als DFHSHUNT angeben. Installieren Sie für jeden dieser Protokoll-datenströme in einer CICS-Region nur ein Journalmodell.

Von CICS bereitgestellte Definitionen für DFHLOG, DFHSHUNT und DFHLGLOG sind in der Gruppe DFHLGMOD in DFHLIST enthalten.

**Für Protokoll der Protokolle**

Um ein Journalmodell (JOURNALMODEL) für das Protokoll der Protokolle zu definieren, müssen Sie als Namen DFHLGLOG angeben. Weitere Informationen zum Zweck eines Protokolls der Protokolle finden Sie in Planning log streams for use by your log of logs (DFHLGLOG).

**Für automatische Journale**

Für automatische Journale (Datei- und Terminalsteuerung) muss der Name das Format DFHJ $nn$  aufweisen. Hierbei steht  $nn$  für eine Zahl im Bereich zwischen 1 und 99. Der Name kann entweder der spezifische Name eines Journals oder ein generischer Name sein.

**Für Benutzerjournale**

Für Benutzerjournale kann der Name bis zu acht Zeichen umfassen. Es kann sich dabei entweder um den spezifischen Namen eines Journals oder einen generischen Namen handeln. Wenn die Kompatibilität mit Releases vor CICS Transaction Server for z/OS erforderlich ist, dann muss der Name das Format DFHJ $nn$  aufweisen, wobei  $nn$  für eine Zahl im Bereich zwischen 1 und 99 steht.

**Für Protokolle für die aktualisierende Wiederherstellung (Nicht-RLS)**

Für Nicht-RLS-Protokolle für die aktualisierende Wiederherstellung muss der Name das Format DFHJ $nn$  aufweisen. Hierbei steht  $nn$  für eine Zahl im Bereich zwischen 1 und 99. Der Name kann entweder der spezifische Name eines Journals oder ein generischer Name sein.

**Anmerkung:** Sie können kein Journalmodell für die Verwendung mit VSAM-RLS-Protokollen für die aktualisierende Wiederherstellung definieren. CICS ruft die vollständig qualifizierte LSN (Log Serial Number; Protokollfolgennummer) direkt aus dem VSAM-Katalog ab und benötigt deshalb kein Journalmodell, um die LSN abzurufen.

Generische Namen können mit den Sondersymbolen %, + und \* wie folgt definiert werden:

- Sie können die Symbole % und + verwenden, um ein beliebiges Einzelzeichen im Journalnamen darzustellen.

- Mit dem Symbol \* am Ende eines Namens können Sie eine beliebige Anzahl von Zeichen darstellen. Ein Standardname, der aus einem einzelnen Stern (\*) besteht, wird verwendet, um eine Übereinstimmung mit allen Journalnamen anzugeben, die mit einem spezifischeren Namen übereinstimmen.

Wenn mehrere JOURNALMODEL-Definitionen installiert sind, die mit einem Journalnamen übereinstimmen, dann wählt CICS die beste Übereinstimmung wie folgt aus:

1. Wenn ein JOURNALMODEL mit einem spezifischen JOURNALNAME vorhanden ist, das eine exakte Übereinstimmung darstellt, dann verwendet CICS dieses Modell.
2. Wenn keine exakte Übereinstimmung vorhanden ist, wird der Journalname mit den übereinstimmenden generischen Einträgen verglichen und es wird der spezifischste Eintrag verwendet.

Beim Vergleichen von Namen, um den Namen mit der präzisesten Übereinstimmung zu ermitteln, werden die Namen Zeichen für Zeichen verglichen. Beim ersten unterschiedlichen Zeichen wird wie folgt vorgegangen:

- Wenn in einem der Namen ein diskretes Zeichen vorhanden ist (nicht %, + oder \*) und der andere Name ein generisches Zeichen (%, + oder \*) enthält, dann wird der Name verwendet, der das diskrete Zeichen enthält.
  - Wenn ein Name das Zeichen % oder + und der andere das Zeichen \* enthält, dann wird der Name mit dem Zeichen % oder + verwendet.
3. Wenn doppelte JOURNALMODEL-Definitionen (Definitionen mit demselben Wert für JOURNALNAME) vorhanden sind, dann verwendet CICS die zuletzt verarbeitete Definition.

**Achtung:** Beim Definieren eines vollständig generischen Journalnamens, bei dem nur der einzelne Stern (\*) verwendet wird, sollten Sie mit Bedacht vorgehen. Dies ist besonders wichtig, wenn Sie kein spezielles Journalmodell für das Systemprotokoll (mit dem Journalnamen DFHLOG) definiert haben und wenn der Protokolldatenstromname ein vollständig qualifizierter Literalname ist. Wenn Sie ein Journalmodell mit der Angabe JOURNALNAME(\*) definieren und für das Systemprotokoll kein Journalmodell definieren, dann verwendet CICS den Protokolldatenstromnamen, der in der generischen Modelldefinition definiert wurde. Dies führt zu Problemen, wenn andere Journale und Protokolle für die aktualisierende Wiederherstellung mithilfe des generischen Journalmodells demselben Protokolldatenstrom zugeordnet sind.

**STREAMNAME**({&USERID..&APPLID..&JNAME. | *datenstromnamenschablone*})

Gibt entweder den expliziten Protokolldatenstromnamen einer MVS-Systemprotokollfunktion oder eine Schablone an, die zum Erstellen des Protokolldatenstromnamens verwendet wird. Die Angabe von STREAMNAME ist nur gültig für Journalmodelle, bei denen für LOGSTREAMTYPE die Einstellung MVS angegeben wurde.

Die vier symbolischen Namen, von denen Sie maximal drei verwenden können, lauten wie folgt:

#### **&USERID.**

Der symbolische Name für die Benutzer-ID der CICS-Region, der bis zu acht Zeichen lang sein kann. Wenn in der Region keine Benutzer-ID vorhanden ist, wird die Zeichenfolge 'CICS' verwendet.

### **&APPLID.**

Der symbolische Name für die Anwendungs-ID (APPLID) der CICS-Region, der im Systeminitialisierungsparameter angegeben ist und der bis zu acht Zeichen lang sein kann.

**Anmerkung:** Wenn Sie mit XRF arbeiten und den Systeminitialisierungsparameter APPLID im Format APPLID=(generische\_applid, spezifische\_applid) angeben, dann verwendet CICS beim Auflösen von &APPLID. die generische Anwendungs-ID.

### **&JNAME.**

Der symbolische Name für einen Journalnamen, der mit einer spezifischen oder generische Übereinstimmung auf diese Journalmodelldefinition verweist. &JNAME. kann bis zu acht Zeichen lang sein.

### **&SYSID.**

Der symbolische Name für die System-ID (SYSID) der CICS-Region, die im Systeminitialisierungsparameter SYSIDNT angegeben ist. Wenn SYSIDNT nicht angegeben ist, wird die Zeichenfolge 'CICS' verwendet.

Die Standardgruppe symbolischer Namen lautet wie folgt: &USERID..  
&APPLID..&JNAME.

**Beispiel:** &USERID..&APPLID..&JNAME. =

CICSHA##.CICSHAA1.DFHJ02

Dabei gilt Folgendes:

#### **CICSHA##**

ist die Benutzer-ID der CICS-Region, die von allen AORs (Application-Owning Regions; Anwendungsverwaltungsregionen) verwendet wird.

#### **CICSHAA1**

ist die Anwendungs-ID einer AOR.

#### **DFHJ02**

ist der Journalname eines automatischen Journals.

Eine alternative Gruppe symbolischer Namen lautet wie folgt:

&SYSID..&APPLID..&JNAME. =  
SYSA.CICSHAA1.DFHJ02

Dabei gilt Folgendes:

**SYSA** ist die Zeichenfolge, die im Systeminitialisierungsparameter SYSIDNT angegeben ist.

#### **CICSHAA1**

ist die Anwendungs-ID einer AOR.

#### **DFHJ02**

ist der Journalname eines automatischen Journals.

CICS installiert die JOURNALMODEL-Ressource wie definiert und einschließlich der symbolischen Namen.

### datenstromnamenschablone

Ein Protokolldatenstromname kann entweder ein nicht qualifizierter Name oder ein qualifizierter Name gemäß der Definition für die MVS-Dateinamen sein:

- **Nicht qualifizierter Name:** Ein bis acht alphanumerische Zeichen oder nationale Sonderzeichen (\$ # @) oder ein Bindestrich. Das erste Zeichen des Namens muss ein alphabetisches Zeichen oder ein nationales Sonderzeichen (A-Z \$ # @) sein.
- **Qualifizierter Name:** Mehrere Namen, die durch Punkte verbunden sind, mit bis zu 26 Zeichen. Jeder Name in einem qualifizierten Namen muss auf Basis der Regeln für einen nicht qualifizierten Namen gebildet werden, wobei jeder qualifizierte Name (außer dem letzten) von einem Punkt gefolgt wird. Beispiel:

name\_1.name\_2...name\_n

Die Anzahl der Namen wird durch das Limit von 26 Zeichen beschränkt.

Weitere Informationen zu den Regeln für qualifizierte und nicht qualifizierte Dateinamen finden Sie in z/OS MVS JCL Reference.

Sie können Protokolldatenstromnamen erstellen, die aus einer Kombination spezieller Zeichen (aus der zulässigen Gruppe) und symbolischen Namen für die Substitution bestehen. Nach der Substitution muss der Name den Regeln für qualifizierte und nicht qualifizierte Protokolldatenstromnamen entsprechen und darf (einschließlich der Punkte) maximal 26 Zeichen lang sein. Wenn jeder Name in einem qualifizierten Namen maximal acht Zeichen umfasst, dann sind Sie auf nur drei Namen beschränkt, wobei der erste und der zweite Name sowie der zweite und der dritte Name durch einen Punkt getrennt sind. Das Beispiel

CICSDA##.CICSDAA1.FWDRECOV

gilt für einen Protokolldatenstrom für die aktualisierende Wiederherstellung. Der Protokolldatenstromname wird durch die symbolische Substitution ermittelt, wenn ein Journalname zum ersten Mal in eine JOURNALMODEL-Definition aufgelöst wird.

Wenn Sie den gleichen Protokolldatenstromnamen für mehrere allgemeine CICS-Protokolle angeben, dann können Sie die Protokolldatenströme aus verschiedenen CICS-Regionen zusammenführen. Sie können allerdings keine allgemeinen Protokolldatenströme mit dem CICS-Systemprotokoll zusammenführen. Auch ist es nicht möglich, Systemprotokolle aus unterschiedlichen CICS-Regionen zusammenzuführen.

Wenn Sie Protokolldatenströme aus unterschiedlichen CICS-Systemen zusammenführen wollen, dann werden die Protokolldatenblöcke in die zugehörigen Protokolldatenströme geschrieben, und zwar strikt in der Reihenfolge der Zeitmarken der MVS-Systemprotokollfunktion. Die einzelnen Datensätze aus unterschiedlichen CICS-Regionen sind jedoch möglicherweise nicht in strikter Reihenfolge der Zeitmarken über mehrere Blöcke angeordnet.

CICS-Protokolldatenströme sollten nicht mit Protokolldatenströmen zusammengeführt werden, die in anderen Produkten generiert wurden, es sei denn, es handelt sich dabei um Programme zum Lesen des Protokolldatenstroms, die zur Verarbeitung der Formate in der Lage sind.

**Sicherheitshinweis:** Wenn Sie einen Protokolldatenstromnamen für CICS und die MVS-Systemprotokollfunktion definiert haben, dann müssen Sie sicherstellen, dass die erforderlichen Sicherheitsberechtigungen in RACF (oder in einem



gleichwertigen externen Sicherheitsmanager) definiert wurden. Diese Sicherheitsberechtigung ist erforderlich, damit Sie versuchen können, eine CICS-Region zu starten, die auf einen neuen Protokolldatenstrom verweist. RACF unterstützt zu diesem Zweck die allgemeine Ressourcenklasse LOGSTRM.

#### **TYPE({DUMMY|MVS|SMF})**

Gibt an, an welche Position die Journalsätze geschrieben werden sollen. Die Angabe kann bis zu fünf Zeichen umfassen und die folgenden Werte aufweisen:

##### **DUMMY**

Es werden keine Protokolldatensätze geschrieben. Sie können diese Option z. B. verwenden, um unerwünschte Protokolldatensätze zu unterdrücken, ohne dass hierzu Änderungen an einer Anwendung bzw. an den Datei- oder Profilressourcendefinitionen durchgeführt werden müssen.

Wenn Sie nicht mit einem Systemprotokoll oder einem Protokoll der Protokolle arbeiten möchten, dann geben Sie in den JOURNALMODEL-Definitionen für DFHLOG, DFHSHUNT und DFHLGLOG bei Bedarf die Option DUMMY an.

**MVS** Die Datensätze werden in den Protokolldatenstrom einer MVS-Systemprotokollfunktion geschrieben. Der Name des Protokolldatenstroms wird im Attribut STREAMNAME angegeben.

**SMF** Journaldatensätze werden im SMF-Format ins MVS-SMF-Protokoll und nicht in den Protokolldatenstrom der MVS-Systemprotokollfunktion geschrieben.

**Anmerkung:** Die Verwendung der Option SMF ist für das CICS-Systemprotokoll und für die Protokolle zur aktualisierenden Wiederherstellung nicht zulässig.

## **JOURNALMODEL-Standardattribute**

Wenn CICS auf einem System keine installierte JOURNALMODEL-Definition feststellen kann, dann wird eine Gruppe von Standardattributen verwendet.

Die in diesem Fall verwendeten Attribute stammen aus der folgenden „integrierten“ Standarddefinition:

```
DEFINE JOURNALMODEL(OTHERS) GROUP(LOGS)
    JOURNALNAME(*)
    STREAMNAME(&USERID..&APPLID..&JNAME.)
    TYPE(MVS)
```

Bei JOURNALNAME(\*) handelt es sich um das Standardjournalmodell, das von CICS verwendet wird, wenn für einen Journalnamen kein übereinstimmender JOURNALMODEL-Eintrag vorhanden ist.

## **Beispiele**

Beispiel für Ressourcendefinitionsbefehle für CICS-Journalmodelle (JOURNALMODELs).

Im Folgenden ist eine Gruppe von Definitionen für eine CICS-Anwendungsverwaltungsregion (AOR = Application-Owning Region) mit der Anwendungs-ID CICS-HAA3 und der Regions-Benutzer-ID CICSHA## aufgeführt:

```
1. DEFINE JOURNALMODEL(USERJNL8) GROUP(LOGS)
    JOURNALNAME(DFHJ08)
    TYPE(SMF)
```

Datensätze, die mit dem Befehl EXEC CICS WRITE JOURNALNAME(DFHJ08)... oder EXEC CICS WRITE JOURNALNUM(08)... ins automatische Journal 08 oder ins Benutzerjournal 08 geschrieben werden, werden im SMF-Format in die MVS-SMF-Datei geschrieben.

```
2. DEFINE JOURNALMODEL(USERJNL9) GROUP(LOGS)
    JOURNALNAME(DFHJ09)
    TYPE(DUMMY)
```

Datensätze, die mit dem Befehl EXEC CICS WRITE JOURNALNAME(DFHJ09)... oder EXEC CICS WRITE JOURNALNUM(09)... ins automatische Journal 09 oder ins Benutzerjournal 09 geschrieben werden, werden in keinen Protokolldatenstrom geschrieben. Das Anwendungsprogramm erhält eine normale Antwort.

```
3. DEFINE JOURNALMODEL(UJ10T019) GROUP(LOGS)
    JOURNALNAME(DFHJ1%)
    STREAMNAME(&USERID..MERGED.USRJRNL)
    TYPE(MVS)
```

Datensätze, die in die Benutzerjournale 10-19 (DFHJ10-DFHJ19) geschrieben werden, werden im Protokolldatenstrom CICS#Merged.USRJRNL mit Datensätzen aus allen anderen CICS-Regionen zusammengeführt, die unter der Benutzer-ID CICS# und mit den gleichen installierten Journalmodellen (JOURNALMODELS) ausgeführt werden.

```
4. DEFINE JOURNALMODEL(LOGOFLAG) GROUP(LOGS)
    JOURNALNAME(DFHLGLOG)
    STREAMNAME(CICSVR.SHARED.DFHLGLOG)
    TYPE(MVS)
```

Dateibelegungsdatensätze und andere Datensätze, die über die Dateisteuerung und den CICS-Protokollmanager ins Journal DFHLGLOG geschrieben wurden, um dann in Produkten zur Durchführung der aktualisierenden Wiederherstellung wie z. B. CICSVR verwendet zu werden, werden in einen gemeinsam genutzten Protokolldatenstrom mit dem Namen CICSVR.SHARED.DFHLGLOG geschrieben. Dieser Protokolldatenstrom wird von allen CICS-Regionen im Sysplex, in denen diese JOURNALMODEL-Ressourcendefinition installiert ist, gemeinsam genutzt.

```
5. DEFINE JOURNALMODEL(JNLMDL1) GROUP(LOGS)
    JOURNALNAME(USERJNL*)
    STREAMNAME(&USERID..ANYCORP.&JNAME..UK)
    TYPE(MVS)
```

Datensätze, die in Benutzerjournale oder automatische Journale geschrieben werden und deren Name mit den Buchstaben USERJNL beginnt, werden in einem Protokolldatenstrom mit einem Namen zusammengeführt, der ermittelt wird, indem für &USERID. die Benutzer-ID der CICS-Region und für &JNAME. der Journalname eingesetzt wird.

Sind nur diese Beispiele installiert, dann verwenden andere Formen der Journalführung (z. B. die automatische Nachrichtenjournalführung für die Terminalsteuerung, die mit PROFILE ... JOURNAL(25) definiert wurden) das standardmäßige Journalmodell (JOURNALMODEL), wobei die Datensätze in den Protokolldatenstrom CICS#CICSHAA3.DFHJ25 geschrieben werden. Weitere Informationen hierzu finden Sie in „JOURNALMODEL-Standardattribute“ auf Seite 95.

---

## JVMSERVER-Ressourcen

Eine JVMSERVER-Ressource definiert die Laufzeitumgebung für einen JVM-Server.

In der JVMSERVER-Ressource definieren Sie die Position des JVM-Profiles und die Language Environment-Optionen, die zum Erstellen einer Language Environment-Enklave und eines JVM-Servers in einer CICS-Region erforderlich sind.

Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in *Working with BAS JVMSERVER resource definitions*.

### JVMSERVER-Ressourcen in CICS-Bundles

Sie können ein CICS-Bundle verwenden, um eine JVMSERVER-Ressourcendefinition zu erstellen, zu bearbeiten und zu installieren. Wenn Sie eine JVMSERVER-Ressource auf diese Weise erstellen, dann müssen Sie zum Verwalten des Lebenszyklus dieser Ressource das CICS-Bundle verwenden. Die unabhängige Verwaltung der Ressource ist nicht möglich.

Sie können eine JVMSERVER-Ressource abfragen, die von einem CICS-Bundle dynamisch generiert wird. Verwenden Sie hierzu den Befehl **EXEC CICS INQUIRE JVMSERVER** oder **CEMT INQUIRE JVMSERVER**.

Der Befehl DISCARD kann für eine JVMSERVER-Ressource, die von einem CICS-Bundle dynamisch generiert wird, nicht abgesetzt werden. Sie müssen das CICS-Bundle löschen. CICS wendet die Operation auf die JVMSERVER-Ressource an.

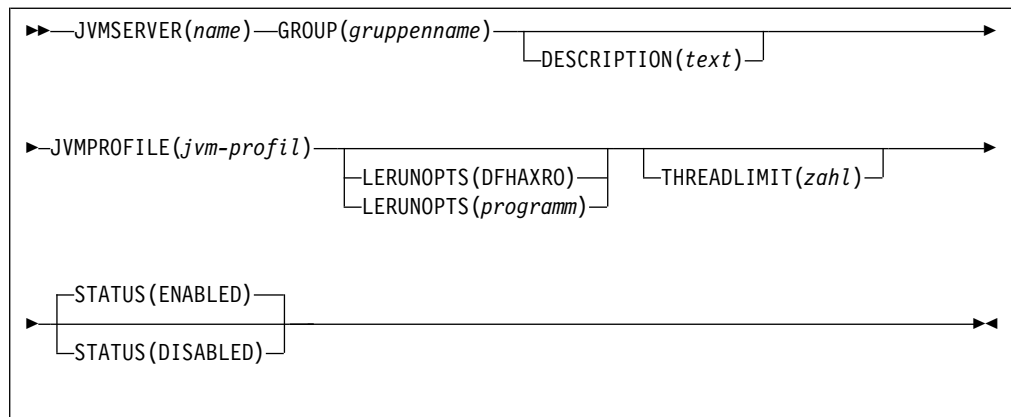
Zum Ändern der Attribute einer JVMSERVER-Ressource, die in einem CICS-Bundle definiert und installiert wurde, können Sie den Ressourceneditor in CICS Explorer verwenden, um die Definition im CICS-Bundle zu ändern und eine neue Version des CICS-Bundles oder der Anwendung zu installieren, über das bzw. die die Bereitstellung durchgeführt wurde. Sie können den Befehl SET JVMSERVER verwenden, um die Attribute der dynamisch generierten Ressource zu ändern. Diese Änderungen werden jedoch nicht katalogisiert und können nach einem Warmstart von CICS nicht wiederhergestellt werden.

Um den Status einer JVMSERVER-Ressource zu ändern, die in einem CICS-Bundle definiert und installiert wurde, müssen Sie das CICS-Bundle aktivieren oder inaktivieren. Wenn Sie das CICS-Bundle inaktiviert haben, jedoch Tasks bereinigen müssen, die auf dem JVM-Server weiterhin ausgeführt werden, dann können Sie den Befehl SET JVMSERVER DISABLED für die dynamisch generierte Ressource absetzen und dabei die Option PURGE, FORCEPURGE oder KILL verwenden, um die Tasks zu bereinigen.

Weitere Informationen zum Definieren von Ressourcen in CICS-Bundles finden Sie im Abschnitt zum Definieren von CICS-Bundles.

### JVMSERVER-Attribute

Beschreibt die Syntax und die Attribute der JVMSERVER-Ressource.



### **JVMSERVER**(*name*)

Gibt den aus einem bis acht Zeichen bestehenden Namen für die JVMSERVER-Ressource an.

#### **Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . - \_ % ? ! : | = , ;

Sie sollten keine Namen verwenden, die mit der Zeichenfolge DFH beginnen, weil diese Zeichen zur Verwendung durch CICS reserviert sind.

CICS kann sie löschen oder verwerfen und durch CICS-Ressourcen ersetzen, wenn Sie ein Upgrade durchführen oder neue Funktionen oder Fixes bereitstellen.

### **DESCRIPTION**(*text*)

In diesem Feld können Sie eine Beschreibung für die Ressource angeben, die Sie definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Es gelten keine Einschränkungen für die Zeichen, die Sie verwenden können. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für jede linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Textcode zwei Hochkommas angeben.

### **GROUP**(*gruppenname*)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

#### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein.

### **JVMPROFILE**(*jvm-profil*)

Gibt den aus einem bis acht Zeichen bestehenden Namen des JVM-Profils an, das die JVM-Optionen für die Ausführung eines JVM-Servers enthält. Die Datei, die das JVM-Profil enthält, muss die Dateierweiterung `.jvmprofile` aufweisen.

- Für ein JVM-Profil für eine JVMSERVER-Ressource, die anhand einer Onlinersourcendefinition definiert wurde, wird der Dateipfad im Systeminitialisierungsparameter **JVMPROFILEDIR** angegeben.

- Für ein JVM-Profil für eine JVMSERVER-Ressource, die in einem CICS-Bundle definiert wurde, ist der Dateipfad relativ zum Stammverzeichnis des CICS-Bundles angegeben. Die Datei, die das JVM-Profil enthält, muss im CICS-Bundle mit der JVMSERVER-Ressource gepackt werden. Weitere Informationen hierzu finden Sie in Referencing zFS artifacts in a bundle.

Der Name des JVM-Profils darf nicht aus einem einzelnen oder einem doppelten Punkt bestehen. Die Gruppe der im JVM-Profilnamen zulässigen Zeichen unterliegt größeren Einschränkungen, wenn die JVMSERVER-Ressource mit CICS Explorer in einem CICS-Bundle definiert wurde.

**Zulässige Zeichen für den JVMPROFILE-Namen in einer JVMSERVER-Onlineresource:**

A-Z a-z 0-9 @ # . - \_ % & ? ! : | " = , ; < >

**Zulässige Zeichen für den JVMPROFILE-Namen in einer JVMSERVER-Ressourcendefinition in einem CICS-Bundle:**

A-Z a-z 0-9 @ # - \_ % ! =

#### **LERUNOPTS(DFHAXRO|programm)**

Gibt den aus einem bis acht Zeichen bestehenden Namen des Programms an, das die Laufzeitoptionen für die Language Environment-Enklave definiert. DFHAXRO ist ein bereitgestelltes Programm, in dem eine Gruppe von Standardwerten enthalten ist. Die Quelle für DFHAXRO befindet sich in der Bibliothek *hlq.SDFHSAMP*, wenn Sie die Standardwerte für eine der Language Environment-Laufzeitoptionen ändern möchten.

Wenn Sie ein anderes Programm verwenden wollen, stellen Sie das Programm in die Bibliothek *hlq.SDFHLOAD* und geben Sie den Programmnamen in Großbuchstaben an.

#### **STATUS(ENABLED|DISABLED)**

Gibt den Anfangsstatus der JVMSERVER-Ressource bei der Installation an.

##### **ENABLED**

Die Laufzeitumgebung des JVM-Servers steht für die Verwendung zur Verfügung.

##### **DISABLED**

Die Laufzeitumgebung des JVM-Servers steht nicht für die Verwendung zur Verfügung.

#### **THREADLIMIT(15|zahl)**

Gibt die maximale Anzahl der Threads an, die in der Language Environment-Enklave für den JVM-Server zulässig sind. Jeder Thread wird in einem T8-Tasksteuerblock (TCB = Task Control Block) ausgeführt. Sie können eine Begrenzung im Bereich von 1 bis 256 Threads angeben.

Wenn Sie eine Threadbegrenzung angeben, die das Maximum von 2000 Threads überschreitet, das für die CICS-Region zulässig ist, dann ordnet CICS unter Berücksichtigung aller anderen aktivierten und inaktivierten JVMSERVER-Ressourcen die bis zur Begrenzung von 2000 verbleibenden Threads der Ressource als Threadbegrenzungswert zu. Wenn in CICS die maximal zulässige Anzahl von JVMSERVER-Threads bereits erreicht ist, dann wird die Ressource im inaktivierten Status installiert.

**Tip:** Liberty-JVM-Server benötigen zur Ausführung mindestens fünf Threads. Wenn Sie für einen Liberty-JVM-Server eine Threadbegrenzung angeben, der

weniger als fünf Threads vorsieht, dann werden die Threadbegrenzung und der Threadzähler automatisch auf fünf erhöht, um zu vermeiden, dass die Liberty-Einheit ausfällt.

---

## LIBRARY-Ressourcen

Verwenden Sie die LIBRARY-Ressource, um die physischen Merkmale und die Verfahrensmerkmale der LIBRARY-Verknüpfung eines dynamischen Programms zu definieren.

Die LIBRARY-Definition umfasst Attribute, die den Namen von Dateien in der LIBRARY-Ressource und weitere Einzelangaben zum Betriebsstatus der LIBRARY-Komponente enthalten.

DFHRPL ist ein spezielles Beispiel einer LIBRARY-Komponente, die in einem aktiven CICS-System nicht geändert werden kann und die in CEDA nicht als Ressource erscheint.

Die Dateien in einer LIBRARY-Verknüpfung eines dynamischen Programms können im EAS (Extended Addressing Space) eines EAV-DASD-Datenträgers (EAV = Extended Address Volume) enthalten sein.

Für LIBRARY-Ressourcen gibt es keine Gruppenfestschreibung (sog. Group-Commit). Jede LIBRARY-Komponente in einer CSD-Gruppe wird separat festgeschrieben, wenn die Gruppe installiert wird. Um eine ähnliche Wirkung zu erzielen wie bei einer Gruppeninstallation, können Sie den Status 'Inaktiviert' verwenden, um eine Gruppe von LIBRARY-Ressourcen zu installieren, die anschließend in den Aktivierungszustand 'Aktiviert' versetzt werden, nachdem alle Ressourcen installiert wurden.

Informationen zur Installation von LIBRARY-Ressourcendefinitionen mit CEMT und CEDA finden Sie in *Installing LIBRARY resource definitions*. Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in *Working with BAS LIBRARY resource definitions*.

### LIBRARY-Ressourcen in CICS-Bundles

Sie können ein CICS-Bundle verwenden, um eine LIBRARY-Ressourcendefinition zu erstellen, zu bearbeiten und zu installieren. Wenn Sie eine LIBRARY-Ressource auf diese Weise erstellen, dann müssen Sie zum Verwalten des Lebenszyklus dieser Ressource das CICS-Bundle verwenden. Die unabhängige Verwaltung der Ressource ist nicht möglich. Sie können eine LIBRARY-Ressource abfragen, die von einem CICS-Bundle dynamisch generiert wird. Verwenden Sie hierzu den Befehl **EXEC CICS INQUIRE LIBRARY** oder **CEMT INQUIRE LIBRARY**. Der Befehl SET oder DISCARD kann jedoch nicht für eine LIBRARY-Ressource ausgeführt werden, die von einem CICS-Bundle dynamisch generiert wird. Sie müssen die Befehle für die BUNDLE-Ressource absetzen und CICS wendet sie dann auf die LIBRARY-Ressource an.

Weitere Informationen zum Definieren von Ressourcen in CICS-Bundles finden Sie in *Defining CICS bundles*.

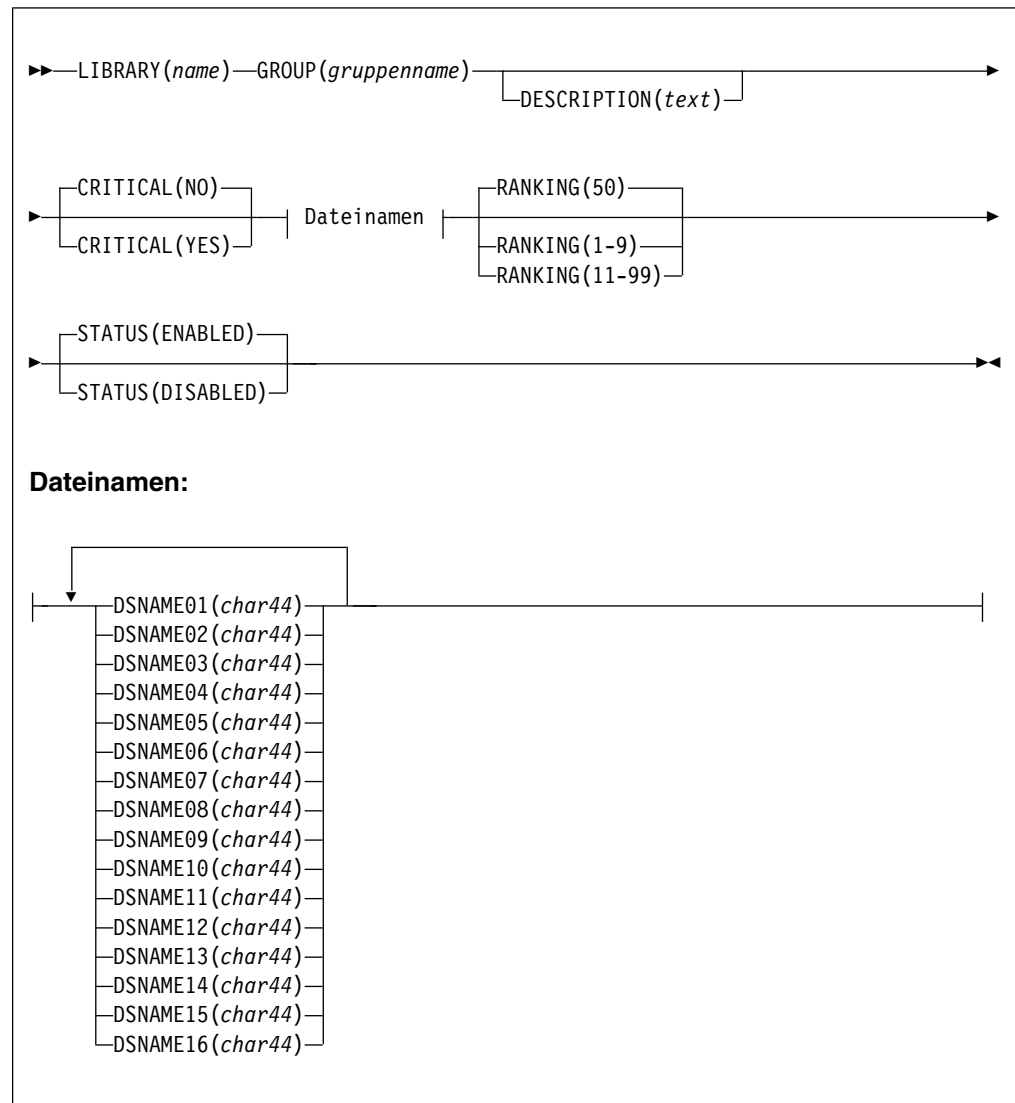
## Private LIBRARY-Ressourcen für Anwendungen

Die LIBRARY-Ressource wird als private Ressource für eine Anwendungsversion unterstützt. Jede Version einer Anwendung sollte eine private LIBRARY-Ressource enthalten, die eine versionsspezifische Datei darstellt, die die Lademodule für die Anwendung enthält.

Weitere Informationen zu privaten Ressourcen finden Sie in Characteristics of bundled resources.

## LIBRARY-Attribute

Beschreibt die Syntax und die Attribute der LIBRARY-Ressource.



### CRITICAL({NO|YES})

Gibt an, ob diese Bibliothek (LIBRARY) für die Ausführung von CICS von kritischer Bedeutung ist. Mit diesem Attribut wird das Verhalten festgelegt, wenn die LIBRARY-Komponente während des Starts nicht installiert werden kann, z. B. wenn eine Datei in der LIBRARY-Definition nicht gefunden oder aus einem der folgenden Gründe nicht zugeordnet werden kann:

- Wenn für LIBRARY das Attribut CRITICAL angegeben ist.
  - Wenn die LIBRARY-Komponente während des CICS-Starts (über die Gruppenlisteninstallation, BAS-Installation, ein PLTPI-Programm oder durch Wiederherstellung aus dem Katalog) erstellt wird, dann wird eine „GO- oder CANCEL“-Nachricht abgesetzt, sodass der Bediener entscheiden kann, ob der kritische Status ignoriert und der Start von CICS zugelassen oder der Start mit einem Fehler abgebrochen werden soll. Wenn der Start fortgesetzt wird, dann wird die LIBRARY-Komponente nicht als NONCRITICAL neu katalogisiert, sondern der kritische Status kann in NONCRITICAL geändert werden, wenn Sie entscheiden, dass die LIBRARY-Komponente in Zukunft nicht mehr als CRITICAL eingestuft werden soll.
  - Wenn die LIBRARY-Komponente nicht während des Starts von CICS erstellt wird, dann hat die Kritikalität der LIBRARY-Komponente keine Auswirkung auf das Verhalten der Installation.
- Wenn für LIBRARY das Attribut CRITICAL nicht angegeben ist.
  - Wenn die LIBRARY-Komponente während des CICS-Starts (über die Gruppenlisteninstallation, BAS-Installation, ein PLTPI-Programm oder durch Wiederherstellung aus dem Katalog) erstellt wird, dann wird die LIBRARY-Komponente zwar installiert, jedoch inaktiviert. Das System gibt eine Warnung aus und der Start von CICS wird fortgesetzt. Zu einem späteren Zeitpunkt kann versucht werden, das Problem zu beheben und die LIBRARY-Komponente zu aktivieren.
  - Wenn die LIBRARY-Komponente nicht während des Starts von CICS installiert wird, dann hat die Kritikalität der LIBRARY-Komponente keine Auswirkung auf das Verhalten der Installation.

Für die statische LIBRARY-Komponente DFHRPL ist die Einstellung CRITICAL vordefiniert. Sie kann nicht geändert werden. Eventuell auftretende Probleme führen zum Fehlschlagen des CICS-Starts und zur Ausgabe einer Fehlermeldung.

Für eine private LIBRARY-Ressource, die in einem CICS-Bundle definiert ist, das als Teil eines Anwendungsbundles oder Anwendungsbindungsbundles gepackt und installiert wird, wird das Attribut CRITICAL ignoriert.

#### **DSNAME01-16 (Name)**

Die Namen von bis zu 16 Dateien, die Programmartefakte enthalten und die die LIBRARY-Verknüpfung bilden sollen. Wenn die LIBRARY-Komponente installiert wird, dann werden die Dateien in der Reihenfolge verknüpft, in der sie angegeben werden, z. B. DSNAME01 vor DSNAME02. Es ist nicht erforderlich, dass alle Dateiattribute sequenziell ausgeführt werden. Für DSNAME01 muss beispielsweise keine Angabe definiert werden. Allerdings muss mindestens eines der Dateiattribute über einen Wert verfügen. Weitere Informationen hierzu finden Sie in Defining a data set naming convention.

#### **GROUP (gruppenname)**

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

#### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.



Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

#### **LIBRARY**(*name*)

Der Name der LIBRARY-Ressource.

Für eine LIBRARY-Ressource, die in einer CICS-Region, einem CICSplex SM-Datenrepository oder einem eigenständigen CICS-Bundle definiert ist, wird der Ressourcename als DD-Name für die LIBRARY-Verknüpfung der Dateien benutzt. Der Name der LIBRARY-Ressource muss deshalb den Regeln für einen gültigen DD-Namen (ein bis acht Zeichen, wobei das erste Zeichen A - Z, \$, # oder @ lauten muss und für die verbleibenden Zeichen A - Z, 1 - 9, \$, # oder @ verwendet werden können) entsprechen. Vermeiden Sie die Verwendung von Namen, die zu Konflikten mit anderen DD-Namen führen, die von der Installation verwendet werden. Die Erstellung einer öffentlichen LIBRARY-Ressource schlägt fehl, wenn der ausgewählte Name zu einem Konflikt mit einem DD-Namen führt, der auf dem System bereits für eine andere öffentliche LIBRARY-Ressource verwendet wird.

Für eine private LIBRARY-Ressource, die in einem CICS-Bundle definiert ist, das als Teil eines Anwendungsbundles oder Anwendungsbindungsbundles gepackt und installiert wird, wird der Name der LIBRARY-Ressource nicht als DD-Name für die LIBRARY-Verknüpfung von Dateien verwendet. Stattdessen fordert CICS einen eindeutigen DD-Namen für die LIBRARY-Verknüpfung der Dateien an, wenn die Anwendung auf der Plattform installiert wird. Der Ressourcename kann deshalb mit den LIBRARY-Namen übereinstimmen, die in anderen Bereichen der Installation oder von anderen Versionen der Anwendung benutzt werden. CICS gibt die Nachricht DFHLD0518 aus, um den DD-Namen anzugeben, der von z/OS für die LIBRARY-Verknüpfung generiert wurde. Sie können die Dateinamen für eine installierte Anwendung auch in CICS Explorer anzeigen.

Die folgenden reservierten Namen werden für eine LIBRARY-Ressource nicht akzeptiert:

- DFHRPL oder ein anderer Name, der mit der Zeichenfolge DFH beginnt.
- Ein beliebiger Name, der mit der Zeichenfolge EYU beginnt.
- CEEDUMP
- DUMMY
- JOBCAT
- JOBLIB
- STEPCAT
- STEPLIB
- SYSABEND
- SYSIN
- SYSMDUMP
- SYSOUT
- SYSUDUMP

LIBRARY-Ressourcennamen, die mit der Zeichenfolge DFH oder EYU beginnen, werden nicht akzeptiert, um sicherzustellen, dass diese Namen für die zukünftige Verwendung durch CICS verfügbar bleiben. LIBRARY-Ressourcennamen, die mit CEE beginnen, sollten mit Bedacht verwendet werden, sie werden vom System jedoch akzeptiert. SYSIN, SYSOUT und DUMMY werden normalerweise als DD-Karten verwendet und sollten deshalb nicht als LIBRARY-Ressourcennamen benutzt werden. Es verbleiben die DD-Namen, die für MVS re-

serviert sind. DFHRPL ist für die Verwendung im JCL-Code für den Start von CICS als Name der statischen LIBRARY-Komponente reserviert.

#### **RANKING(1-99)**

Im Attribut RANKING wird ein Vollwort-Binärwert angegeben, der eine Dezimalzahl enthält, die festlegt, wo diese LIBRARY-Komponente in der Gesamtsuchreihenfolge für LIBRARY steht. Eine niedrigere Nummer gibt an, dass diese LIBRARY-Komponente nach Programmen durchsucht wird, die geladen werden müssen, bevor andere LIBRARY-Ressourcen mit höherrangigen Nummern durchsucht werden. Die Rangfolge ist analog zur Verknüpfungsnummer einer Datei in einer LIBRARY-Verknüpfung, obwohl bestimmte Unterschiede bestehen.

Für RANKING können Werte im Bereich zwischen 1 und 99 angegeben werden. Der Standardwert ist 50. Der Wert 10 ist für die statische LIBRARY-Komponente DFHRPL reserviert und darf nicht angegeben werden.

Der DFHRPL-Verknüpfung ist die vordefinierte Rangfolge 10 zugeordnet, die nicht geändert werden kann. Dieser Wert ermöglicht es Ihnen, dynamisch definierte LIBRARY-Ressourcen in der Gesamtsuchreihenfolge vor der DFHRPL-Verknüpfung zu platzieren, indem diesen Ressourcen ein Rangfolgewert kleiner als 10 zugeordnet wird.

#### **Anmerkung:**

- LIBRARY-Ressourcen, die eine Rangfolge haben, die sie in der Suchreihenfolge vor DFHRPL platziert, sollten nur als vorübergehende Lösung betrachtet werden.
- Obwohl die vordefinierte DFHRPL-Rangfolge 10 dazu dienen soll, die Platzierung von LIBRARY-Ressourcen vor DFHRPL in der Suchreihenfolge zu vermeiden, wird dadurch nicht die Gesamtanzahl der LIBRARY-Ressourcen beschränkt, die vor DFHRPL platziert werden können, sofern die Rangfolge zwischen den LIBRARY-Ressourcen selbst nicht von Bedeutung ist. Wenn Sie einen RANKING-Wert kleiner als 10 angeben, werden Sie in einer Nachricht darüber informiert, dass die betreffende LIBRARY-Komponente in der Suchreihenfolge vor DFHRPL erscheint.
- Die Änderung der Rangfolge tritt erst in Kraft, wenn eine NEWCOPY- oder PHASEIN-Anforderung abgesetzt wird oder das Programm zum ersten Mal geladen wird, wenn es noch nicht geladen wurde.
- Die Suchreihenfolge von LIBRARY-Komponenten mit derselben Rangfolge wird nicht definiert und impliziert, dass die relative Suchreihenfolge der LIBRARY-Ressourcen nicht von Bedeutung ist. In der Praxis ist die Suchreihenfolge die Reihenfolge, in der die LIBRARY-Komponenten installiert wurden.
- Eine LIBRARY-Ressource, die in einem CICS-Bundle als Teil einer Anwendung definiert wurde, die auf einer Plattform installiert ist, wird in Bezug auf diese Version der Anwendung als privat eingestuft. Diese LIBRARY-Verknüpfungen werden nur nach Tasks für diese Version der Anwendung durchsucht.
- LIBRARY-Komponenten können die gleiche Rangfolge aufweisen, Sie sollten jedoch keine gleichen Rangfolgewerte für zwei LIBRARY-Ressourcen verwenden, wenn ihre relative Suchposition von Bedeutung ist.
- Die Suchpositionen von zwei LIBRARY-Ressourcen mit derselben Rangfolge sind nicht zuverlässig. Die Suchreihenfolge bezieht sich auf den Zeitpunkt, zu dem jede Ressource installiert wurde und kann nach einem COLD- oder INITIAL-Start oder beim Ändern der Rangfolge einer LIBRARY-Ressource geändert werden. Die zuerst installierte Ressource erscheint in der Suchreihenfolge vor der anderen LIBRARY-Komponente.

#### **STATUS({ENABLED|DISABLED})**

Gibt an, ob die LIBRARY-Komponente aktiviert oder inaktiviert werden soll, wenn sie erstellt wird. Wenn eine LIBRARY-Komponente als aktiviert erstellt wird, versucht CICS, die Dateien zuzuordnen und dann zu verknüpfen, bevor die LIBRARY-Verknüpfung geöffnet wird. Wenn einer dieser Schritte fehlschlägt, dann werden die bereits erfolgreich ausgeführten Schritte rückgängig gemacht und die LIBRARY-Komponente wird als inaktiviert installiert. Es wird eine Nachricht ausgegeben, in der angegeben ist, welcher Schritt fehlgeschlagen ist.

Wenn eine LIBRARY-Komponente als inaktiviert erstellt wird, dann versucht CICS nicht, die Dateien zuzuordnen oder zu verknüpfen.

Dieses Attribut wird für LIBRARY-Ressourcen ignoriert, die dynamisch von einem CICS-Bundle generiert werden. Der Anfangsstatus einer LIBRARY-Ressource wird aus dem Anfangsstatus des Bundles abgeleitet, das die Ressource definiert.

---

## **LSRPOOL-Ressourcen**

Die LSRPOOL-Ressource definiert die Größe und die Merkmale des LSR-Pools (LSR = Local Shared Resources; gemeinsam genutzte lokale Ressourcen). Der LSR-Pool stellt eine Reserve von Datenpuffern, Zeichenfolgen und Hiperespace-Puffern dar, die von VSAM verwendet werden, wenn Zugriffsanforderungen für bestimmte Dateien verarbeitet werden.

Ein Hiperespace-Puffer ist ein Hochleistungsspeicherbereich im MVS-Image. Dieser Bereich wird zum Lesen und Schreiben von 4-KB-Seiten verwendet. Der von VSAM verwendete Hiperespace-Typ befindet sich vollständig im Expansionspeicher, einem zusätzlichen Prozessorspeicher, der ausschließlich für das Paging in den und aus dem Realspeicher benutzt wird.

Bis zu 255 LSR-Pools können gleichzeitig auf einem System definiert werden, wobei jeder Pool durch seinen LSRPOOLNUM-Wert identifiziert werden kann. Dieser LSRPOOLNUM-Wert wird verwendet, um eine FILE-Komponente einem LSR-Pool zuzuordnen, wenn diese Datei gemeinsam genutzte Ressourcen verwenden soll.

Wenn die LSRPOOL-Definition im aktiven System installiert ist, dann werden die zugehörigen Informationen gespeichert und verwendet, wenn der Pool mit der angegebenen ID zum nächsten Mal erstellt wird. Ein Pool wird erstellt, wenn die erste Datei, die einen bestimmten LSR-Pool verwendet, geöffnet wird. Die Zuordnung wird nur dann dynamisch aufgehoben, wenn momentan keine Dateien für diesen Pool geöffnet sind. Dies bedeutet, dass eine LSRPOOL-Definition, die auf dem System installiert wird, möglicherweise nicht sofort wirksam wird.

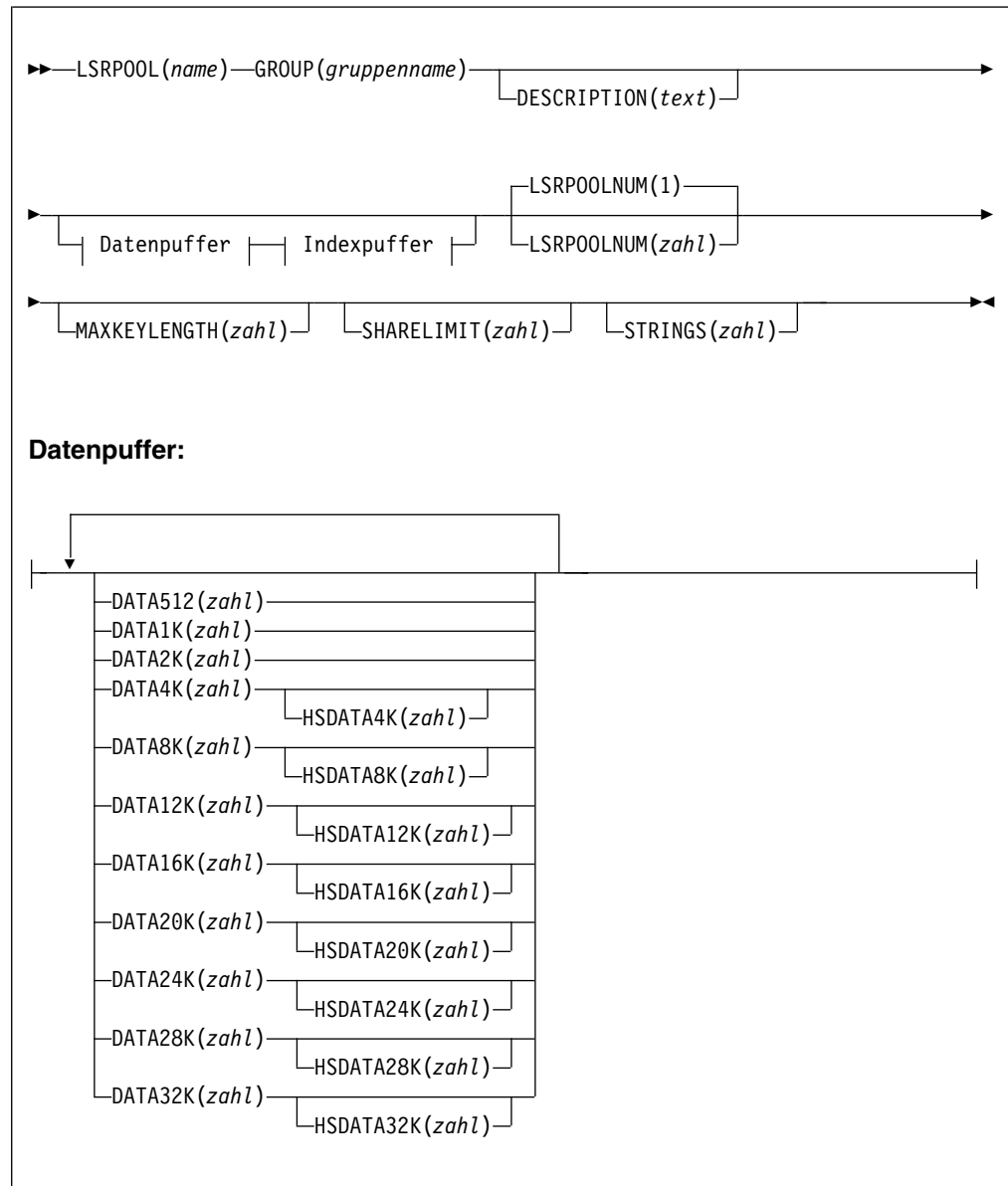
CICS legt Standardattribute fest, wenn kein LSRPOOL-Wert definiert wurde. Aus Gründen der Systemleistung ist es jedoch zu empfehlen, den LSRPOOL-Wert dennoch zu definieren. Auf einem Produktionssystem kann es z. B. zu Verzögerungen kommen, während die Poolanforderungen von CICS berechnet werden. Ein weiteres Problem kann möglicherweise auftreten, wenn Dateien während der Erstellung des Pools nicht zugeordnet werden. In diesem Fall sind die Dateinamen für CICS nicht bekannt. Der Pool wird in einer solchen Situation auf Basis der verfügbaren Informationen erstellt, nachfolgend kann es jedoch zu Leistungseinbußen auf dem System kommen oder das Öffnen von Dateien kann fehlschlagen.

Sie können die CSD-Datei einem bestimmten LSRPOOL zuordnen, indem Sie den Systeminitialisierungsparameter **CSDL SRNO** angeben. Der Standardwert ist Pool 1.

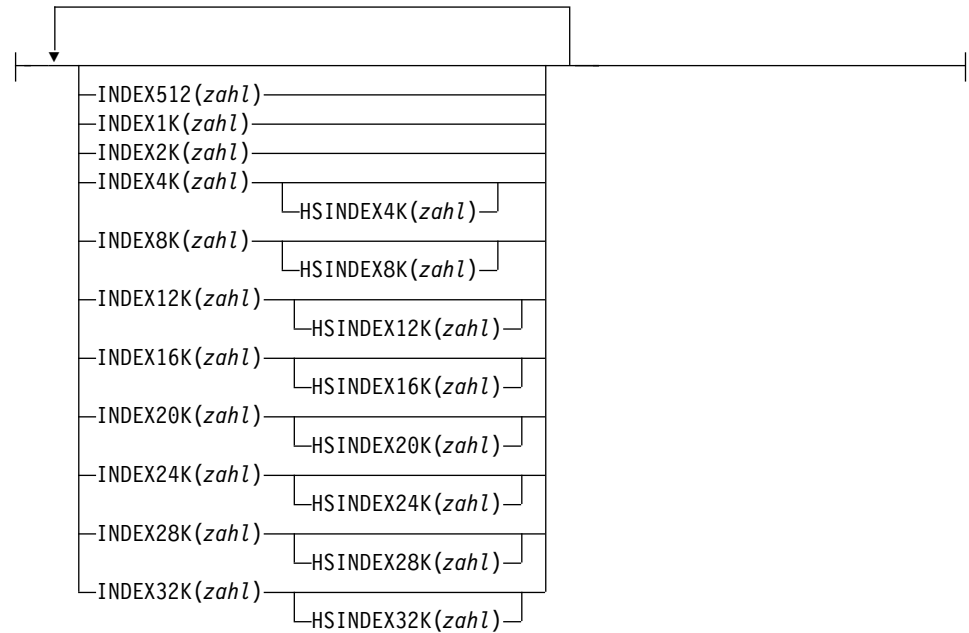
Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in [Working with BAS LSR pool resource definitions](#).

## LSRPOOL-Attribute

Beschreibt die Syntax und die Attribute der LSRPOOL-Ressource.



### Indexpuffer:



### DATA BUFFERS

Gibt die Anzahl der Datenpuffer in allen von Ihnen benötigten Größen im Bereich zwischen 3 und 32767 an. Für diese Attribute sind keine Standardwerte vorhanden.

#### DATA512(zahl)

Gibt die Anzahl der 512-Byte-Datenpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

#### DATA1K(zahl)

Gibt die Anzahl der 1-KB-Datenpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

#### DATA2K(zahl)

Gibt die Anzahl der 2-KB-Datenpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

#### DATA4K(zahl)

Gibt die Anzahl der 4-KB-Datenpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

#### DATA8K(zahl)

Gibt die Anzahl der 8-KB-Datenpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

#### DATA12K(zahl)

Gibt die Anzahl der 12-KB-Datenpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

#### DATA16K(zahl)

Gibt die Anzahl der 16-KB-Datenpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

**DATA20K(zahl)**

Gibt die Anzahl der 20-KB-Datenpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

**DATA24K(zahl)**

Gibt die Anzahl der 24-KB-Datenpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

**DATA28K(zahl)**

Gibt die Anzahl der 28-KB-Datenpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

**DATA32K(zahl)**

Gibt die Anzahl der 32-KB-Datenpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

**DESCRIPTION(text)**

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

**GROUP(gruppenname)**

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

**HIPERSPACE DATA BUFFERS**

Gibt die Anzahl der Hiperspace-Datenpuffer in allen von Ihnen benötigten Größen im Bereich zwischen 0 und 16777215 an. Wenn Sie diese Felder leer lassen, gibt es keine Standardwerte.

**Anmerkung:** Wenn Sie einen Wert für einen Hiperspace-Datenpuffer einer bestimmten Größe angeben, dann müssen Sie auch einen Wert für den Datenpuffer mit derselben Größe angeben.

**HSDATA4K(zahl)**

Gibt die Anzahl der 4-KB-Hiperspace-Datenpuffer im Bereich zwischen 0 und 16777215 an, die Sie benötigen.

**HSDATA8K(zahl)**

Gibt die Anzahl der 8-KB-Hiperspace-Datenpuffer im Bereich zwischen 0 und 16777215 an, die Sie benötigen.

**HSDATA12K(zahl)**

Gibt die Anzahl der 12-KB-Hiperspace-Datenpuffer im Bereich zwischen 0 und 16777215 an, die Sie benötigen.

**HSDATA16K(zahl)**

Gibt die Anzahl der 16-KB-Hiperspace-Datenpuffer im Bereich zwischen 0 und 16777215 an, die Sie benötigen.

**HSDATA20K(zahl)**

Gibt die Anzahl der 20-KB-Hiperspace-Datenpuffer im Bereich zwischen 0 und 16777215 an, die Sie benötigen.

**HSDATA24K(zahl)**

Gibt die Anzahl der 24-KB-Hiperspace-Datenpuffer im Bereich zwischen 0 und 16777215 an, die Sie benötigen.

**HSDATA28K(zahl)**

Gibt die Anzahl der 28-KB-Hiperspace-Datenpuffer im Bereich zwischen 0 und 16777215 an, die Sie benötigen.

**HSDATA32K(zahl)**

Gibt die Anzahl der 32-KB-Hiperspace-Datenpuffer im Bereich zwischen 0 und 16777215 an, die Sie benötigen.

**HIPERSPACE INDEX BUFFERS**

Gibt die Anzahl der Hiperspace-Indexpuffer in allen von Ihnen benötigten Größen im Bereich zwischen 0 und 16777215 an. Wenn Sie diese Felder leer lassen, gibt es keine Standardwerte.

**Anmerkung:** Wenn Sie einen Wert für einen Hiperspace-Indexpuffer einer bestimmten Größe angeben, dann müssen Sie auch einen Wert für den Indexpuffer mit derselben Größe angeben.

**HSINDEX4K(zahl)**

Gibt die Anzahl der 4-KB-Hiperspace-Indexpuffer im Bereich zwischen 0 und 16777215 an, die Sie benötigen.

**HSINDEX8K(zahl)**

Gibt die Anzahl der 8-KB-Hiperspace-Indexpuffer im Bereich zwischen 0 und 16777215 an, die Sie benötigen.

**HSINDEX12K(zahl)**

Gibt die Anzahl der 12-KB-Hiperspace-Indexpuffer im Bereich zwischen 0 und 16777215 an, die Sie benötigen.

**HSINDEX16K(zahl)**

Gibt die Anzahl der 16-KB-Hiperspace-Indexpuffer im Bereich zwischen 0 und 16777215 an, die Sie benötigen.

**HSINDEX20K(zahl)**

Gibt die Anzahl der 20-KB-Hiperspace-Indexpuffer im Bereich zwischen 0 und 16777215 an, die Sie benötigen.

**HSINDEX24K(zahl)**

Gibt die Anzahl der 24-KB-Hiperspace-Indexpuffer im Bereich zwischen 0 und 16777215 an, die Sie benötigen.

**HSINDEX28K(zahl)**

Gibt die Anzahl der 28-KB-Hiperspace-Indexpuffer im Bereich zwischen 0 und 16777215 an, die Sie benötigen.

**HSINDEX32K(zahl)**

Gibt die Anzahl der 32-KB-Hiperspace-Indexpuffer im Bereich zwischen 0 und 16777215 an, die Sie benötigen.

## INDEX BUFFERS

Gibt die Anzahl der Indexpuffer in allen von Ihnen benötigten Größen im Bereich zwischen 3 und 32767 an. Wenn Sie diese Felder leer lassen, gibt es keine Standardwerte.

### INDEX512(*zahl*)

Gibt die Anzahl der 512-Byte-Indexpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

### INDEX1K(*zahl*)

Gibt die Anzahl der 1-KB-Indexpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

### INDEX2K(*zahl*)

Gibt die Anzahl der 2-KB-Indexpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

### INDEX4K(*zahl*)

Gibt die Anzahl der 4-KB-Indexpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

### INDEX8K(*zahl*)

Gibt die Anzahl der 8-KB-Indexpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

### INDEX12K(*zahl*)

Gibt die Anzahl der 12-KB-Indexpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

### INDEX16K(*zahl*)

Gibt die Anzahl der 16-KB-Indexpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

### INDEX20K(*zahl*)

Gibt die Anzahl der 20-KB-Indexpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

### INDEX24K(*zahl*)

Gibt die Anzahl der 24-KB-Indexpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

### INDEX28K(*zahl*)

Gibt die Anzahl der 28-KB-Indexpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

### INDEX32K(*zahl*)

Gibt die Anzahl der 32-KB-Indexpuffer im Bereich zwischen 3 und 32767 an, die Sie benötigen.

## LSRPOOL(*name*)

Gibt den Namen des lokalen gemeinsam genutzten Ressourcenpools an, der definiert wird. Der Name kann bis zu acht Zeichen lang sein.

Wenn nur DATA BUFFERS angegeben ist, wird eine Gruppe von Puffern für den Pool erstellt, der sowohl für die Index- als auch für die Datenkomponenten einer VSAM-KSDS-Datei verwendet werden soll.

Wenn keine Datenpuffer angegeben sind, berechnet CICS die Puffer, die für Daten- und Indexkomponenten benötigt werden. Beide Komponenten nutzen dieselbe Gruppe von Puffern gemeinsam.



Wenn INDEX BUFFERS angegeben wird, werden zwei Teile des Pools erstellt, wobei eines der Teile für Datenpuffer und das andere für Indexpuffer benutzt wird. Wenn Sie INDEX BUFFERS angeben, müssen Sie auch DATA BUFFERS angeben.

**LSRPOOLID**({1}|*lsr-pool*)

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt.

Der für LSRPOOLID in vorhandenen Definitionen angegebene Wert wird an die neue Option LSRPOOLNUM übertragen.

**LSRPOOLNUM**({1}|*zahl*)

Gibt die Kennung des lokalen gemeinsam genutzten Ressourcenpools an, der definiert wird. Der Wert muss im Bereich zwischen 1 und 255 liegen.

**MAXKEYLENGTH**(*zahl*)

Gibt die maximale Schlüssellänge einer der Dateien an, die Ressourcen gemeinsam nutzen sollen. Der Wert muss im Bereich zwischen 0 und 255 liegen. Dieser Wert überschreibt einen Teil der Ressourcenberechnung von CICS. Wird er nicht angegeben, ermittelt CICS die maximal zulässige Schlüssellänge und berechnet sie immer dann neu, wenn die LSR neu erstellt wird.

**SHARELIMIT**(*zahl*)

Gibt den Prozentsatz des maximalen Volumens an VSAM-Ressourcen, die zugeordnet werden, als Ganzzahl an. Als Anzahl kann ein beliebiger Wert zwischen 1 und 100 angegeben werden.

Geben Sie das Attribut SHARELIMIT an, wenn CICS das maximale Volumen an Ressourcen berechnen soll, das von den VSAM-Dateien benötigt wird, die diese Ressourcen gemeinsam nutzen sollen. Da diese Ressourcen gemeinsam genutzt werden sollen, muss ein gewisser Prozentsatz dieses maximalen Volumens an Ressourcen zugeordnet werden. Wenn dieses Attribut nicht angegeben wird, dann werden 50 Prozent des maximalen Volumens an Ressourcen zugeordnet.

Wenn sowohl das Attribut STRINGS als auch das Attribut SIZE angegeben ist, hat die Angabe von SHARELIMIT keine Auswirkungen.

**STRINGS**(*zahl*)

Gibt das Limit (im Bereich zwischen 1 und 255) für die Gesamtzahl der Zeichenfolgen der Dateien im Pool an. Weitere Informationen hierzu finden Sie in Number of buffers and strings for LSR and NSR pools.

---

## MAPSET-Ressourcen

Eine MAPSET-Ressource definiert eine BMS-Maskengruppe.

Jede interaktive Anwendung, die mit einer Bildschirmeinheit arbeitet, kann spezielle Anzeigenlayouts oder Masken verwenden. Diese Elemente werden nicht im Programm selbst angegeben. Stattdessen verwenden Sie BMS (Basic Mapping Support). Hierdurch erhalten Sie mehr Flexibilität und die Möglichkeit, die Masken bei mehreren Aufrufen desselben Programms oder bei verschiedenen Programmen zu verwenden. Sie geben Masken und auch die darin enthaltenen Felder an. Verwenden Sie hierzu die Makros DFHMSD, DFHMDI und DFHMDF. Weitere Anleitungen zu diesem Thema finden Sie in Creating the map.

Anstelle mit den BMS-Maskendefinitionsmakros können Sie Masken auch interaktiv mit dem Lizenzprogramm Screen Definition Facility II (SDF II) mit den Programmnummern 5665-366 (für MVS) und 5664-307 (für VM) definieren. Mithilfe

von SDF II können Sie eine Bildschirmanzeige interaktiv gestalten. Anschließend können Sie die Anzeige generieren, um ein gleichwertiges Element zur CICS/BMS-Maskengruppe zu erhalten. Die Testfunktionen von SDF II ermöglichen Ihnen außerdem die Anzeige Ihrer Maske in ihrer Laufzeitdarstellung. Hintergrundinformationen hierzu finden Sie in Screen Definition Facility II Library.

Eine Anwendung kann während der Interaktion mit dem Benutzer zu unterschiedlichen Zeiten eine Reihe zusammengehöriger Masken verwenden. Außerdem kann sie verschiedene zusammengehörige Masken gleichzeitig und in derselben Anzeige verwenden, um eine vollständige Anzeige zu erstellen.

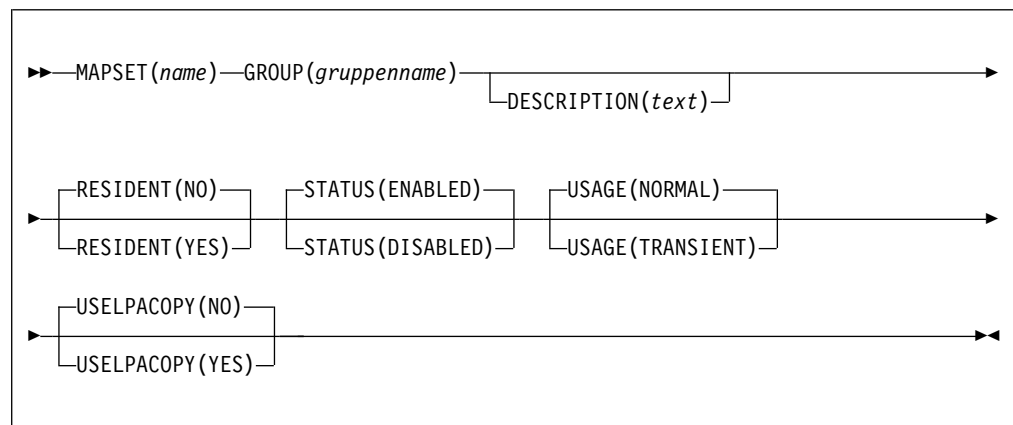
Diese zusammengehörigen Masken gehören zu einer Maskengruppe, die Sie in einer MAPSET-Definition angeben. Auch wenn Ihr Programm über nur eine Maske verfügt, muss diese einer Maskengruppe zugeordnet werden. Sie können Ihre MAPSETs entweder mit CEDA oder DFHCSDUP oder aber durch Festlegen der entsprechenden Systeminitialisierungsparameter definieren, um sie automatisch installieren zu lassen. Weitere Informationen zur automatischen Installation finden Sie in Autoinstalling programs, map sets, and partition sets in Configuring.

Zwischen einem Programm und seinen Maskengruppen besteht keine Verbindung über Ressourcendefinitionen. Stattdessen geben Sie den MAPSET-Namen in den BMS-Befehlen SEND MAP und RECEIVE MAP in Ihrem Programm an.

Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with BAS map set resource definitions.

## MAPSET-Attribute

Beschreibt die Syntax und die Attribute der MAPSET-Ressource.



### DESCRIPTION(text)

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

**GROUP**(*gruppenname*)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

**MAPSET**(*name*)

Dieser Operand gibt den Namen dieser MAPSET-Definition an. Der Name kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Verwenden Sie keine Maskengruppennamen, die mit der Zeichenfolge DFH beginnen, weil diese Zeichen zur Verwendung durch CICS reserviert sind.

Für eine einheitenabhängige BMS-Maskengruppe muss der Maskengruppennamen abgeleitet werden, indem das Maskengruppensuffix zum ursprünglichen (ein bis sieben Zeichen langen) Maskengruppennamen hinzugefügt wird. Das Suffix ist von dem Parameter abhängig, der im Operanden TERM der Makroinstruktion DFHMSD angegeben ist, mit der die Maskengruppe definiert wurde.

Zur Verwendung der einheitenabhängigen Suffixe müssen Sie als Systeminitialisierungsparameter BMS=(,,DDS) angeben. Informationen zu Maskengruppensuffixen finden Sie in Device dependent support.

**RESIDENT**(({NO|YES}))

Gibt den Verfügbarkeitsstatus der Maskengruppe an.

**NO** Die Maskengruppe soll nicht permanent resident sein.

**YES** Die Maskengruppe soll beim ersten Verweis geladen werden und dann permanent im virtuellen Speicher resident bleiben, sie muss jedoch vom Betriebssystem umgelagert werden können.

**RSL**

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt. Weitere Informationen hierzu finden Sie in Obsolete attributes.

**STATUS**(({ENABLED|DISABLED}))

Gibt den Maskengruppenstatus an.

**DISABLED**

Die Maskengruppe kann nicht verwendet werden.

**ENABLED**

Die Maskengruppe kann verwendet werden.

**USAGE**(({NORMAL|TRANSIENT}))

Gibt an, wann der Speicher für diese Maskengruppe freigegeben wird.

### NORMAL

Wenn der Nutzungszähler der Maskengruppe den Wert null erreicht, dann wird sie für die Entfernung aus dem Speicher im Rahmen des normalen dynamischen Speicherkomprimierungsprozesses auswählbar.

### TRANSIENT

Wenn der Nutzungszähler für diese Maskengruppe den Wert null erreicht, dann wird der Speicher für diese Maskengruppe freigegeben. Dieser Wert sollte für Maskengruppen verwendet werden, auf die selten verwiesen wird.

### USELPACOPY({NO|YES})

Gibt an, ob die Maskengruppe über den Link-Pack-Bereich (LPA = Link Pack Area) verwendet werden soll.

#### NO

Die Maskengruppe soll nicht über den LPA verwendet werden. Sie wird in die CICS-Partition geladen.

#### YES

Die Maskengruppe kann über den LPA verwendet werden, wenn LPA=YES als Systeminitialisierungsparameter angegeben ist. Zur Verwendung der Maskengruppe über den LPA ist es erforderlich, dass sie dort installiert wurde und dass die Maskengruppe nicht in der Startoption PRVMOD angegeben ist. Weitere Anleitungen hierzu finden Sie in Controlling the use of modules from the MVS link pack area.

---

## MQCONN-Ressourcen

Eine MQCONN-Ressource definiert die Attribute der Verbindung zwischen CICS und IBM MQ. Sie können eine MQCONN-Ressource nur dann installieren oder verwerfen, wenn CICS nicht mit IBM MQ verbunden ist.

In einer CICS-Region kann immer nur eine MQCONN-Ressource gleichzeitig installiert werden. Wenn Sie eine MQCONN-Ressource definieren, dann überprüft CICS, ob nur eine MQCONN-Ressource in der Gruppe oder Liste definiert ist. Werden mehrere solcher Ressourcen gefunden, wird eine Warnung ausgegeben. Dies gilt auch im Falle unterschiedlicher Namen. Wenn Sie eine zweite MQCONN-Ressource installieren, dann verwirft CICS die vorhandene MQCONN-Ressource implizit, bevor mit der Installation fortgefahren wird, es sei denn, Sie führen eine Neuinstallation einer MQCONN-Ressource mit demselben Namen durch.

Wenn Sie die MQCONN-Ressource installiert haben, dann können Sie den Befehl **CEMT SET MQCONN** oder **EXEC CICS SET MQCONN**, CICSplex SM oder CICS Explorer verwenden, um die Verbindung zwischen CICS und IBM MQ zu starten.

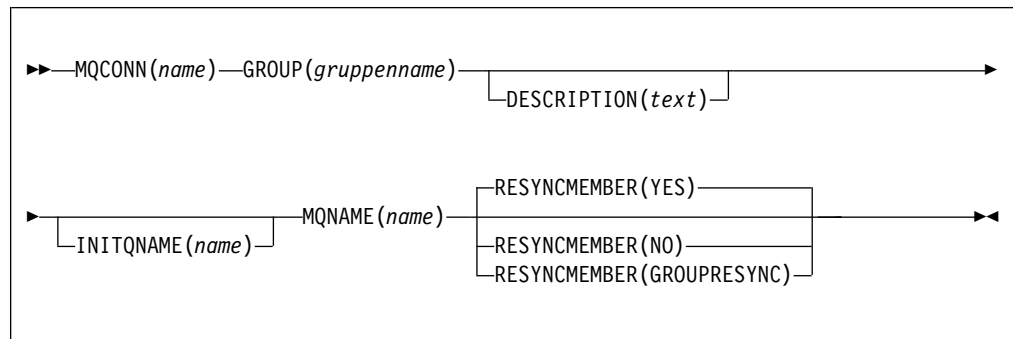
Wenn Sie eine MQCONN-Ressource installieren, die einen Wert im Attribut **INITQNAME** angibt, erstellt und installiert CICS außerdem dynamisch eine MQMONITOR-Ressource mit dem reservierten Namen DFHMQINI, der für die Standard-Initialisierungswarteschlange steht.

Detaillierte Informationen zu DFHMQINI finden Sie in MQMONITOR resources.

. Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with BAS MQCONN resource definitions.

## MQCONN-Attribute

Beschreibt die Syntax und die Attribute der MQCONN-Ressource.



### Attribute

#### DESCRIPTION(text)

In diesem Feld können Sie eine Beschreibung für die Ressource angeben, die Sie definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Es gelten keine Einschränkungen für die Zeichen, die Sie verwenden können. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für jede linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Textcode zwei Hochkommas angeben.

#### GROUP(gruppenname)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

##### Zulässige Zeichen:

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein.

#### INITQNAME(name)

Gibt den aus 1 bis 48 Zeichen bestehenden Namen der standardmäßigen Initialisierungswarteschlange für diese CICS-MQ-Verbindung an.

##### Zulässige Zeichen:

A-Z a-z 0-9 . / \_ %

Wenn Sie das Attribut INITQNAME angeben, dann können Sie einen symbolischen Parameter mit dem Namen *&APPLID*. verwenden, um die Anwendungs-ID (APPLID) einer CICS-Region zu identifizieren. Beispiel: *&APPLID..EXAMPLE.QUEUE.NAME*. Wenn Sie die Ressource in einer CICS-Region installieren, dann wird *&APPLID*. durch die Anwendungs-ID (APPLID) der lokalen Region ersetzt.

Wenn Sie die MQCONN-Ressourcendefinition installieren und ein belegtes Attribut INITQNAME angeben, dann installiert CICS eine MQMONITOR-Ressource mit dem reservierten Namen DFHMQINI, der für die standardmäßige Initialisierungswarteschlange steht. Das Attribut QNAME von DFHMQINI gibt

den Namen der standardmäßigen Initialisierungwarteschlange an. Wenn Sie das Attribut QNAME ändern wollen, dann müssen Sie das Attribut INITQNAME der MQCONN-Ressourcendefinition ändern und dann die MQCONN-Ressourcendefinition neu installieren.

Wenn Sie die CICS-MQ-Verbindung mit dem Befehl **CKQC START** starten und den Namen einer standardmäßigen Initialisierungwarteschlange angeben, dann wird der Wert für INITQNAME der installierten MQCONN-Ressourcendefinition und der Wert für QNAME der installierten MQMONITOR-Ressourcendefinition durch die von Ihnen angegebenen Einstellungen ersetzt.

#### **MQCONN**(*name*)

Gibt den aus acht Zeichen bestehenden Namen dieser Ressourcendefinition an.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

#### **MQNAME**(*name*)

Gibt den aus einem bis vier Zeichen bestehenden Namen eines einzelnen IBM MQ-Warteschlangenmanagers oder einer Gruppe von IBM MQ-Warteschlangenmanagern mit gemeinsamer Warteschlange an.

- Wenn Sie einen einzelnen Warteschlangenmanager angeben und die CICS-MQ-Verbindung starten, dann stellt CICS ausschließlich eine Verbindung zu diesem Warteschlangenmanager her.
- Wenn Sie eine Gruppe mit gemeinsamer Warteschlange angeben und die Verbindung starten, dann stellt CICS eine Verbindung zu jedem aktiven Mitglied dieser Gruppe innerhalb derselben logischen Partition (LPAR) her. Verwenden Sie das Attribut RESYNCMEMBER, um anzugeben, welche Aktionen ausgeführt werden sollen, wenn CICS ausstehende Arbeitseinheiten für den letzten Warteschlangenmanager vorhält, zu dem über die Gruppe mit gemeinsamer Warteschlange eine Verbindung hergestellt wurde.
- Wenn Sie RESYNCMEMBER(GROUPRESYNC) angeben, dann muss im Attribut MQNAME der Name einer Gruppe mit gemeinsamer Warteschlange angegeben sein.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Der Name des Warteschlangenmanagers oder der Gruppe mit gemeinsamer Warteschlange darf nicht mit einem numerischen Zeichen beginnen.

IBM MQ verwendet das Symbol @ zum Auffüllen der Namen von Gruppen mit gemeinsamer Warteschlange, die weniger als vier Zeichen lang sind. Wenn Sie eine Gruppe mit gemeinsamer Warteschlange angeben, deren Name weniger als vier Zeichen lang ist, dann dürfen Sie das Symbol @ am Ende nicht angeben. CICS fügt die Auffüllungszeichen hinzu, bevor die Gruppe mit gemeinsamer Warteschlange abgefragt wird. Beispiel: Wenn die Gruppe mit gemeinsamer Warteschlange für IBM MQ als DEV@ definiert wurde, dann geben Sie DEV als den Wert für das Attribut MQNAME an.

Wenn Sie den Namen eines alternativen IBM MQ-Warteschlangenmanagers oder einer entsprechenden Gruppe mit gemeinsamer Warteschlange im Befehl CEMT oder EXEC CICS SET MQCONN angeben, oder wenn Sie den Namen eines Warteschlangenmanagers beim Starten der CICS-MQ-Verbindung mithilfe

des Befehls CKQC START angeben, dann stellt CICS eine Verbindung zu diesem alternativen Warteschlangenmanager oder zu dieser alternativen Gruppe mit gemeinsamer Warteschlange her. Darüber hinaus wird Ihre Einstellung für das Attribut MQNAME in der installierten MQCONN-Definition durch den Namen des Warteschlangenmanagers oder der Gruppe mit gemeinsamer Warteschlange ersetzt, die Sie in dem Befehl angegeben haben. Wenn Sie das System auf den ursprünglichen Warteschlangenmanager oder die ursprüngliche Gruppe mit gemeinsamer Warteschlange zurücksetzen wollen, dann geben Sie erneut MQNAME an.

#### **RESYNCMEMBER({YES|NO|GROUPRESYNC})**

Dieses Attribut ist nur anwendbar, wenn Sie das Attribut MQNAME zur Angabe einer IBM MQ-Gruppe mit gemeinsamer Warteschlange verwenden.

RESYNCMEMBER gibt die Strategie an, die von CICS angewendet wird, wenn ausstehende Arbeitseinheiten für den letzten Warteschlangenmanager gehalten werden, mit dem CICS über die Gruppe mit gemeinsamer Warteschlange verbunden war.

Die Änderung der Einstellung für RESYNCMEMBER darf nur dann durchgeführt werden, wenn alle Ressourcen sich in einem konsistenten Status befinden. Dies bedeutet, dass keine unbestätigten Arbeitseinheiten ausstehen. Andernfalls kann CICS die Resynchronisation der IBM MQ-Arbeitseinheiten nicht durchführen. Es ist wichtig sicherzustellen, dass alle Ressourcen sich in einem konsistenten Status befinden, bevor RESYNCMEMBER in GROUPRESYNC oder umgekehrt geändert wird.

**YES** CICS stellt eine Verbindung zu demselben Warteschlangenmanager her und wartet dann ggf., bis der Warteschlangenmanager aktiviert wird, um die unbestätigten Arbeitseinheiten aufzulösen. Der Standardwert ist YES.

**NO** CICS versucht ein Mal, eine Verbindung zu demselben Warteschlangenmanager herzustellen. Schlägt dieser Versuch fehl, dann stellt CICS eine Verbindung zu einem beliebigen Mitglied der Gruppe mit gemeinsamer Warteschlange her und gibt die Warnung DFHM2064 aus, in der Sie über die ausstehenden Arbeitseinheiten informiert werden.

#### **GROUPRESYNC**

CICS stellt eine Verbindung zu einem beliebigen Mitglied der Gruppe mit gemeinsamer Warteschlange her. Der Warteschlangenmanager wird von IBM MQ ausgewählt und fordert CICS auf, unbestätigte Arbeitseinheiten für alle auswählbaren Warteschlangenmanager in der Gruppe mit gemeinsamer Warteschlange aufzulösen. Diese Funktion wird als *Arbeitseinheit mit Wiederherstellung für eine Gruppe* bezeichnet. Die Option GROUPRESYNC kann nur verwendet werden, wenn Sie mit WebSphere MQ 7.1 oder höher arbeiten, wenn die von Ihnen verwendete MQ-Version die Arbeitseinheit mit Wiederherstellung für Gruppen für CICS unterstützt und wenn die Arbeitseinheit mit Wiederherstellung für Gruppen in den Warteschlangenmanagern aktiviert wurde.

Hauptsächlich wird die Option GROUPRESYNC für gemeinsam genutzte Warteschlangen verwendet, sie kann jedoch auch für private Warteschlangen benutzt werden. Nachrichten für gemeinsam genutzte Warteschlangen werden sofort aufgelöst, die Auflösung für private Warteschlangen wird jedoch erst dann ausgeführt, wenn der Warteschlangenmanager, der Eigner ist, neu gestartet wird. Wenn Sie mit privaten Warteschlangen arbeiten, dann kann es sinnvoll sein, die Warteschlangen mit QSGDISP(GROUP) zu definieren. QSGDISP(GROUP) stellt sicher, dass eine Instanz der privaten Warteschlange in jedem Ma-

nager in der Gruppe mit gemeinsamer Warteschlange vorhanden ist, sodass bei Auftreten eines Fehlers mit anschließender Neuherstellung der Verbindung zu einem anderen Warteschlangenmanager die Anwendung weiterhin die neue Warteschlangeninstanz nutzen kann. Die Anwendung muss in der Lage sein, die Verwendung einer neuen Warteschlangeninstanz zu tolerieren.

Ändern Sie die Einstellung für RESYNCMEMBER nicht, wenn noch ausstehende Arbeitseinheiten in IBM MQ vorhanden sind, weil Arbeitseinheiten nicht aufgelöst werden können. Eine Arbeitseinheit, die in CICS gehalten wird, wird anhand einer Ressourcenmanagerkennung identifiziert. Wenn RESYNCMEMBER(GROUPRESYNC) verwendet wird, dann wird als Kennung der Name der Gruppe mit gemeinsamer Warteschlange verwendet, andernfalls der Name des einzelnen Warteschlangenmanagers.

Arbeitseinheiten, die als unbestätigt verzögert eingestuft werden, werden nicht in diesen Prozess eingebunden, weil CICS selbst nicht in der Lage ist, diese Arbeitseinheiten zum aktuellen Zeitpunkt aufzulösen. Die Resynchronisation für diese Arbeitseinheiten wird durchgeführt, nachdem CICS die Resynchronisation mit dem fernen Koordinator abgeschlossen hat.

Wenn Sie eine CICS-Systemdefinitionsdatei (CSD) mit einem früheren Release von CICS verwenden, dann wird eine MQCONN-Definition mit der Angabe RESYNCMEMBER(GROUPRESYNC) als RESYNCMEMBER(NO) installiert.

---

## MQMONITOR-Ressourcen

Eine MQMONITOR-Ressource dient zum Definieren von Attributen für IBM MQ-Nachrichtenkonsumenten wie beispielsweise der Transaktion CKTI zum Auslösen der Überwachung. Mit MQMONITOR kann die zugehörige Transaktion, die eine MQ-Warteschlange unterstützt, automatisch neu gestartet werden, wenn die Verbindung zum IBM MQ-Warteschlangenmanager hergestellt wird.

Sie können MQMONITOR-Ressourcen zur Überwachung von MQ-Warteschlangen einschließlich der Initialisierungswarteschlangen einsetzen. Sie können mehrere MQMONITOR-Ressourcen zur Überwachung einer MQ-Warteschlange oder einer MQ-Initialisierungswarteschlange verwenden.

Eine installierte und gestartete MQMONITOR-Ressource überwacht die MQ-Warteschlange, die entweder im Ressourcenattribut **QNAME** angegeben oder für die Transaktion über das Attribut **MONDATA** bereitgestellt wird.

Damit CICS versuchen kann, die Transaktion, die einer MQMONITOR-Ressource zugeordnet ist, automatisch zu starten, muss die MQMONITOR-Ressource mit den Attributen AUTOSTART(YES) und STATUS(ENABLED) installiert worden sein. Wenn die Verbindung zwischen CICS und MQ hergestellt wurde, startet CICS MQ-Monitore, die mit den Attributen AUTOSTART(YES) und STATUS(ENABLED) installiert wurden, sofern die Benutzer-ID, die der Task zugeordnet ist, die die MQCONN-Ressource auf CONNECTED gesetzt hat, über eine ausreichende Berechtigung zum Starten der zugeordneten Transaktionen verfügt.

MQMONITORs werden automatisch gestoppt, wenn CICS die Verbindung zum IBM MQ-Warteschlangenmanager trennt.

Wenn der Zustandsservice von z/OS Workload Manager in der CICS-Region aktiv ist, reagieren die MQMONITORs auf den Zustand der Region und können von der



Festlegung des offenen Status für den Zustandsservice von z/OS Workload Manager abhängig sein. Weitere Informationen finden Sie unter Effect of z/OS Workload Manager Health service on MQMONITORs.

Wenn CICS eine MXT-Bedingung erkennt, wird eine Einschränkung in Bezug auf die Anzahl der MQGET-Aufrufe festgelegt, die eine MQMONITOR-Ressource pro Sekunde absetzen kann, während die Bedingung existiert. Weitere Informationen hierzu finden Sie in Alert monitor (CKAM).

**Anmerkung:** Um festzustellen, ob die MXT-Ausfilterung stattgefunden hat, sollten Sie die Informationen in Interpreting transaction statistics lesen.

Wenn Sie eine MQMONITOR-Ressource für die Überwachung einer MQ-Initialisierungswarteschlange definieren, dann müssen Sie sicherstellen, dass im Attribut **TRANSACTION** entweder CKTI oder kein Wert angegeben ist und dass das Attribut **QNAME** den Namen der MQ-Initialisierungswarteschlange angibt, die überwacht werden soll.

**Anmerkung:**

- Sie können eine MQMONITOR-Ressource nur dann installieren, wenn Sie zuvor eine MQCONN-Ressource installiert haben. Anweisungen hierzu finden Sie in Defining and installing MQMONITOR resources.
- Wenn Sie eine MQMONITOR-Ressource definieren, dann überprüft CICS das System auf Konsistenz mit anderen Ressourcendefinitionen in derselben Gruppe oder Liste. Weitere Informationen hierzu finden Sie in Checks on definitions of MQMONITOR resources.

Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with BAS MQMONITOR resource definitions.

## Sicherheitsaspekte

### Benutzer-ID für Transaktion

Wenn die Sicherheitsprüfung aktiv ist (Systeminitialisierungsparameter **SEC** wurde auf YES gesetzt), dann wird die Benutzer-ID, die der Transaktion zugeordnet ist, anhand des Attributs **MONUSERID** der MQMONITOR-Ressource abgerufen.

Wenn die Sicherheitsprüfung inaktiviert ist (d. h. **SEC** auf 'NO' gesetzt ist), ist die Benutzer-ID, die der MQ-Überwachungstransaktion zugeordnet werden soll, die Benutzer-ID der Transaktion, die den Status der MQMONITOR-Ressource auf "gestartet" gesetzt hat.

### Von CICS beim Festlegen des MQ-Überwachungsstatus "Gestartet" ausgeführte Sicherheitsprüfungen

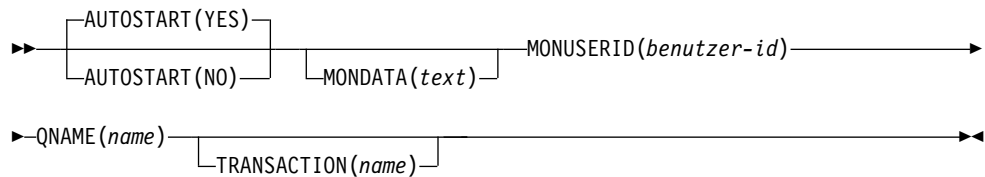
Wenn die Sicherheitsprüfung aktiv ist und der MQ-Überwachungsstatus auf "Gestartet" gesetzt wird, führt CICS die Sicherheitsprüfungen für die Benutzer-ID der Transaktion aus, mit der versucht wird, den MQ-Überwachungsstatus wie folgt auf "Gestartet" zu setzen:

**Anmerkung:** Wird der MQ-Überwachungsstatus über eine CICSplex SM-API-Schnittstelle (wie beispielsweise CICS Explorer) festgelegt, ist die Benutzer-ID, die der MQ-Überwachungstransaktion zugeordnet werden muss, die Benutzer-ID der Region oder die Benutzer-ID für PLTPUIUSR (sofern sie angegeben ist).

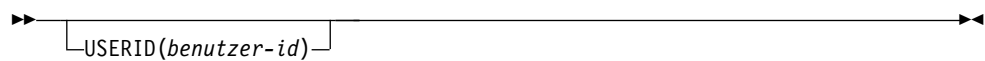
- CICS überprüft, ob die Benutzer-ID ein Ersatz für die Benutzer-ID ist, die in **MONUSERID** definiert wurde.



## Überwachungsattribute für Warteschlangen



## Anwendungsattribute



## Attribute

### AUTOSTART({YES|NO})

Gibt an, ob die MQ-Überwachung automatisch gestartet werden soll, wenn die Verbindung zur IBM MQ-Warteschlange hergestellt wird.

YES Die MQ-Überwachung wird automatisch gestartet, wenn die Verbindung zur IBM MQ-Warteschlange hergestellt wird.

**NO** Die MQ-Überwachung wird nicht automatisch gestartet.

### DESCRIPTION(text)

In diesem Feld können Sie eine Beschreibung für die Ressource angeben, die Sie definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Es gelten keine Einschränkungen für die Zeichen, die Sie verwenden können. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für jede linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Textcode zwei Hochkommas angeben.

### GROUP(gruppenname)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

#### Zulässige Zeichen:

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein.

### MQMONITOR(name)

Gibt den aus acht Zeichen bestehenden Namen dieser Ressourcendefinition an.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

**MONDATA(text)**

Sie können Daten angeben, die an die Transaktion übergeben werden sollen, die die MQ-Warteschlange überwacht. Sie können dieses Feld z. B. verwenden, um Parameter an die MQ-Brückenüberwachungstransaktion CKBR zu übergeben. Das Attribut darf bis zu 200 Zeichen in gemischter Groß-/Kleinschreibung umfassen.

**Anmerkung:**

Wenn die MONDATA-Daten von der Überwachungstask angezeigt und abgerufen werden, werden ihnen folgenden 18 Byte vorangestellt:

Byte 1: < (spitze Klammer links)

Byte 2 - 9: *Name der MQMONITOR-Ressource*

Byte 10 - 17: *USERID*

Byte 18: > (spitze Klammer rechts)

Bytes 19 - 218 enthalten MONDATA wie vom Benutzer eingegeben.

Daher müssen benutzerdefinierte Programme eine maximale Länge von 218 Byte für den Abruf von **MONDATA** zulassen und den in Byte 2-9 der abgerufenen **MONDATA** angegebenen MQMONITOR-Namen verwenden, um das Attribut MONSTATUS des MQ-Monitors zu definieren und dadurch seinen aktuellen Status anzuzeigen. Beachten Sie außerdem, dass bei aktiver Sicherheitsprüfung CICS Sicherheitsprüfungen an der Benutzer-ID durchführt, die der Transaktion zugeordnet ist, welche versucht, den MQ-Überwachungsstatus auf "gestartet" zu setzen. Weitere Informationen finden Sie in der Beschreibung der Sicherheitsaspekte unter MQMONITOR resources.

Alle benutzerdefinierten Zeichenfolgen &applid. oder &APPLID. werden durch die Anwendungs-ID (APPLID) der lokalen Region ersetzt, wenn MQMONITOR installiert wird. Dadurch wird die generische Verwendung dieser Ressource ermöglicht.

**MONUSERID(benutzer-id)**

Gibt die Benutzer-ID an, die der Transaktion zur Überwachung der MQ-Warteschlange zugeordnet werden soll. Dieses Attribut ist nur dann wirksam, wenn die Sicherheitsprüfung aktiviert ist (d. h., der Systeminitialisierungsparameter **SEC** auf 'YES' gesetzt ist). Wenn die Sicherheitsprüfung inaktiviert ist (d. h. **SEC** auf 'NO' gesetzt ist), ist die Benutzer-ID, die der MQ-Überwachungstransaktion zugeordnet werden soll, die Benutzer-ID der Transaktion, die den Status der MQMONITOR-Ressource auf "gestartet" gesetzt hat.

**Anmerkung:** Wenn die Sicherheitsprüfung aktiv ist, dann überprüft CICS, ob die Benutzer-ID, die der Transaktion zugeordnet ist, die den MQ-Überwachungsstatus als "Gestartet" festlegen will, ein Ersatz für die Benutzer-ID ist, die in **MONUSERID** definiert wurde und ob die Berechtigung zum Starten der Transaktionen vorliegt, die **MONUSERID** zugeordnet sind. Wird der MQ-Überwachungsstatus über eine CICSplex SM-API-Schnittstelle (wie beispielsweise CICS

Explorer) festgelegt, ist die Benutzer-ID, die der MQ-Überwachungstransaktion zugeordnet werden muss, die Benutzer-ID der Region oder die Benutzer-ID für PLTPUSR (sofern sie angegeben ist).

**QNAME** (*name*)

Gibt den Namen der MQ-Warteschlange an, die überwacht werden soll.

Wenn Sie das Attribut **QNAME** angeben, dann können Sie einen symbolischen Parameter mit dem Namen *&APPLID.* als Präfix in dem Wert verwenden, um die Anwendungs-ID (APPLID) einer CICS-Region zu identifizieren. Beispiel: *&APPLID..EXAMPLE.QUEUE.NAME*. Wenn Sie die Ressource in einer CICS-Region installieren, dann wird *&APPLID.* durch die Anwendungs-ID (APPLID) der lokalen Region ersetzt.

Wenn **QNAME** ausgelassen wird, wird der Wert während der Installation der Ressource standardmäßig auf *&APPLID..INITIATION.QUEUE* gesetzt.

**STATUS** ({**ENABLED**|**DISABLED**})

Gibt den Anfangsstatus der MQMONITOR-Ressource bei der Installation an.

**DISABLED**

Die MQMONITOR-Ressource steht für die Verwendung nicht zur Verfügung.

**ENABLED**

Die MQMONITOR-Ressource steht für die Verwendung zur Verfügung.

**TRANSACTION** (*name*)

Gibt die aus vier Zeichen bestehende ID der CICS-Transaktion an, die von der Task verwendet wird, die die MQ-Warteschlange überwacht.

Wenn **TRANSACTION** nicht angegeben wird, dann wird standardmäßig der Wert CKTI festgelegt.

**USERID** (*id*)

Gibt die Benutzer-ID an, die standardmäßig zum Starten der Transaktion verwendet wird, die der Anwendung zugeordnet ist, wenn keine passende Benutzer-ID aus einer anderen Quelle verfügbar ist.

Wenn für **USERID** keine Angabe gemacht wird, dann wird die CICS-Standard-Benutzer-ID verwendet.

---

## PARTITIONSET-Ressourcen

Eine PARTITIONSET-Ressource dient zur Definition einer Partitionsgruppe. Hierbei handelt es sich um eine Tabelle, die für CICS beschreibt, wie eine Bildschirmanzeige aufgeteilt werden soll.

Partitionsgruppen werden erstellt, indem eine Reihe von Befehlen angegeben und assembliert wird.

Die Anzeigebereiche bestimmter Bildschirmseinheiten (z. B. 8775 Display Terminal und IBM 3290 Information Panel) können in **Partitionen** unterteilt werden, die als mehrere unterschiedliche kleine Anzeigen behandelt werden können. Unterschiedliche Programme oder Programmschritte einer Transaktion können dann Eingabedaten in unterschiedliche Partitionen schreiben oder aus unterschiedlichen Partitionen empfangen.

Sie geben die Partitionsgruppe mit den Makros DFHPSD und DFHPDI an, deren Verwendung für die Programmierung in Partition set definition erläutert wird.

Sie geben jede unterschiedliche Partitionskonfiguration als PARTITIONSET an. PARTITIONSET-Definitionen werden mit CEDA oder DFHCSDUP erstellt oder können automatisch installiert werden, wenn die entsprechenden Systeminitialisierungsparameter festgelegt wurden. Informationen hierzu finden Sie in *Autoinstalling programs, map sets, and partition sets in Configuring*.

Sie können der PARTITIONSET-Ressource, die von der Transaktion verwendet werden soll, in der TRANSACTION-Definition einen Namen zuordnen. Wenn die Transaktion gestartet wird, dann werden die Informationen in den internen Puffer der Bildschirmeinheit geladen.

Alternativ hierzu kann CICS, wenn keine PARTITIONSET-Ressource angegeben wird, die Bildschirmeinheit auf den Basisstatus setzen, bevor die Transaktion eingeleitet wird.

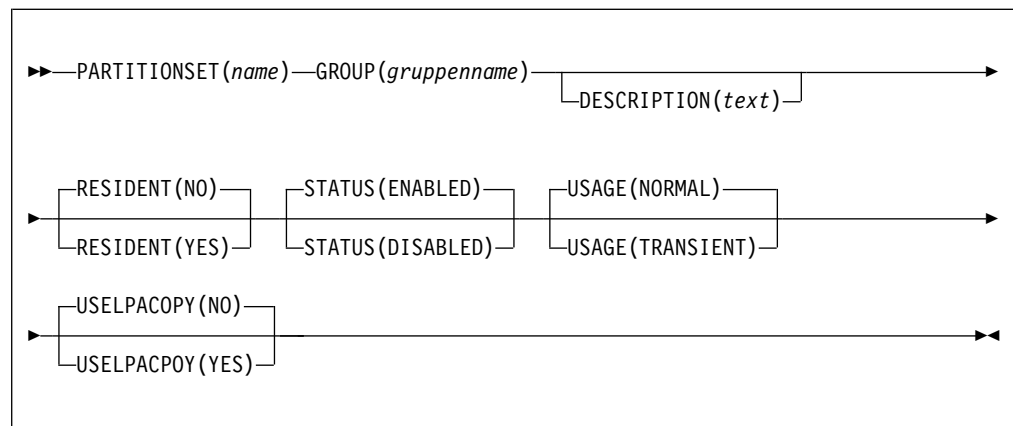
Für die Transaktion muss CICS möglicherweise eine PARTITIONSET-Komponente laden oder zu einer neuen PARTITIONSET-Komponente wechseln, indem der BMS-Befehl SEND PARTNSET abgesetzt wird. Dadurch wird die Partitionsgruppe dynamisch geladen, wenn ihre Definition auf dem aktiven CICS-System installiert wurde.

Anstelle mit den BMS-Partitionsdefinitions makros können Sie Partitionen auch interaktiv mit dem Lizenzprogramm Screen Definition Facility II (SDF II) mit den Programmnummern 5665-366 (für MVS) und 5664-307 (für VM) definieren. Mithilfe von SDF II können Sie eine Bildschirmanzeige interaktiv gestalten. Anschließend können Sie die Anzeige generieren, um ein gleichwertiges Element zur CICS/BMS-Partitionsgruppe zu erhalten. Die Testfunktionen von SDF II ermöglichen Ihnen außerdem die Anzeige Ihrer Partition in ihrer Laufzeitdarstellung. Allgemeine Informationen hierzu finden Sie in *Screen Definition Facility II Library*.

Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in *Working with BAS partition set resource definitions*.

## PARTITIONSET-Attribute

Beschreibt die Syntax und die Attribute der PARTITIONSET-Ressource.



### DESCRIPTION (text)

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie je-

doch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

#### **GROUP**(*gruppenname*)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

#### **PARTITIONSET**(*name*)

Gibt den Namen dieser PARTITIONSET-Definition an. Der Name kann bis zu acht Zeichen lang sein.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Sie sollten keine Namen verwenden, die mit der Zeichenfolge DFH beginnen, weil diese Zeichen zur Verwendung durch CICS reserviert sind.

CICS kann sie löschen oder verwerfen und durch CICS-Ressourcen ersetzen, wenn Sie ein Upgrade durchführen oder neue Funktionen oder Fixes bereitstellen.

Für eine einheitenabhängige Partitionsgruppe muss der Partitionsgruppenname abgeleitet werden, indem das Partitionsgruppensuffix zum ursprünglichen (ein bis sechs Zeichen langen) Partitionsgruppennamen hinzugefügt wird. Das Suffix ist von dem Parameter abhängig, der im Operanden SUFFIX der Makroinstuktion DFHPSD angegeben ist, mit der die Partitionsgruppe definiert wurde.

Zur Verwendung der einheitenabhängigen Suffixe müssen Sie als Systeminitialisierungsparameter BMS=(,,DDS) angeben.

#### **RESIDENT**(**{NO|YES}**)

Gibt den Verfügbarkeitsstatus der Partitionsgruppe an.

**NO** Die Partitionsgruppe soll nicht permanent resident sein.

**YES** Die Partitionsgruppe soll beim ersten Verweis geladen werden und dann permanent im virtuellen Speicher resident bleiben, sie muss jedoch vom Betriebssystem umgelagert werden können.

#### **RSL**

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt. Weitere Informationen hierzu finden Sie in *Obsolete attributes*.

#### **STATUS**(**{ENABLED|DISABLED}**)

Gibt den Partitionsgruppenstatus an.

### **DISABLED**

Die Partitionsgruppe kann möglicherweise nicht verwendet werden.

### **ENABLED**

Die Partitionsgruppe kann verwendet werden.

### **USAGE ( { NORMAL | TRANSIENT } )**

Gibt an, wann der Speicher für diese Partitionsgruppe freigegeben wird.

#### **NORMAL**

Wenn der Nutzungszähler dieser Partitionsgruppe den Wert null erreicht, dann wird sie für die Entfernung aus dem Speicher im Rahmen des normalen dynamischen Programmkomprimierungsprozesses auswählbar.

#### **TRANSIENT**

Wenn der Nutzungszähler für diese Partitionsgruppe den Wert null erreicht, dann wird der Speicher für diese Partitionsgruppe freigegeben. Dieser Wert sollte für Partitionsgruppen verwendet werden, auf die selten verwiesen wird.

### **USELPACOPY ( { NO | YES } )**

Gibt an, ob die Partitionsgruppe über den Link-Pack-Bereich (LPA = Link Pack Area) verwendet werden soll.

#### **NO**

Die Partitionsgruppe soll nicht über den LPA verwendet werden. Sie wird in die CICS-Partition geladen.

#### **YES**

Die Partitionsgruppe kann über den LPA verwendet werden, wenn LPA=YES als Systeminitialisierungsparameter angegeben ist. Zur Verwendung der Partitionsgruppe über den LPA ist es erforderlich, dass sie dort installiert wurde und dass die Partitionsgruppe nicht in der Startoption PRVMOD angegeben ist. Weitere Informationen hierzu finden Sie in Controlling the use of modules from the MVS link pack area.

---

## **PARTNER-Ressourcen**

Eine PARTNER-Ressource ermöglicht CICS-Anwendungsprogrammen über APPC-Protokolle die Kommunikation mit einem Partneranwendungsprogramm, das auf einer fernen logischen Einheit ausgeführt wird.

Die Interaktion zwischen einem CICS-Anwendungsprogramm und einem Partneranwendungsprogramm wird als *Datenaustausch* bezeichnet.

Die PARTNER-Definition ermöglicht außerdem die Verwendung des Aufrufs an die Schnittstelle mit dem SAA-Kommunikationselement (SAA = System Application Architecture). Weitere Informationen zur SAA-Kommunikationsschnittstelle finden Sie in Common Programming Interface Communications (CPI-C) Reference.

Zur Verwendung der SAA-Kommunikationsschnittstelle müssen Sie die folgenden Ressourcen angeben:

- Eine PROFILE-Definition (siehe hierzu „PROFILE-Ressourcen“ auf Seite 136)
- Eine CONNECTION-Definition (siehe hierzu „CONNECTION-Ressourcen“ auf Seite 6)
- Eine SESSIONS-Definition (siehe hierzu „SESSIONS-Ressourcen“ auf Seite 158)

Sie können Ihre CICS-Partnerinformationen auf die folgenden beiden Arten definieren:

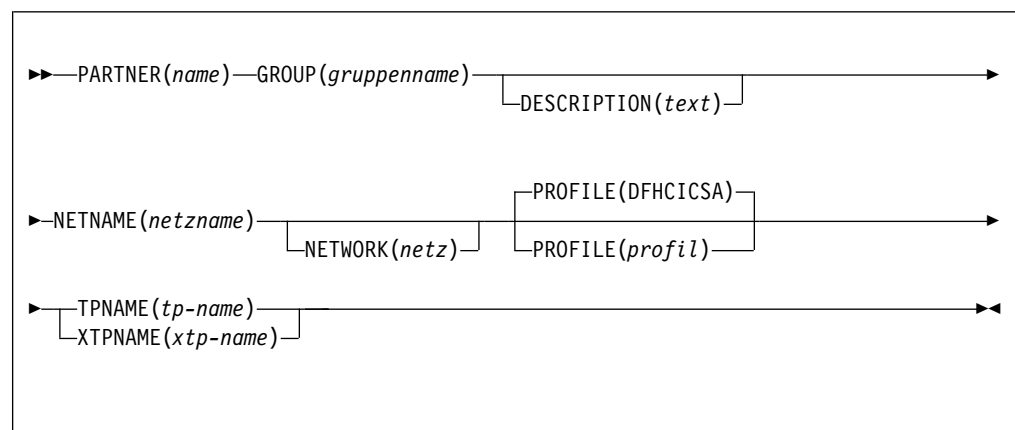


- Erstellen Sie eine PARTNER-Definition.
- Legen Sie in einem Anwendungsprogramm für SYMDESTNAME einen Nullwert fest und setzen Sie die entsprechenden CPI SET-Aufrufe ab. Weitere Detailinformationen finden Sie in Common Programming Interface Communications (CPI-C) Reference.

Informationen zur Installation von PARTNER-Ressourcendefinitionen mit CEMT und CEDA finden Sie in Installing PARTNER resource definitions. Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with BAS partner resource definitions.

## PARTNER-Attribute

Beschreibt die Syntax und die Attribute der PARTNER-Ressource.



### DESCRIPTION(text)

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

### GROUP(gruppenname)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

#### Zulässige Zeichen:

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

### NETNAME(netzname)

Dies ist der Netzname der logischen Einheit, auf der das Partneranwendungs-

programm ausgeführt wird. Er stimmt mit dem Wert des Attributs NETNAME in der CONNECTION-Definition überein. Der Name kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

**NETWORK**(*netz*)

Sie können dieses optionale Attribut verwenden, um den Namen des Netzes anzugeben, in dem sich die Partner-LU befindet. Der Name kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

**PARTNER**(*name*)

Gibt den Namen dieser PARTNER-Definition an. Der Name kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Sie sollten keine Partnernamen verwenden, die mit der Zeichenfolge DFH beginnen, weil diese Zeichen zur Verwendung durch CICS reserviert sind.

CICS kann sie löschen oder verwerfen und durch CICS-Ressourcen ersetzen, wenn Sie ein Upgrade durchführen oder neue Funktionen oder Fixes bereitstellen.

Eine Partnerdefinition gibt die Informationen der SAA-Kommunikationsschnittstelle an, die zum Einrichten eines Datenaustausches mit einem Partnerprogramm benötigt werden. Weitere Anleitungen zu diesem Thema finden Sie in Common Programming Interface Communications (CPI-C) Reference.

**PROFILE**(*profil*)

Gibt den Namen der PROFILE-Definition an, die für die Sitzung und den Datenaustausch verwendet werden soll. Der Standardwert für PROFILE ist DFHCICSA. Der Name kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

**TPNAME**(*tp-name*)

Gibt den Namen eines fernen Transaktionsprogramms an, das auf der Partner-LU ausgeführt werden soll. Die Definition eines fernen TP-Namens ist zwingend erforderlich. Sie müssen entweder TPNAME oder alternativ XTPNAME angeben. Dieser Name kann bis zu 64 Zeichen lang sein.

**Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Wenn dieser Zeichenbereich für einen Namen, den Sie angeben wollen, nicht ausreicht, dann können Sie anstelle von TPNAME das Attribut XTPNAME verwenden.

**XTPNAME**(*xtp-name*)

Dieses Attribut kann alternativ zu TPNAME verwendet werden. Sie **müssen** eines der beiden Attribute angeben, weil die Definition eines fernen TP-Namens zwingend erforderlich ist.

Geben Sie eine hexadezimale Zeichenfolge mit einer Länge von bis zu 128 Zeichen ein, die für den Namen des fernen Transaktionsprogramms steht, das auf der Partner-LU ausgeführt wird. Alle hexadezimalen Kombinationen **mit Ausnahme von X'40'** sind zulässig.

Um einen XTPNAME-Wert mit mehr als 72 Zeichen in DFHCSDUP anzugeben, geben Sie in Spalte 72 einen Stern ein. Dies führt dazu, dass die folgende Zeile verknüpft wird.

---

## PIPELINE-Ressourcen

Eine PIPELINE-Ressource wird verwendet, wenn der CICS-Anwendung die Rolle eines Web-Service-Providers oder Web-Service-Requesters zugeordnet wurde. Sie enthält Informationen zu den Nachrichtenhandlerprogrammen, die eine Serviceanforderung verarbeiten, und zu den entsprechenden Antworten. Normalerweise definiert eine einzelne PIPELINE-Definition eine Infrastruktur, die von zahlreichen Anwendungen genutzt werden kann.

Die Informationen zu den Verarbeitungsknoten werden indirekt zur Verfügung gestellt. Die PIPELINE gibt dabei den Namen einer z/OS UNIX-Datei an, die eine XML-Beschreibung der Knoten sowie deren Konfiguration enthält.

Eine eingehende Web-Service-Anforderung (Anforderung, mit der der Client einen Web-Service in CICS aufruft) wird einer PIPELINE-Ressource mithilfe der URIMAP-Ressource zugeordnet. Die URIMAP-Ressource identifiziert die PIPELINE-Ressource, die für den URI gilt, der der Anforderung zugeordnet ist. Die PIPELINE-Ressource gibt die Verarbeitung an, die für die Nachricht ausgeführt werden soll.

Für Service-Provider, die mit dem CICS-Web-Service-Assistenten bereitgestellt werden, werden die URIMAP-Ressourcen automatisch erstellt, wenn das Abholverzeichnis durchsucht wird. Dieses Scanning wird ausgeführt, wenn die PIPELINE-Ressource installiert wird oder wenn der Befehl **PERFORM PIPELINE SCAN** abgesetzt wird. Die URIMAP-Ressource, die für CICS die Informationen für die Zuordnung der WEBSERVICE-Ressource zu einem bestimmten URI bereitstellt, ist eine erforderliche Ressource. Die Attribute für diese Ressource werden durch eine Datei für Web-Service-Bindungen im Abholverzeichnis angegeben. Die URIMAP-Ressource, die für CICS die Informationen für die Zuordnung der WSDL-Archivdatei oder des WSDL-Dokuments zu einem bestimmten URI bereitstellt, ist eine optionale Ressource. Sie wird erstellt, wenn entweder eine WSDL-Datei oder eine WSDL-Archivdatei im Abholverzeichnis vorhanden ist.

Für Serviceanforderer erstellt CICS nicht automatisch URIMAP-Ressourcen, wenn die PIPELINE-Ressource installiert wird oder wenn der Befehl **PERFORM PIPELINE SCAN** abgesetzt wird.

Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with BAS pipeline resource definitions.

## PIPELINE-Ressourcen in CICS-Bundles

Sie können ein CICS-Bundle verwenden, um eine PIPELINE-Ressourcendefinition zu erstellen, zu bearbeiten und zu installieren. Wenn Sie eine PIPELINE-Ressource auf diese Weise erstellen, dann müssen Sie zum Verwalten des Lebenszyklus dieser Ressource das CICS-Bundle verwenden. Die unabhängige Verwaltung der Ressource ist nicht möglich.

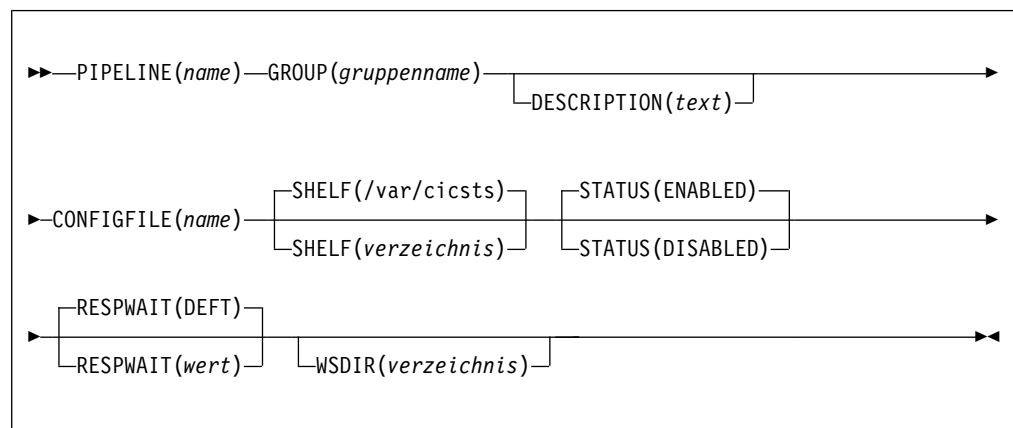
Sie können eine PIPELINE-Ressource abfragen, die von einem CICS-Bundle dynamisch generiert wird. Verwenden Sie hierzu den Befehl **EXEC CICS INQUIRE PIPELINE** oder **CEMT INQUIRE PIPELINE**. Allerdings kann ein Befehl **DISCARD** nicht für eine dynamisch generierte PIPELINE-Ressource verwendet werden. Auch kann der Befehl **SET** nicht zum Ändern des Status der Ressource angegeben werden. Sie müssen die Befehle für die BUNDLE-Ressource absetzen und CICS wendet sie dann auf die PIPELINE-Ressource an.

Sie können die Attribute einer dynamisch generierten PIPELINE-Ressource ändern. Diese Änderungen werden jedoch nicht katalogisiert und können nach einem Warmstart von CICS nicht wiederhergestellt werden. Wenn Sie ein Attribut einer PIPELINE-Ressource ändern wollen, die mithilfe eines Bundles installiert wurde, dann müssen Sie das CICS-Bundle inaktivieren und verwerfen und eine neue Version des Bundles mit den erforderlichen Änderungen installieren.

Weitere Informationen zum Definieren von Ressourcen in CICS-Bundles finden Sie in Defining CICS bundles.

## PIPELINE-Attribute

Beschreibt die Syntax und die Attribute der PIPELINE-Ressource.



### CONFIGFILE(name)

Gibt den vollständig qualifizierten oder relativen Namen einer z/OS UNIX-Datei an, die Informationen zu den Verarbeitungsknoten, die bei Empfang einer Serviceanforderung reagieren, und zu der jeweiligen Antwort enthält.

- Für eine Konfigurationsdatei einer PIPELINE-Ressource, die mithilfe der Onlinersourcendefinition definiert wurde, wird entweder ein vollständig qua-

lizierter Dateipfad (mit einem Schrägstrich (/) als Präfix) oder ein Dateipfad relativ zum Ausgangsverzeichnis der Benutzer-ID für die CICS-Region angeben.

- Für eine Konfigurationsdatei für eine PIPELINE-Ressource, die in einem CICS-Bundle definiert ist, wird der Dateipfad relativ zum Stammverzeichnis des CICS-Bundles angegeben. Die zFS-Datei muss im CICS-Bundle mit der PIPELINE-Ressource gepackt werden. Weitere Informationen hierzu finden Sie in Referencing zFS artifacts in a bundle.

Der angegebene Wert muss ein gültiger Name einer UNIX-Datei sein:

- Er darf keine eingebetteten Leerzeichen enthalten.
- Er darf keine aufeinanderfolgenden Vorkommen des Zeichens / enthalten.
- Die Groß-/Kleinschreibung muss beachtet werden.

**Zulässige Zeichen:**

A-Z a-z 0-9 . / \_ # @ -

**DESCRIPTION**(*text*)

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

**GROUP**(*gruppenname*)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

**PIPELINE**(*name*)

Gibt den Namen dieser PIPELINE-Komponente an. Der Name kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

**RESPWAIT**(*wert*)

Gibt die Anzahl der Sekunden an, die ein Anwendungsprogramm auf eine Antwortnachricht eines fernen Web-Service warten soll. Der Wert kann zwischen 0 und 9999 Sekunden liegen.

Zur Verwendung des Standardzeitlimitwerts des Übertragungsprotokolls geben Sie DEFT an. Der Standardzeitlimitwert wird außerdem auch verwendet, wenn Sie für RESPWAIT keinen Wert angeben.

- Der Standardzeitlimitwert für HTTP ist 10 Sekunden.
- Der Standardzeitlimitwert für WebSphere MQ ist 60 Sekunden.

#### **SHELF({/var/cicsts/|verzeichnis})**

Gibt den aus 1 – 255 Zeichen bestehenden, vollständig qualifizierten Namen eines Verzeichnisses (sog. *Ablage*, primär für Dateien für Web-Service-Bindungen) unter z/OS UNIX an.

Der angegebene Wert muss ein gültiger Name einer UNIX-Datei sein:

- Er darf keine eingebetteten Leerzeichen enthalten.
- Er darf keine aufeinanderfolgenden Vorkommen des Zeichens / enthalten.
- Die Groß-/Kleinschreibung muss beachtet werden.

##### **Zulässige Zeichen:**

A-Z a-z 0-9 . / \_ # @ -

CICS-Regionen, in denen die PIPELINE-Definition installiert ist, müssen über umfassende Berechtigungen auf das Ablageverzeichnis zur Ausführung von Lese- und Schreiboperationen sowie zum Erstellen von Unterverzeichnissen verfügen.

Ein einzelnes Ablageverzeichnis kann von mehreren CICS-Regionen und mehreren PIPELINE-Definitionen gemeinsam genutzt werden. In einem Ablageverzeichnis verwendet jede CICS-Region ein separates Unterverzeichnis, um ihre Dateien getrennt von den Dateien anderer CICS-Regionen aufzubewahren. Innerhalb des Verzeichnisses der einzelnen Regionen verwendet jede PIPELINE ein separates Unterverzeichnis.

Die Inhalte eines Ablagezeichnisses, auf das durch eine installierte PIPELINE-Definition verwiesen wird, dürfen nicht geändert werden. Andernfalls kann es zu unvorhersehbaren Auswirkungen kommen.

Eine PIPELINE-Ressource, die in einem CICS-Bundle mit CICS Explorer definiert wurde, verwendet kein Ablageverzeichnis.

#### **STATUS({ENABLED|DISABLED})**

Gibt den Anfangsstatus der PIPELINE-Ressource bei der Installation an:

##### **ENABLED**

Die Web-Service-Anforderungen für diese PIPELINE werden normal verarbeitet.

##### **DISABLED**

Die Web-Service-Anforderungen für diese PIPELINE können nicht verarbeitet werden.

#### **WSDIR(verzeichnis)**

Gibt den aus 1 – 255 Zeichen bestehenden, vollständig qualifizierten Namen des Verzeichnisses für Web-Service-Bindungen (sog. *Abholverzeichnis*) unter z/OS UNIX an. Für jede PIPELINE, die in einer CICS-Region installiert ist, muss ein anderes Verzeichnis für Web-Service-Bindungen angegeben sein.

Der angegebene Wert muss ein gültiger Name einer UNIX-Datei sein:

- Er darf keine eingebetteten Leerzeichen enthalten.
- Er darf keine aufeinanderfolgenden Vorkommen des Zeichens / enthalten.
- Die Groß-/Kleinschreibung muss beachtet werden.

##### **Zulässige Zeichen:**

A-Z a-z 0-9 . / \_ # @ -

Das Verzeichnis für Web-Service-Bindungen enthält Dateien für Web-Service-Bindungen, die einer PIPELINE zugeordnet sind und die automatisch mithilfe des CICS-Scanning-Mechanismus installiert werden. Wenn die PIPELINE-Definition installiert wird, dann durchsucht CICS das Verzeichnis und installiert automatisch alle Dateien für Web-Service-Bindungen, die dort vorgefunden werden. Beachten Sie hierbei, dass dieser Vorgang unabhängig davon ausgeführt wird, ob die PIPELINE im aktivierten oder inaktivierten Status installiert wird.

Wenn Sie einen Wert für das Attribut WSDIR angeben, dann muss dieser Wert auf ein gültiges z/OS UNIX-Verzeichnis verweisen, auf das die CICS-Region mindestens über Lesezugriff verfügt. Wenn dies nicht der Fall ist, dann schlägt die Installation der PIPELINE-Ressource fehl.

Wenn Sie für WSDIR keinen Wert angeben, dann kann das automatische Durchsuchen (Scanning) bei der Installation der PIPELINE nicht durchgeführt werden.

---

## PROCESSTYPE-Ressourcen

Eine PROCESSTYPE-Ressource dient zur Definition eines BTS-Prozesstyps. Sie benennt die CICS-Datei, die der physischen VSAM-Datei (Repository) zugeordnet ist, in der Details zu allen Prozessen dieses Typs (und der zugehörigen Aktivitätsinstanzen) gespeichert werden.

Mit der CICS BTS-API (BTS = Business Transaction Services) können Sie komplexe Geschäftsanwendungen (sog. *Prozesse*) definieren und ausführen. Ein Prozess wird im Speicher als Speicherblock dargestellt, der die für die Ausführung relevanten Informationen enthält. Darüber hinaus ist dem Prozess mindestens ein zusätzlicher Block mit Informationen zugeordnet (sog. Aktivitätsinstanz). Wenn die Ausführung nicht unter der Steuerung der CICS BTS-Domäne erfolgt, dann werden ein Prozess und seine Aktivitätsinstanzen in eine Datei (sog. *Repository*) geschrieben.

Sie können Ihre BTS-Prozesse bestimmten Prozesstypen zuordnen und so in Kategorien einteilen. Diese Vorgehensweise ist z. B. für Suchvorgänge nützlich. Die Aktivitäten, aus denen ein Prozess besteht, haben denselben Prozesstyp wie der Prozess selbst.

**Anmerkung:** Datensätze für mehrere Prozesstypen können in dieselbe Repository-Datei geschrieben werden.

In Abb. 2 auf Seite 134 wird die Beziehung zwischen PROCESSTYPE-Definitionen, FILE-Definitionen und den BTS-Repository-Dateien dargestellt:

- Mehrere PROCESSTYPE-Ressourcen können auf dasselbe FILE-Element verweisen.
- Mehrere FILE-Elemente können auf dieselbe BTS-Repository-Datei verweisen.

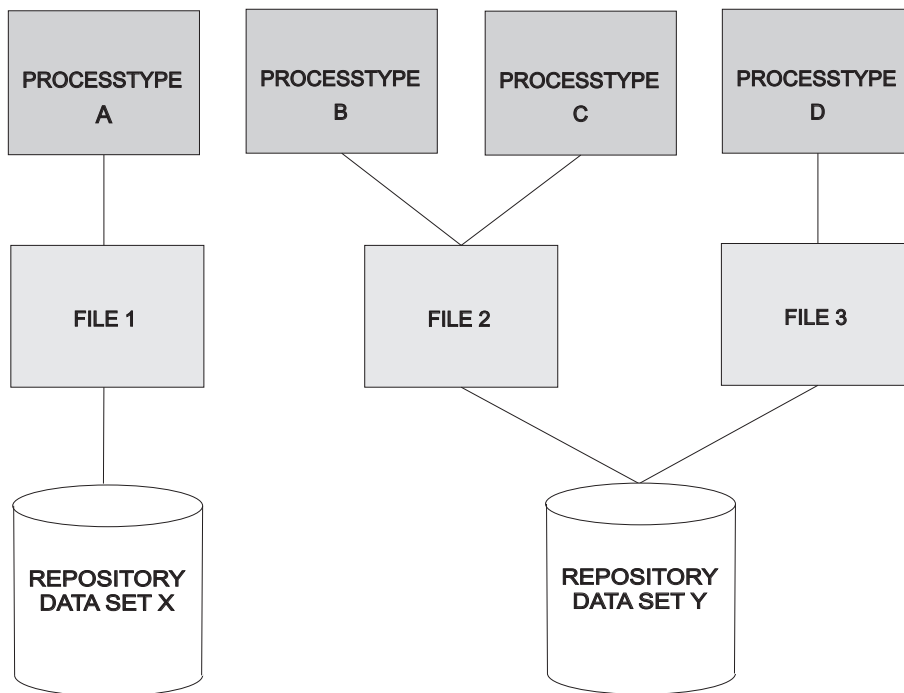


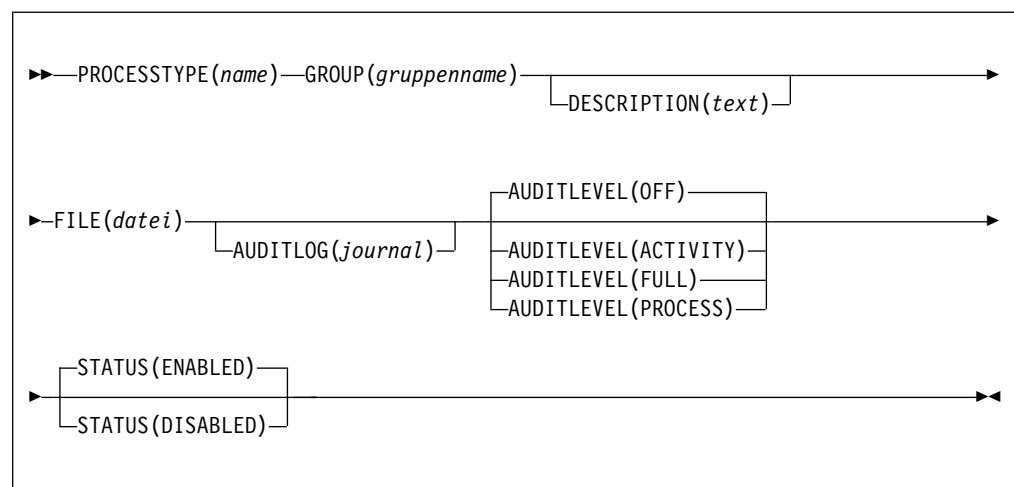
Abbildung 2. PROCESSTYPE-Definitionen, FILE-Definitionen und BTS-Repository-Dateien

Sie können den Fortschritt der BTS-Prozesse und -Aktivitäten zu Prüfzwecken und zur Vereinfachung der Fehlerdiagnose in BTS-Anwendungen aufzeichnen. In diesem Fall können Sie das CICS-Journal angeben, in das die Auditdatensätze geschrieben werden sollen, und die erforderliche Protokollierungsstufe für Prozesse des angegebenen Typs angeben.

Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in *Working with BAS PROCESSTYPE resource definitions*.

## PROCESSTYPE-Attribute

Beschreibt die Syntax und die Attribute der PROCESSTYPE-Ressource.





**AUDITLEVEL({OFF|PROCESS|ACTIVITY|FULL})**

Gibt die Ausgangsstufe der Auditprotokollierung für Prozesse dieses Typs an. Wenn Sie einen anderen Wert als OFF angeben, dann müssen Sie auch die Option AUDITLOG angeben.

**ACTIVITY**

Prüfung auf Aktivitätsebene. Auditdatensätze werden über folgende Komponenten geschrieben:

1. Auditpunkte des Prozesses.
2. Primäre Auditpunkte der Aktivität.

**FULL** Vollständige Prüfung. Auditdatensätze werden über folgende Komponenten geschrieben:

1. Auditpunkte des Prozesses.
2. Primäre *und* sekundäre Auditpunkte der Aktivität.

**OFF** Es werden keine Datensätze für das Prüfprotokoll geschrieben.

**PROCESS**

Prüfung auf Prozessebene. Auditdatensätze werden ausschließlich über die Auditpunkte des Prozesses geschrieben:

Detaillierte Informationen zu den Datensätzen, die über die Auditpunkte des Prozesses sowie die primären und sekundären Auditpunkte der Aktivität geschrieben werden, finden Sie in Specifying the level of audit logging.

**AUDITLOG(*journal*)**

Gibt den Namen eines CICS-Journals an, in das Prüfprotokolldatensätze für Prozesse dieses Typs sowie für die Teilaktivitäten geschrieben werden sollen, aus denen diese Prozesse bestehen. Der Name kann bis zu acht Zeichen lang sein. Wenn Sie kein Auditprotokoll angeben, dann werden keine Auditdatensätze für Prozesse dieses Typs aufbewahrt.

**DESCRIPTION(*text*)**

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

**FILE(*datei*)**

Gibt den Namen der CICS-Dateidefinition an, die zum Schreiben der Prozess- und Aktivitätsdatensätze dieses Prozesstyps in die zugehörige Repository-Datei verwendet wird. Der Name kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

**GROUP(*gruppenname*)**

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

**PROCESSTYPE** (*name*)

Gibt den Namen dieser PROCESSTYPE-Definition an.

**Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % &amp; ? ! : | " = ~ , ; &lt; &gt;

Der Name kann bis zu acht Zeichen lang sein. Führende und eingebettete Leerzeichen sind nicht zulässig. Wenn der angegebene Name weniger als acht Zeichen umfasst, dann wird er mit nachfolgenden Leerzeichen auf eine Länge von acht Zeichen aufgefüllt.

**STATUS** ({**ENABLED**|**DISABLED**})

Gibt den Anfangsstatus des Prozesstyps nach einer CICS-Initialisierung mit START=COLD oder START=INITIAL an. Nach der Initialisierung können Sie den Befehl CEMT SET PROCESSTYPE verwenden, um den Status des Prozesstyps zu ändern. Der Status des Prozesstyps nach einem Neustart wird auf den Status beim vorherigen Systemabschluss zurückgesetzt.

**DISABLED**

Prozesse dieses Typs können nicht erstellt werden. Eine EXEC CICS DEFINE PROCESS-Anforderung, mit der versucht wird, einen Prozess dieses Typs zu erstellen, führt zum Rückgabe der Bedingung INVREQ an das Anwendungsprogramm.

**ENABLED**

Prozesse dieses Typs können erstellt werden.

---

## PROFILE-Ressourcen

Eine PROFILE-Ressource gibt Optionen an, die die Interaktionen zwischen Transaktionen und Terminals oder logischen Einheiten steuern. PROFILE bietet eine Möglichkeit zur Standardisierung der Verwendung von Optionen wie z. B. für die Anzeigegröße und die Druckerkompatibilität und der Verwendung von Funktionen wie zur Journalführung für Nachrichten und des Knotenfehlerprogramms.

**MODENAME**

Ein Profil ist der Kommunikation zwischen einer Transaktion und einer LUTYPE6.1- oder APPC-Sitzung mit einem anderen System zugeordnet. Für APPC-Sitzungen verweisen Sie in der PROFILE-Definition auf das Attribut MODENAME, das auch in den SESSIONS-Definitionen angegeben ist. Dieses Attribut MODENAME gibt den Namen des Modussets an, zu dem die Sitzungen gehören. Weitere Informationen hierzu finden Sie in CONNECTION resources.

Bei Installation in CICS werden die Informationen aus der PROFILE-Definition zum Erstellen eines Eintrags in der Profiltabelle verwendet. Dieser Eintrag wird später von jeder Transaktion verwendet, die auf dieses PROFILE verweist.

Es gibt von CICS bereitgestellte PROFILE-Definitionen, die für die meisten Zwecke geeignet sind. Jede TRANSACTION-Definition benennt den zu verwendenden

PROFILE-Wert. Wenn Sie keinen Wert für PROFILE angeben, dann verwendet die Transaktion den PROFILE-Wert, der für die standardmäßige Verwendung eines Terminals bereitgestellt wurde.

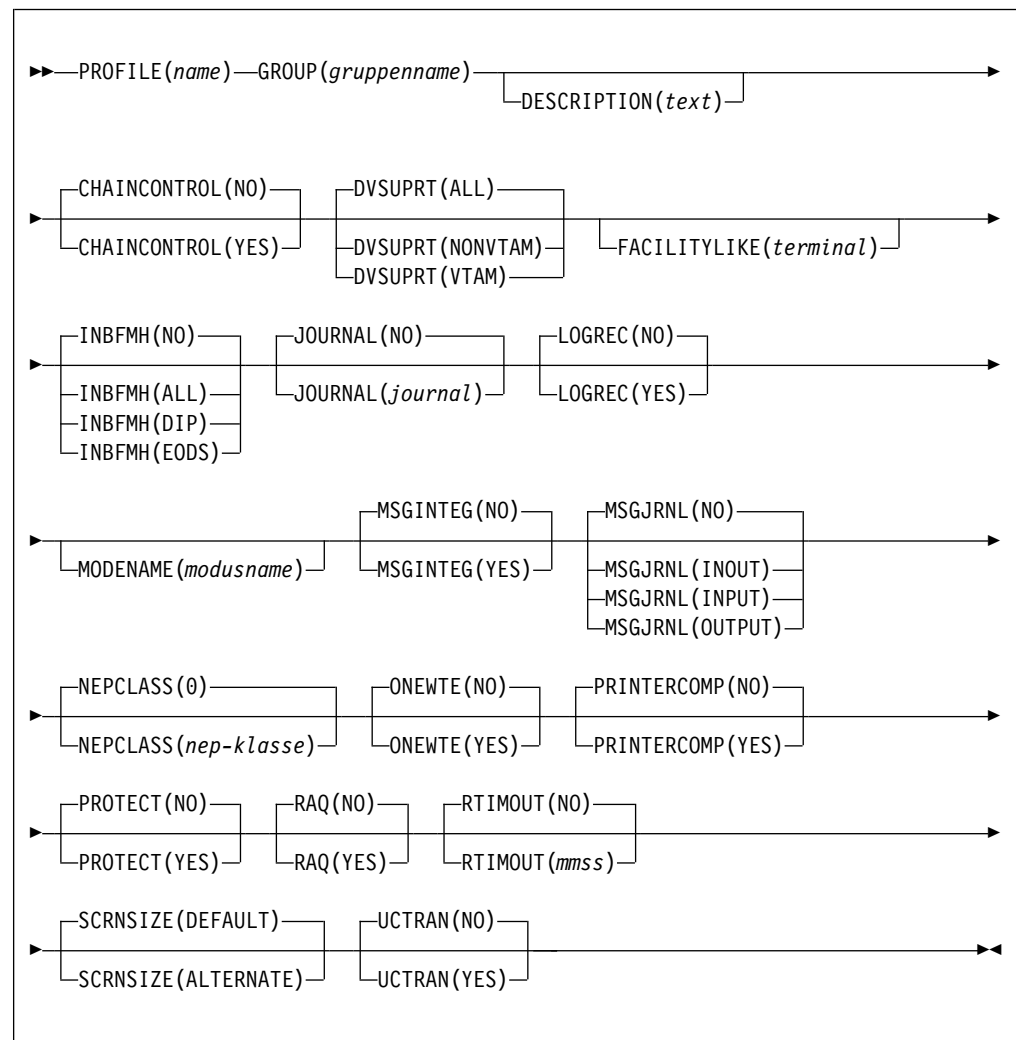
Bei den CICS-Einrichtungen für die übergreifende Kommunikation (z. B. Funktionsverlagerung) wird ein PROFILE-Wert für die Kommunikation zwischen der Transaktion und der Sitzung benötigt. Die Attribute der von CICS bereitgestellten Profile werden in PROFILE definitions in group DFHISC dargestellt. In Defining communication profiles sind weiterführende Informationen zu den von CICS bereitgestellten PROFILES und Erläuterungen zum Definieren eigener Profile enthalten.

Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with BAS profile resource definitions.

## PROFILE-Attribute

Beschreibt die Syntax und die Attribute der PROFILE-Ressource.

**Anmerkung:** VTAM wird jetzt als z/OS Communications Server bezeichnet.



## Attribute

### CHAINCONTROL({NO|YES})

Gibt an, ob das Anwendungsprogramm die abgehende Verkettung von Anforderungseinheiten steuern kann. Wenn Sie CHAINCONTROL(YES) angeben, dann bedeutet ONEWTE(YES) eine Verkettung und nicht eine Ausgabeanforderung für die Terminalsteuerung.

### DESCRIPTION(*text*)

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

### DVSUPRT({ALL|NONVTAM|VTAM})

Gibt die Einheiten (Terminals oder logischen Einheiten) an, die unterstützt werden sollen. Die Zugriffsmethode, die von einem bestimmten Terminal oder einer bestimmten logischen Einheit verwendet wird, wird im zugehörigen Terminaleintrag der Terminalsteuertabelle (TCTTE = Terminal Control Table Terminal Entry) angegeben.

ALL Das Profil kann mit einem beliebigen Terminal oder einer beliebigen logischen Einheit verwendet werden.

#### NONVTAM

Das Profil kann nur mit Terminals verwendet werden, bei denen es sich nicht um z/OS Communications Server-Terminals handelt.

#### VTAM

Das Profil kann nur mit logischen Einheiten verwendet werden.

### FACILITYLIKE(*terminal*)

Dies ist ein optionales Attribut, das den Namen einer bereits existierenden (vier Zeichen langen) Terminalressourcendefinition angibt, die als Schablone für die Brückeneinrichtung verwendet wird. Der Wert kann überschrieben werden, indem im Brückenexit FACILITYLIKE angegeben wird.

Für dieses Attribut gibt es keinen Standardwert.

Wenn Sie die Verarbeitung in einem CICS-System ausführen, das mit dem Systeminitialisierungsparameter (SIT) VTAM=NO gestartet wurde, dann muss die in FACILITYLIKE angegebene Ressourcendefinition als fernes Terminal definiert werden.

### GROUP(*gruppenname*)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

#### Zulässige Zeichen:

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

**INBFMH({NO|ALL|DIP|EODS}) (nur SNA-LUs)**

Gibt für die mit logischen Einheiten benutzten Profile an, ob ein Funktionsverwaltungsheader (FMH = Function Management Header), der von einer logischen Einheit empfangen wurde, an das Anwendungsprogramm übergeben werden soll.

- ALL** Alle FMHs (mit Ausnahme der APPC-FMHs, LU6.1 ATTACH- und SYNCPOINT-FMHs, die von CICS verarbeitet werden) werden an das Anwendungsprogramm übergeben. Dieser Wert ist für Funktionsverlagerungstransaktionen wie z. B. CSMI, Transaktionen, die die verteilte Transaktionsverarbeitung nutzen, und für DPL-Anforderungen (DPL = Distributed Program Link; Verbindung zu verteilten Programmen) erforderlich.
- DIP** Eingehende FMHs werden mit DFHDIP (Batch Data Interchange Program; Stapeldatenaustauschprogramm) verarbeitet. BMS gibt eine Empfangsanforderung für den Stapeldatenaustausch aus, wenn eine BMS-Empfangsanforderung abgesetzt wurde. Außerdem wird eine Empfangsanforderung für den Stapeldatenaustausch anstelle einer Empfangsanforderung für die Terminalsteuerung abgesetzt.
- EODS** Ein FMH wird an das Anwendungsprogramm nur dann übergeben, wenn er einen EODS-Anzeiger (EODS = End of Dataset; Dateiende) enthält.
- NO** Die FMHs werden verworfen.

**JOURNAL({NO|*journal*})**

Gibt an, dass Sie die automatische Journalführung für Nachrichten aktivieren möchten, indem Sie die Kennung des Journals angeben.

- NO** Das System führt keine automatische Journalführung für Nachrichten aus.

*journal* Die Journalkennung, die für die automatische Journalführung verwendet werden soll. Hier kann eine beliebige Zahl im Bereich zwischen 01 und 99 angegeben werden. Diese Zahl wird an die Buchstaben DFHJ angefügt, um eine Journalkennung im Format DFHJnn zu erhalten, die einem allgemeinen Protokollstrom der MVS-Systemprotokollfunktion zugeordnet wird.

**Anmerkung:** In CICS Transaction Server for z/OS stellt DFHJ01 nicht das Systemprotokoll dar.

In einer Umgebung mit Transaktionsrouting wird die Journalführung für Nachrichten in der Anwendungsverwaltungsregion (AOR = Application-Owning Region) ausgeführt. Aus diesem Grund sollten Sie das Attribut JOURNAL im Transaktionsprofil der AOR angeben.

**LOGREC({NO|YES})**

Gibt an, ob es für das Design der Anwendung erforderlich ist, dass jede EXEC CICS RECEIVE-Anforderung mit einem logischen Datensatz beantwortet wird. Diese Option erlaubt den Anschluss bereits vorhandener 2770- und 2780-basierter Anwendungsprogramme an eine logische Stapelverarbeitungseinheit (z. B. 3790 oder 8100), ohne dass hierzu Änderungen am Programm erforderlich werden.

**MODENAME(modusname)**

Gibt den Namen an, der eine Gruppe von Sitzungen zur Verwendung über eine APPC-Verbindung identifiziert. Der Name darf eine Länge von maximal acht Zeichen haben und muss den Namen eines LOGMODE-Eintrags von

z/OS Communications Server angeben, der in z/OS Communications Server definiert wurde. Der reservierte Name SNASVCMG darf hier nicht verwendet werden. Wenn Sie den Modusnamen nicht angeben, werden standardmäßig Leerzeichen angenommen. Weitere Informationen zu z/OS Communications Server-Modusnamen finden Sie in Defining groups of APPC sessions.

Wenn eine Transaktion, die dieses Profil angibt, mit dem Befehl EXEC CICS START gestartet wurde, dann wird MODENAME für die Zuordnung der Hauptfunktion verwendet. Wenn eine Transaktion einen Befehl EXEC CICS ALLOCATE ausführt, in dem dieses Profil angegeben ist, dann wird MODENAME für die Zuordnung der alternativen Funktion verwendet.

Wenn Sie MODENAME nicht angeben, dann wählt CICS eine Sitzung aus einem der Modussets aus, die definiert wurden.

Das von CICS bereitgestellte Profil DFHCICSA wird verwendet, wenn PROFILE im Befehl EXEC CICS ALLOCATE nicht angegeben wird. Für die Funktionsverlagerung wird immer das Profil DFHCICSF verwendet. MODENAME wird in keiner der Definitionen für diese Profile angegeben, Sie können MODENAME jedoch hinzufügen, wenn Sie eine eigene Kopie erstellen. Sie müssen dann sicherstellen, dass die Modussets, die mit Ihrem MODENAME-Wert arbeiten, in der TERMINAL- oder SESSIONS-Definition für alle Systeme definiert wurden, mit denen über APPC kommuniziert wird.

Wenn ein Wert für MODENAME angegeben wird und wenn Sie diesen Wert entfernen wollen, dann müssen Sie den zuvor angegebenen Wert vollständig löschen, indem Sie die Taste für ERASE EOF drücken.

#### **MSGINTEG({NO|YES})) (nur SNA-LUs)**

Gibt an, ob eine unbedingte Antwort mit einer Ausgabeanforderung an eine logische Einheit angefordert werden muss. Für eine Pipelinetransaktion darf YES nicht angegeben werden.

#### **MSGJRNL({NO|INPUT|OUTPUT|INOUT}))**

Gibt an, welche Nachrichten automatisch im Journal aufgezeichnet werden sollen. Wenn Sie einen anderen Wert als NO angeben, müssen Sie auch einen Wert für das Attribut JOURNAL angeben.

**NO** Es ist keine Journalführung für Nachrichten erforderlich.

#### **INPUT**

Die Journalführung ist für Eingabenachrichten erforderlich.

#### **OUTPUT**

Die Journalführung muss für Ausgabenachrichten ausgeführt werden.

#### **INOUT**

Die Journalführung muss für Ein- und Ausgabenachrichten ausgeführt werden.

In einer Umgebung mit Transaktionsrouting wird die Journalführung für Nachrichten bei APPC-Sitzungen mit Routing (LU Type 6.2-Sitzungen) in der Anwendungsverwaltungsregion (AOR = Application-Owning Region) ausgeführt. Sie müssen das Attribut MSGJRNL im Transaktionsprofil der Anwendungsverwaltungsregion angeben. Für andere Sitzungen mit Routing wird die Journalführung für Nachrichten in der Terminalverwaltungsregion (TOR = Terminal-Owning Region) ausgeführt. In diesem Fall muss das Attribut MSGJRNL im Transaktionsprofil der Terminalverwaltungsregion angegeben werden.

#### **NEPCLASS({0|nep-klasse})) (nur z/OS Communications Server)**

Gibt die NEP-Transaktionsklasse (NEP = Node Error Program; Knotenfehler-

programm) an. Dieser Wert überschreibt den Wert, der in den TYPETERM- und SESSION-Definitionen angegeben ist.

**0** Die Angabe dieses Werts führt dazu, dass eine Verbindung zum standardmäßigen NEP-Modul für z/OS Communications Server-Einheiten hergestellt wird, oder es handelt sich um den Standardwert für Nicht-z/OS Communications Server-Einheiten.

*wert* Die Transaktionsklasse für das (nicht standardmäßige) NEP-Modul. Der Wert kann im Bereich zwischen 1 und 255 liegen. Informationen zur Programmierung für das Knotenfehlerprogramm (NEP) finden Sie in Writing a node error program.

Das Attribut NEPCCLASS gilt nur für Benutzertransaktionen und wird für SNASVCMGR-Sitzungen ignoriert.

#### **ONEWTE({NO|YES})**

Gibt an, ob die Transaktion während der Ausführung nur eine Schreiboperation oder einen Befehl EXEC CICS SEND verwenden darf. Die Angabe von YES hat zur Folge, dass in der ersten Schreiboperation der Transaktion zwingend die Option LAST verwendet wird. Alle zusätzlichen Schreibanforderungen werden als Fehler behandelt und die Task wird für die abnormale Beendigung vorbereitet.

Für eine PIPELINE-Transaktion muss YES angegeben werden.

#### **PRINTERCOMP({NO|YES})**

Gibt den Grad der Kompatibilität an, die für die Generierung von Datenströmen zur Unterstützung der Druckerkompatibilitätsoption für den BMS-Befehl SEND TEXT erforderlich ist.

**NO** Jede Zeile der Ausgabe beginnt mit einem Leerzeichen, sodass das Format gleichbedeutend mit dem Format in einer 3270-Anzeige ist, in der jeder Zeile ein Attributbyte vorangestellt wird.

**YES** Es wird kein Leerzeichen eingefügt, sodass Formularvorschubzeichen, die als erstes Zeichen Ihrer Daten eingebunden wurden, berücksichtigt werden und die volle Breite des Druckers für Ihre Daten zur Verfügung steht.

Wenn Sie die BMS-Option für den Formularvorschub verwenden, dann geben Sie YES an.

#### **PROFILE(name)**

Gibt den Namen dieser PROFILE-Definition an. Der Name kann bis zu acht Zeichen lang sein.

##### **Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Sie sollten keine Namen verwenden, die mit der Zeichenfolge DFH beginnen, weil diese Zeichen zur Verwendung durch CICS reserviert sind.

CICS kann sie löschen oder verwerfen und durch CICS-Ressourcen ersetzen, wenn Sie ein Upgrade durchführen oder neue Funktionen oder Fixes bereitstellen.

**Anmerkung:** Wenn Sie in einem Namen ein Komma (,) verwenden, dann können Befehle wie der folgende nicht verwendet werden:

CEMT INQUIRE PROFILE(*wert1,wert2*)

Hierbei dient das Komma als Listenbegrenzer. Informationen zur Verwendung von Listen mit Ressourcen-IDs finden Sie in List of resource identifiers.

In einem Profil werden die Optionen angegeben, die die Interaktion zwischen CICS und einem Terminal oder einer logischen Einheit steuern. Ein Profilname wird in der Transaktionsdefinition angegeben, um die Optionsgruppe anzugeben, die die Kommunikation zwischen der Transaktion und dem Hauptterminal steuert. Sie können einen Profilnamen auch im Befehl EXEC CICS ALLOCATE angeben, um die Optionen anzugeben, die die Kommunikation zwischen der Transaktion und der zugeordneten Sitzung steuern.

CICS stellt eine Reihe von Profildefinitionen bereit, die für die meisten Zwecke geeignet sind. Weitere Informationen hierzu finden Sie in Defining communication profiles.

#### **PROTECT**

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt. Weitere Informationen hierzu finden Sie in Obsolete attributes.

#### **RAQ({NO|YES})) (nur SNA-Terminals)**

Gibt an, ob die Option 'Read ahead queuing' erforderlich ist.

**NO** Die Transaktion unterliegt den SNA-Protokollen und die Befehle SEND und RECEIVE können nur dann ausgeführt werden, wenn sich das System im korrekten Modus befindet. Wird das Protokoll nicht beachtet, dann erfolgt möglicherweise eine abnormale Beendigung der Verarbeitung mit dem Code ATCV.

**YES** Die Transaktion arbeitet möglicherweise nicht auf Basis der SNA-Protokolle und eingehende Daten werden von CICS möglicherweise in die TS-Warteschlange eingestellt, bis die Daten von der Transaktion gezielt angefordert werden. RAQ(YES) wird nur aus Gründen der Kompatibilität mit Transaktionen bereitgestellt, die sowohl bisynchrone Einheiten als auch logische Einheiten unterstützen. Die Verwendung wird allerdings nicht empfohlen.

#### **RTIMOUT({NO|mmss}))**

Gibt den Zeitlimitwert für die Funktion für Lesezeitlimits an. Für Tasks, die HTTP-Clientanforderungen mit der EXEC CICS WEB-API erstellen, wenn dieses Intervall beendet wird, gibt CICS eine TIMEDOUT-Antwort an die Anwendung zurück. Für andere Tasks gibt der fehlgeschlagene Befehl entweder TERMERR zurück oder die Task, die das Zeitlimit überschritten hat, empfängt eine abnormale Beendigung mit dem Code AKCT, AZCT oder AZIG.

**NO** Die Funktion für Lesezeitlimits ist nicht erforderlich.

**wert** Dies ist ein Intervall (MMSS für Minuten und Sekunden), nach dessen Ablauf die Task beendet wird, wenn keine Eingabedaten vom Terminal empfangen wurden. Der maximal zulässige Wert sind 70 Minuten. Wenn ein Wert angegeben ist und wenn Sie den Standardwert NO verwenden möchten, dann müssen Sie den zuvor angegebenen Wert vollständig löschen.

#### **SCRNSIZE({DEFAULT|ALTERNATE}))**

Gibt an, ob die standardmäßige oder alternative Puffergröße (DEFAULT oder ALTERNATE) für eine 3270-Anzeige oder einen entsprechenden Drucker verwendet werden soll. Weitere Informationen zur Auswahl von Anzeigegrößen und Puffergrößen finden Sie in den Attributen ALTSCREEN und DEFSCREEN in der TYPETERM-Definition.



Der SCRNSIZE-Wert wird ignoriert, wenn in der TYPETERM-Definition ALTSCREEN(0,0) und DEFSCREEN(0,0) angegeben sind. Dies bedeutet, dass die Anzeigegröße aus dem zugehörigen Attribut TERMMODEL in der TYPETERM-Definition abgeleitet wird. Die Seitengröße wird anhand von PAGESIZE ermittelt und der Wert für ALTPAGE wird ignoriert. Der 3270-Befehl EW (Erase Write) wird für Ausgabeanforderungen mit der Option ERASE eingefügt.

#### **ALTERNATE**

Wenn die TYPETERM-Definition einen Wert ungleich null für ALTSCREEN enthält, dann wird der Modus für die alternative Anzeigegröße angewendet. Dazu wird der Befehl EWA (Erase Write Alternate) benutzt. Dies bedeutet, dass bei Absetzen einer Terminalausgabeanforderung mit der Option ERASE der 3270-Befehl EWA in den Datenstrom eingefügt wird. Der Wert für ALTSCREEN wird als Anzeigegröße angenommen und BMS verwendet den Wert in ALTPAGE als Seitengröße.

SCRNSIZE(ALTERNATE) kann für alle CICS-Servicetransaktionen (z. B. CSMT) verwendet werden.

#### **DEFAULT**

Wenn die TYPETERM-Definition einen Wert ungleich null für ALTSCREEN oder DEFSCREEN enthält, dann wird der Modus für die Standardanzeigegröße angewendet. Dazu wird der Befehl EW (Erase Write) benutzt. Dies bedeutet, dass bei Absetzen einer Terminalausgabeanforderung durch das Terminal mit der Option ERASE der 3270-Befehl EW in den Datenstrom eingefügt wird. Die Anzeigegröße, die im Attribut DEFSCREEN angegeben ist, wird verwendet und BMS benutzt den im Attribut PAGESIZE festgelegten Wert als Seitengröße.

**Anmerkung:** Sowohl DEFAULT als auch ALTERNATE können von den Optionen DEFAULT und ALTERNATE in den Befehlen SEND MAP, SEND TEXT und SEND CONTROL überschrieben werden.

#### **UCTRAN({NO|YES}) (nur z/OS Communications Server)**

Gibt an, ob die Terminaleingabe in Großbuchstaben umgesetzt werden muss, bevor sie für die Transaktion, die dieses Profil verwendet, an Programme übergeben wird.

Sie können die Umsetzung in Großbuchstaben auch auf Terminalebene in der zugehörigen TYPETERM-Definition anfordern (siehe hierzu „TYPETERM-Attribute“ auf Seite 279). Dabei müssen Sie jedoch die folgenden Faktoren berücksichtigen:

- Die Definition von TYPETERM UCTRAN(YES) überschreibt die Definition von PROFILE UCTRAN(NO). Wenn Sie also TYPETERM UCTRAN(YES) angeben, dann hat PROFILE UCTRAN(NO) keine Auswirkung.
- Die Definition von PROFILE UCTRAN(YES) überschreibt die Definition von TYPETERM UCTRAN(NO).
- Die Angabe von TYPETERM UCTRAN(TRANID) führt dazu, dass die Transaktions-ID (tranid) in Großbuchstaben umgesetzt wird, sodass CICS die Transaktionsdefinition lokalisieren kann. Alle anderen Eingaben, die von der Anwendung empfangen werden, werden entsprechend der für PROFILE UCTRAN angegebenen Einstellung umgesetzt.
- UCTRAN(YES) in einer Profildefinition führt erst dann zur Umsetzung der Eingabedaten, wenn ein Befehl EXEC CICS RECEIVE oder CONVERSE ausgeführt wird. Dies bedeutet, dass beim Routing einer Transaktion über ein Programm für dynamisches Routing (z. B. DFHDYP) die Kopie der Eingabe-

daten, die an das Routing-Programm übergeben wird, nicht von der Angabe der Option UCTRAN der PROFILE-Definition beeinflusst wird.

**Anmerkung:** In einer Umgebung mit Transaktionsrouting, in der die z/OS Communications Server-Terminals über eine ferne Definition in der Anwendungsverwaltungsregion (AOR = Application-Owning Region) verfügen und die AOR über einen anderen Wert für UCTRAN aus der Terminalverwaltungsregion (TOR = Terminal-Owning Region) verfügt, der TOR-Wert von UCTRANST (wie im Befehl EXEC CICS SET TERMINAL angegeben) den entsprechenden Wert der AOR überschreibt.

In Tabelle 2 wird gezeigt, welcher Teil der Terminaleingabe (Transaktions-ID und/oder Daten) anhand der Einstellung von UCTRAN in der PROFILE- und der TYPETERM-Ressourcendefinition umgesetzt wird.

*Tabelle 2. Auswirkung von UCTRAN-Attributen auf Umsetzung von Transaktions-IDs (TRANID) und Daten*

UCTRAN in PROFILE	UCTRAN in TYPETERM	TRANID umgesetzt?	Daten umgesetzt?
YES	YES	Ja	Ja
YES	NO	Nein	Ja
YES	TRANID	Ja	Ja
NO	YES	Ja	Ja
NO	NO	Nein	Nein
NO	TRANID	Ja	Nein

Bestimmte nationale Sonderzeichen werden nicht automatisch umgesetzt, wenn UCTRAN(YES) angegeben wird. In diesem Fall können Sie eine der in Translating national characters to uppercase beschriebenen Methoden verwenden.

---

## PROGRAM-Ressourcen

PROGRAM-Ressourcen steuern Informationen für ein Programm, das in der Programmbibliothek gespeichert ist und zum Verarbeiten einer Transaktion oder eines Teils einer Transaktion verwendet wird.

Sie können PROGRAM-Definitionen entweder mit CEDA oder DFHCSDUP oder aber durch Festlegen der entsprechenden Systeminitialisierungsparameter erstellen, um sie automatisch installieren zu lassen. Informationen zur automatischen Installation von Programmen finden Sie in Autoinstalling programs, map sets, and partition sets in Configuring.

Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with BAS PROGRAM resource definitions.

### PROGRAM-Ressourcen in CICS-Bundles

Sie können ein CICS-Bundle verwenden, um eine PROGRAM-Ressourcendefinition zu erstellen, zu bearbeiten und zu installieren. Wenn Sie eine PROGRAM-Ressource auf diese Weise erstellen, dann müssen Sie zum Verwalten des Lebenszyklus dieser Ressource das CICS-Bundle verwenden. Die unabhängige Verwaltung der Ressource ist nicht möglich. Sie können eine PROGRAM-Ressource abfragen, die von einem CICS-Bundle dynamisch generiert wird. Verwenden Sie hierzu den Befehl **EXEC CICS INQUIRE PROGRAM** oder **CEMT INQUIRE PROGRAM**. Der Befehl SET oder

DISCARD kann jedoch nicht für eine PROGRAM-Ressource ausgeführt werden, die von einem CICS-Bundle dynamisch generiert wird. Sie müssen die Befehle für die BUNDLE-Ressource absetzen und CICS wendet sie dann auf die PROGRAM-Ressource an.

Weitere Informationen zum Definieren von Ressourcen in CICS-Bundles finden Sie in Defining CICS bundles.

## PROGRAM-Ressourcen als Anwendungseinstiegspunkte

Die PROGRAM-Ressource kann als Anwendungseinstiegspunkt definiert werden. Weitere Informationen zum Definieren des Einstiegspunkts finden Sie in Defining application entry points in the CICS Explorer product documentation.

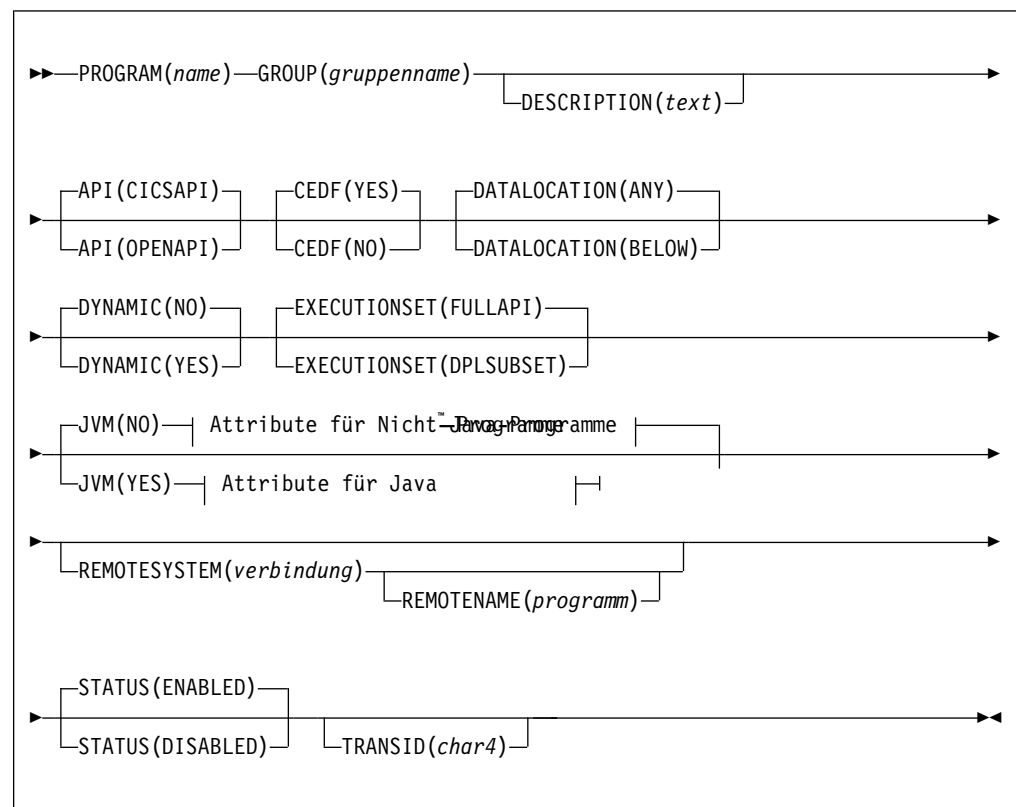
## Private PROGRAM-Ressourcen für Anwendungen

Die PROGRAM-Ressource wird als private Ressource für eine Anwendungsversion unterstützt. Ein Programm, das von einer Task automatisch für eine Anwendung installiert wird, die auf einer Plattform bereitgestellt wird, wird in Bezug auf diese Version der Anwendung ebenfalls als privat eingestuft.

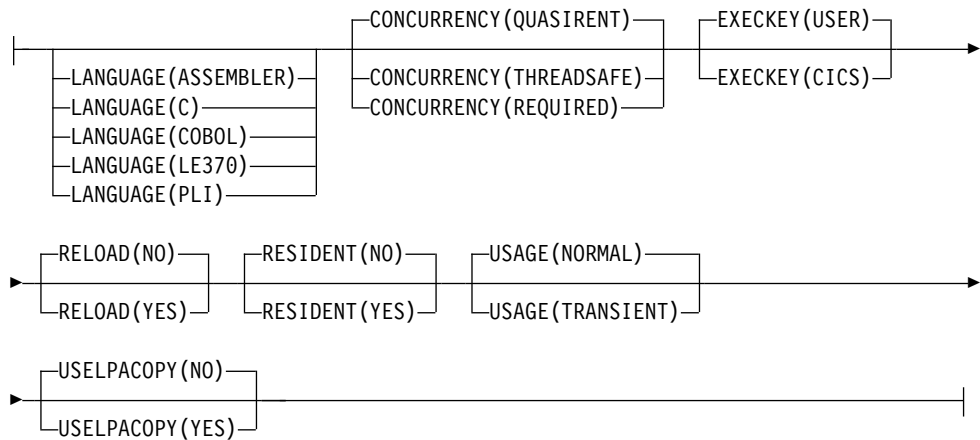
Weitere Informationen zu privaten Ressourcen finden Sie in Private resources for application versions.

## PROGRAM-Attribute

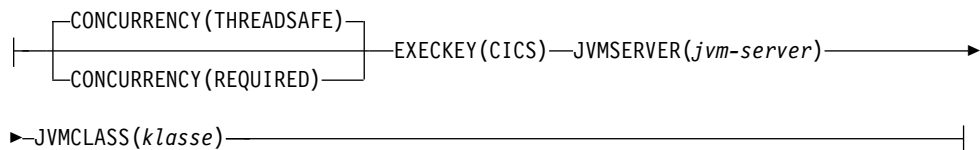
Beschreibt die Syntax und die Attribute der PROGRAM-Ressource.



### Attribute für Nicht-Java-Programme:



### Attribute für Java-Programme:



### API({CICSAPI|OPENAPI})

Gibt an, welche API von dem Programm verwendet werden soll. Das Attribut API gilt für die folgenden Programme:

- Benutzeranwendungsprogramme.
- PLT-Programme.
- Durch den Benutzer austauschbare Programme.
- Taskbezogene Benutzerexitprogramme. Weitere Informationen hierzu finden Sie in Task-related user exit programs.

Das Attribut API gilt nicht für globale Benutzerexits.

#### CICSAPI

Das Programm ist auf die ausschließliche Verwendung der von CICS erlaubten Anwendungsprogrammierschnittstellen beschränkt.

Wenn das Programm mit CONCURRENCY(QUASIRENT) definiert ist, wird es immer im QR-Tasksteuerblock (QR = Quasi-Reentrant; quasi-wiedereintrittsfähig) ausgeführt. Wenn das Programm mit CONCURRENCY(THREADSAFE) definiert ist, dann wird es in dem Tasksteuerblock ausgeführt, der von CICS zum entsprechenden Zeitpunkt verwendet wird. Wenn das Programm mit CONCURRENCY(REQUIRED) definiert ist, wird es immer über einen offenen Tasksteuerblock ausgeführt.

#### OPENAPI

Das Programm ist nicht auf die CICS-Anwendungsprogrammierschnittstellen beschränkt.

CICS führt das Programm über den eigenen offenen Tasksteuerblock aus. Der Typ des offenen Tasksteuerblocks ist von dem Wert des Attributs EXECKEY und der Sprache des Programms abhängig.

Wenn für CICS während der Ausführung eines Befehls ein Wechsel zum QR-Tasksteuerblock erforderlich ist, dann kehrt das Programm zum offenen Tasksteuerblock zurück, bevor die Steuerung an das Anwendungsprogramm zurückgegeben wird.

Zur Angabe des Attributs OPENAPI muss Ihr Programm auf Basis der threadsicheren Standards codiert und mit CONCURRENCY(REQUIRED) definiert werden. Die primäre Verwendung für OPENAPI-Programme besteht in der Verschiebung von Anwendungsworkloads aus dem QR-Tasksteuerblock in mehrere offene Tasksteuerblöcke. So können Sie die Serverressourcen besser nutzen, um bessere Durchsätze zu erzielen.

**Anmerkung:** Die Kombination von CONCURRENCY(THREADSAFE) und API(OPENAPI), die in früheren Releases unterstützt wurde, ist veraltet und wird nur aus Gründen der Kompatibilität weiterhin beibehalten. Sie führt zu demselben Verhalten wie die Angabe von CONCURRENCY(REQUIRED) und API(OPENAPI).

Die Verwendung anderer Nicht-CICS-APIs in OPENAPI-Programmen ist möglich. Wenn ein offener Tasksteuerblock von einem Wartezyklus des Betriebssystems blockiert wird, ist davon nur die jeweilige Anwendung und nicht das gesamte CICS-System betroffen. Solche OPENAPI-Programme dürfen aufgrund dieses Risikos, dass der Tasksteuerblock durch einen Wartezyklus des Betriebssystems blockiert wird und so das gesamte CICS-System beeinträchtigt, nicht im QR-Tasksteuerblock ausgeführt werden. Allerdings haben OPENAPI-Programme weiterhin Verpflichtungen gegenüber dem gesamten CICS-System. Weitere Informationen hierzu finden Sie in Multithreading: Reentrant, quasi-reentrant, and threadsafe programs .

**Important:** Die Verwendung anderer Nicht-CICS-APIs in CICS unterliegt der alleinigen Verantwortung des Benutzers. Es wurden keine Tests für solche Nicht-CICS-APIs in CICS ausgeführt und die Verwendung dieser APIs wird vom IBM Service nicht unterstützt.

#### **CEDF({YES|NO})**

Gibt die Aktion der Execution Diagnostic Facility (EDF) an, wenn das Programm unter EDF-Steuerung ausgeführt wird.

**NO** Die EDF-Diagnoseanzeigen werden nicht angezeigt.

**YES** Die EDF-Diagnoseanzeigen werden angezeigt. Wenn das Programm mit der Option NOEDF umgesetzt wird, werden nur die EDF-Anzeigen für die Initialisierung und Beendigung des Programms angezeigt. Weitere Informationen hierzu finden Sie in Tabelle 3.

*Tabelle 3. Auswirkungen auf Programme mit CEDF(NO) und NOEDF*

Option CEDF in PROGRAM	EDF für Umsetzer angegeben	NOEDF für Umsetzer angegeben
YES	ALLE EDF-Anzeigen	Nur Anzeigen für die Programminitialisierung und -beendigung
NO	KEINE EDF-Anzeigen	KEINE EDF-Anzeigen

Tabelle 3. Auswirkungen auf Programme mit CEDF(NO) und NOEDF (Forts.)

Option CEDF in PROGRAM	EDF für Umsetzer angegeben	NOEDF für Umsetzer angegeben
<b>Anmerkung:</b> In der Tabelle wird gezeigt, wie die Option CEDF in der Programmressourcendefinition mit der Option EDF interagiert, die für den Umsetzer angegeben wurde.		

#### CONCURRENCY ({QUASIRENT|THREADSAFE|REQUIRED})

Gibt an, ob das Programm auf Basis threadsicherer Standards geschrieben wurde oder lediglich quasiwiedereintrittsfähig ist. Sie können das Attribut CONCURRENCY für alle ausführbaren CICS-Programmobjekte angeben:

- Benutzeranwendungsprogramme.
- PLT-Programme.
- Durch den Benutzer austauschbare Programme.
- Globale Benutzerexitprogramme. Weitere Informationen hierzu finden Sie in Global user exit programs.
- Taskbezogene Benutzerexitprogramme. Weitere Informationen hierzu finden Sie in Task-related user exit programs.

#### QUASIRENT

Das Programm ist lediglich quasiwiedereintrittsfähig und arbeitet auf Basis der Serialisierung, die von CICS beim Zugriff auf gemeinsam genutzte Ressourcen bereitgestellt wird.

Das Programm ist auf die für CICS zulässigen Programmierschnittstellen beschränkt und muss mit den CICS-QR-Regeln konform sein. Weitere Informationen hierzu finden Sie in Multithreading: Reentrant, quasi-reentrant, and threadsafe programs .

Dieser Wert wird für alle ausführbaren Programme unterstützt.

CICS stellt sicher, dass das Programm immer unter dem QR-Tasksteuerblock ausgeführt wird, und zwar auch dann, wenn die Steuerung nach dem Start einer JVM oder eines taskbezogenen Open API-Benutzerexits oder bei Interaktion mit threadsicheren Programmen zurückgegeben wird.

#### THREADSAFE

Das Programm wird auf Basis threadsicherer Standards geschrieben. Wenn es auf gemeinsam genutzte Ressourcen zugreift, dann wird die Möglichkeit berücksichtigt, dass andere Programme gleichzeitig ausgeführt werden und versuchen, dieselbe Ressource zu ändern. Das Programm verwendet daher geeignete Serialisierungstechniken beim Zugriff auf gemeinsam genutzte Ressourcen.

JVM-Programme und alle C- und C++-Programme, die mit der Option XPLink kompiliert wurden, müssen als threadsicher definiert werden. Aus Gründen der Kompatibilität mit früheren Releases stellt CONCURRENCY(THREADSAFE) den Standardwert für Java-Programme dar, die bevorzugte Option ist jedoch CONCURRENCY(REQUIRED).

Informationen zu den CICS-Db2-Anwendungsprogrammen finden Sie in Preparing CICS Db2 programs for execution and production.

Informationen zum Schreiben threadsicherer Anwendungsprogramme finden Sie in Threadsafep programs.

Dieser Wert wird für alle ausführbaren Programme unterstützt. Threadsichere Programme müssen mit Language Environment konform sein oder es muss sich um Assemblerprogramme handeln.

## REQUIRED

Das Programm wird auf Basis threadsicherer Standards geschrieben. CICS startet das Programm über einen offenen Tasksteuerblock und stellt sicher, dass das Programm immer über einen offenen Tasksteuerblock ausgeführt wird. Wenn CICS zum QR-Tasksteuerblock wechselt, um einen CICS-Befehl auszuführen, dann kehrt es zu dem offenen Tasksteuerblock zurück, bevor die Steuerung wieder an das Anwendungsprogramm zurückgegeben wird. Der Typ des verwendeten offenen Tasksteuerblocks ist von der Einstellung für API und der Sprache des Programms abhängig.

- Java-Programme und OSGi-Bundles, die auf einem JVM-Server ausgeführt werden, benutzen einen T8-Tasksteuerblock.
- C- oder C++-XPLink-Programme arbeiten wie OPENAPI-Programme und verwenden einen X8-Tasksteuerblock, wenn der CICS-Schlüssel gesetzt ist, und einen X9-Tasksteuerblock, wenn der Benutzerschlüssel gesetzt ist.
- COBOL, PL/I, Nicht-XPLink-C- oder -C++- sowie Assemblerprogramme, für die auch API(CICSAPI) angegeben ist, verwenden einen L8-Tasksteuerblock, weil CICS-Befehle in diesem Tasksteuerblock unabhängig vom Ausführungsschlüssel des Programms eingesetzt werden können.

REQUIRED gilt für Benutzeranwendungsprogramme, PLT-Programme und durch den Benutzer austauschbare Programme und stellt die bevorzugte Option für Java-Programme dar. Die Einstellung von REQUIRED ist auch für Programme geeignet, die auf Ressourcenmanager wie Db2 und IBM MQ zugreifen, für die auch ein L8-Tasksteuerblock erforderlich ist. Für OPENAPI-Programme muss CICS allerdings einen L9-Tasksteuerblock für Benutzerschlüsselprogramme und einen L8-Tasksteuerblock für CICS-Schlüsselprogramme verwenden, sodass Nicht-CICS-API-Befehle wie beispielsweise MVS-Anforderungen korrekt arbeiten. Der taskbezogene CICS-Db2-Benutzerexit verwendet einen L8-Tasksteuerblock, wenn die Ausführung mit CONCURRENCY(REQUIRED) und API(OPENAPI) erfolgt. Alternativ wird er über jeden auswählbaren offenen Tasksteuerblock mit dem Schlüssel 8 ausgeführt, wenn mit den Optionen CONCURRENCY(REQUIRED) und API(OPENAPI) gearbeitet wird.

Mit einem Exit für die automatische Programminstallation können Sie auch das Attribut CONCURRENCY des Programms angeben, wenn die automatische Programminstallation aktiv ist.

## DATALOCATION({ANY|BELOW})

Befehle, die mit der Option SET arbeiten, können eine Datenadresse an ein Anwendungsprogramm zurückgeben. Dieser Operand gibt die Position an, unter der die Daten gespeichert sind. Beispiel: Im Befehl **EXEC CICS RECEIVE SET(ptr-ref)** hat *ptr-ref* einen Wert von weniger als 16 MB, wenn DATALOCATION(BELOW) angegeben ist, kann jedoch größer als 16 MB sein, wenn DATALOCATION(ANY) angegeben wurde. Die Angabe für DATALOCATION hat keine Auswirkungen auf die Ausführung des Befehls GETMAIN. Weitere Informationen dazu, welchen Speicher CICS in Beantwortung dieses Befehls anfordert, finden Sie in GETMAIN.

ANY Das Programm kann 31-Bit-Adressen verarbeiten. Dies ist der Standardwert. Die Adresse der Daten kann ober- oder unterhalb der 16-MB-Grenze liegen. Die für das Attribut DATALOCATION angegebenen Werte gelten unabhängig vom Adressierungsmodus des Programms

mit Programmverbindung. Programme mit Programmverbindung, die im Adressierungsmodus AMODE=24 arbeiten, können nicht auf Daten oberhalb der 16-MB-Grenze zugreifen. Stellen Sie sicher, dass der von Ihnen angegebene Wert mit dem Adressierungsmodus des Anwendungsprogramms mit Programmverbindung kompatibel ist.

- Geben Sie für alle 31-Bit- und 64-Bit-Programme ANY an, es sei denn, dass CICS-Datenadressen an andere 24-Bit-Programme übergeben werden sollen.
- Geben Sie für ein Programm mit der Angabe AMODE=24 die Einstellung DATALOCATION(BELOW) an, es sei denn, die Speicheradressen werden an ein Programm übergeben, das auf den Speicher oberhalb der 16-MB-Grenze zugreifen kann, oder das Programm wechselt explizit den Adressiermodus.

#### **BELOW**

Das Programm kann nur 24-Bit-Adressen verarbeiten und darf daher nur Daten unterhalb der 16-MB-Grenze erhalten. Falls erforderlich, werden die Daten an eine Speicherposition unterhalb der 16-MB-Grenze kopiert, bevor die zugehörigen Adressen an das Anwendungsprogramm übergeben werden.

Die Angabe für DATALOCATION hat keine Auswirkungen auf die Ausführung der Befehle GETMAIN64 oder GET64 CONTAINER. Weitere Informationen zu diesen Befehlen finden Sie in GETMAIN64 und in GET64 CONTAINER.

#### **DESCRIPTION(text)**

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

#### **DYNAMIC({NO|YES})**

Gibt an, ob die Anforderung dynamisch weitergeleitet werden kann, wenn das Programm in einer Anforderung zur Programmverknüpfung angegeben wurde.

**NO** Wenn das Programm in einer Anforderung zur Programmverknüpfung angegeben wurde, dann wird das Programm für dynamisches Routing nicht gestartet.

Für eine DPL-Anforderung (DPL = Distributed Program Link; Verbindung zu verteilten Programmen) muss die Serverregion, in der das Programm ausgeführt werden soll, explizit im Attribut REMOTESYSTEM oder in der Option SYSID des Befehls **EXEC CICS LINK** angegeben werden. Andernfalls wird standardmäßig die lokale Region verwendet.

**YES** Wenn das Programm in einer Anforderung zur Programmverknüpfung angegeben wurde, dann wird das CICS-Programm für dynamisches Routing gestartet. Wenn in der Option SYSID des Befehls **EXEC CICS LINK** keine ferne Region angegeben ist, dann kann das Routing-Programm die Anforderung an die Region weiterleiten, in der das Programm ausgeführt werden soll.

Das Attribut DYNAMIC hat Vorrang vor dem Attribut REMOTESYSTEM (siehe hierzu REMOTESYSTEM).



Information für den Benutzer zum dynamischen Routing von DPL-Anforderungen finden Sie in Dynamically routing DPL requests.

#### **EXECKEY({USER|CICS})**

Gibt den Schlüssel an, in dem CICS dem Programm die Steuerung überträgt, und legt fest, ob das Programm den CICS-Schlüsselspeicher ändern kann. Außer bei wiedereintrittsfähigen Programmen (Programmen mit Programmverbindung, für die das Attribut RENT angegeben ist) definiert die Option EXECKEY auch anhand des Residenzmodus, in welche DSAs CICS das Programm lädt.

**CICS** CICS übergibt die Steuerung an das Programm, das im CICS-Schlüssel aufgeführt ist, wenn es aufgerufen wird. CICS lädt das Programm abhängig vom Residenzmodus des Programms in einen der DSAs für CICS-Schlüssel - entweder den CDSA oder den ECDSA.

In einer CICS-Region mit aktivem Speicherschutz hat ein CICS-Schlüsselprogramm Lese- und Schreibzugriff auf den CICS-Schlüsselspeicher und den Benutzerschlüsselspeicher der eigenen Task und aller anderen Tasks. Dies gilt unabhängig davon, ob die Transaktionsisolation aktiv ist.

**USER** CICS übergibt die Steuerung an das Programm, das im Benutzerschlüssel aufgeführt ist, wenn es aufgerufen wird. CICS lädt das Programm abhängig vom Residenzmodus des Programms in einen der gemeinsam genutzten DSAs für Benutzerschlüssel - entweder den SDSA oder den ESDSA.

In einer CICS-Region, in der nur der Speicherschutz aktiv ist, verfügt ein Benutzerschlüsselprogramm über Schreib- und Lesezugriff auf den gesamten Benutzerschlüsselspeicher, jedoch nur über Lesezugriff auf den CICS-Schlüsselspeicher.

In einer Speicherschutz- und Transaktionsisolationsumgebung verfügt ein Benutzerschlüsselprogramm über Lese- und Schreibzugriff auf den Benutzerschlüsselspeicher für die Lebensdauer der Task nur für die eigene Task und auf den gemeinsam genutzten DSA-Speicher, wenn die Transaktion mit ISOLATE(YES) definiert ist.

Wenn eine Transaktion in einer Transaktionsisolationsumgebung mit ISOLATE(NO) definiert wurde, dann verfügen auch die zugehörigen Benutzerschlüsselprogramme über Lese- und Schreibzugriff auf den Benutzerschlüsselspeicher für die Lebensdauer der Task anderer Transaktionen, die mit ISOLATE(NO) definiert wurden.

Benutzerschlüsselprogramme verfügen immer über Lesezugriff auf den CICS-Schlüsselspeicher.

Das Attribut EXECKEY wird in folgenden Fällen ignoriert:

- Globale Benutzerexitprogramme der ersten Ebene, taskbezogene Benutzerexitprogramme, durch den Benutzer austauschbare Programme und PLT-Programme werden immer im CICS-Schlüssel ausgeführt.
- Wenn es sich um ein Programm mit Programmverbindung handelt, für das das Attribut RENT angegeben wurde, dann lädt CICS das Programm abhängig vom Residenzmodus des Programms in einen der schreibgeschützten DSAs - entweder den RDSA oder den ERDSA. Die schreibgeschützten DSAs werden aus dem Nur-Lese-Speicher nur dann zugeordnet, wenn als Systeminitialisierungsparameter RENTPGM=PROTECT angegeben wird.
- Programme, die mit der dynamischen COBOL-Funktion CALL aufgerufen werden, werden immer in demselben Schlüssel wie der Aufrufende ausgeführt.

**EXECUTIONSET({FULLAPI|DPLSUBSET})**

Gibt an, ob CICS eine Verbindung zu einem Programm herstellen und ein Programm in der Weise ausführen soll, in der Programme in einer fernen CICS-Region ausgeführt werden.

**DPLSUBSET**

Geben Sie DPLSUBSET an, wenn Sie mit CICS eine Verbindung zu dem Programm herstellen und es mit den API-Einschränkungen eines fernen DPL-Programms ausführen möchten. Detaillierte Informationen zu den API-Einschränkungen für ein DPL-Programm finden Sie in Exception conditions for LINK command.

**FULLAPI**

Geben Sie FULLAPI an, wenn Sie mit CICS eine Verbindung zu dem Programm herstellen und es ohne die API-Einschränkungen eines DPL-Programms ausführen möchten. Das Programm kann die vollständige CICS-API verwenden.

Das Attribut EXECUTIONSET gilt nur in folgenden Fällen:

- Für Programme, zu denen eine Verbindung hergestellt wird, und nicht für die Programme, die von einer Transaktion als erstes die Steuerung erhalten.
- Wenn der in REMOTESYSTEM angegebene Name mit dem Namen der lokalen CICS-Region identisch ist. Dies dient zum Testen von Programmen in einer lokalen CICS-Umgebung unter den gleichen Bedingungen wie bei Ausführung als DPL-Programme.

**GROUP(*gruppenname*)**

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

**JVM({NO|YES})**

Gibt an, ob es sich bei dem Programm um ein Java-Programm handelt, das in einer JVM (Java Virtual Machine) ausgeführt werden muss.

**NO** Das Programm wird nicht in einer JVM ausgeführt.

**YES** Das Programm wird in einer JVM ausgeführt. Geben Sie im Attribut JVMCLASS einen Klassennamen an, wenn Sie JVM(YES) angeben.

Zusätzlich zu YES und NO können Sie auch DEBUG angeben, was jedoch nur im Kompatibilitätsmodus zulässig ist (siehe hierzu Sharing the CSD between different releases of CICS).

Wenn Sie für JVM die Einstellung YES angeben und auch für JVMPROFILE (veraltetes Attribut) ein Wert festgelegt wird, dann wird die Programminstallation verhindert.

**JVMCLASS(*klasse*)**

Gibt den (bis zu 255 Zeichen langen) Namen des Service für ein Java-Programm an.

- Für OSGi-Bundles, die in einem JVM-Server ausgeführt werden, entspricht dieser Wert dem Namen des OSGi-Service. Der OSGi-Service wird registriert, wenn Sie die BUNDLE-Ressource installieren, die das OSGi-Bundle enthält. Der Name des OSGi-Service ist in der Ansicht für die Bundlekomponenten (Bundle Parts) in CICS Explorer ersichtlich.
- Für Java-Programme, die in einem JVM-Server ausgeführt werden, entspricht dieser Wert dem Klassennamen, der durch den Paketnamen qualifiziert wird.
- Für Anwendungen in einem Liberty-JVM-Server lautet dieser Wert `wlp:className#methodName`. Beispiel:  
`wlp:com.ibm.cics.server.examples.tsq.ClassOne#myLinkMethod`. Der Methodenname ist optional.

Bei den Namen muss die Groß-/Kleinschreibung beachtet werden und sie müssen mit der korrekten Kombination aus Groß- und Kleinbuchstaben angegeben werden. Wenn Sie ein Terminal verwenden, müssen Sie sicherstellen, dass die Umsetzung in Großbuchstaben unterdrückt wird.

Der Wert für JVMCLASS kann die folgenden Zeichen enthalten:

**Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Dieses Attribut gilt nur für Java-Anwendungen, die unter der Steuerung einer JVM ausgeführt werden. Wenn Sie JVM(NO) angeben, dann ignoriert CICS alle Werte, die in JVMCLASS angegeben werden.

**JVMPROFILE**(*jvm-profil*)

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt. Um ein Java-Programm in einem JVM-Server ausführen zu können, müssen Sie stattdessen das Attribut JVMSERVER verwenden.

**JVMSERVER**(*jvm-server*)

Gibt den (bis zu acht Zeichen langen) Namen der JVMSERVER-Ressource an, die den OSGi-Service enthält. Eine JVMSERVER-Ressource stellt die JVM-Server-Laufzeitumgebung in CICS dar. Der JVM-Server führt alle Programme im CICS-Schlüssel aus.

**Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . - \_ % ? ! : | = , ;

**LANGUAGE**(**{COBOL|ASSEMBLER|LE370|C|PLI}**)

Gibt die Programmiersprache an.

**ASSEMBLER**

Ein Assemblerprogramm, das nicht mit der Umsetzungsoption LEASM umgesetzt wurde. LEASM wird verwendet, um die Assemblerprogramme umzusetzen, bei denen es sich um MAIN-Programme handelt, die mit Language Environment konform sind.

**C** Ein C- oder C++-Programm, das nicht mit einem Language Environment-konformen Compiler kompiliert wurde.

**COBOL**

Ein COBOL-Programm.

**LE370** Ein Programm, das die Unterstützung für mehrere Sprachen verwendet

oder mit einem Language Environment-konformen Compiler kompiliert wurde, oder bei dem es sich um ein Assembler-MAIN-Programm handelt, das mit der Option LEASM umgesetzt wurde, um ein Language Environment-konformes Programm zu erstellen.

**PLI** Ein PL/I-Programm.

In den meisten Fällen müssen Sie das Attribut LANGUAGE nicht angeben, weil der CICS-Programmmanger die korrekte Sprache ableiten kann und den von Ihnen angegebenen Wert ignoriert. Allerdings kann CICS für in Assembler geschriebene Programme, für die kein DFHEAI- oder DFHEA-Stub vorhanden ist, die Sprache nicht ableiten. In diesem Fall müssen Sie den geeigneten Wert angeben. Wenn die Sprache nicht angegeben wird und CICS sie nicht ableiten kann, dann werden Transaktionen, die versuchen, das Programm zu verwenden, mit dem Code ALIG abnormal beendet.

Obwohl Sie normalerweise keinen Wert für dieses Attribut angeben müssen, sollten Sie beachten, dass der angegebene Wert in den Optionen LANGDEDUCED und LANGUAGE des Befehls INQUIRE PROGRAM zurückgegeben wird. Programme, die diesen Befehl verwenden, können von einer Änderung des Wertes für dieses Attribut betroffen sein.

Dieses Attribut gilt nicht für JVM-Programme. CICS leitet ab, dass es sich bei dem Programm um ein Java-Programm handelt, das unter der Steuerung einer JVM ausgeführt wird, wenn JVM(YES) angegeben wurde.

#### **PROGRAM(name)**

Gibt den Namen dieser PROGRAM-Definition an. Der Name kann bis zu acht Zeichen lang sein.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Sie sollten keine Namen verwenden, die mit der Zeichenfolge DFH beginnen, weil diese Zeichen zur Verwendung durch CICS reserviert sind.

CICS kann sie löschen oder verwerfen und durch CICS-Ressourcen ersetzen, wenn Sie ein Upgrade durchführen oder neue Funktionen oder Fixes bereitstellen.

Zur Verwendung des Programms in einer aktiven CICS-Region muss eine Programmverbindung zu einer der Bibliotheken hergestellt werden, die als Teil von DFHRPL oder der dynamischen LIBRARY-Verknüpfung angegeben wurden. Wenn es sich um ein wiedereintrittsfähiges Programm handelt, können Sie es im Link-Pack-Bereich (LPA = Link Pack Area) platzieren. Weitere Informationen zum Installieren von Anwendungsprogrammen finden Sie in *Installing application programs*.

#### **RELOAD({NO|YES})**

Gibt an, ob eine Anforderung für eine Programmsteuerungsverbindung, eine Ladeanforderung oder XCTL-Anforderung eine neue Kopie eines Programms bereitstellen muss. Dieses Attribut gilt nicht für JVM-Programme.

**NO** Jede zulässige Kopie des momentan im Speicher befindlichen Programms wird für die Anforderung wiederverwendet.

**YES** Für jede Anforderung wird eine neue Kopie des Programms in den Speicher eingefügt. Darüber hinaus muss jede dieser Programmkopien

explizit mit der Speichersteuerungsanforderung FREEMAIN aus dem Speicher entfernt werden, wenn sie nicht mehr benötigt wird und bevor die Transaktion beendet wird. Wenn die relevanten FREEMAIN-Anforderungen nicht abgesetzt werden, dann werden Bereiche des DSA/EDSA mit nicht adressierbaren Programmkopien belegt, wodurch es zu Speicherengpässen oder zur Fragmentierung des Speichers kommen kann.

**Anmerkung:** Wenn eine neue Version des Programms in der LIBRARY-Verknüpfung platziert wurde, dann muss der Befehl NEWCOPY oder PHASEIN für das Programm abgesetzt werden, bevor die neue Version geladen wird.

Sie können RELOAD(YES) verwenden, um Tabellen oder Steuerblöcke zu laden, die durch die Ausführung der zugeordneten Programme verändert wurden. Geben Sie diesen Wert nicht für das erste Programm an, das für eine Task geladen wird, da die Task keine Möglichkeit zum Absetzen einer FREEMAIN-Anforderung für das Programm hat.

Für nicht wiedereintrittsfähige Programme müssen Sie RELOAD(YES) angeben.

Weitere Informationen zum Attribut RELOAD finden Sie in Controlling the use of modules from the MVS link pack area.

#### **REMOTENAME**(*programm*)

Gibt den Namen des Programms in der fernen CICS-Region an.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Wenn Sie REMOTESYSTEM angeben, REMOTENAME jedoch nicht, dann wird für das Attribut REMOTENAME standardmäßig der lokale Name verwendet, also der Programmname in dieser Ressourcendefinition.

#### **REMOTESYSTEM**(*verbindung*)

Gibt den Namen einer IPCONN- oder CONNECTION-Ressource an, die eine Verbindung zu der fernen CICS-Region definiert, in der sich das Programm befindet. Geben Sie dieses Attribut an, wenn Sie möchten, dass CICS eine DPL-Anforderung (DPL = Distributed Program Link) an eine andere CICS-Region sendet.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Neben dem Attribut REMOTESYSTEM der Programmdefinition kann die DPL-Serverregion auch wie folgt angegeben werden:

- Im Anwendungsprogramm unter Verwendung der Option SYSID des Befehls **EXEC CICS LINK PROGRAM.**
- Mit dem Programm für das dynamische Routing.

Die Rangfolge ist wie folgt geregelt:

1. Wenn ein Anwendungsprogramm eine DPL-Anforderung absetzt und in der Option SYSID im Befehl **LINK** eine ferne CICS-Region angegeben ist, dann leitet CICS die Anforderung an die ferne Region weiter.

Wenn in der installierten Programmdefinition DYNAMIC(YES) angegeben wurde oder wenn keine installierte Programmdefinition vorhanden ist, dann wird das Programm für dynamisches Routing nur zur Benachrichtigung aufgerufen. Die Umleitung der Anforderung ist hingegen nicht möglich.

2. Wenn ein Anwendungsprogramm eine DPL-Anforderung absetzt, der Wert für SYSID jedoch identisch mit dem Namen der lokalen CICS-Region ist oder die Option SYSID nicht angegeben wurde, gilt Folgendes:

- a. Wenn in der installierten Programmdefinition DYNAMIC(YES) angegeben wurde oder wenn keine installierte Programmdefinition vorhanden ist, dann wird das Programm für dynamisches Routing aufgerufen. Die Weiterleitung der Anforderung ist möglich.

Im Attribut REMOTESYSTEM der Programmdefinition wird (sofern angegeben) die standardmäßige Serverregion angegeben, die an das Programm für dynamisches Routing übergeben wird.

- b. Wenn in der installierten Programmdefinition DYNAMIC(NO) angegeben ist, dann sendet CICS die Anforderung an das ferne System, das im Attribut REMOTESYSTEM angegeben ist. Wurde REMOTESYSTEM nicht angegeben, dann führt CICS das Programm lokal aus.

Die Regeln für die Angabe des Namens für das ferne System stimmen mit denen für das Attribut CONNECTION der CONNECTION-Ressourcendefinition überein.

**Anmerkung:** Sie dürfen keine fernen Attribute für vom Benutzer geschriebene CICS-Programme (z. B. Programme für dynamisches Transaktionsrouting oder Benutzerprogramme mit automatischer Installation) angeben.

#### **RESIDENT({NO|YES})**

Gibt den Verfügbarkeitsstatus des Programms an. Dieses Attribut gilt nicht für JVM-Programme.

**NO** Das Programm soll nicht permanent resident sein. Dieser Wert muss angegeben werden, wenn RELOAD(YES) angegeben wurde.

**YES** Das Programm soll beim ersten Verweis geladen werden und dann permanent im virtuellen Speicher resident bleiben, es muss jedoch vom Betriebssystem umgelagert werden können. Wenn Sie RESIDENT(YES) angeben, dann geht CICS davon aus, dass USAGE(NORMAL) festgelegt wurde.

Weitere Informationen zu den Auswirkungen des Attributs RESIDENT finden Sie in Defining programs as resident, nonresident, or transient.

#### **RSL**

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt. Weitere Informationen hierzu finden Sie in Obsolete attributes.

#### **STATUS({ENABLED|DISABLED})**

Gibt den Programmstatus an.

##### **DISABLED**

Das Programm kann nicht verwendet werden.

### **ENABLED**

Das Programm kann verwendet werden.

Dieses Attribut wird für PROGRAM-Ressourcen ignoriert, die dynamisch von einem CICS-Bundle generiert werden. Der Anfangsstatus einer PROGRAM-Ressource wird aus dem Anfangsstatus des Bundles abgeleitet, das die Ressource definiert.

### **TRANSID(*name*)**

Wenn es sich um ein dynamisches Programm handelt, dann gibt dieser Wert die standardmäßige Transaktions-ID (TRANSID) an, die für die DPL-Anforderung verwendet wird. Wenn es sich nicht um ein dynamisches Programm handelt, dann gibt dieser Wert den Namen der Transaktion an, für die eine Zuordnung zum fernen CICS-System erstellt werden soll und unter der das ferne Programm ausgeführt werden soll.

Wenn Sie im Attribut TRANSID keinen Transaktionsnamen angeben, dann führt die ferne Region das DPL-Programm unter einer der folgenden standardmäßigen Spiegeltransaktionen aus, die von CICS bereitgestellt werden. Wenn Sie eine Benutzertransaktion in der fernen Region definiert haben, um das Spiegelprogramm aufzurufen, dann wird das DPL-Programm unter dieser Transaktions-ID ausgeführt.

**CPMI** Die Spiegeltransaktion, die für LU6.2-Verbindungen, für die eine Datenkonvertierung erforderlich ist, und für TCP/IP- und IPIC-Anforderungen aus CICS Transaction Gateway for Multiplatforms verwendet wird.

**CSMI** Die CICS-Spiegeltransaktion für MRO- und LU6.2-Verbindungen mit der Synchronisationsebene 2 und für EXCI- und IPIC-Anforderungen aus CICS Transaction Gateway for z/OS.

### **USAGE({NORMAL|TRANSIENT})**

Gibt an, wann der Speicher für dieses Programm freigegeben wird. Dieses Attribut gilt nicht für JVM-Programme.

#### **NORMAL**

Wenn der Nutzungszähler für residente Einheiten (RESCOUNT) für dieses Programm den Wert null erreicht, dann wird es für die Entfernung aus dem Speicher im Rahmen eines normalen Komprimierungsprozesses für den dynamischen Programmspeicher auswählbar.

Dieser Wert muss angegeben werden, wenn RELOAD(YES) angegeben wurde.

#### **TRANSIENT**

Wenn der Nutzungszähler für residente Einheiten (RESCOUNT) für dieses Programm den Wert null erreicht, dann wird der Speicher für dieses Programm freigegeben. Geben Sie diesen Wert für Programme an, auf die selten verwiesen wird.

### **USELPACOPY({NO|YES})**

Gibt an, ob das Programm über den Link-Pack-Bereich (LPA = Link Pack Area) verwendet werden soll. Dieses Attribut gilt nicht für JVM-Programme.

**NO** Das Programm soll nicht über den LPA verwendet werden. Es wird in den CICS-Adressraum geladen.

**YES** Das Programm kann über den LPA verwendet werden, wenn LPA=YES als Systeminitialisierungsparameter angegeben ist. Zur Verwendung des Programms über den LPA ist es erforderlich, dass es dort installiert

wurde und dass das Programm nicht im Systeminitialisierungsparameter PRVMOD angegeben ist. Weitere Informationen hierzu finden Sie in Controlling the use of modules from the MVS link pack area.

---

## SESSIONS-Ressourcen

Eine SESSIONS-Ressource definiert die logische Verbindung zwischen zwei CICS-Systemen, die über ISC (Intersystem Communication; systemübergreifende Kommunikation) oder über MRO (Multiregion Operation; Mehrregionenbetrieb) kommunizieren.

Bevor zwei Systeme über ISC oder MRO kommunizieren können, müssen Sie über mindestens eine Sitzung logisch miteinander verbunden sein. Die Art der logischen Verbindung bestimmt, wie sie kommunizieren können. CICS verwendet den SESSIONS-Namen **nicht**, nachdem die Definition auf dem aktiven System installiert wurde. Dieser Name wird nur verwendet, um die Definition in der CSD-Datei zu identifizieren.

Mit dem Attribut CONNECTION der SESSIONS-Ressource können Sie die Verbindung (CONNECTION) benennen, denen diese Sitzungen (SESSIONS) zugeordnet werden, wenn sie auf dem aktiven System installiert werden.

Zu den unterschiedlichen Sitzungstypen sind die folgenden Hinweise zu beachten:

### MRO-Verbindungen und -Sitzungen

Wenn Sie eine SESSIONS-Definition für MRO installieren, dann informieren Sie CICS über eine Gruppe parallel ausgeführter Sitzungen zwischen diesem CICS-System und einem anderen CICS-System. Die Anzahl der Sitzungen wird in den Attributen SENDCOUNT und RECEIVECOUNT festgelegt. Die SEND-Sitzungen werden anhand der Namen bestimmt, die mit den Attributen SENDPFX und SENDCOUNT erstellt werden. Die RECEIVE-Sitzungen werden anhand der Namen bestimmt, die mit den Attributen RECEIVEPFX und RECEIVECOUNT erstellt werden.

### APPC-Verbindungen (LUTYPE6.2) und Parallelsitzungen

Wenn Sie die SESSIONS-Definition installieren, dann werden die Sitzungen (zum Vorteil von z/OS Communications Server) in einem Modusset zusammengefasst, das im Attribut MODENAME angegeben wird. Die einzelnen Sitzungen werden mithilfe eines Zählers benannt. Der ersten erstellten Sitzung wird der Name -999 zugeordnet, der zweiten der Name -998 usw. Der Wert dieses Zählers wird auch nach einem Warm- oder Notfallstart beibehalten. Die Anzahl der Sitzungen, die erstellt werden, wird durch das Attribut MAXIMUM in der SESSIONS-Definition gesteuert.

### LUTYPE6.1-CICS-CICS-ISC-Verbindungen und -Sitzungen

Die Art und Weise, in der Sitzungen von CICS angegeben werden, hängt von der Vorgehensweise beim Definieren der Sitzungen ab. Sie können die Attribute SENDPFX, SENDCOUNT, RECEIVEPFX und RECEIVECOUNT (MRO-Sitzungen) oder das Attribut SESSNAME (CICS-IMS-Sitzungen) verwenden.

**Anmerkung:** Verwenden Sie APPC für alle neuen CICS-CICS-ISC-Verbindungen.

### LUTYPE6.1-CICS-IMS-Verbindungen und -Sitzungen

Wenn Sie die SESSIONS-Definitionen auf dem aktiven CICS-System installieren, dann identifiziert CICS jede Sitzung anhand des Attributs SESSNAME.



## INDIRECT-Verbindungen

Da die Zuordnung zwischen einer INDIRECT-Verbindung und den Transitsystemen, die für die Kommunikation verwendet werden, während der Installation hergestellt wird, sollten Sie die Definition für die Transitsysteme installieren, bevor die Definition für die INDIRECT-Verbindung installiert wird. Wenn Sie die INDIRECT-Verbindung zuerst installieren, bleibt sie im Ruhemodus, bis die Definition für das Transitsystem installiert wurde und bis alle anderen bereits installierten Verbindungen, die auf sie verweisen, aufgelöst wurden. Beispiel: System A ist über System B indirekt mit System C verbunden. Auf System A müssen Sie die folgenden Definitionen in der hier angegebenen Reihenfolge installieren:

1. Das Transitsystem:

```
CONNECTION(B) NETNAME(B) ACCESSMETHOD(IRC) ...
```

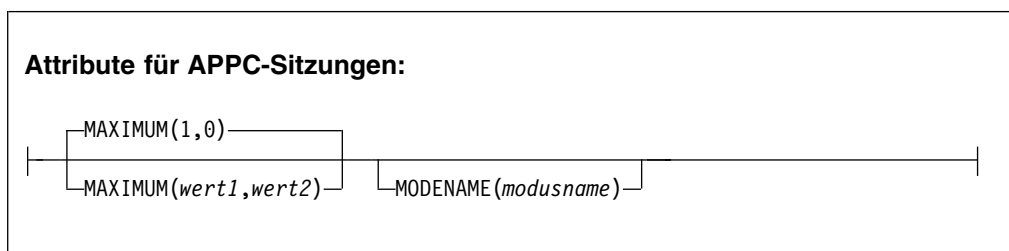
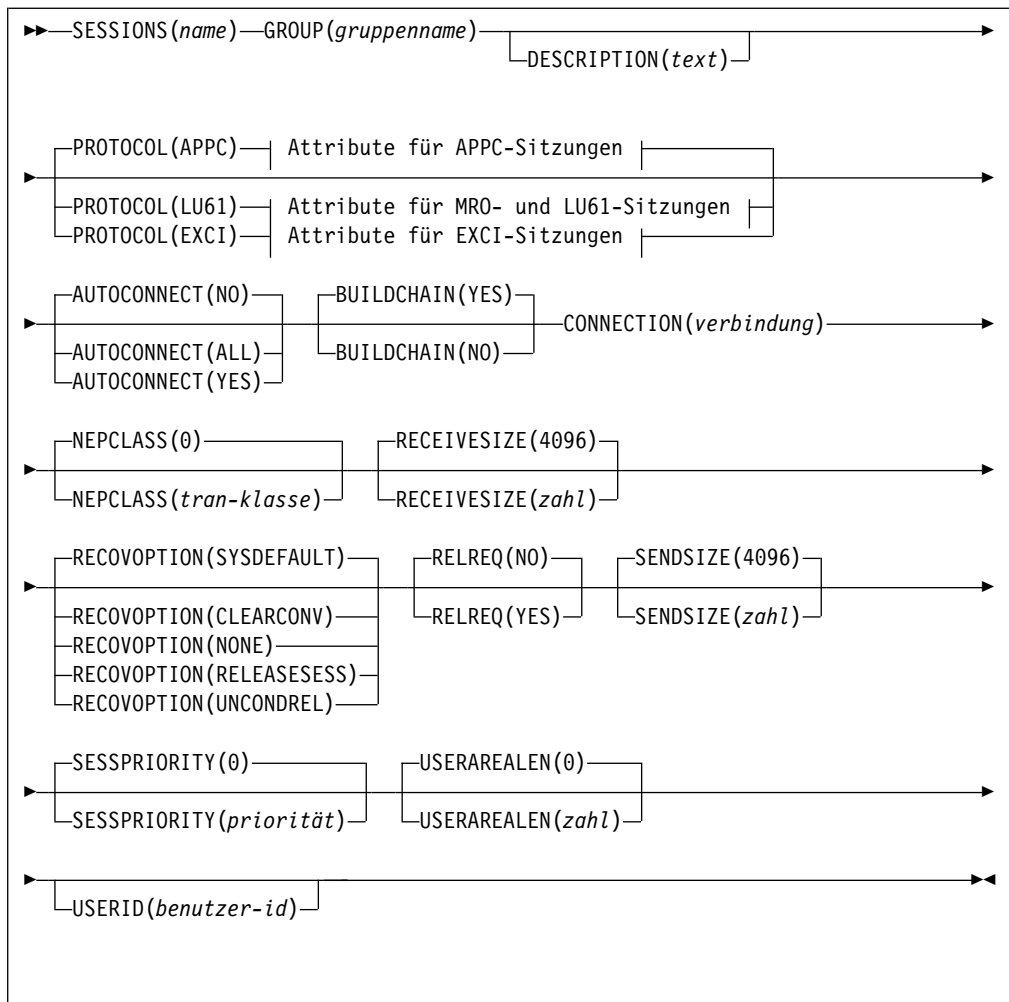
2. Die INDIRECT-Verbindung

```
CONNECTION(C) NETNAME(C) ACCESSMETHOD(INDIRECT)  
INDSYS(B) ...
```

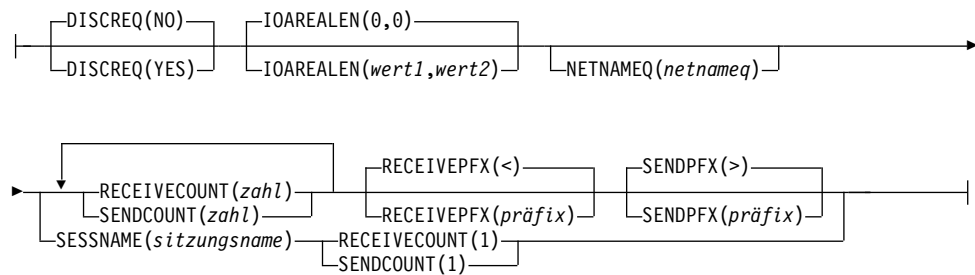
Informationen zur Installation von SESSIONS-Ressourcendefinitionen mit CEMT und CEDA finden Sie in *Installing SESSIONS resource definitions*. Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in *Working with BAS session resource definitions*.

## SESSIONS-Attribute

Beschreibt die Syntax und die Attribute der SESSIONS-Ressource.



### Attribute für MRO- und LU61-Sitzungen:



### Attribute für EXCI-Sitzungen:



### AUTOCONNECT({NO|YES|ALL})

Gibt an, wie Verbindungen eingerichtet werden können. Im Folgenden werden die Elemente erläutert, die für LU6.1- und APPC-Sitzungen angegeben werden müssen:

#### APPC-Sitzungen

Für ein über z/OS Communications Server angeschlossenes System, dessen Verbindungsdefinition die Angabe AUTOCONNECT(YES) oder (ALL) enthält, gilt Folgendes:

**NO** CICS versucht nicht, bei der Verbindungsherstellung Sitzungsbindungen durchzuführen. Allerdings können einzelne oder mehrere Benutzersitzungen im Rahmen der ausgeführten ACQUIRE CONNECTION-Verarbeitung zugeordnet werden.

#### YES oder ALL

Die Konfliktgewinnersitzung wird während der CICS-Initialisierung (durch Ausführung von BIND) oder beim Starten der Kommunikation mit z/OS Communications Server mithilfe des Befehls CEMT SET VTAM OPEN eingerichtet. Wenn zu diesem Zeitpunkt keine Verbindung hergestellt werden kann, weil das ferne System nicht verfügbar ist, dann muss die Verbindung nachfolgend mit dem Befehl CEMT SET CONNECTION(sysid) INSERVICE ACQUIRED angefordert werden, es sei denn, das ferne System wird in der Zwischenzeit wieder verfügbar und kann die Kommunikation selbst einleiten.

AUTOCONNECT(ALL) sollte für Sitzungen zu anderen CICS-Systemen nicht angegeben werden, weil dies zu einer bind-race-Situation führen kann.

Für ein über z/OS Communications Server angeschlossenes System, dessen CONNECTION-Definition die Angabe AUTOCONNECT(NO) enthält, gilt Folgendes:

**ALL** Alle Sitzungen (und nicht nur die Konfliktgewinner) werden einge-

richtet, wenn die Verbindung mithilfe des Befehls CEMT SET CONNECTION(name) ACQUIRED angefordert wird oder wenn das ferne System selbst die Kommunikation einleitet.

- NO** CICS versucht nicht, bei der Verbindungsherstellung Sitzungsbindungen durchzuführen. Allerdings können einzelne oder mehrere Benutzersitzungen im Rahmen der ausgeführten ACQUIRE CONNECTION-Verarbeitung zugeordnet werden.
- YES** Konfliktgewinnersitzungen werden eingerichtet, wenn die Verbindung mithilfe des Befehls CEMT SET CONNECTION(sysid) ACQUIRED angefordert wird oder wenn das ferne System selbst die Kommunikation einleitet.

#### **LU6.1-Sitzungen**

Geben Sie AUTOCONNECT(YES) in der SESSIONS-Definition an, wenn die Verbindung während der Initialisierung oder bei der CEDA-Installation eingerichtet werden soll.

Geben Sie AUTOCONNECT(NO) in der SESSIONS-Definition an, wenn die Verbindung nicht während der Initialisierung oder bei der CEDA-Installation eingerichtet werden soll.

#### **BUILDCHAIN({YES|NO})**

Gibt an, ob CICS die Kettenassemblierung ausführen soll, bevor die Eingabedaten an das Anwendungsprogramm übergeben werden.

- NO** Jeder Terminal-E/A-Bereich (TIOA = Terminal Input/Output Area), der durch ein Anwendungsprogramm von dieser logischen Einheit empfangen wird, enthält eine Anforderungseinheit (RU = Request Unit).
- YES** Jeder Terminal-E/A-Bereich (TIOA), der durch ein Anwendungsprogramm von dieser logischen Einheit empfangen wird, enthält eine komplette Kette.

#### **CONNECTION(verbindung)**

Gibt den Namen der Verbindungsdefinition an, die mit dieser Sitzungsdefinition verwendet werden soll. Der Name kann bis zu vier Zeichen lang sein.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Beachten Sie, dass die CONNECTION-Definition in derselben Gruppe (GROUP) wie die SESSIONS-Definition enthalten sein muss.

#### **DESCRIPTION(text)**

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

#### **DISCREQ({NO|YES})**

Gibt an, ob Unterbrechungsanforderungen berücksichtigt werden sollen. DISCREQ gilt für LUTYPE6.1-ISC-Sitzungen, jedoch nicht für MRO-Sitzungen, bei denen CICS nicht mit z/OS Communications Server-Einheiten arbeitet.

DISCREQ gilt nicht für APPC-Sitzungen (LUTYPE6.2). Wenn APPC verwendet wird, werden einzelne Sitzungen bei Bedarf als Transaktionen angefordert und anschließend wieder freigegeben. Da es möglich ist, mehrere Sitzungen zwischen logischen APPC-Einheiten herzustellen, sollte kein Problem auftreten, weil eine Anforderung eine andere aufhält. Einzelne APPC-Sitzungen können nicht unterbrochen werden. Stattdessen können Sie den Befehl CEMT SET CONNECTION RELEASED absetzen.

**NO** CICS berücksichtigt eine Unterbrechungsanforderung für eine z/OS Communications Server-Einheit nicht.

**YES** CICS berücksichtigt eine Unterbrechungsanforderung für eine z/OS Communications Server-Einheit und setzt eine z/OS Communications Server-Makroinstruktion CLSDST ab, um die z/OS Communications Server-Sitzung mit dieser logischen Einheit zu beenden.

Der Befehl CESF LOGOFF oder GOODNIGHT, der von einem Terminal abgesetzt wird, führt ebenfalls zur Trennung der Verbindung, wenn Sie DISCREQ(YES) angeben.

#### **GROUP**(*gruppenname*)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

#### **INSERVICE**

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt.

#### **IOAREALEN**(**{0|wert1}**,**{0|wert2}**)

Gibt die Länge (in Byte) des Terminal-E/A-Bereichs an, der für die Verarbeitung der Nachrichten verwendet werden soll, die über die MRO-Verbindung übertragen werden.

*wert1* In *wert1* wird die minimale Größe eines Terminal-E/A-Bereichs angegeben, die an ein Anwendungsprogramm übergeben wird, wenn ein Befehl RECEIVE abgesetzt wird.

*wert2* Wenn *wert2* nicht angegeben wird oder wenn der Wert niedriger als der Wert für *wert1* ist, dann wird standardmäßig der Wert von *wert1* verwendet.

Sie können für *wert2* einen Wert größer-gleich dem Wert von *wert1* angeben. In diesem Fall verwendet CICS, wenn die Größe einer Eingabemessage den Wert von *wert1* überschreitet, einen Terminal-E/A-Bereich (TIOA), der *wert2* Byte lang ist. Wenn eine Transaktion über eine MRO-Verbindung angeschlossen ist, dann verwendet CICS einen TIOA, der lang genug ist, um die ursprüngliche Eingabemessage zu enthalten. Übersteigt die Größe der Eingabemessage auch den Wert für *wert2*, sendet das Programm für abnormale Bedingungen des Knotens eine Ausnahmeantwort an das Terminal.

Sie vergeuden sowohl Realspeicher als auch virtuellen Speicher, indem Sie einen IOAREALEN-Wert angeben, der zu groß für die meisten Nachrichten ist, die über Ihre MRO-Verbindung übertragen werden. Andererseits werden auf dem System bei Angabe eines IOAREALEN-Werts, der null oder kleiner als die Mehrzahl Ihrer Nachrichten ist, übermäßig viele FREEMAIN- und GETMAIN-Aktivitäten ausgeführt. Dies führt zu einer weiteren Erhöhung der Prozessoranforderungen.

#### **MAXIMUM({1|wert1},{0|wert2}) (nur APPC)**

Gibt die maximale Anzahl von Sitzungen an, die für das Modusset unterstützt werden. Der Wert für *wert1* muss größer-gleich dem Wert für *wert2* sein.

##### **1|wert1**

Die maximale Anzahl von Sitzungen in der Gruppe. Dieser Wert kann im Bereich zwischen 1 und 999 liegen. Der Standardwert ist 1.

##### **0|wert2**

Die maximale Anzahl von Sitzungen, die als Konfliktgewinner unterstützt werden sollen. Dieser Wert kann im Bereich zwischen 0 und 999 liegen. Der Standardwert ist 0. Beachten Sie, dass dieser Operand für eine Einzelsitzungsverbindung keine Bedeutung hat.

Unter SNA können bestimmte Ressourcen (z. B. Wählleitungen) im Netz als **eingeschränkte Ressourcen** definiert werden. Beim Binden gibt z/OS Communications Server für CICS an, ob die Bindung über eine eingeschränkte Ressource ausgeführt wird. Wenn eine CICS-Task eine Sitzung über eine eingeschränkte Ressource freigibt, dann hebt CICS die Bindung der Sitzung auf, wenn keine andere Task sie verwenden will.

Wenn die Sitzungen mit eingeschränkten Ressourcen arbeiten sollen, dann geben Sie **MAXIMUM(wert1,0)** an. Dies führt dazu, dass alle aufgelösten Sitzungen zurückgesetzt werden, sodass beide Seiten dann eine Bindung als Gewinner herstellen können, wenn die nächste Anforderung abgesetzt wird.

Weitere Informationen zu den Auswirkungen der Option MAXIMUM und zur Verwendung eingeschränkter Ressourcen finden Sie in Effects of the MAXIMUM option of the SESSIONS resource.

#### **MODENAME(modusname) (nur APPC)**

Gibt den Namen an, der eine Gruppe von Sitzungen zur Verwendung über eine APPC-Verbindung identifiziert. Der Name darf eine Länge von maximal acht Zeichen haben und muss den Namen eines LOGMODE-Eintrags von z/OS Communications Server angeben, der in z/OS Communications Server definiert wurde. Der reservierte Name SNASVCMG darf hier nicht verwendet werden. Wenn Sie den Modusnamen nicht angeben, werden standardmäßig Leerzeichen angenommen. Weitere Informationen zu z/OS Communications Server-Modusnamen finden Sie in Defining groups of APPC sessions.

Das Attribut MODENAME muss für jede Sitzungsgruppe, die für eine systemübergreifende Verbindung definiert wurde, eindeutig sein. Dies bedeutet, dass MODENAME innerhalb der SESSIONS-Definitionen, die sich auf eine CONNECTION-Definition beziehen, eindeutig sein muss. Das Attribut wird als LOGMODE-Name an z/OS Communications Server übergeben.

#### **NEPCCLASS({0|transaktionsklasse})**

Gibt die Transaktionsklasse für das Knotenfehlerprogramm an. Dieser Wert stellt den Standardwert dar.

##### **0**

Die Angabe dieses Werts führt dazu, dass eine Verbindung zum Standardmodul des Knotenfehlerprogramms (NEP = Node Error Program) hergestellt wird.

#### *transaktionsklasse*

Die Transaktionsklasse für das (nicht standardmäßige) NEP-Modul. Der Wert kann im Bereich zwischen 1 und 255 liegen. Informationen zur Programmierung für das Knotenfehlerprogramm (NEP) finden Sie in Writing a node error program.

Das Attribut NEPCCLASS wird für SNASVCMGR-Sitzungen ignoriert.

#### **NETNAMEQ**(*netnameq*)

Gibt den Namen an, unter dem das ferne IMS-System diese spezielle Sitzung erkennt. Dieser Name wird für CICS-IMS-Sitzungen verwendet. Der Name kann bis zu acht Zeichen lang sein. Die folgenden Zeichen sind zulässig: A-Z 0-9 \$ @ # . / - \_ % & ¢ ? ! : | " = ~ , ; < und >. Kleinbuchstaben werden in Großbuchstaben konvertiert. Eine Ausnahme bildet hierbei der Befehl CREATE, bei dem keine Konvertierung stattfindet.

#### **OPERID**

#### **OPERPRIORITY**

#### **OPERRSL**

#### **OPERSECURITY**

Diese Attribute sind veraltet, werden jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt.

#### **PROTOCOL**(**{APPC|LU61|EXCI}**)

Gibt den Typ des Protokolls an, das für eine übergreifende Kommunikationsverbindung (ISC oder MRO) verwendet werden soll.

##### **APPC** (**LUTYPE6.2**)

Das Protokoll Advanced Program-to-Program Communication (APPC). Geben Sie diesen Wert für die CICS-CICS-ISC an.

**EXCI** Die externe CICS-Schnittstelle. Definieren Sie diese Option, um anzugeben, dass die Sitzungen durch ein Nicht-CICS-Clientprogramm über die externe CICS-Schnittstelle (EXCI = External CICS Interface) verwendet werden.

**LU61** Das Protokoll LUTYPE6.1. Geben Sie diese Option für CICS-CICS-ISC, für CICS-IMS oder für MRO an.

#### **RECEIVECOUNT**(*zahl*)

Für MRO- und z/OS Communications Server LU6.1-Sitzungen sowie für Sitzungen mit EXCI-Clients gibt diese Option die Anzahl der *Empfangssitzungen* an. Dabei handelt es sich um die Sitzungen, die normalerweise vor dem Sendevorgang Daten empfangen:

- MRO-Empfangssitzungen (einschließlich Sitzungen mit EXCI-Clients) empfangen Daten immer vor dem Sendevorgang.
- z/OS Communications Server LU6.1-Empfangssitzungen empfangen Daten normalerweise vor dem Sendevorgang, können aber auch vor dem Empfang senden, wenn ein Engpass bei den geeigneten Sendesitzungen besteht.

Wenn Sie das Attribut RECEIVECOUNT nicht angeben, gibt es keine Empfangssitzungen.

Die Anzahl der Empfangssitzungen, die Sie angeben können, hängt von der Länge des im Attribut RECEIVEPFX angegebenen Präfix ab:

- Wenn Sie das Standardempfangspräfix (<) oder ein eigenes einstelliges Präfix verwenden, können Sie zwischen 1 und 999 Empfangssitzungen angeben.
- Wenn Sie ein zweistelliges Präfix verwenden, dann können Sie zwischen 1 und 99 Empfangssitzungen angeben.

Darüber hinaus sollten Sie sicherstellen, dass der angegebene Wert mit der Anzahl der Sendesitzungen auf dem Partnersystem übereinstimmt:

- Wenn der Partner ein anderes CICS-System ist, dann sollte der Wert mit dem Wert für SENDCOUNT übereinstimmen, der auf dem Partnersystem angegeben wurde.
- Wenn der Partner ein EXCI-Client ist, dann können Sie die Anzahl der Sendesitzungen auf dem Partnersystem nicht angeben. Es gibt jedoch eine Obergrenze für Sendesitzungen in einem EXCI-Adressraum. Wenn dieser Grenzwert erreicht ist, dann weist IRP weitere Anforderungen für eine Sitzung mit dem Ursachencode 608 (SYSTEM\_ERROR) zurück. Informationen zur Vorgehensweise bei der Angabe des Grenzwerts für das CICS-System finden Sie in EXCI pipe allocation in Installing.

#### **RECEIVEPFX({<|präfix})**

Gibt das aus einem bis zwei Zeichen bestehende Präfix an, das von CICS für die ersten ein bis zwei Zeichen im Namen von Empfangssitzungen (Namen der Terminaleinträge einer Terminalsteuertabelle (TCTTEs) für die Sitzungen) verwendet werden soll.

Präfixe dürfen keinen Konflikt mit einer vorhandenen Verbindung oder einem vorhandenen Terminalnamen verursachen.

#### **< (MRO- und EXCI-Sitzungen)**

Wenn Sie für MRO- und EXCI-Sitzungen kein eigenes Empfangspräfix angeben, dann erzwingt CICS die Verwendung des Standardpräfix. Hierbei handelt es sich um das Kleiner-als-Symbol (<), das zusammen mit dem Empfangszähler verwendet wird, um den Namen von Empfangssitzungen zu generieren.

CICS erstellt die letzten drei Zeichen der Sitzungsnamen. Zulässige Zeichen sind A-Z sowie 1-9. Diese aus drei Zeichen bestehenden Kennungen beginnen mit den Buchstaben AAA und werden fortlaufend in aufsteigender Reihenfolge erstellt, bis die Anzahl der Sitzungseinträge den im Wert für RECEIVECOUNT angegebenen Grenzwert erreicht. Beachten Sie, dass die Namen von Empfangssitzungen **nach** den Sendesitzungen erstellt werden und dass für sie die gleiche Reihenfolge gilt.

Wenn der zuletzt für die Sendesitzungen generierte Sitzungsname >AAJ lautet, dann werden bei Verwendung des standardmäßigen Sendeprefix (>) von CICS die folgenden Empfangssitzungsnamen generiert: <AAK, <AAL, <AAM usw. (Diese Methode der Generierung von Sitzungskennungen ist mit der entsprechenden Methode für APPC-Sitzungen identisch. Eine Ausnahme bildet hierbei allerdings das erste Präfixsymbol.)

Wenn Sie mehr als 46656 Sitzungen (<AAA - <999) verwenden, dann ordnet CICS den nächsten Bereich von AAA< - 999< in ähnlicher Weise für die APPC-Sitzungen zu.

In einer Region mit mehr als 46656 Sitzungen kommt es möglicherweise zu schlechten Leistungswerten. Sie sollten prüfen, ob alternativ die Erhöhung der Anzahl der CICS-Regionen sinnvoll ist.

Obwohl Sie bis zu 93312 MRO-Sitzungen definieren können, besteht eine aktuelle Beschränkung, die verhindert, dass mehr als 65535 Sitzungen in einem Versuch angefordert werden. Dieser Fall kann eintreten, wenn CICS gestartet wird oder auch bei einer CEDA-Installation



für mehr als 65536 Sitzungen, wenn ALLE Partnerregionen betriebsbereit sind. Weitere Sitzungen können später angefordert werden.

**Anmerkung:** Wenn Sie ein eigenes Präfix angeben, dann generiert CICS die Sitzungsnamen auf dieselbe Weise wie bei LUTYPE6.1-Sitzungen.

*präfix* (**LUTYPE6.1-Sitzungen**)

Wenn die Sitzungen über LUTYPE6.1-ISC-Verbindungen eingerichtet werden, müssen Sie ein aus einem bis zwei Zeichen bestehendes Präfix angeben. Verwenden Sie für LUTYPE6.1-Sitzungen nicht das Standardsymbol (<).

Für LUTYPE6.1-Sitzungen (und MRO-Sitzungen, wenn Sie ein eigenes ein- bis zweistelliges Präfix angeben) generiert CICS Sitzungsnamen, indem eine Zahl an das Präfix angefügt wird. Möglich sind Zahlen im Bereich zwischen 1 und 99 oder im Bereich zwischen 1 und 999. Die Zahl beginnt mit 1 und wird jeweils um den Wert 1 erhöht, bis der in RECEIVECOUNT angegebene Wert erreicht wird.

**RECEIVSIZE**(**{4096|zahl}**)

Gibt die maximale Größe einer z/OS Communications Server-Anforderungseinheit (RU = Request Unit) an, die von diesen Sitzungen empfangen werden kann. Der Wert muss zwischen 1 und 30720 (LU61-Sitzungen) oder zwischen 256 und 30720 (APPC-Sitzungen) liegen. Der Standardwert ist 4096.

Der angegebene Wert wird an die verbundene logische Einheit übertragen. Dieser Wert kann von CICS abhängig von dem von Ihnen angegebenen Wert abgerundet werden, weil der Wert in einer architekturbasierten Form übertragen werden muss. Der Wert kann während der Bindung (BIND) weiter nach unten korrigiert werden.

Wenn CICS die sekundäre LU-Sitzung darstellt, dann gibt dieser Wert die maximale Größe für die z/OS Communications Server-Anforderungseinheit (RU) an, die von diesen Sitzungen gesendet werden kann.

**RECOVNOTIFY**

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt.

**RECOVOPTION**(**{SYSDEFAULT|CLEARCONV| RELEASESESS|UNCONDREL|NONE}**)

Diese Option gilt für die Wiederherstellung von Sitzungen in einer CICS-Region, die mit persistenten z/OS Communications Server-Sitzungen oder mit XRF ausgeführt wird.

**Persistente z/OS Communications Server-Sitzungen:** In einer CICS-Region, die mit Unterstützung für persistente Sitzungen ausgeführt wird, gibt diese Option an, wie CICS bei der Wiederherstellung der Sitzung vorgehen und das Terminal beim Neustart des Systems innerhalb des Verzögerungsintervalls für persistente Sitzungen wieder aktivieren soll.

**XRF:** In einer CICS-Region, die mit XRF-Unterstützung ausgeführt wird, gibt diese Option an, wie CICS bei der Wiederherstellung der Sitzung vorgehen und das Terminal nach einer XRF-Übernahme wieder aktivieren soll.

Für alle Wiederherstellungsoptionen außer NONE: Wenn die ausgeführte Aktion eine z/OS Communications Server-Aktion zum Auflösen der Bindung (UNBIND) ist, dann folgt auf UNBIND eine z/OS Communications Server-Aktion SIMLOGON.

## **CLEARCONV**

**Persistente z/OS Communications Server-Sitzungen:** CLEARCONV wird für APPC-Sitzungen nicht unterstützt. Standardmäßig wird SYS-DEFAULT verwendet.

**XRF:** Wenn AUTOCONNECT(YES) angegeben wird, dann wird die Sitzung erneut gestartet. Wenn AUTOCONNECT(NO) angegeben wird, dann wird die Bindung der Sitzung aufgelöst.

## **NONE**

**Persistente z/OS Communications Server-Sitzungen:** In einer CICS-Region, die mit der Unterstützung für persistente Sitzungen ausgeführt wird, gibt diese Option an, dass die Sitzung beim Neustart des Systems nicht innerhalb des Verzögerungsintervalls für persistente Sitzungen wiederhergestellt werden soll. Die Sitzungen in der Modusgruppe verfügen also nicht über Unterstützung für persistente Sitzungen. Die Bindung der LU6.2-Sitzungen wird aufgelöst und der CNOS-Wert der Modusgruppe wird auf null zurückgesetzt. Nach dem Neustart des Systems wird die Sitzung automatisch wieder verbunden, wenn Sie AUTOCONNECT(YES) angeben.

**XRF:** In einer CICS-Region, die mit XRF-Unterstützung ausgeführt wird, gibt diese Option an, dass der Anmeldestatus vom alternativen System nicht verfolgt wird. Die Terminalsitzung wird nach einer Übernahme nicht automatisch wiederhergestellt. Dies bedeutet, dass das Terminal nicht über XRF-Unterstützung verfügt. Nach der Übernahme wird das Terminal automatisch vom alternativen System wieder verbunden, wenn Sie AUTOCONNECT(YES) angeben.

## **RELEASESESS**

**Persistente z/OS Communications Server-Sitzungen:** RELEASESESS wird für APPC-Sitzungen nicht unterstützt. Standardmäßig wird SYS-DEFAULT verwendet.

**XRF:** Wenn AUTOCONNECT(YES) angegeben wird, dann wird die Sitzung erneut gestartet. Wenn AUTOCONNECT(NO) angegeben wird, dann wird die Bindung der Sitzung aufgelöst.

## **SYSDEFAULT**

**Persistente z/OS Communications Server-Sitzungen:** In einer CICS-Region, die mit der Unterstützung für persistente Sitzungen ausgeführt wird, gibt diese Option an, dass CICS die optimale Prozedur zur Wiederherstellung einer Sitzung beim Neustart des Systems innerhalb des Verzögerungsintervalls für persistente Sitzungen abhängig von der Sitzungsaktivität und den Merkmalen des Terminals auswählen soll.

Obwohl die Sitzungen wiederhergestellt werden, werden zum Fehlerzeitpunkt ausgeführte Operationen abgebrochen und nicht wiederhergestellt. Transaktionen werden ebenfalls abgebrochen, wenn die wiederhergestellte Sitzung momentan von einer anderen CICS-Region über eine APPC-Verbindung benutzt wird.

CICS stellt die Sitzung mit den geringstmöglichen Auswirkungen wie folgt wieder her:

- Wenn die Sitzung zum Fehlerzeitpunkt von CICS nicht belegt war, ist keine Aktion erforderlich.
- Wenn die Sitzung zum Fehlerzeitpunkt von CICS aktiv war, dann setzt CICS die Anweisung DEALLOCATE(ABEND) (gleichbedeutend

mit EXEC CICS ISSUE ABEND) für den APPC-Datenaustausch ab, der zum Fehlerzeitpunkt aktiv war.

- Wenn keines dieser Ereignisse zutrifft, dann wird die Bindung der Sitzung aufgelöst.

**XRF:** Wenn AUTOCONNECT(YES) angegeben wird, dann wird die Sitzung erneut gestartet. Wenn AUTOCONNECT(NO) angegeben wird, dann wird die Bindung der Sitzung aufgelöst.

#### **UNCONDREL**

Fordert CICS zum Senden einer UNBIND-Anforderung für die Freigabe der aktiven Sitzung auf. Die UNBIND-Anforderung wird unabhängig davon gesendet, ob die Sitzung während des Neustarts des Systems belegt war (bei Unterstützung für persistente Sitzungen) oder ob eine Übernahme stattgefunden hat (bei XRF).

#### **RELREQ({NO|YES})**

Gibt an, ob CICS die logische Einheit bei einer entsprechenden Anforderung eines anderen z/OS Communications Server-Anwendungsprogramms freigeben soll.

#### **SEND COUNT(zahl)**

Nur für MRO- und z/OS Communications Server-LU6.1-Sitzungen gibt diese Option die Anzahl der *Sendesitzungen* an. Dabei handelt es sich um die Sitzungen, die normalerweise vor dem Empfangsvorgang Daten senden:

- MRO-Sendesitzungen senden immer vor dem Empfang.
- z/OS Communications Server-LU6.1-Sendesitzungen senden Daten normalerweise vor dem Empfangsvorgang, können aber auch vor dem Senden Daten empfangen, wenn ein Engpass bei den geeigneten Empfangssitzungen besteht.

Die Anzahl der Sendesitzungen, die Sie angeben können, hängt von der Länge des im Attribut SENDPFX angegebenen Präfix ab:

- Wenn Sie das Standardsendepräfix (>) oder ein eigenes einstelliges Präfix verwenden, können Sie zwischen 1 und 999 Sendesitzungen angeben.
- Wenn Sie ein zweistelliges Präfix verwenden, dann können Sie zwischen 1 und 99 Sendesitzungen angeben.

Darüber hinaus sollten Sie sicherstellen, dass der angegebene Wert mit der Anzahl der Empfangssitzungen auf dem Partnersystem übereinstimmt:

- Wenn der Partner ein anderes CICS-System ist, dann sollte der Wert mit dem Wert für RECEIVECOUNT übereinstimmen, der auf dem Partnersystem angegeben wurde.

Wenn Sie das Attribut SENDCOUNT nicht angeben, gibt es keine Sendesitzungen. Geben Sie das Attribut SENDCOUNT nicht an, wenn der Partner ein EXCI-Client ist.

#### **SENDPFX({>|präfix})**

Gibt das aus einem bis zwei Zeichen bestehende Präfix an, das von CICS für die ersten ein bis zwei Zeichen im Namen von Sendesitzungen (Namen der Terminaleinträge einer Terminalsteuertabelle (TCTTEs) für die Sitzungen) verwendet werden soll.

Präfixe dürfen keinen Konflikt mit einer vorhandenen Verbindung oder einem vorhandenen Terminalnamen verursachen.

#### **> (MRO-Sitzungen)**

Wenn Sie für MRO-Sitzungen kein eigenes Sendeprefix angeben, dann erzwingt CICS die Verwendung des Standardpräfix. Hierbei handelt es

sich um das Größer-als-Symbol (>), das zusammen mit dem Sendungszähler verwendet wird, um den Namen von Sendesitzungen zu generieren.

CICS erstellt die letzten drei Zeichen der Sitzungsnamen anhand der alphanumerischen Zeichen A bis Z und 1 bis 9. Diese aus drei Zeichen bestehenden Kennungen beginnen mit den Buchstaben AAA und werden fortlaufend in aufsteigender Reihenfolge erstellt, bis die Anzahl der Sitzungseinträge den im Wert für SENDCOUNT angegebenen Grenzwert erreicht.

Wenn Sie z. B. das Standardpräfix (>) verwenden, dann generiert CICS Sitzungsnamen im Format >AAA, >AAB, >AAC usw. Wenn Sie mehr als 46656 Sitzungen (>AAA - >999) verwenden, dann ordnet CICS den nächsten Bereich von AAA> - 999> zu. (Diese Methode der Generierung von Sitzungskennungen ist mit der entsprechenden Methode für APPC-Sitzungen identisch. Eine Ausnahme bildet hierbei allerdings das erste Symbol).

In einer Region mit mehr als 46656 Sitzungen kommt es möglicherweise zu schlechten Leistungswerten. Sie sollten prüfen, ob alternativ die Erhöhung der Anzahl der CICS-Regionen sinnvoll ist.

Obwohl Sie bis zu 93312 MRO-Sitzungen definieren können, besteht eine aktuelle Beschränkung, die verhindert, dass mehr als 65535 Sitzungen in einem Versuch angefordert werden. Dieser Fall kann eintreten, wenn CICS gestartet wird oder auch bei einer CEDA-Installation für mehr als 65536 Sitzungen, wenn ALLE Partnerregionen betriebsbereit sind. Weitere Sitzungen können später angefordert werden.

**Anmerkung:** Wenn Sie ein eigenes Präfix angeben, dann generiert CICS die Sitzungsnamen auf dieselbe Weise wie bei LUTYPE6.1-Sitzungen.

#### *präfix* (für LUTYPE6.1-Sitzungen)

Wenn die Sitzungen über LUTYPE6.1-ISC-Verbindungen eingerichtet werden, müssen Sie ein aus einem bis zwei Zeichen bestehendes Präfix angeben. Verwenden Sie für LUTYPE6.1-Sitzungen nicht das Standardsymbol (>).

Für LUTYPE6.1-Sitzungen (und MRO-Sitzungen, wenn Sie ein eigenes ein- bis zweistelliges Präfix angeben) generiert CICS Sitzungsnamen, indem eine Zahl an das Präfix angefügt wird. Möglich sind Zahlen im Bereich zwischen 1 und 99 oder im Bereich zwischen 1 und 999. Die Zahl beginnt mit 1 und wird jeweils um den Wert 1 erhöht, bis der in SENDCOUNT angegebene Wert erreicht wird.

#### **SENDSIZE** ({4096|**zahl**})

Gibt die maximale Größe einer z/OS Communications Server-Anforderungseinheit (RU = Request Unit) an, die von diesen Sitzungen gesendet werden kann. Der Wert muss zwischen 1 und 30720 (LU61-Sitzungen) oder zwischen 256 und 30720 (APPC-Sitzungen) liegen. Der Standardwert ist 4096. Der Wert kann während der Bindung nach unten korrigiert werden. Die Erhöhung des Werts von SENDSIZE führt dazu, dass mehr Speicher für die Sitzung zugeordnet wird, kann jedoch zur Reduzierung der Anzahl der physischen Nachrichten führen, die zwischen den beiden Knoten gesendet werden.

Wenn CICS die sekundäre LU-Sitzung darstellt, dann gibt dieses Attribut die maximale Größe für die z/OS Communications Server-Anforderungseinheit (RU) an, die von diesen Sitzungen empfangen werden kann. Der Wert muss zwischen 256 und 30720 liegen.

#### **SESSIONS**(*name*)

Gibt den Namen dieser SESSIONS-Definition an. Der Name kann bis zu acht Zeichen lang sein.

##### **Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Dieser Name wird verwendet, um die SESSIONS-Definition in der CSD-Datei zu identifizieren. Innerhalb des aktiven CICS-Systems wird er nicht verwendet.

#### **SESSNAME**(*sitzungsname*)

Gibt die symbolische Kennung an, die als lokale Hälfte eines Paares mit Sitzungsqualifikationsmerkmalen in einer parallelen CICS-Sitzung für die übergreifende Kommunikation verwendet werden soll. Der Name kann bis zu vier Zeichen lang sein.

##### **Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

#### **SESSPRIORITY**(**{0|***priorität***}**)

Gibt die Terminalpriorität an. Dieser Dezimalwert (0 bis 255) wird verwendet, um die übergreifende Priorität für die Transaktionsverarbeitung einzurichten. (Die Priorität für die Transaktionsverarbeitung entspricht der Summe der Terminalpriorität, Transaktionspriorität und Bedienerpriorität und darf den Wert 255 nicht überschreiten.)

#### **TRANSACTION**

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt.

#### **USERAREALEN**(**{0|***zahl***}**)

Geben Sie die Länge (in Byte) des Benutzerbereichs für diese Sitzung im Bereich zwischen 0 und 255 an. Der Wert sollte möglichst niedrig sein. Der Terminalbenutzerbereich wird bei der Installation der Sitzung mit Nullen initialisiert.

Der Terminalbenutzerbereich kann sich im 31-Bit-Speicher oder im 24-Bit-Speicher befinden. Die Speicherposition ist von dem Wert abhängig, der im Systeminitialisierungsparameter **TCTUALOC** angegeben ist. Der Standardwert legt fest, dass sich der Terminalbenutzerbereich sowohl im 31-Bit-Speicher als auch im 24-Bit-Speicher befinden kann, CICS verwendet jedoch nach Möglichkeit immer den 31-Bit-Speicher. Wenn der Terminalbenutzerbereich unbedingt im 24-Bit-Speicher angelegt werden muss, weil Sie mit Anwendungsprogrammen arbeiten, die die 31-Bit-Adressierung nicht unterstützen, dann müssen Sie für die CICS-Region den Systeminitialisierungsparameter **TCTUALOC=BELOW** angeben.

#### **USERID**(*benutzer-id*)

Gibt eine Benutzer-ID an, die für die Anmeldung verwendet wird (SEC=YES oder MIGRATE) und auf die in Sicherheitsfehlernachrichten, Sicherheitsverstößnachrichten und im Prüfprotokoll verwiesen wird. Hierbei muss es sich um eine gültige Benutzer-ID handeln, die für den Sicherheitsmanager definiert wurde, da sich die Bediener andernfalls nicht anmelden können. Der gesamte Zugriff auf geschützte Ressourcen ist vom Operanden USERID abhängig.

Diese Benutzer-ID (USERID) überschreibt einen Wert für SECURITYNAME, der in der CONNECTION-Definition angegeben wurde.

Der Name kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

---

## TCPIPSERVICE-Ressourcen

Eine TCPIPSERVICE-Ressource definiert, welche TCP/IP-Services die interne Socketunterstützung von CICS verwenden sollen.

Für TCPIPSERVICE-Ressourcen können die CICS-Services ECI over TCP/IP (für CICS-Clients), die CICS-Webunterstützung (HTTP), IPIC oder ein benutzerdefiniertes Protokoll definiert werden. Die TCPIPSERVICE-Definition ermöglicht Ihnen die Verwaltung dieser internen CICS-Schnittstellen, wobei CICS mit unterschiedlichen ECI-Varianten, mit der CICS-Webunterstützung oder mit dem benutzerdefinierten Protokoll an unterschiedlichen Ports empfangsbereit ist.

TCPIPSERVICE-Definitionen dürfen ausschließlich mit den von CICS bereitgestellten TCP/IP-Services verwendet werden und sind von der IP CICS Sockets-Schnittstelle von z/OS Communications Server getrennt. Die TCP/IP-Socketschnittstelle für CICS gehört zum Lieferumfang von z/OS Communications Server, das integraler Bestandteil von z/OS ist und nicht mit der CICS-SO-Domäne arbeitet.

Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with BAS TCPIPSERVICE resource definitions.

### TCPIPSERVICE-Ressourcen in CICS-Bundles

Sie können ein CICS-Bundle verwenden, um eine TCPIPSERVICE-Ressourcendefinition zu erstellen, zu bearbeiten und zu installieren. Wenn Sie eine TCPIPSERVICE-Ressource auf diese Weise erstellen, dann müssen Sie zum Verwalten des Lebenszyklus dieser Ressource das CICS-Bundle verwenden. Die unabhängige Verwaltung der Ressource ist nicht möglich.

Sie können eine TCPIPSERVICE-Ressource abfragen, die von einem CICS-Bundle dynamisch generiert wird. Verwenden Sie hierzu den Befehl **EXEC CICS INQUIRE TCPIPSERVICE** oder **CEMT INQUIRE TCPIPSERVICE**.

Der Befehl DISCARD kann für eine TCPIPSERVICE-Ressource, die von einem CICS-Bundle dynamisch generiert wird, nicht abgesetzt werden. Sie müssen das CICS-Bundle löschen. CICS wendet die Operation auf die TCPIPSERVICE-Ressource an.

Zum Ändern der Attribute einer TCPIPSERVICE-Ressource, die in einem CICS-Bundle definiert und installiert wurde, können Sie den Ressourceneditor in CICS Explorer verwenden, um die Definition im CICS-Bundle zu ändern und eine neue Version des CICS-Bundles oder der Anwendung zu installieren, über das bzw. die die Bereitstellung durchgeführt wurde. Sie können den Befehl SET TCPIPSERVICE verwenden, um die Attribute der dynamisch generierten Ressource zu ändern. Die-

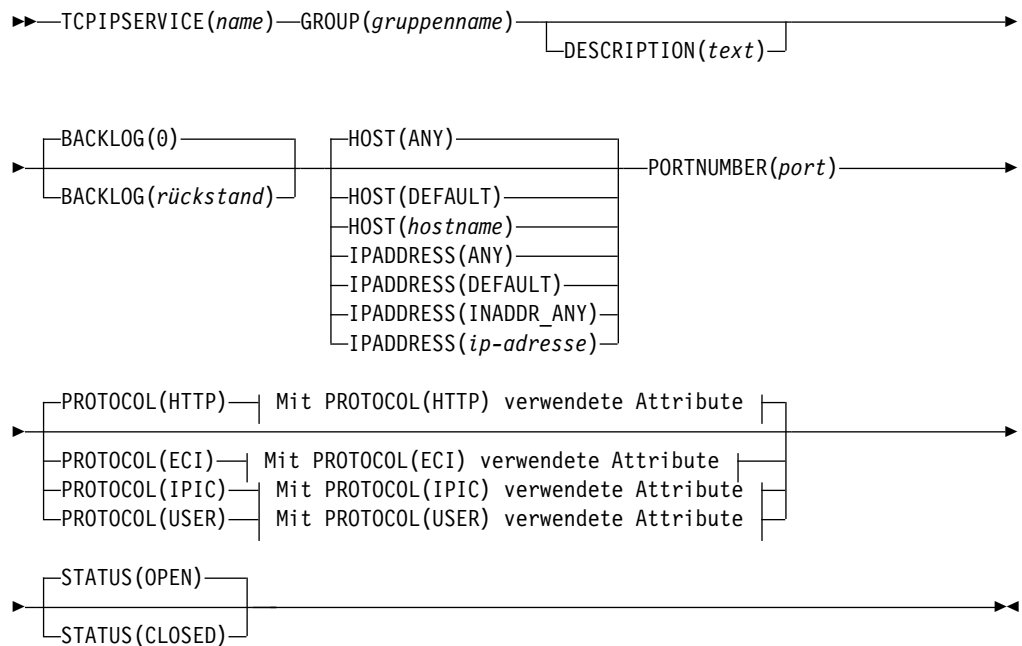
se Änderungen werden jedoch nicht katalogisiert und können nach einem Warmstart von CICS nicht wiederhergestellt werden.

Um den Status einer TCPIPService-Ressource zu ändern, die in einem CICS-Bundle definiert und installiert wurde, müssen Sie das CICS-Bundle aktivieren oder inaktivieren. Wenn Sie das CICS-Bundle inaktiviert haben, der Service jedoch noch nicht geschlossen wurde, dann können Sie den Befehl **SET TCPIPService IMMClose** für die dynamisch generierte Ressource absetzen, um den Service sofort zu schließen.

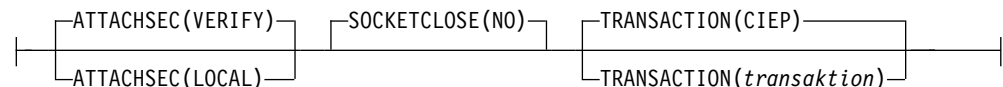
Weitere Informationen zum Definieren von Ressourcen in CICS-Bundles finden Sie im Abschnitt zum Definieren von CICS-Bundles.

## TCPIPService-Attribute

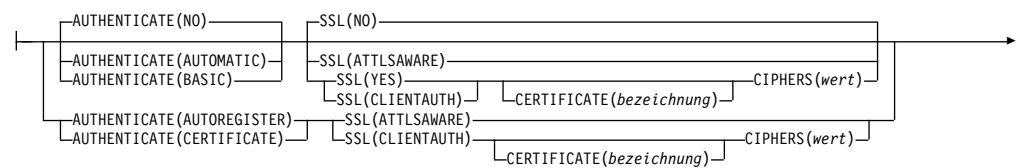
Beschreibt die Syntax und die Attribute der TCPIPService-Ressource.

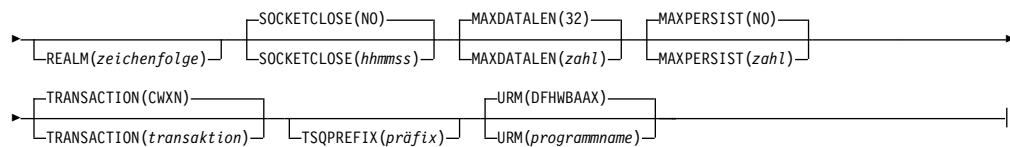


### Mit PROTOCOL(ECI) verwendete Attribute:

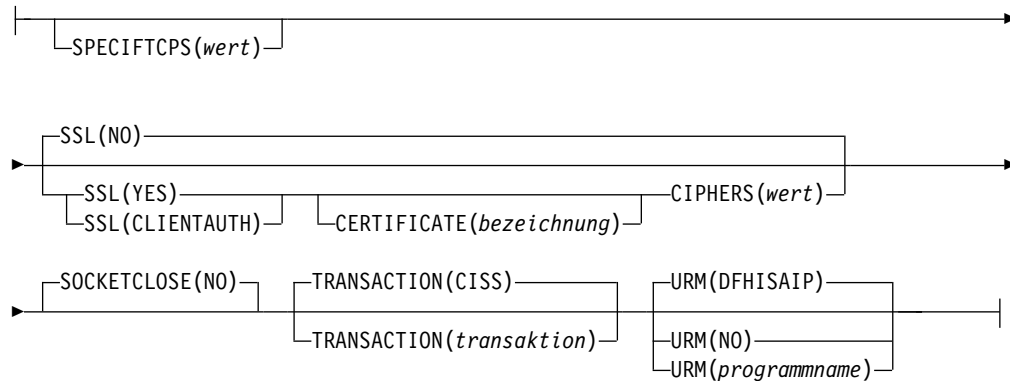


### Mit PROTOCOL(HTTP) verwendete Attribute:

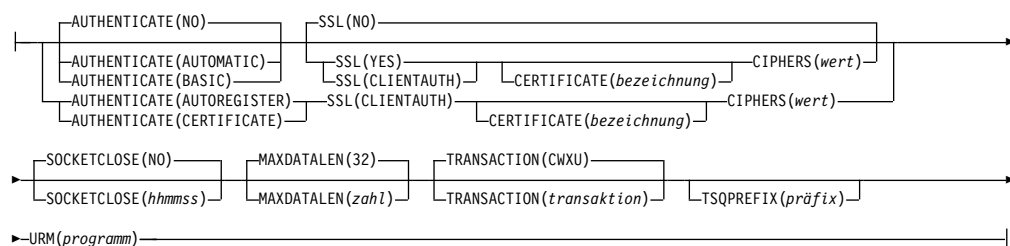




### Mit PROTOCOL(IPIC) verwendete Attribute:



### Mit PROTOCOL(USER) verwendete Attribute:



### ATTACHSEC({LOCAL|VERIFY})

Gibt die erforderliche Stufe der Benutzersicherheit an, die für die Herstellung dieser Verbindung erforderlich ist.

Diese Option ist nur für PROTOCOL(ECI) gültig.

#### LOCAL

Gibt an, dass CICS keine Benutzer-ID und kein Kennwort (bzw. keine Kennwortphrase) von den Clients benötigt.

#### VERIFY

Gibt an, dass eingehende Verbindungsanforderungen eine Angabe zur Benutzer-ID und ein Benutzerkennwort oder eine Kennwortphrase umfassen müssen. Geben Sie VERIFY an, wenn die Systeme, die eine Verbindung herstellen, nicht identifiziert werden können und nicht vertrauenswürdig sind.

### AUTHENTICATE({NO|ASSERTED|AUTOMATIC|AUTOREGISTER|BASIC|CERTIFICATE})

Gibt das Authentifizierungs- und Identifikationsschema an, das für eingehende TCP/IP-Verbindungen bei den Protokollen HTTP und USER verwendet werden soll. Für das IPIC-Protokoll kann dieses Attribut nicht verwendet werden. Für das ECI-Protokoll ist dieses Attribut ungültig. Weitere Informationen zur Authentifizierung finden Sie in Identification and authentication.



## ASSERTED

Das Attribut ASSERTED in TCPIPSERVICE-Definitionen ist veraltet, wird jedoch aus Kompatibilitätsgründen beibehalten.

**Wenn PROTOCOL(HTTP) oder PROTOCOL(USER) angegeben ist, gilt Folgendes:**

**NO** Der Client muss keine Authentifizierungs- oder Identifikationsinformationen senden. Wenn der Client jedoch ein gültiges Zertifikat sendet, das bereits im Sicherheitsmanager registriert und einer Benutzer-ID zugeordnet ist, dann identifiziert diese Benutzer-ID den Client.

## BASIC

Die HTTP-Basisauthentifizierung wird zum Abrufen einer Benutzer-ID und eines Kennworts oder einer Kennwortphrase vom Client verwendet.

Wenn der Client einen Berechtigungsheader gesendet hat, dann wird sein Inhalt als Benutzer-ID und Kennwort oder Kennwortphrase decodiert. Wenn diese Angaben nicht gültig sind, dann gibt das System die Antwort HTTP 401 zusammen mit einem WWW-Authentifizierungsheader zurück, wodurch das Clientprogramm veranlasst wird, den Benutzer zur Eingabe einer neuen Benutzer-ID und eines entsprechenden Kennworts oder einer entsprechenden Kennwortphrase aufzufordern. Dieser Prozess wird so lange fortgeführt, bis der Client entweder eine gültige Benutzer-ID und ein Kennwort oder eine Kennwortphrase angibt oder die Verbindung abbricht.

Nachdem der Benutzer erfolgreich authentifiziert wurde, identifiziert die angegebene Benutzer-ID den Client.

## AUTOREGISTER

Die Authentifizierung mit einem SSL-Clientzertifikat wird zur Authentifizierung des Clients verwendet.

- Wenn der Client ein gültiges Zertifikat sendet, das bereits im Sicherheitsmanager registriert und einer Benutzer-ID zugeordnet ist, dann identifiziert diese Benutzer-ID den Client.
- Wenn der Client ein gültiges Zertifikat sendet, das nicht im Sicherheitsmanager registriert ist, dann wird die HTTP-Basisauthentifizierung zum Abrufen einer Benutzer-ID und eines Kennworts oder einer Kennwortphrase vom Client verwendet. Wenn das Kennwort oder die Kennwortphrase gültig ist, registriert CICS das Zertifikat beim Sicherheitsmanager und ordnet es der Benutzer-ID zu. Die Benutzer-ID dient zur Identifikation des Clients.
- Damit ein Zertifikat registriert werden kann, benötigt der Client die entsprechende Zugriffsdefinition in RACF. Es wird Lesezugriff (READ) auf das Profil IRR.DIGTCERT.ADD in der Klasse FACILITY benötigt.

**Anmerkung:** Wenn Sie AUTHENTICATE(AUTOREGISTER) angeben, dann müssen Sie auch SSL(CLIENTAUTH | ATTLISAWARE) angeben.

## AUTOMATIC

Diese Funktion kombiniert die Funktionen AUTOREGISTER und BASIC.

- Wenn der Client ein Zertifikat sendet, wird die Verarbeitung wie im Abschnitt zu AUTOREGISTER beschrieben fortgesetzt.

- Wenn der Client kein Zertifikat sendet, wird die Verarbeitung wie im Abschnitt zu BASIC beschrieben fortgesetzt.

## CERTIFICATE

Die Authentifizierung mit einem SSL-Clientzertifikat wird zur Authentifizierung und Identifikation des Clients verwendet. Der Client muss ein gültiges Zertifikat senden, das bereits im Sicherheitsmanager registriert und einer Benutzer-ID zugeordnet ist. Wenn kein gültiges Zertifikat empfangen wurde oder wenn das Zertifikat nicht einer bestimmten Benutzer-ID zugeordnet ist, dann weist das System die Verbindung zurück.

Nachdem der Benutzer erfolgreich authentifiziert wurde, identifiziert die Benutzer-ID, die dem Zertifikat zugeordnet wurde, den Client.

**Anmerkung:** Wenn Sie AUTHENTICATE(CERTIFICATE) angeben, dann müssen Sie auch SSL(CLIENTAUTH | ATTLAWARE) angeben.

**Anmerkung:** Für das HTTP- oder das USER-Protokoll kann das Analyseprogramm (angegeben im Attribut URM) die Benutzer-ID ändern, die während des Authentifizierungsprozesses angegeben wurde. Wenn während des Authentifizierungsprozesses keine Benutzer-ID angegeben wird, dann kann die Benutzer-ID über das Analyseprogramm oder die URIMAP-Definition bereitgestellt werden. Andernfalls wird die CICS-Standard-Benutzer-ID verwendet.

## BACKLOG(10|rückstand)

Gibt die maximale Anzahl eingehender Anforderungen (im Bereich zwischen 1 und 32767) zum Öffnen einer TCP/IP-Verbindung an, die unter TCP/IP in die Warteschlange eingestellt werden können, um auf die Verarbeitung durch CICS zu warten. Wenn das Maximum erreicht ist, weist TCP/IP weitere Anforderungen zurück. Wenn der Wert für BACKLOG null ist, dann wird der Wert im TCP/IP-Attribut SOMAXCONN verwendet. Wenn der Wert für BACKLOG größer als der Wert für SOMAXCONN ist, dann wird der Wert für SOMAXCONN verwendet. Für IPIC-Verbindungen müssen Sie für diesen Wert die maximal zulässige Anzahl von Anforderungen für gleichzeitige Verbindungen festlegen, die mithilfe dieser TCPIPService-Komponente hergestellt werden können.

Gibt die maximale Anzahl eingehender Anforderungen für TCP/IP-Verbindungen an, die unter TCP/IP in die Warteschlange eingestellt werden können, um auf die Verarbeitung durch CICS zu warten. Wenn das Maximum erreicht ist, weist TCP/IP weitere Verbindungsanforderungen zurück. Geben Sie für diesen Wert die maximal zulässige Anzahl von Anforderungen für gleichzeitige Verbindungen an, die mithilfe dieser TCPIPService-Komponente hergestellt werden können. Wenn der Wert von BACKLOG höher als der TCP/IP-Konfigurationswert für SOMAXCONN ist, dann verwendet TCP/IP den Wert, der im Attribut SOMAXCONN angegeben ist. Mit dem Wert null werden eingehende Verbindungsanforderungen inaktiviert.

Wenn die Leistungsoptimierung für HTTP-Verbindungen aktiviert ist, wenn CICS die maximale Kapazität ausgeschöpft hat, werden alle eingehenden Anforderungen zum Öffnen einer HTTP-Verbindung außerhalb von CICS in der Rückstandswarteschlange der empfangsbereiten Verbindung von TCPIPService eingestellt. Stellen Sie sicher, dass der Wert für BACKLOG hoch genug ist, da Verbindungsanforderungen zurückgewiesen werden, wenn diese Warteschlange vollständig belegt ist. Zum Anzeigen der Rückstandswerte für eine empfangsbereite Verbindung verwenden Sie die CICS-TCP/IP-Services: Ressourcenstatistik oder den Befehl NETSTAT ALL, um Informationen zum Status

des lokalen Hosts abzurufen, die auch Informationen zum Rückstand der empfangsbereiten Verbindung umfassen.

Wenn der Verbindungslastausgleich aktiviert ist, dann wird die Tiefe der Rückstandswarteschlange in die Verarbeitung einbezogen, mit der die am besten geeignete Empfangsfunktion ermittelt wird. Aus diesem Grund sollten Sie die Verwendung des Attributs BACKLOG für alle TCPIP SERVICES in CICS-Regionen berücksichtigen, die einen Port gemeinsam nutzen.

#### **CERTIFICATE**(*bezeichnung*)

Gibt die Bezeichnung eines X.509-Zertifikats an, das während des SSL-Handshakes beim Anfordern der Verbindung als Serverzertifikat verwendet wird. Wird dieses Attribut nicht angegeben, dann wird das Standardzertifikat verwendet, das im Schlüsselring der Benutzer-ID der CICS-Region angegeben ist.

Zertifikatsbezeichnungen können bis zu 32 Byte lang sein.

Das Zertifikat muss in einem Schlüsselring in der Datenbank des externen Sicherheitsmanagers gespeichert werden. Weitere Informationen finden Sie in *Building a key ring manually*.

Dieses Attribut kann nur angegeben werden, wenn auch SSL(YES) oder SSL(CLIEN TAUTH) angegeben wird.

#### **CIPHERS**(*wert*)

Das Attribut CIPHERS kann auf eine der beiden folgenden Arten angegeben werden:

- Eine aus maximal 56 Hexadezimalziffern bestehende Zeichenfolge, die als Liste mit bis zu 28 zweistelligen Cipher-Suite-Codes interpretiert wird.
- Der Name der Spezifikationsdatei für die SSL-Cipher-Suite, bei der es sich um eine z/OS UNIX-Datei im Unterverzeichnis security/ciphers des Verzeichnisses handelt, das im Systeminitialisierungsparameter **USSCONFIG** angegeben ist. Wenn **USSCONFIG** z. B. auf den Wert /var/cicsts/dfhconfig gesetzt wird und **CIPHERS** auf den Wert strongciphers.xml, dann lautet der vollständig qualifizierte Dateiname /var/cicsts/dfhconfig/security/ciphers/strongciphers.xml. Weitere Informationen hierzu finden Sie in *Creating an SSL cipher suite specification file*.

Wenn Sie die CEDA-Transaktion zum Definieren der Ressource verwenden, dann wird das Attribut von CICS automatisch mit einer Standardliste zulässiger Codes initialisiert. Damit CICS das Attribut initialisieren kann, muss der Systeminitialisierungsparameter KEYRING in der CICS-Region angegeben werden, in der CEDA ausgeführt wird. Wird KEYRING nicht angegeben, dann kann CICS das Attribut nicht initialisieren. Die Standardliste der Codes trägt die Bezeichnung 35363738392F303132330A1613100D15120F0C, es sei denn, der Systeminitialisierungsparameter **NISTSP800131A=CHECK** wird festgelegt. In diesem Fall lautet die Bezeichnung 35363738392F303132330A1613100D.

Sie können die Cipher-Codes anders sortieren oder aus der ursprünglichen Liste entfernen. Allerdings ist es nicht möglich, Cipher-Codes hinzuzufügen, die sich nicht in der Standardliste für die angegebene Verschlüsselungsstufe befinden. Zum Zurücksetzen des Werts auf die standardmäßige Codeliste müssen Sie alle Cipher-Suite-Codes löschen. Das Feld wird dann automatisch erneut mit der Standardliste gefüllt.

Weitere Informationen hierzu finden Sie in *Cipher suites and cipher suite specification files*.

#### **DESCRIPTION**(*text*)

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem

Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

**DNSGROUP** (*dns-gruppe*)

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt.

**GROUP** (*gruppenname*)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

**GRPCRITICAL** ({**NO**|**YES**})

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt.

**HOST** ({**ANY**|**DEFAULT**|*hostname*})

Gibt die aus 116 Zeichen bestehende IPv4- oder IPv6-Adresse oder den Hostnamen an, über die bzw. den CICS für eingehende Verbindungen empfangsbereit ist. Verwenden Sie HOST anstelle von IPADDRESS, wenn Sie neue Ressourcen definieren wollen. Geben Sie HOST und IPADDRESS nicht zusammen an, weil HOST immer Vorrang vor IPADDRESS hat. IPADDRESS wird für vorhandene Programme unterstützt, die die IPv4-Funktion angeben.

Folgende Werte sind zulässig:

**ANY** Die Option ANY hat die gleiche Funktion wie die Optionen ANY und INADDR\_ANY von IPADDRESS. Die Option ANY gibt an, dass CICS an jeder der Adressen empfangsbereit ist, die in TCP/IP für das Hostsystem bekannt sind. Sie können für einen Host mehrere IP-Adressen definieren. Durch Angabe von ANY ermöglichen Sie außerdem die gemeinsame Nutzung der TCPIPSERVICE-Definition durch verschiedene CICS-Server. Wenn Sie ANY angeben, dann versucht CICS, eine Bindung zu dem Port in jedem Stack herzustellen, in dem es definiert ist. Wenn Sie außerdem möchten, dass mehrere CICS-Regionen eine Bindung zu dem Port erhalten, dann müssen Sie die Option SHAREPORT in jedem Stack angeben, in dem der Port definiert ist. Andernfalls kann nur eine CICS-Region eine Bindung zu der Portnummer in diesen Stacks erhalten, für die die Option SHAREPORT nicht angegeben ist. Nachfolgende Versuche anderer Regionen zur Herstellung einer Bindung zu jedem Stack schlagen fehl und CICS gibt eine Nachricht aus, in der Sie darüber informiert werden, dass der Port belegt ist.

Wenn Sie die Option ANY in einer Dualmodusumgebung (IPv4 und IPv6) angeben, dann versucht CICS, die neueste IPv4- oder IPv6-Adresse erneut zu verwenden. Handelt es sich dabei um die erste Verbin-

dung und kann CICS keine Adresse abrufen, dann wird 0.0.0.0 zurückgegeben. Dabei wird keine Affinität zugeordnet.

#### DEFAULT

Die Option DEFAULT ordnet die Affinität dem TCP/IP-Stack zu, der in einer CINET-Multistack-Umgebung als Standardwert definiert wurde.

Wenn die Option DEFAULT in einer Dualmodusumgebung (IPv4 und IPv6) verwendet wird, dann wird die Affinität der IPv4-Umgebung zugeordnet, weil die Option DEFAULT in der IPv4-Umgebung angewendet wird.

Wenn DEFAULT in einer Nicht-CINET-Umgebung verwendet wird oder wenn kein standardmäßiger TCP/IP-Stack vorhanden ist, dann wird ein Ausnahmebedingungstrace geschrieben, der Wert 0.0.0.0 zurückgegeben und keine Affinität zugeordnet.

**Important:** Wenn Sie in einer Dualmodusumgebung (IPv4 und IPv6) arbeiten, erzwingt die Angabe von HOST(DEFAULT) die Abwicklung des gesamten Datenverkehrs über die IPv4-Netzverbindung.

#### *hostname*

In *hostname* kann ein zeichenbasierter Hostname, eine IPv4-Adresse oder eine IPv6-Adresse angegeben werden.

Sie können eine Adresse als Zeichenname angeben, der auf dem Domänennamensserver abgefragt werden kann. Der Hostname kann in Großbuchstaben, Kleinbuchstaben oder in gemischter Groß-/Kleinschreibung eingegeben werden. Wenn jedoch anstelle einer IP-Adresse ein Hostname angegeben wird, wird dieser Hostname in der TCPIPSERVICE-Definition in Kleinbuchstaben konvertiert.

Verwenden Sie keinen zeichenbasierten Hostnamen, wenn Sie über eine Liste mit Adressen auf dem Domänennamensserver verfügen, weil *hostname* nur für die erste IP-Adresse in der Liste aufgelöst wird. (Dies bedeutet, dass der Server nicht an einer der IP-Adressen in der Liste für diesen Hostnamen empfangsbereit ist.) Wenn Sie in einer Liste auf dem Domänennamensserver eine spezielle IP-Adresse benötigen, dann definieren Sie die IP-Adresse explizit in *hostname*.

Wenn Sie eine IPv6-Adresse (oder einen Hostnamen, der in eine IPv6-Adresse aufgelöst wird) angeben, dann müssen Sie sicherstellen, dass Sie in einer Dualmodusumgebung (IPv4 und IPv6) arbeiten und dass der Client oder der Server, mit dem Sie kommunizieren, ebenfalls in einer Dualmodusumgebung (IPv4 und IPv6) arbeitet. Weitere Informationen zu IPv6 finden Sie in Understanding IPv6 and CICS.

Sie können IPv4- und IPv6-Adressen in verschiedenen zulässigen Formaten angeben. Weitere Informationen zu den Adressformaten finden Sie in IP addresses.

#### IPADDRESS({ANY|INADDR\_ANY|DEFAULT|*ip-adresse*})

Gibt die IPv4-Adresse in der Schreibweise mit Trennzeichen an, unter der dieser TCPIPSERVICE für eingehende Verbindungen empfangsbereit ist. Die Adresse muss das Format nnn.nnn.nnn.nnn aufweisen, wobei nnn für einen Wert zwischen 0 und 255 steht. Sie können das Attribut HOST verwenden, um die gleichen Informationen wie in IPADDRESS anzugeben, wobei HOST allerdings auch eine Adresse im IPv6-Format und einen zeichenbasierten Hostnamen unterstützt. Wenn Sie IPv6-Verbindungen nutzen, dann müssen Sie das Attribut

HOST anstelle des Attributs IPADDRESS für die Definitionen verwenden. HOST hat immer Vorrang vor IPADDRESS. Das Attribut IPADDRESS interagiert mit dem Attribut HOST auf verschiedene Weise:

- Wenn Sie HOST angeben, dann wird IPADDRESS immer mit einem Wert überschrieben, der von den Inhalten von HOST abhängig ist:
  - Wenn Sie eine IPv4-Adresse, ANY oder DEFAULT in HOST angeben, dann wird IPADDRESS mit den Inhalten von HOST überschrieben.
  - Wenn Sie eine IPv6-Adresse oder einen zeichenbasierten Hostnamen in HOST angeben, dann wird IPADDRESS mit Leerzeichen überschrieben.
- Wenn Sie sowohl HOST als auch IPADDRESS angeben, dann wird immer der Wert für HOST verwendet:
  - Wenn HOST eine IPv4-Adresse, ANY oder DEFAULT enthält, dann wird IPADDRESS mit den Inhalten von HOST gefüllt.
  - Wenn HOST eine IPv6-Adresse enthält, dann wird IPADDRESS mit Leerzeichen überschrieben.
- Wenn Sie IPADDRESS (nicht aber HOST) angeben, wird HOST mit den Inhalten von IPADDRESS gefüllt.

Wenn Sie die IP-Adresse 0.0.0.0 angeben und für die Option HOST keine Angabe machen, dann gibt das System eine Warnung aus und nimmt den Wert ANY an.

IPADDRESS wird nur für bereits vorhandene IPv4-Funktionen unterstützt. Verwenden Sie die Option HOST für neue Ressourcen.

Folgende Werte sind zulässig:

#### **ANY oder INADDR\_ANY**

Die TCPIPService-Komponente ist an jeder der Adressen empfangsbereit, die in TCP/IP für das Hostsystem bekannt sind. Sie können für einen Host mehrere IP-Adressen definieren. Durch Angabe von ANY oder INADDR\_ANY ermöglichen Sie außerdem die gemeinsame Nutzung der TCPIPService-Definition durch verschiedene CICS-Server.

Wenn Sie ANY oder INADDR\_ANY angeben, dann versucht CICS, eine Bindung zu dem Port in jedem Stack herzustellen, in dem es definiert ist. Wenn Sie außerdem möchten, dass mehrere CICS-Regionen eine Bindung zu dem Port erhalten, dann müssen Sie die Option SHAREPORT in jedem Stack angeben, in dem der Port definiert ist. Andernfalls kann nur eine CICS-Region eine Bindung zu der Portnummer in diesen Stacks erhalten, für die die Option SHAREPORT nicht angegeben ist. Nachfolgende Versuche anderer Regionen zur Herstellung einer Bindung zu jedem Stack schlagen fehl und CICS gibt eine Nachricht aus, in der Sie darüber informiert werden, dass der Port belegt ist. Informationen zur Option SHAREPORT finden Sie in *z/OS Communications Server: IP Configuration Reference*.

#### **DEFAULT**

Die Option DEFAULT ordnet die Affinität dem TCP/IP-Stack zu, der in einer CINET-Multistack-Umgebung als Standardwert definiert wurde. Wenn DEFAULT in einer Nicht-CINET-Umgebung verwendet wird oder wenn kein standardmäßiger TCP/IP-Stack vorhanden ist, dann wird ein Ausnahmebedingungscode geschrieben und keine Affinität zugeordnet.

#### ***ip-adresse***

Die TCPIPService-Komponente akzeptiert Verbindungen zu dieser bestimmten Adresse. Wenn die angegebene Adresse in TCP/IP auf

dem Hostsystem nicht bekannt ist, dann kann die TCPIPSERVICE-Komponente nicht geöffnet werden. Wenn Sie hier eine spezifische Adresse eingeben, dann ist diese Definition möglicherweise für CICS-Server, die in anderen Regionen ausgeführt werden, nicht gültig. In diesem Fall sind Sie eventuell nicht in der Lage, die Definition mit diesen Servern gemeinsam zu nutzen.

#### **MAXDATALEN({32|zahl})**

Gibt die maximal zulässige Datenlänge in Kilobyte an, die von CICS als HTTP-Server über das HTTP-Protokoll oder das USER-Protokoll empfangen werden kann. Der Standardwert ist 32 Kilobyte (KB). Der Mindestwert ist 3 KB und der Maximalwert 524.288 KB. Verwenden Sie MAXDATALEN zum Schutz gegen Denial-of-Service-Attacks, bei denen große Mengen von Daten übertragen werden.

#### **MAXPERSIST({NO|zahl})**

Gibt die maximale Anzahl persistenter Verbindungen von Web-Clients an, die die CICS-Region für diesen Port zu einem angegebenen Zeitpunkt zulässt. Diese Einstellung gilt nur für das HTTP-Protokoll.

- Der Standardwert NO bedeutet, dass in Bezug auf die Anzahl der persistenten Verbindungen keine Beschränkung besteht.
- In einer CICS-Region, bei der die Gefahr einer Überlastung mit persistenten Verbindungen besteht, können Sie einen geeigneten Wert (bis zu dem theoretisch möglichen Maximum von 65535) auf Basis der Anzahl persistenter Verbindungen angeben, die von der CICS-Region gleichzeitig verarbeitet werden können. Wenn dieser Grenzwert erreicht wird und weitere Web-Clients eine Verbindung zu diesem Port herstellen, dann fordert CICS die neuen Clients auf, die Verbindung zu schließen, nachdem eine Antwort empfangen wurde. Wenn die neuen Clients eine Verbindung wiederherstellen und dabei eine Verbindung zu einer anderen CICS-Region hergestellt wird, die den Port gemeinsam mit anderen Einheiten nutzt und den Grenzwert noch nicht erreicht hat, dann können diese Clients eine persistente Verbindung dort aufrechterhalten. Ein HTTP/1.1-Server lässt normalerweise persistente Verbindungen zu, sodass Sie diese Option nur in einer CICS-Region festlegen sollten, in der bereits Leistungsprobleme aufgrund persistenter Verbindungen von Web-Clients mit langer Lebensdauer aufgetreten sind.
- Wenn Sie für diese Option den Wert null angeben, lässt die CICS-Region keine persistenten Verbindungen zu und setzt voraus, dass jeder Web-Client die Verbindung schließt, nachdem die Antwort empfangen wurde. Die Einstellung null für MAXPERSIST stimmt nicht mit der HTTP/1.1-Spezifikation überein, sodass Sie diese Einstellung nur verwenden sollten, wenn eine spezielle Anforderung in einer CICS-Region besteht, die momentan keine externen Anforderungen verarbeitet, z. B. in einer Testumgebung.

#### **PORTNUMBER(port)**

Gibt eine Dezimalzahl im Bereich zwischen 1 und 65535 für den Port an, an dem CICS für eingehende Clientanforderungen empfangsbereit ist.

Die bekannten Ports liegen im Bereich zwischen 1 und 1023. Es ist empfehlenswert, die bekannten Portnummern nur für die Services zu verwenden, denen sie normalerweise zugeordnet sind. Die bekannten Ports für die von CICS unterstützten Services lauten wie folgt:

<b>80</b>	HTTP (nicht SSL)
<b>443</b>	HTTP mit SSL
<b>1435</b>	ECI (registrierte Portnummer)

Konflikte mit anderen Servern in demselben MVS-Image, die möglicherweise die bekannten Ports verwenden, sollten unbedingt behoben werden.

Die gemeinsame Portnutzung muss für alle Ports aktiviert werden, die auf CICS-Systemen innerhalb eines MVS-Images gemeinsam verwendet werden sollen. Weitere Informationen hierzu finden Sie in z/OS Communications Server: IP Configuration Reference.

#### **PRIVACY**

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt. Weitere Informationen hierzu finden Sie in *Obsolete attributes*.

#### **PROTOCOL({HTTP|IIOP|ECI|IPIC|USER})**

Gibt das Protokoll auf Anwendungsebene an, das für den TCP/IP-Port verwendet wird.

**HTTP** Das HTTP-Protokoll wird verwendet. Das HTTP-Protokoll wird von der CICS-Webunterstützung bearbeitet. CICS führt die grundlegenden Annahmeprüfungen für Nachrichten durch, die über dieses Protokoll gesendet und empfangen werden. Dieses Protokoll ist für die bekannten Ports 80 (für HTTP ohne SSL) und 443 (für HTTP mit SSL) erforderlich. HTTP-Anforderungen kommen möglicherweise für die Verarbeitung durch direkt angehängte Benutzertransaktionen in Betracht, wobei die Webanhängetask umgangen wird, die im Attribut TRANSACTION von TCPIPSERVICE definiert wird. Weitere Informationen hierzu finden Sie in *HTTP requests are processed by directly attached user transactions*.

**IIOP** Das Attribut IIOP in TCPIPSERVICE-Definitionen ist veraltet, wird jedoch aus Kompatibilitätsgründen beibehalten.

**ECI** Das CICS-ECI-Protokoll wird verwendet.

**IPIC** Das IPIC-Protokoll wird verwendet. Geben Sie für TCPIPSERVICE-Ressourcen, die für IPIC-Verbindungen (IPIC = IP Interconnectivity; IP-Interkonnektivität) verwendet werden sollen, IPIC an.

**USER** Das benutzerdefinierte Protokoll wird verwendet. Nachrichten werden als Nicht-HTTP-Nachrichten verarbeitet. Sie werden als Nicht-HTTP-Nachrichten gekennzeichnet und unverändert an das Analyseprogramm für die TCPIPSERVICE-Ressource übergeben. Die CICS-Webunterstützungsfunktionen werden für die Verarbeitung der Anforderung verwendet, es werden jedoch keine Annahmeprüfungen für Nachrichten durchgeführt, die über dieses Protokoll gesendet und empfangen werden. Die Verarbeitung aller Nicht-HTTP-Anforderungen muss über das USER-Protokoll ausgeführt werden, sodass sie in Bezug auf die grundlegenden Annahmeprüfungen geschützt sind, die CICS mit dem HTTP-Protokoll für Anforderungen ausführt. Wenn eine HTTP-Nachricht vom USER-Protokoll verarbeitet wird, dann sind Sie für die Überprüfung ihrer Gültigkeit verantwortlich.

#### **REALM(*zeichenfolge*)**

Gibt das Realm an, das für die HTTP-Basisauthentifizierung verwendet wird. Sie können dieses Attribut nur für das HTTP-Protokoll angeben.

Das Realm wird von CICS im WWW-Authentifizierungsheader bereitgestellt und für den Benutzer während des Basisauthentifizierungsprozesses angezeigt. Es identifiziert die Gruppe von Ressourcen, für die die angeforderten Authentifizierungsinformationen (Benutzer-ID und Kennwort oder Kennwortphrase) gelten.



Wenn Sie kein Realm angeben, dann verwendet CICS standardmäßig CICS application *aaaaaaaa*. Hierbei steht *aaaaaaaa* für die Anwendungs-ID (APP-LID) der CICS-Region.

Das Realm kann bis zu 56 Zeichen umfassen und eingebettete Leerzeichen enthalten. Es wird in gemischter Groß-/Kleinschreibung angegeben, wobei die gewählte Groß-/Kleinschreibung beibehalten wird. Geben Sie keine linken und rechten Anführungszeichen an, da CICS sie bei der Assemblierung des WWW-Authentifizierungsheaders bereitstellt.

**Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . - \_ % & ? ! : | ' = ~ + \* , ; < > ( )

Leerzeichen sind ebenfalls zulässig. Wenn Klammern verwendet werden, dann müssen Sie sie immer als Paare aus linken und rechten Klammern verwenden.

**SOCKETCLOSE({NO|*hhmmss*})**

Gibt an, ob und für wie lange CICS wartet, bevor es das Socket schließt. Das Attribut SOCKETCLOSE gilt nicht für die erste Empfangsanforderung, die nach der Herstellung einer Verbindung abgesetzt wird. Bei der ersten Empfangsanforderung für die ECI- und USER-Protokolle wartet CICS 30 Sekunden lang auf Daten, bevor das Socket geschlossen wird. Bei der ersten Empfangsanforderung für das HTTP-Protokoll wartet CICS für die Zeitdauer, die im Wert für DTIMEOUT angegeben wurde, der in TCPIPSERVICE angegebenen Transaktion zugeordnet ist. Wenn dieser DTIMEOUT-Wert null ist, dann wartet CICS 30 Sekunden lang.

Das Intervall wird ab dem Zeitpunkt gemessen, zu dem die erste Empfangsanforderung für eingehende Daten an dieses Socket abgesetzt wurde.

**NO** Das Socket bleibt so lange geöffnet, bis es vom Client oder einem Benutzeranwendungsprogramm in CICS geschlossen wird.

***hhmmss***

Das Intervall (im Format HHMMSS) wird ab dem Zeitpunkt gemessen, zu dem die erste Empfangsanforderung für eingehende Daten abgesetzt wurde. Nach Ablauf des Intervalls wird in CICS eine Zeitlimitüberschreitung für das Socket generiert. Wählen Sie einen Wert aus, der für die Reaktionsfähigkeit des Clients und die Zuverlässigkeit des Netzes geeignet ist. Bei Angabe von 000000 wird das Socket sofort geschlossen, wenn keine Daten für Empfangsanforderungen außer der ersten Empfangsanforderung verfügbar sind.

Wenn Sie diese TCPIPSERVICE-Ressource für die CICS-Webunterstützung mit dem HTTP-Protokoll verwenden, dann bedeutet die Einstellung null für SOCKETCLOSE, dass CICS die Verbindung sofort nach dem Empfang der Daten vom Web-Client schließt, sofern keine weiteren Daten zur Übertragung anstehen. Diese Einstellung bedeutet, dass persistente Verbindungen nicht beibehalten werden können. Es ist keine Konformität mit der HTTP/1.1-Spezifikation gegeben. Die Einstellung null für SOCKETCLOSE mit dem HTTP-Protokoll sollte verwendet werden, wenn in einer CICS-Region, die momentan keine externen Anforderungen verarbeitet (z. B. in einer Testumgebung) eine spezielle Anforderung besteht.

Wenn Sie diese TCPIPSERVICE-Ressource für die CICS-Webunterstützung mit dem HTTP-Protokoll verwenden, dann wirkt sich der SOCKETCLOSE-Wert für TCPIPSERVICE darauf aus, wie bald CICS persistente HTTP-Verbindungen

schließt, wenn der Befehl **SET TCPIPSERVICE OPENSTATUS(CLOSED)** abgesetzt wird. CICS versucht, die persistenten HTTP-Verbindungen wie folgt zu schließen:

- Wenn Anforderungen für persistente HTTP-Verbindungen vorhanden sind, lässt CICS die Verarbeitung ihrer nächsten Anforderung zu. Nach Abschluss der Verarbeitung sendet CICS einen Header für das Schließen an den Client und schließt dann die persistenten HTTP-Verbindungen.
- Wenn keine Anforderungen für persistente HTTP-Verbindungen vorhanden sind, dann schließt CICS die persistenten HTTP-Verbindungen innerhalb von 30 Sekunden oder innerhalb des Zeitraums, der im Attribut **SOCKETCLOSE** von **TCPIPSERVICE** angegeben ist, wenn dieser Wert kleiner als 30 Sekunden ist.

Wenn Sie **PROTOCOL(ECI)** oder **PROTOCOL(IPIC)** angeben, dann muss auch **SOCKETCLOSE(NO)** angegeben werden.

Wenn Sie **PROTOCOL(USER)** angeben, dann werden persistente Sitzungen nicht unterstützt und Sie sollten **SOCKETCLOSE(000000)** angeben.

Nachdem die **TCPIPSERVICE**-Ressource installiert wurde, können Sie diesen Wert mit **CEMT** nicht mehr ändern. Sie müssen die **TCPIPSERVICE**-Ressource inaktivieren und sie dann mit der geänderten Definition erneut installieren.

#### **SPECIFTCPS**

Dieser Parameter gibt den aus acht Zeichen bestehenden Namen der **TCPIP-SERVICE**-Komponente an, der beim Definieren einer generischen **TCPIP-SERVICE**-Komponente in einem Hochverfügbarkeitscluster festgelegt wird. Dieser Parameter ist nur dann gültig, wenn das **IPIC**-Protokoll verwendet wird und der mit den Attributen **PORT** und **HOSTNAME** oder **IPADDRESS** definierte IP-Endpunkt den generischen Endpunkt eines **HA**-Clusters der **CICS**-Regionen darstellt.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl **CREATE** verwenden, werden alle Kleinbuchstaben in Großbuchstaben konvertiert.

#### **SSL({NO|YES|CLIENTAUTH|ATTL\$AWARE})**

Gibt an, ob der **TCP/IP**-Service für die Verschlüsselung und Authentifizierung **SSL** (Secure Socket Layer) verwenden soll. Sie können dieses Attribut für das **HTTP**-, **USER**- und das **IPIC**-Protokoll, nicht jedoch für das **ECI**-Protokoll angeben. Hinweis: Bei Verwendung von **SSL** muss für **MAXSSLTCBs** ein Wert angegeben werden, der mindestens dem Doppelten der Anzahl der **IPIC**-Verbindungen entspricht, die **SSL** in der **CICS**-Region verwenden.

**NO** **SSL** soll nicht verwendet werden. Beim Anfordern der Verbindung werden keine Sicherheitsprüfungen durchgeführt. Für abgehende Nachrichten wird keine Verschlüsselung durchgeführt. **HTTP**-Anforderungen mit den Optionen **SSL(NO)** und **PROTOCOL(HTTP)** kommen möglicherweise für die Verarbeitung durch direkt angehängte Benutzertransaktionen in Betracht, wobei die Webanhangetask umgangen wird. Weitere Informationen hierzu finden Sie in **HTTP requests are processed by directly attached user transactions**.

**YES** Eine **SSL**-Sitzung soll verwendet werden. **CICS** sendet ein Serverzertifikat an den Client. Für alle Nachrichten, die über diesen Port eingehen,

wird die SSL-Entschlüsselung durchgeführt. Die Verschlüsselungsstufe, die für eingehende Nachricht angewendet wird, ist dem Wert des Attributs CIPHERS zu entnehmen.

#### CLIENTAUTH

Eine SSL-Sitzung soll verwendet werden. CICS sendet ein Serverzertifikat an den Client. CICS erwartet den Empfang eines Clientzertifikats vom Partnersystem während eines SSL-Handshakes, wenn die Verbindung angefordert wird.

#### ATTLSAWARE

CICS fragt die Clientverbindung ab, um zu ermitteln, ob AT-TLS aktiv ist. CICS ruft ein Clientzertifikat aus TCP/IP ab, wenn ein solches Zertifikat vom Partnersystem bereitgestellt wurde. ATTLSAWARE muss zusammen mit PROTOCOL(HTTP) TCPIP SERVICES verwendet werden. HTTP-Anforderungen mit den Optionen SSL(ATTLSAWARE) und PROTOCOL(HTTP) kommen möglicherweise für die Verarbeitung durch direkt angehängte Benutzertransaktionen in Betracht, wobei die Webanhangetask umgangen wird. Weitere Informationen hierzu finden Sie in HTTP requests are processed by directly attached user transactions.

#### STATUS({OPEN|CLOSED})

Gibt den Anfangsstatus des Service nach der Installation an. Geben Sie OPEN an, wenn CICS nach der Installation die Empfangsbereitschaft für diesen Service aufnehmen soll. Legen Sie CLOSE fest, wenn CICS nach der Installation nicht für diesen Service empfangsbereit sein soll.

Dieses Attribut wird für TCPIP SERVICE-Ressourcen ignoriert, die dynamisch von einem CICS-Bundle generiert werden. Der Anfangsstatus einer TCPIP SERVICE-Ressource wird aus dem Anfangsstatus des Bundles abgeleitet, das die Ressource definiert.

#### TCPIP SERVICE(*name*)

Gibt den aus acht Zeichen bestehenden Namen dieses Service an.

##### Zulässige Zeichen:

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

#### TRANSACTION(*transaktion*)

Gibt die aus vier Zeichen bestehende ID der CICS-Transaktion an, die zur Verarbeitung neuer Anforderungen, die für diesen Service empfangen werden, verbunden wurde.

- Für eine ECI over TCP/IP-TCPIP SERVICE-Ressource müssen Sie CIEP (oder eine andere Transaktion, die das Programm DFHIEP ausführt) angeben.
- Für eine TCPIP SERVICE-Ressource mit HTTP müssen Sie CWXN (oder eine andere Transaktion, die das Programm DFHWBXN ausführt) angeben. HTTP-Anforderungen kommen möglicherweise für die Verarbeitung durch direkt angehängte Benutzertransaktionen in Betracht, wobei die Webanhangetask umgangen wird. Weitere Informationen hierzu finden Sie in HTTP requests are processed by directly attached user transactions.
- Für eine TCPIP SERVICE-Ressource mit IPIC müssen Sie CISS (oder eine andere Transaktion, die das Programm DFHISCOP ausführt) angeben.

- Für eine TCPIPService-Ressource mit USER müssen Sie CWXU (oder eine andere Transaktion, die das Programm DFHWBXN ausführt) angeben.

#### **TSQPREFIX**(*präfix*)

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt. Weitere Informationen hierzu finden Sie in *Obsolete attributes*.

#### **URM**(**{NO|***programmname***}**)

Gibt den Namen eines durch den Benutzer austauschbaren Programms an, das von diesem Service gestartet werden soll.

**NO** Die automatische Installation ist für diese TCPIPService-Ressource nicht zulässig. Diese Einstellung gilt nur für PROTOCOL(IPIC).

#### *programmname*

Für die Protokolle, für die URM ein erforderliches Attribut darstellt, hängt der Standardprogrammname von dem Wert für das Attribut PROTOCOL ab:

- Geben Sie für das HTTP-Protokoll den Namen eines Analyseprogramms an, das dieser TCPIPService-Ressource zugeordnet werden soll. Standardmäßig wird das von CICS bereitgestellte Analyseprogramm DFHWBAAX verwendet. DFHWBAAX stellt die grundlegende Fehlerbehandlung zur Verfügung, wenn alle Anforderungen für den Port von URIMAP-Definitionen (z. B. Web-Service-Anforderungen) verarbeitet werden sollen. Wenn Sie Unterstützung für Anforderungen bereitstellen müssen, die nicht mithilfe von URIMAP-Definitionen verarbeitet werden, dann muss das für Ihre TCPIPService-Ressource angegebene Analyseprogramm das von CICS bereitgestellte Beispielanalyseprogramm DFHWBADX oder ein eigenes benutzerdefiniertes Analyseprogramm sein. Weitere Informationen zu den Analyseprogrammen finden Sie in *Analyzer programs in Developing system programs*.
- Geben Sie für das IPIC-Protokoll bei Bedarf den Namen des Benutzerprogramms für die automatische Installation für IPIC-Verbindungen an. Wenn Sie für PROTOCOL(IPIC) dieses Attribut nicht angeben, dann verwendet CICS das von CICS standardmäßig bereitgestellte Benutzerprogramm DFHISAIP für die automatische Installation.
- Geben Sie für das USER-Protokoll den Namen eines Analyseprogramms an, das dieser TCPIPService-Definition zugeordnet werden soll. Das Analyseprogramm muss vorhanden sein, und es verarbeitet alle Anforderungen in diesem Protokoll. Das von CICS bereitgestellte Beispielanalyseprogramm DFHWBADX ist geeignet. Weitere Informationen zu Analyseprogrammen finden Sie in *Analyzer programs in Developing system programs*.

## **TCPIPService: In Wechselbeziehung zueinander stehende Attribute**

Die Attribute und die Werte, die Sie angeben können, hängen in einigen Fällen davon ab, welche anderen Attribute Sie angegeben haben.

Für TCPIPService-Ressourcen:

- Die Werte, die Sie für das Attribut AUTHENTICATE angeben können, hängen von dem Wert des Attributs PROTOCOL ab:

AUTHENTICATE	PROTOCOL(ECI) oder PROTOCOL(IPIC)	PROTOCOL(HTTP) oder PROTOCOL(USER)
NO	Ungültig	Gültig
BASIC	Ungültig	Gültig
CERTIFICATE	Ungültig	Gültig
AUTOREGISTER	Ungültig	Gültig
AUTOMATIC	Ungültig	Gültig

- Wenn Sie PROTOCOL(HTTP) oder PROTOCOL(USER) angeben, darf für ATTACHSEC keine Angabe gemacht werden. Sie können das Attribut ATTACHSEC nur angeben, wenn Sie PROTOCOL(ECI) angeben.
- Wenn Sie PROTOCOL(HTTP) angeben, ist der Standardwert für URM das von CICS bereitgestellte standardmäßige Analyseprogramm DFHWBAAX.
- Wenn Sie PROTOCOL(IPIC) angeben, ist der Standardwert für URM das von CICS bereitgestellte standardmäßige Programm für die automatische Installation für IP-Verbindungen DFHISAIP.
- Wenn Sie AUTHENTICATE(CERTIFICATE) oder AUTHENTICATE(AUTOREGISTER) angeben, dann müssen Sie SSL(CLIENTAUTH) oder SSL(ATTLSAWARE) angeben.
- Wenn Sie im Attribut PORTNUMBER eine bekannte Portnummer angeben, legt CICS die Werte anderer Attribute fest:

PORT	PROTOCOL	TRANSACTION	SSL
80	HTTP	CWXN	NO
443	HTTP	CWXN	YES
1435	ECI	CIEP	NO

- Wenn Sie das Attribut PROTOCOL angeben, nicht jedoch das Attribut TRANSACTION, legt CICS das Attribut TRANSACTION fest. Wenn Sie außerdem einen der folgenden Werte im Attribut TRANSACTION angeben, nicht jedoch das Attribut PROTOCOL, legt CICS das Attribut PROTOCOL fest:

PROTOCOL	TRANSACTION
ECI	CIEP
HTTP	CWXN
IPIC	CISS
USER	CWXU

- Sie können den Wert, den CICS für das Attribut TRANSACTION bereitstellt, ändern. Abhängig von dem Wert, den Sie für das Attribut PROTOCOL angeben, sind einige Werte nicht zulässig:

PROTOCOL	TRANSACTION (CWXN)	TRANSACTION (CIEP)	TRANSACTION (CWXU)	TRANSACTION (CISS)
HTTP	Standardwert	Ungültig	Ungültig	Ungültig
USER	Ungültig	Ungültig	Standardwert	Ungültig
ECI	Ungültig	Standardwert	Ungültig	Ungültig
IPIC	Ungültig	Ungültig	Ungültig	Standardwert

- Wenn Sie PROTOCOL(ECI) oder PROTOCOL(IPIC) angeben, dann muss auch SOCKETCLOSE(NO) angegeben werden.

---

## TDQUEUE-Ressourcen

In einer TDQUEUE-Definition werden die Attribute einer Warteschlange mit transienten Daten definiert.

Die folgenden Ressourcen für transiente Daten können mit RDO verwaltet werden:

- Partitionsinterne Ressourcen
- Partitionsübergreifende Ressourcen
- Indirekte Ressourcen
- Ferne Ressourcen

**Partitionsinterne Definitionen** enthalten Attribute, die Informationen zu Wiederherstellungsmerkmalen, Auslöseebenen, zugeordneten Transaktionen, Funktionen und Benutzer-IDs bereitstellen.

**Partitionsübergreifende Definitionen** enthalten Informationen über die zugeordnete QSAM-Datei und die Anzahl der Puffer, die verwendet werden sollen.

**Indirekte Definitionen** geben den zugrunde liegenden Warteschlangennamen an.

**Ferne Definitionen** enthalten den Namen des fernen Systems und den Namen, unter dem die Warteschlange auf diesem fernen System bekannt ist.

Bevor eine Warteschlange mit transienten Daten von einem aktiven CICS-System verwendet werden kann, müssen Sie die zugehörige Definition auf dem aktiven System installieren. CICS verwendet die Definition für den Zugriff auf die Datei, die der Warteschlange zugeordnet ist, und zeichnet die Anzahl der Lese- und Schreiboperationen in der Warteschlange auf.

Ferne Warteschlangen mit transienten Daten können mit der CEDA-Transaktion auf eine der beiden folgenden Arten definiert werden:

- Wenn die Warteschlange TYPE nicht angegeben wird und die Daten nur im Abschnitt REMOTE ATTRIBUTES der Definition eingegeben werden, wird eine ferne Definition erstellt.
- Wenn der TYPE-Wert INTRA oder EXTRA angegeben wird und der Abschnitt REMOTE ATTRIBUTES abgeschlossen ist, werden gleichzeitig eine lokale und eine ferne Ressource eingerichtet.

Ein Beispiel für das Definieren einer Ressourcendefinition für transiente Daten mit Doppelfunktion finden Sie in Abb. 3 auf Seite 189.

Informationen zum Installieren, Ersetzen oder Inaktivieren von TDQUEUE-Ressourcendefinitionen finden Sie in Installing TDQUEUE resource definitions, Replacing TDQUEUE resource definitions und Disabling TDQUEUE resource definitions. Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with BAS transient data queue resource definitions.

## Ressourcendefinition mit Doppelfunktion für transiente Daten

Für Warteschlangen mit transienten Daten darf TYPE=REMOTE nicht angegeben werden.

Stattdessen können Sie den Einsatz einer Ressourcendefinition mit Doppelfunktion (siehe hierzu Shared resources for intercommunication) in Erwägung ziehen. Res-

sourcendefinitionen mit Doppelfunktion können zusammen mit Definitionen für transiente Daten verwendet werden. Abb. 3 enthält ein Beispiel einer Ressourcendefinition mit Doppelfunktion für Ressourcen mit transienten Daten.

Wenn die in Abb. 3 dargestellte Definition auf einem System mit dem Namen CICQ installiert wird, dann wird diese Definition als lokale partitionsinterne Warteschlange (REMOTESYSTEM-Wert CICQ) definiert.

Wenn die in Abb. 3 dargestellte Definition auf einem anderen System als CICQ installiert wird, dann wird die Definition als ferne Warteschlange (REMOTE) definiert.

```

TDqueue      : TDQ1
Group        : Example
Description  ==>
TYPE         ==> Intra                      Extra | INTra | INDirect
EXTRA PARTITION PARAMETERS
Databuffers  :                               1-255
DDname       :
DSname       :
Sysoutclass  :
Erroroption  :                               Ignore | Skip
Opentime     :                               Initial | Deferred
REWInd       :                               Leave | Reread
TYPEFile     :                               Input | Output | Rdback
RECORDSize   :                               0-32767
BLOCKSize    :                               0-32767
RECORDFormat :                               Fixed | Variable
BLOCKFormat  :                               Blocked | Unblocked
Printcontrol :                               A | M
Disposition  :                               Shr | Old | Mod
INTRA PARTITION PARAMETERS
Atifacility  ==> Terminal                    Terminal | File | System
RECOVstatus  ==> Logical                     No | Physical | Logical
Facilityid   ==> FR1
TRAnsId      ==>
TRiggerlevel ==> 00001                       0-32767
UserId       ==>

INDOUBT ATTRIBUTES
WAIT         ==> Yes                         Yes|No
WAITAction   ==> Reject                      Queue|Reject

INDIRECT PARAMETERS
Indirectname :

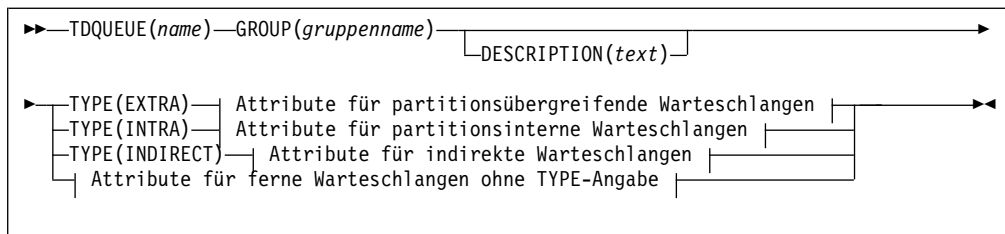
REMOTE PARAMETERS
REMOTENAME   ==> FR1
REMOTESystem ==> CICQ
REMOTELength ==>                               0-32767

```

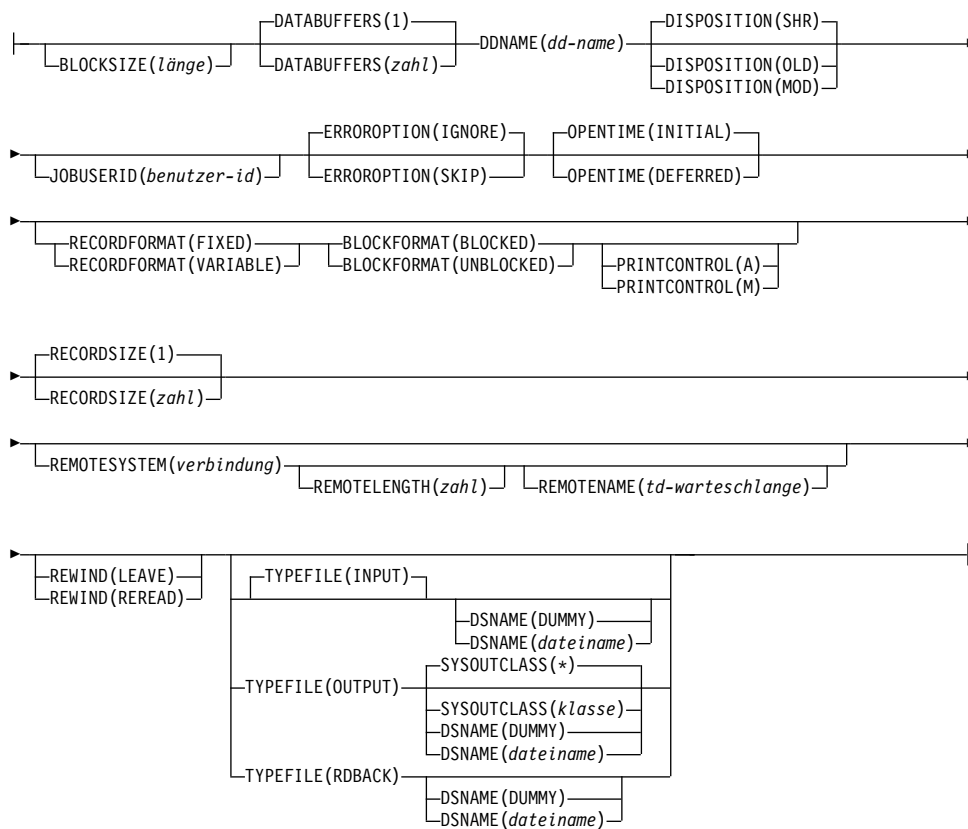
Abbildung 3. Ressourcendefinition mit Doppelfunktion für transiente Daten

## TDQUEUE-Attribute

Beschreibt die Syntax und die Attribute der TDQUEUE-Ressource.

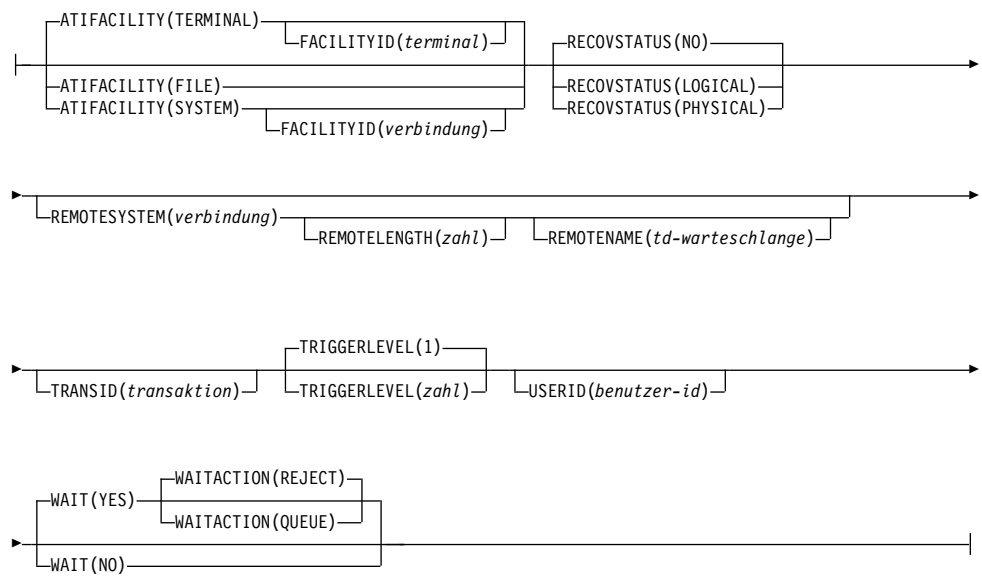


### Attribute für partitionsübergreifende Warteschlangen:

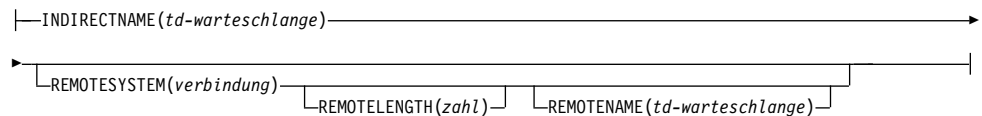




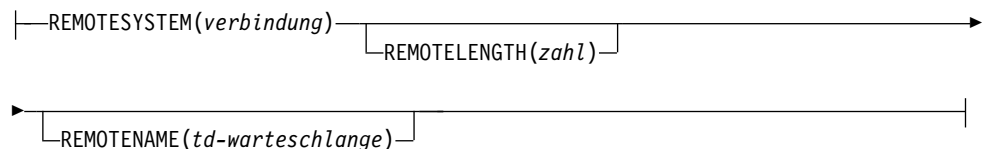
### Attribute für partitionsinterne Warteschlangen:



### TDQUEUE-Attribute für indirekte Warteschlangen:



### TDQUEUE-Attribute für ferne Warteschlangen ohne TYPE-Angabe:



### ATIFACILITY({TERMINAL|FILE|SYSTEM}) (nur partitionsinterne Warteschlangen)

Gibt den Typ des Ziels für die Warteschlange an.

**FILE** Die Warteschlange mit transienten Daten wird als Datei für Datensätze verwendet, die keinem bestimmten Terminal oder System zugeordnet sind. Für die ATI (Automatic Transaction Initiation; automatische Transaktionsinitialisierung) muss kein Terminal verfügbar sein.

#### SYSTEM

Die Warteschlange mit transienten Daten muss der angegebenen Sys-

tem-ID zugeordnet werden. Das System muss auf dem lokalen CICS-System mit einer RDO-CONNECTION-Definition definiert werden.

Durch Angabe von ATIFACILITY(SYSTEM) wird eine DTP-Sitzung (DTP = Distributed Transaction Processing; verteilte Transaktionsverarbeitung) eingeleitet. Weitere Informationen zu den für DTP bei der Anwendungsprogrammierung zu berücksichtigenden Faktoren finden Sie im Abschnitt zu Distributed Transaction Processing (DTP).

### **TERMINAL**

Die Warteschlange mit transienten Daten wird dem Terminal zugeordnet. Das Terminal muss für CICS definiert werden. Wenn Sie für TERMINAL keinen Wert angeben, dann wird standardmäßig der Wert von FACILITYID verwendet. Wenn die ATI wie in den Attributen TRANSID und TRIGGERLEVEL angegeben verwendet wird, dann wird die eingeleitete Transaktion dem angegebenen Terminal zugeordnet, das verfügbar sein muss, um die Transaktion einleiten zu können.

### **BLOCKFORMAT({BLOCKED|UNBLOCKED|*leer*}) (nur partitionsübergreifende Warteschlangen)**

Gibt das Blockformat der Datei an. Es gibt keinen Standardwert. Wenn Sie das Datensatzformat (Attribut RECORDFORMAT) als "Nicht definiert" angeben (oder den Standardwert übernehmen), dann darf für das Attribut BLOCKFORMAT kein Wert angegeben werden.

*leer*      Gibt an, dass für diese Datei kein Blockformat definiert ist. Lassen Sie dieses Feld leer, wenn Sie für RECORDFORMAT keinen Wert angeben.

### **BLOCKED**

Das geblockte Datensatzformat.

### **UNBLOCKED**

Das entblockte Datensatzformat.

Für partitionsübergreifende Warteschlangen, die als Schnittstelle zum internen JES-Reader verwendet werden, sollte unbedingt ein entblocktes Datensatzformat angegeben werden. Wenn Sie ein geblocktes Datensatzformat verwenden, dann wird Ihr Job in der SYSOUT-Datei gehalten und nicht direkt an JES gesendet, solange nicht eine der folgenden Aktionen ausgeführt wird:

- Sie geben nach der JES-Steueranweisung /\*EOF eine zweite Anweisung /\*EOF an.
- Ihre Anwendung schreibt einen anderen Job in dieselbe Warteschlange.
- Sie schließen die Warteschlange nach der Jobschreiboperation explizit.
- Sie führen einen normalen CICS-Systemabschluss durch.

### **BLOCKSIZE({*länge*}) (nur partitionsübergreifende Warteschlangen)**

Gibt die Länge des Blocks im Bereich zwischen 0 und 32767 (in Byte) an.

Der Maximalwert, den Sie angeben können, ist davon abhängig, ob das Attribut SYSOUTCLASS entweder explizit oder standardmäßig angegeben wird. Außerdem ist dieser Wert abhängig davon, ob für RECORDFORMAT die Option FIXED oder VARIABLE angegeben wird.

- Wenn Sie SYSOUTCLASS angeben, beträgt der maximale Wert für RECORDSIZE 8968.
- Jeder Block in einer Datei mit variablem Format besteht aus einem Blockdeskriptorwort gefolgt von mindestens einem logischen Datensatz. Wenn Sie RECORDFORMAT(VARIABLE) angeben, dann muss der für BLOCKSIZE an-

gegebene Wert also vier Byte für das Blockdeskriptorwort umfassen und außerdem genügend Speicherplatz für den größten möglichen logischen Datensatz berücksichtigen.

Diese Grenzwerte sind in der folgenden Tabelle zusammengefasst:

Wurde SYSOUTCLASS angegeben?	Ja	Ja	Nein	Nein
RECORDFORMAT	VARIABLE	FIXED	VARIABLE	FIXED
Maximaler Wert für RECORDSIZE	8968	8968	32763	32767
Maximaler Wert für BLOCKSIZE	8972	8968	32767	32767

#### **DATABUFFERS({1|zahl}) (nur partitionsübergreifende Warteschlangen)**

Gibt die Anzahl der bereitgestellten Puffer bis zu einem Maximum von 255 an.

#### **DDNAME(dd-name)**

Gibt einen aus einem bis acht Zeichen bestehenden Wert an, der auf eine Datei verweist, die im JCL-Code für den Systemstart definiert wurde. Der Name darf nicht mit den Zeichen „DFH“ beginnen, da diese zur Verwendung durch CICS reserviert sind, es sei denn, der Name beschreibt eines der Standardziele.

#### **DESCRIPTION(text)**

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

#### **DISPOSITION({SHR|OLD|MOD}) (nur partitionsübergreifende Warteschlangen)**

Gibt die Disposition der Datei an.

**MOD** CICS geht zunächst davon aus, dass die Datei vorhanden ist. Für eine vorhandene sequenzielle Datei wird durch Angabe von MOD der Schreib-/Lesemechanismus nach dem letzten Datensatz in der Datei positioniert. Der Schreib-/Lesemechanismus wird nach dem letzten Datensatz positioniert, sobald die Datei für die Ausgabe geöffnet wird.

Wenn CICS keine Datenträgerinformationen für die Datei in der Datendefinitionsanweisung, im Katalog oder als Übergabe mit der Datei aus einem vorherigen Schritt finden kann, dann geht das Programm davon aus, dass die Datei im aktuellen Jobschritt erstellt wird. Eine auf diese Weise dynamisch zugeordnete Datei wird gelöscht, wenn die Warteschlange geschlossen wird. Alle Datensätze gehen verloren.

Für eine neue Datei wird der Schreib-/Lesemechanismus durch Angabe von MOD am Anfang der Datei positioniert.

**OLD** Die Datei war bereits vor diesem Jobschritt vorhanden.

**SHR** Die Datei war bereits vor diesem Jobschritt vorhanden und kann von anderen gleichzeitig ablaufenden Jobs gelesen werden.

**DSNAME({*dateiname*|DUMMY}) (nur partitionsübergreifende Warteschlangen)**

Gibt den Namen der QSAM-Datei an, die zum Speichern von Datensätzen verwendet werden soll, die in diese partitionsübergreifende Warteschlange geschrieben werden.

Wenn CICS eine Anforderung zum Öffnen einer partitionsübergreifenden Warteschlange mit transienten Daten erhält, dann wird auf den JCL-Code für den Systemstart verwiesen, um zu überprüfen, ob eine Dateidefinition erstellt wurde. Wenn keine gefunden wird, dann wird der aus 44 Zeichen bestehende Name, der im Attribut DSNAME angegeben wurde, zur dynamischen Zuordnung der erforderlichen Datei verwendet.

Wenn Sie über JCL-Code verfügen, der für diese Warteschlange den DSCNAME-Wert vorab einem DSNAME-Wert zuordnet, wird der DSNAME-Wert in der Ressourcendefinition mit dem DSNAME-Wert aus dem JCL-Code überschrieben. Die JCL-Zuordnung hat immer Priorität.

Partitionierte Dateien (PDS) werden im Attribut DSNAME nicht unterstützt. Wenn Sie ein PDS-Member für eine Datei einer partitionsübergreifenden Warteschlange verwenden wollen, dann müssen Sie es im JCL-Code explizit angeben. Beachten Sie dabei, dass bei Abfragen für diese Warteschlange der DSCNAME-Wert, der zurückgegeben wird, keine Angabe zum Membernamen enthält.

**DUMMY**

Der Name einer Pseudodatei.

*name* Der aus 44 Zeichen bestehende Name einer physischen Datei.

**Achtung:** Wenn Sie Protokolldatenströme für die Dateien partitionsübergreifender Warteschlangen verwenden, können unvorhersehbare Ergebnisse auftreten.

**ERROROPTION({IGNORE|SKIP}) (nur partitionsübergreifende Warteschlangen)**

Gibt die Aktion an, die ausgeführt werden soll, wenn ein E/A-Fehler auftritt. Hierbei kann es sich um eine der folgenden Aktionen handeln:

**IGNORE**

Der Block, der den Fehler verursacht hat, wird akzeptiert.

**SKIP** Der Block, der den Fehler verursacht hat, wird übersprungen.

**FACILITYID(*terminal*|*verbindung*) (nur partitionsinterne Warteschlangen)**

Gibt ein aus vier Zeichen bestehendes Feld an, dass folgende Informationen enthalten kann:

- Die System-ID für eine partitionsinterne Warteschlange, in der ATIFACILITY(SYSTEM) angegeben ist.
- Die Terminalkennung, für die ATIFACILITY(TERMINAL) angegeben ist.

Wenn Sie im Feld FACILITYID nichts angeben, wird standardmäßig in jedem Fall der Name der Warteschlange verwendet.

Wenn ATIFACILITY(FILE) angegeben wird, muss das Feld FACILITYID leer bleiben.

**GROUP(*gruppenname*)**

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

**INDIRECTNAME(td-warteschlange) (nur indirekte Warteschlangen)**

Gibt den Namen der Warteschlange mit transienten Daten an, an die diese indirekte Warteschlange Daten weiterleitet. Die Warteschlange mit transienten Daten muss in CICS definiert werden. Es kann sich dabei um eine partitionsinterne, partitionsübergreifende, ferne oder indirekte Warteschlange handeln.

**JOBUSERID(benutzer-id) (nur partitionsübergreifende Warteschlangen)**

Gibt eine Benutzer-ID an, unter der der an den internen Reader übergebene JCL-Job ausgeführt wird, wenn der JCL-Code eine JOB-Karte enthält, ohne dass der Parameter USER angegeben wurde.

Wenn JOBUSERID nicht definiert wird, dann wird standardmäßig die Benutzer-ID der CICS-Region verwendet.

**OPENTIME({INITIAL|DEFERRED}) (nur partitionsübergreifende Warteschlangen)**

Gibt den Anfangsstatus der Datei an. Der Anfangsstatus kann folgende Werte angeben:

**DEFERRED**

Die Datei bleibt geschlossen, bis Sie angeben, dass Sie sie mit dem Befehl CEMT INQUIRE|SET TDQUEUE öffnen wollen.

**INITIAL**

Die Datei soll während der Installation geöffnet werden. Wenn das Attribut DSNNAME nicht angegeben wird und der Dateiname in der Datendefinitionsanweisung im JCL-Code für den Systemstart nicht definiert wird, dann wird die Warteschlange mit transienten Daten dem JES (Job Entry Subsystem; Jobeingabesubsystem) während des Systemstarts von CICS zugeordnet.

**PRINTCONTROL({ASA|MACHINE|leer}) (nur partitionsübergreifende Warteschlangen)**

Gibt die Steuerzeichen an, die verwendet werden sollen. Es gibt keinen Standardwert.

Wenn Sie für RECORDFORMAT den Standardwert 'leer' übernehmen, dann dürfen Sie im Feld PRINTCONTROL nichts angeben. Sie können die folgenden Zeichen verwenden:

**ASA** ASA-Steuerzeichen.

*leer* Es dürfen keine Steuerzeichen verwendet werden.

**MACHINE**

Die Maschinensteuerzeichen.

**RECORDFORMAT({FIXED|VARIABLE|leer}) (nur partitionsübergreifende Warteschlangen)**

Gibt das Datensatzformat der Datei an.

**leer** Wird RECORDFORMAT nicht angegeben (und leer bleibt), dürfen auch in den Feldern BLOCKFORMAT und PRINTCONTROL keine Werte angegeben werden. Wenn in der Ressourcendefinition RECORDFORMAT nicht angegeben wird, dann versucht TD, dieses Attribut aus

dem JCL-Code für den Systemstart von CICS oder aus der QSAM-Dateidefinition abzurufen, wenn versucht wird, die Warteschlange zu öffnen. Die Öffnungsanforderung schlägt fehl, wenn diese Informationen nicht aus einer dieser Quellen abgeleitet werden können.

#### **FIXED**

Feste Datensätze. Wenn Sie RECORDFormat(Fixed) angeben, müssen Sie auch ein Blockformat angeben.

#### **VARIABLE**

Variable Datensätze. Wenn Sie RECORDFormat(Variable) angeben, dann müssen Sie auch ein Blockformat angeben.

#### **RECORDSIZE({1|zahl}) (partitionsübergreifende und ferne Warteschlangen)**

Gibt die Datensatzlänge im Bereich zwischen 0 und 32767 (in Byte) an.

**1** Die Standarddatensatzlänge ist 1 Byte.

*zahl* Die Datensatzlänge (in Byte) bis maximal 32767.

Der Maximalwert, den Sie angeben können, ist davon abhängig, ob das Attribut SYSOUTCLASS entweder explizit oder standardmäßig angegeben wird. Außerdem ist dieser Wert abhängig davon, ob für RECORDFORMAT die Option FIXED oder VARIABLE angegeben wird.

- Wenn Sie SYSOUTCLASS angeben, beträgt der maximale Wert für RECORDSIZE 8968.
- Für Dateien im V-Format besteht jeder logische Datensatz aus einem Datensatzdeskriptor, auf den ein Datensatz folgt. Der Wert, der im Feld RECORDSIZE eingegeben wird, muss vier Byte für das Datensatzdeskriptorwort (oder LLBB) enthalten. Außerdem sollte genügend Speicherplatz für den längsten möglichen Datensatz berücksichtigt werden.

Diese Grenzwerte werden in der Beschreibung des Attributs BLOCKSIZE zusammengefasst:

#### **RECOVSTATUS({NO|PHYSICAL|LOGICAL}) (nur partitionsinterne Warteschlangen)**

Gibt die Wiederherstellbarkeitsattribute der Warteschlange im Fall einer abnormalen Beendigung von CICS oder der Transaktion an, die die Warteschlange verarbeitet. Die Wiederherstellbarkeitsattribute lauten wie folgt:

#### **LOGICAL**

Diese Warteschlange ist logisch wiederherstellbar. Die automatische Protokollierung wird ausgeführt, um die Zugriffe von Anwendungsprogrammen zu überwachen. Wenn eine Transaktion, die auf diese Warteschlange zugegriffen hat, zum Zeitpunkt der abnormalen Beendigung, während des nachfolgenden Wiederanlaufs nach dem Systemabsturz oder während des dynamischen Zurücksetzens von Transaktionen ausgeführt wurde, dann wird die Warteschlange in dem Status wiederhergestellt, in dem sie sich vor der Änderung durch die momentan ausgeführte Arbeitseinheit befunden hat.

Wenn auf diese Warteschlange zugegriffen wird, dann wird die Task, die den Befehl DELETEQ TD, WRITEQ TD oder READQ TD abgesetzt hat, am Ein- und/oder Ausgabeendpunkt der Warteschlange mit transienten Daten eingestellt. Die Einreihung in die Warteschlange (sog. Enqueue) wird beibehalten, bis die Task abgeschlossen wird (oder eine Synchronisationspunktanforderung absetzt, um das Ende einer Arbeitseinheit zu signalisieren). Auf diese Weise wird die Integrität der Daten sichergestellt, auf die zugegriffen wird. Dies bedeutet, dass die Enqueues für längere Zeit beibehalten werden können und zu einer War-

teschlängenaussperrung führen können, wenn ein Anwendungsprogramm, das auf die Warteschlange zugreift, mehrere Arbeitseinheiten für die Warteschlange ausführt, ohne dass jede separate Arbeitseinheit in CICS durch Absetzen einer Synchronisationspunktanforderung definiert wird.

Darüber hinaus werden, wenn eine Anforderung DELETEQ für eine logisch wiederherstellbare Warteschlange abgesetzt wird, sowohl am Ein- als auch am Ausgabeende der Warteschlange Enqueues vorgenommen. Dadurch kann sich das Risiko einer Enqueue-Aussperrung erhöhen.

**Anmerkung:** CICS stellt eine Enqueue-Schutzfunktion für logisch wiederherstellbare (im Gegensatz zu physisch wiederherstellbaren) Warteschlangen für transiente Daten ähnlich wie bei wiederherstellbaren Dateien zur Verfügung. Allerdings stuft CICS jedes logisch wiederherstellbare Ziel als zwei separate wiederherstellbare Ressourcen ein, wobei eines für Schreib- und das andere für Leseoperationen verwendet wird.

Im Falle eines Dateidatensatzes wird ein Datensatz als eine einzige Ressource behandelt und erfordert nur eine Sperre. Die Warteschlange für transiente Daten verfügt jedoch über zwei 'Enden' (Leseende und Schreibende), an denen unabhängig voneinander Enqueues (und somit Sperrungen) vorgenommen werden können. Dies ist darauf zurückzuführen, dass CICS zur Steuerung von (gleichzeitigen) Schreib- und Leseoperationen in der Warteschlange für transiente Daten zwei Zeiger (Cursor) verwalten muss, und zwar einen zum Lesen und einen zum Schreiben. Diese müssen gegen Transaktionen geschützt werden, die einen Konflikt verursachen.

Warteschlangendatensätze werden in mindestens einem Steuerintervall (CI = Control Interval) gehalten. Jedes CI wird zur Freigabe markiert, sobald der letzte Datensatz darin gelesen wurde. Allerdings findet die Freigabe erst statt, wenn das Ende der Task oder der nächste Synchronisationspunkt für Benutzer erreicht wird.

**NO** Diese Warteschlange ist nicht wiederherstellbar. Die automatische Protokollierung wird nicht ausgeführt, um Zugriffe auf diese Warteschlange zu überwachen. Warteschlangendatensätze werden in mindestens einem Steuerintervall (CI = Control Interval) gehalten. Jedes CI wird freigegeben, sobald der letzte Datensatz darin gelesen wurde.

#### **PHYSICAL**

Diese Warteschlange ist physisch wiederherstellbar. Die automatische Protokollierung wird ausgeführt, um die Zugriffe von Anwendungsprogrammen zu überwachen. Wenn ein Wiederanlauf nach einem Systemabsturz durchgeführt wird, muss diese Warteschlange in dem Status wiederhergestellt werden, in dem sie sich bei der Beendigung von CICS befunden hat.

Die Warteschlange wird **nicht** in dem Status wiederhergestellt, in dem Sie sich bei der Beendigung von CICS befunden hat, wenn die letzte Aktion in der Warteschlange eine READQ-Anforderung war und wenn die zugeordnete Arbeitseinheit (UOW) die Änderungen nicht festgeschrieben hat. Bei einem Wiederanlauf nach einem Systemabsturz wird die letzte Leseoperation zurückgesetzt, sodass der Anschein entsteht, als habe sie niemals stattgefunden.

Warteschlangendatensätze werden in mindestens einem Steuerintervall (CI = Control Interval) gehalten. Jedes CI wird freigegeben, sobald der letzte Datensatz darin gelesen wurde.

**REMOTENAME(*td-warteschlange*) (nur ferne Warteschlangen)**

Gibt an, ob die Warteschlange mit transienten Daten sich auf einem fernen System befindet. Außerdem gibt diese Option den aus vier Zeichen bestehenden Namen an, unter dem die Warteschlange in dem System oder in der Region bekannt ist, auf dem bzw. in der sie sich befindet.

**Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

**REMOTELength({1|*zahl*}) (nur ferne Warteschlangen)**

Gibt die Länge im Bereich zwischen 1 und 32767 (in Byte) an.

Für SYSOUT-Dateien darf der im Feld REMOTELength eingegebene Wert nicht größer als 8968 Byte sein (wenn das Attribut SYSOUTCLASS angegeben wurde).

**1** Die Länge beträgt 1 Byte.

*zahl* Die Länge (in Byte) bis maximal 32767.

Wenn die Warteschlange mit TYPE=EXTRA definiert ist und für REMOTELength kein Wert angegeben wurde, dann wird während der Installation der Wert des Attributs RECORDSIZE verwendet.

**REMOTESYSTEM(*verbindung*)**

Gibt den aus vier Zeichen bestehenden alphanumerischen Namen des Systems oder der Region an, auf dem bzw. in der sich die ferne Warteschlange mit transienten Daten befindet. Der eingegebene Name muss mit dem Namen übereinstimmen, der in RDO CONNECTION angegeben wurde, oder aber mit den ersten vier Zeichen des Namens in einer RDO-IPCONN-Definition. Weitere Informationen zu den Verbindungsdefinitionen finden Sie in „CONNECTION-Attribute“ auf Seite 10 oder „IPCONN-Attribute“ auf Seite 77.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Wenn die Definition der Warteschlange mit transienten Daten installiert wird, dann vergleicht das System den im Attribut REMOTESYSTEM eingegebenen Namen mit der System-ID. Wenn die Namen unterschiedlich sind, befindet sich das System oder die Region an einem fernen Standort. Wenn die Namen identisch sind, wird der im Attribut TYPE angegebene Wert verwendet. Wenn für das Attribut TYPE kein Wert angegeben wurde, dann schlägt die Installation fehl.

**REWIND({LEAVE|REREAD}) (nur partitionsübergreifende Warteschlangen)**

Gibt die Disposition einer Banddatei an. Die Disposition kann wie folgt lauten:

**LEAVE**

Das aktuelle Band ist am logischen Ende der Datei positioniert.

**REREAD**

Das aktuelle Band ist am logischen Beginn der Datei positioniert.



### **SYSOUTCLASS({A..Z|0..9|\*|*leer*}) (nur partitionsübergreifende Warteschlangen)**

Anstatt eine partitionsübergreifende Warteschlange einer physischen Datei zuzuordnen, können Sie sie einer Systemausgabedatei (SYSOUT) zuordnen.

Verwenden Sie das Attribut SYSOUTCLASS, um die Klasse für die SYSOUT-Datei anzugeben.

#### **A..Z|0..9**

Ein einzelnes alphabetisches oder numerisches Zeichen, das für eine Ausgabeklasse steht, die auf dem MVS-System definiert wurde, auf dem der CICS-Job ausgeführt werden soll.

#### **Zulässige Zeichen:**

A-Z 0-9

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

**\*** Dies ist die Standardklasse. Für SYSOUTCLASS wird standardmäßig ein Stern (\*) angenommen, wenn Sie für das Attribut DSNNAME nichts angeben und im Typefile-Feld OUTPUT angeben.

*leer* Für SYSOUTCLASS wird standardmäßig ein Leerzeichen angenommen, wenn Sie für das Attribut DSNNAME nichts angeben und für das Typefile-Attribut INPUT oder RDBACK angeben. Im zweiten Fall schlägt die Operation zum Öffnen fehl, weil ein Wert für DSNNAME bei TYPEFILE=INPUT oder TYPEFILE=RDBACK angegeben werden **muss**.

Sie können SYSOUTCLASS als Alternative für DSNNAME verwenden. Wie bei DSNNAME wurde die Warteschlange SYSOUT möglicherweise bereits vorab mit einer JCL-Datendefinitionsanweisung zugeordnet. Eine JCL-Datendefinitionsanweisung überschreibt jede Angabe, die mithilfe der TDQUEUE-Ressourcendefinition gemacht wurde.

Wenn CICS eine Anforderung zum Öffnen einer partitionsübergreifenden Warteschlange mit transienten Daten erhält, dann wird auf den JCL-Code für den Systemstart verwiesen, um zu überprüfen, ob eine Dateidefinition erstellt wurde. Wenn keine gefunden wird, dann wird der aus 44 Zeichen bestehende Name, der im Attribut DSNNAME angegeben wurde, zur dynamischen Zuordnung der erforderlichen Datei verwendet.

Wenn im JCL-Code für eine Warteschlange SYSOUT angegeben wurde, dann können auch andere Attribute als die Klasse (z. B. Formulartypen) angegeben werden.

**Anmerkung:** Die Angabe von SYSOUT-Dateien mit RDO bietet nur Unterstützung für den Klassenparameter. Wenn Sie andere Parameter benötigen, dann müssen Sie die SYSOUT-Dateien im JCL-Code angeben.

Weitere Informationen zu SYSOUT und zu den zugehörigen Klassen finden Sie im Handbuch z/OS MVS JCL User's Guide.

### **TDQUEUE(*name*)**

Gibt den aus einem bis vier Zeichen bestehenden Namen einer Warteschlange mit transienten Daten an.

#### **Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Wenn der angegebene Name weniger als vier Zeichen umfasst, dann wird er linksbündig ausgerichtet und mit Leerzeichen auf eine Länge von vier Zeichen aufgefüllt.

**Anmerkung:**

1. Wenn Sie in einem Namen ein Komma (,) verwenden, dann können Befehle wie der folgende nicht verwendet werden:

```
CEMT INQUIRE TDQUEUE(wert1,wert2)
CEMT SET      TDQUEUE(wert1,wert2)
```

Hierbei dient das Komma als Listenbegrenzer. Informationen zur Verwendung von Listen mit Ressourcen-IDs finden Sie in .

2. Wenn Sie die Warteschlangen mit transienten Daten mithilfe von RACF schützen, dann vermeiden Sie im Namen die Verwendung der Zeichen % und &. In den RACF-Befehlen haben diese Zeichen eine Sonderbedeutung, wenn Sie in Profilnamen verwendet werden. Weitere Informationen hierzu finden Sie in .

**TRANSID(transaktion) (nur partitionsinterne Warteschlangen)**

Gibt den Namen der Transaktion an, die automatisch eingeleitet werden soll, wenn die Auslöseebene erreicht wird. Transaktionen werden auf diese Weise eingeleitet, um Datensätze aus der Warteschlange zu lesen. Wenn das Attribut TRANSID nicht (oder aber TRIGGERLEVEL(0)) angegeben wird, dann müssen Sie eine andere Methode zum Terminieren von Transaktionen für das Lesen von Datensätzen aus Warteschlangen mit transienten Daten verwenden.

**Die angegebene Transaktion darf sich nicht auf einem fernen CICS-System befinden.** Andernfalls schlägt die Initialisierung der Transaktion fehl und an der Konsole wird eine Warnung ausgegeben.

**TRIGGERLEVEL({1|zahl}) (nur partitionsinterne Warteschlangen)**

Gibt die Anzahl der Datensätze an, die aufgelaufen sein müssen, bevor eine Task automatisch zu deren Verarbeitung initialisiert wird. (Diese Anzahl wird als Auslöseebene bezeichnet.)

Wenn Sie das Attribut TRANSID angeben, dann wird für TRIGGERLEVEL der Standardwert 1 angenommen. Geben Sie die Auslöseebene 0 an, wenn die ATI-Verarbeitung inaktiviert werden soll. Wenn Sie keine Transaktions-ID angeben, wird die Auslöseebene ignoriert.

Wenn Sie ATIFACILITY(TERMINAL) angegeben haben, wird die Task erst initialisiert, wenn das angegebene Terminal verfügbar ist. Wenn Sie ATIFACILITY(FILE) angegeben haben, ist für die Initialisierung der Task kein Terminal erforderlich.

Wenn bei Verwendung der maximalen Anzahl von Tasks eine Speicherengpassbedingung (short-on-storage) oder eine Bedingung aufgrund nicht vorhandenen Speicherplatzes (no-space) für ein Ziel, bei dem es sich nicht um ein Terminal handelt, auftritt, dann wird die Task nicht initialisiert. Dies ist auch in den Phasen 1 und 2 der Initialisierung und während der Abschlussphase des Systemabschlusses der Fall. Die Task wird initialisiert, wenn die Stressbedingung nicht mehr vorhanden ist und eine nachfolgende TD WRITE-Operation auftritt.

Bei logisch wiederherstellbaren Warteschlangen mit transienten Daten wird die ATI-Task erst dann zugeordnet, wenn durch die Task eine Forward-Commit-Operation ausgeführt wird. Dies kann bedeuten, dass die Auslöseebene weit überschritten wurde, bevor die ATI auftritt.

Wenn ein z/OS Communications Server-Terminal für die CSTL-Transaktion auf zwei ISC-CICS-Systemen mit der Auslöseebene 1 definiert wird, kann dies zu einem Leistungsproblem führen, wenn beide Systeme das Terminal wiederholt anfordern und freigeben, um Nachrichten für den Sitzungsstart und das Sitzungsende zu schreiben.

Sie können die Auslöseebene ändern, wenn CICS mit der CEMT-Transaktion ausgeführt wird. Wenn Sie die Auslöseebene auf einen Wert reduzieren, der gleich (oder kleiner) der Anzahl der Datensätze ist, die bislang aufgelaufen sind, dann wird die Task initialisiert, wenn der nächste Datensatz erfolgreich in die Warteschlange eingestellt wird.

**1** Es kann nur ein Datensatz auflaufen.

*zahl* Die Anzahl an Datensätzen, die (bis zum Maximum von 32767) auflaufen kann, bevor die ATI auftritt.

#### **TYPE({EXTRA|INTRA|INDIRECT})**

Gibt die folgenden Typen von Warteschlangen mit transienten Daten an:

##### **EXTRA**

Eine Warteschlange, die sich außerhalb der CICS-Region befindet, wird CICS zugeordnet.

Partitionsübergreifende Warteschlangen werden für folgende Aktionen verwendet:

- Senden von Daten außerhalb der CICS-Region: Hierbei kann es sich z. B. um Daten handeln, die von einer Transaktion zur Verarbeitung durch ein Stapelverarbeitungsprogramm erstellt werden.
- Abrufen von Daten außerhalb der Region: Hierbei kann es sich z. B. um Daten handeln, die von Terminals als Eingabe für eine Transaktion empfangen wurden.

Partitionsübergreifende Daten sind sequenziell und werden von QSAM verwaltet.

##### **INDIRECT**

Eine indirekte Warteschlange stellt eine Warteschlange dar, die nicht auf eine tatsächliche Datei, sondern auf eine andere Warteschlange verweist. Bei einer indirekten Warteschlange kann es sich um eine partitionsübergreifende, partitionsinterne oder ferne Warteschlange oder auch um eine weitere indirekte Warteschlange handeln.

Sie können beispielsweise einen anderen symbolischen Namen (INDIRECTDEST) für jeden der verschiedenen Nachrichtentypen angeben. Sie können dann diese Nachrichtentypen an dieselbe physische Warteschlange (INDIRECTDEST) oder an unterschiedliche physische Warteschlangen senden.

Im Beispielprogramm DFH\$TDWT wird gezeigt, wie indirekte Warteschlangen zum Senden unterschiedlicher Kategorien von Nachrichten an dasselbe Terminal verwendet werden können. Informationen zur Programmierung bezüglich DFH\$TDWT finden Sie in Using the transient data write-to-terminal program (DFH\$TDWT). Im Handbuch *CICS/ESA 4.1 Sample Applications Guide* sind DFH\$TDWT-Beispieldefinitionen aufgeführt.

Wenn im Operanden QUEUE eines Befehls EXEC CICS WRITEQ TD, EXEC CICS READQ oder eines EXEC CICS DELETEQ eine indirekte

Warteschlange angegeben ist, dann werden die Zugriffsberechtigungen anhand der Sicherheitseinstellung der letzten Zielwarteschlange ermittelt.

#### **INTRA**

Eine Warteschlange für Daten, die temporär gespeichert werden sollen.

Eine partitionsinterne Zielwarteschlange kann ein Terminal, eine Datei oder ein anderes System sein. Eine einzelne Datei, die von VSAM verwaltet wird, wird zur Speicherung der Daten für alle partitionsinternen Warteschlangen verwendet.

Sie können eine Transaktion angeben, die zur Verarbeitung der Datensätze verwendet werden soll, und außerdem eine Auslöseebene für jede partitionsinterne Warteschlange. Die Auslöseebene steht für die Anzahl der Datensätze, die auflaufen dürfen, bevor die angegebene Transaktion eingeleitet wird. In der Beschreibung zum Attribut TRIGGERLEVEL finden Sie weitere Informationen zu den Auslöseebenen.

Die partitionsinterne Warteschlange kann als logisch wiederherstellbare, physisch wiederherstellbare oder nicht wiederherstellbare Warteschlange definiert werden.

Eine logisch wiederherstellbare Warteschlange wird (nach dem Fehlschlagen einer einzelnen Transaktion oder einem vollständigen Systemausfall) in dem Status wiederhergestellt werden, in dem sie sich am Ende der letzten abgeschlossenen Arbeitseinheit (UOW) befunden hat. (Eine Arbeitseinheit beginnt am Taskanfang oder an einem Synchronisationspunkt und endet am Taskende oder ebenfalls an einem Synchronisationspunkt).

Physisch wiederherstellbare Warteschlangen werden (nach einem vollständigen Systemausfall) in dem Status wiederhergestellt, in dem sie sich beim Auftreten des Systemausfalls befanden.

Die Angabe TYPE=REMOTE darf in der TDQUEUE-Ressourcendefinition nicht festgelegt werden. Wenn Sie eine ferne Warteschlange mit transienten Daten definieren wollen, dann lassen Sie das Attribut TYPE leer und geben Sie Werte für die fernen Attribute REMOTELength, REMOTENAME und REMOTESYSTEM an. Alternativ hierzu können Sie die fernen Attribute als Teil der Ressourcendefinitionen für die anderen Typen von Warteschlangen mit transienten Daten einbinden.

#### **TYPEFILE({INPUT|OUTPUT|RDBACK})**

Gibt den Typ der Datei an, der die Warteschlange zugeordnet werden soll.

##### **INPUT**

Eine Eingabedatei.

##### **OUTPUT**

Eine Ausgabedatei.

##### **RDBACK**

Eine Eingabedatei, die rückwärts gelesen werden soll.

**Anmerkung:** Diese Vorgehensweise ist nur für Dateien möglich, die auf Magnetbändern definiert wurden.

Eine partitionsübergreifende Warteschlange kann eine Ein- oder Ausgabewarteschlange, nicht jedoch beides sein.

Weitere Informationen zu den DCB-Makrofeldern finden Sie in z/OS DFSMS Macro Instructions for Data Sets.

**USERID(*benutzer-id*) (nur partitionsinterne Warteschlangen)**

Gibt die Benutzer-ID an, die von CICS für die Sicherheitsprüfung verwendet werden soll, wenn die Transaktion für die Auslöseebene überprüft wird, die im Feld TRANSID angegeben ist. Die Benutzer-ID kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Der in das Feld USERID eingegebene Wert ist nur gültig, wenn auch ATIFACILITY(FILE) angegeben wird.

Wenn die Sicherheitsfunktion aktiviert wurde, dann wird die Transaktion für die Auslöseebene unter der angegebenen Benutzer-ID ausgeführt. Diese Benutzer-ID muss für alle Ressourcen berechtigt sein, die von der Transaktion für die Auslöseebene verwendet werden.

Wenn Sie die Benutzer-ID aus einer Definition für Warteschlangen mit transienten Daten nicht angeben, dann verwendet CICS die CICS-Standard-Benutzer-ID, die im Systeminitialisierungsparameter DFLTUSER angegeben ist. Die Sicherheitsprüfung findet statt, wenn Sie eine partitionsinterne Definition installieren, die eine Benutzer-ID umfasst. Wenn die Sicherheitsprüfung fehlschlägt, wird die Ressourcendefinition für diese partitionsinterne Warteschlange nicht installiert.

Weitere Informationen zur Ersatzbenutzersicherheit finden Sie in Transient data trigger-level transactions.

**WAIT({YES|NO})) (nur partitionsinterne Warteschlangen)**

Gibt an, ob eine unbestätigte Arbeitseinheit (UOW), die zur Änderung einer logisch wiederherstellbaren Warteschlange verwendet wurde, auf die Resynchronisation mit dem zugehörigen Koordinator warten soll, um zu ermitteln, ob die Änderungen festgeschrieben oder zurückgesetzt werden sollen.

**NO** Die Arbeitseinheit soll nicht warten. Alle Änderungen, die an wiederherstellbaren Ressourcen durchgeführt wurden, sollen wie im Attribut ACTION der TRANSACTION-Ressourcendefinition angegeben zurückgesetzt oder festgeschrieben werden.

**YES** Die Arbeitseinheit soll warten. Alle während des Wartezeitraums erforderlichen Aktionen werden mithilfe des Werts im Attribut WAITACTION ermittelt.

Dieses Attribut überschreibt die Angabe im Attribut WAIT, das in der Transaktionsdefinition der Arbeitseinheit definiert wurde. Eine Erläuterung zu den Interaktionen, die für unbestätigte Attribute in der TDQUEUE- und der TRANSACTION-Definition vorhanden sein können, finden Sie in Tabelle 4 auf Seite 256.

**WAITACTION({REJECT|QUEUE})) (nur partitionsinterne Warteschlangen)**

Gibt die Aktion an, die von CICS für eine unbestätigte Arbeitseinheit (UOW) ausgeführt werden soll, wenn in der Definition für diese Warteschlange die Angabe WAIT(YES) angegeben ist. Folgende Aktionen sind möglich:

## QUEUE

Die Arbeitseinheit (UOW) ist unbestätigt und befindet sich im Wartestatus. Eventuell vorhandene Sperren, die von der UOW für diese Warteschlange gehalten werden, bleiben so lange aktiv, bis der abschließende Status der UOW bekannt ist. Dies bedeutet, dass Tasks ausgesetzt werden, anstatt die Antwort LOCKED zu empfangen. Wenn der Endstatus der UOW bekannt ist, werden alle vorgenommenen Änderungen festgeschrieben oder zurückgesetzt. Bis zu diesem Zeitpunkt müssen alle weiteren Anforderungen der folgenden Typen, die eine der aktiven Sperren benötigen, warten:

- READQ, wenn die unbestätigte Arbeitseinheit READQ- oder DELETEQ-Anforderungen abgesetzt hat.
- WRITEQ, wenn die unbestätigte Arbeitseinheit WRITEQ- oder DELETEQ-Anforderungen abgesetzt hat.
- DELETEQ, wenn die unbestätigte Arbeitseinheit READQ-, WRITEQ- oder DELETEQ-Anforderungen abgesetzt hat.

## REJECT

Die UOW ist unbestätigt und befindet sich im Wartestatus. Alle von der UOW für diese Warteschlange gehaltenen Sperren werden beibehalten, bis der Endstatus der UOW bekannt ist. Wenn der Endstatus bekannt ist, werden alle in der UOW durchgeführten Änderungen entweder festgeschrieben oder zurückgesetzt. Bis zu diesem Zeitpunkt werden weitere Anforderungen, die eine der gehaltenen Sperren benötigen, zurückgewiesen. Das System gibt in diesem Fall die Antwort LOCKED aus. Die Angabe WAITACTION=REJECT führt zur Ausgabe von LOCKED unter den gleichen Bedingungen, unter denen auch QUEUE zum Warten einer Transaktion führt.

## Erforderliche TDQUEUE-Definitionen

CICS-Services verwenden bestimmte Warteschlangen mit transienten Daten und die zugehörigen TDQUEUE-Definitionen müssen so bald wie möglich während eines Kaltstarts installiert werden.

Um sicherzustellen, dass die erforderlichen TDQUEUE-Definitionen möglichst bald installiert werden, müssen Sie den Systeminitialisierungsparameter **GRPLIST** auf eine der folgenden Arten angeben:

- Geben Sie DFHLIST als die erste Liste in **GRPLIST** an. Die DFHLIST-Liste enthält die Gruppe DFHDCTG, die die relevanten TDQUEUE-Definitionen enthält.
- Geben Sie an, dass die Gruppe DFHDCTG die erste Gruppe in der ersten Liste in **GRPLIST** ist.

Die Warteschlangen mit transienten Daten in der Gruppe DFHDCTG lauten wie folgt:

### **CADL (erforderlich zur Protokollierung der z/OS Communications Server-Ressourcendefinitionen)**

Für z/OS Communications Server-Ressourcen führt dieses Ziel ein Protokoll aller RDO-Definitionen, die auf dem aktiven CICS-System installiert sind. Das Protokoll zeichnet sowohl die Installation von Einträgen in der Terminalsteuer-tabelle (TCT = Terminal Control Table) als auch die Löschung automatisch installierter Einträge aus der TCT auf. Es zeichnet die installierten Definitionen wie folgt auf:

- Nach automatischer Installation
- Mit CEDA INSTALL

- Bei der Systeminitialisierung

Detaillierte Informationen zum Definieren von Dateien für Warteschlangen mit transienten Daten finden Sie in Defining data sets.

**CADO (erforderlich für CICS Development Deployment Tool-Nachrichten)**

CADO ist eine Warteschlange mit transienten Daten, in der Nachrichten protokolliert werden, die von CICS Development Deployment Tool stammen.

**CADS (erforderlich für Nachrichten, die von Programmierschnittstellenbefehlen des Prüfsystems generiert werden)**

CADS ist die Warteschlange mit transienten Daten, die die Nachrichten protokolliert, die bei der Prüfung der Befehle der Systemprogrammierschnittstelle generiert werden.

**CAIL (erforderlich zur Protokollierung von Definitionen des Terminalmodells für die automatische Installation)**

AITM (Autoinstall Terminal Model Manager) verwendet dieses Ziel zum Protokollieren aller Einträge des Terminalmodells für die automatische Installation, die in der Terminalsteuertabelle installiert und aus dieser gelöscht werden.

**CCPI (erforderlich für CPI Communications-Nachrichten)**

Die einheitliche Anwendungsunterstützung für DFV (CPI Communications) schreibt Nachrichten in dieses Ziel.

**CCSE (erforderlich zur Unterstützung der Programmiersprache C)**

CICS leitet die C-Standarddatenströme in Warteschlangen mit transienten Daten um. (Warteschlangennamen sind in CICS festgelegt. Aus diesem Grund können die C-Standarddatenströme nicht in andere Warteschlangen umgeleitet werden.) C-Programme schreiben in die Warteschlange CCSE, indem sie einen Schreibvorgang in die Standardfehlerausgabe (stderr) durchführen. Sie können dieses Ziel als partitionsübergreifend, partitionsintern oder indirekt definieren. Wenn Sie keine TDQUEUE-Definition für die CCSE-Warteschlange angeben, schlägt in C-Programmen das Schreiben in die Standardfehlerausgabe fehl.

**CCSI (optional, zur Unterstützung der Programmiersprache C)**

Die Warteschlange CCSI ist für die Standardeingabe (stdin), dem C-Standarddatenstrom für Eingabedaten, reserviert. Obwohl der Name der Warteschlange CCSI für die Standardeingabe (stdin) reserviert ist, führt der Versuch, aus der Standardeingabe zu lesen, in CICS zur Ausgabe von EOF. Aus diesem Grund ist dieses Ziel optional. Sie können es als partitionsübergreifend, partitionsintern oder indirekt definieren.

**CCSO (erforderlich zur Unterstützung der Programmiersprache C)**

CCSO ist der Standardausgabe (stdout), dem C-Standarddatenstrom für Ausgabedaten, zugeordnet. Sie können dieses Ziel als partitionsübergreifend, partitionsintern oder indirekt definieren. Wenn Sie die Warteschlange CCSO nicht definieren, schlägt das Schreiben in die Standardausgabe in C-Programmen fehl.

**CCZM (erforderlich für CICS-Klassennachrichten)**

CCZM ist die Warteschlange mit transienten Daten, in der Nachrichten protokolliert werden, die von den CICS-Basisklassen stammen.

**CDBC (erforderlich für DBCTL-Nachrichten des Typs DFHDB81xx)**

CDBC wird als indirekte Warteschlange definiert, die auf die partitionsübergreifende Warteschlange CSML verweist. Nur DBCTL-Nachrichten des Typs DFHDB81xx verwenden dieses Datenprotokoll, wohingegen andere Nachrichten entweder das Terminal oder die Konsole verwenden.

**CDB2 (erforderlich bei Verwendung von Db2)**

CDB2 ist die Warteschlange mit transienten Daten für Nachrichten und Statistikdaten aus der CICS-Db2-Anschlussfunktion.

**CDEP (erforderlich für Nachrichten zur Einstellung der Unterstützung)**

CDEP ist die Warteschlange mit transienten Daten für Nachrichten zur Einstellung der Unterstützung.

**CDUL (erforderlich für Nachrichten zu Transaktionsspeicherausügen)**

CDUL ist das Ziel für Nachrichten zu Transaktionsspeicherausügen. Wenn ein Transaktionsspeicherauszug angefordert wird, z. B. nach einer abnormalen Beendigung einer Transaktion, wird eine Nachricht an dieses Ziel geschrieben, um anzuzeigen, dass ein Speicherauszug erstellt wurde oder um den Grund dafür anzugeben, aus dem der Speicherauszug unterdrückt wurde.

**CECO (erforderlich für Nachrichten zur Ereigniserfassung und -ausgabe)**

CECO ist die Warteschlange mit transienten Daten für Nachrichten der EC-Komponente (EC = Event Capture and Emission; Ereigniserfassung und Ausgabe).

**CEJL (erforderlich für Enterprise Java-Nachrichten)**

CEJL ist die Warteschlange mit transienten Daten für Nachrichten der Enterprise Java-Domäne.

**CEPO (erforderlich für Nachrichten der Ereignisverarbeitung)**

CEPO ist die Warteschlange mit transienten Daten für Nachrichten der EP-Domäne (EP = Event Processing; Ereignisverarbeitung).

**CESE (erforderlich für die Language Environment-Laufzeitausgabe)**

Für Language Environment wird die gesamte Laufzeitausgabe in diese Warteschlange mit transienten Daten geschrieben. Weitere Informationen hierzu finden Sie in z/OS Language Environment Programming Guide.

**CESO (erforderlich für die Language Environment-Laufzeitausgabe)**

Für Language Environment wird die gesamte Laufzeitausgabe in diese Warteschlange mit transienten Daten geschrieben. Weitere Informationen hierzu finden Sie in z/OS Language Environment Programming Guide.

**CIEO (erforderlich für IP-ECI-Nachrichten)**

CIEO ist die Warteschlange mit transienten Daten für IP-ECI-Nachrichten (IP = Internet Protocol; ECI = External Call Interface) aus der IE-Domäne (IE = IP ECI).

**CISL (erforderlich zum Protokollieren von IPCONN-Ressourcendefinitionen)**

Dieses Ziel stellt ein Protokoll der installierten IPCONN-Ressourcendefinitionen bereit. Während der frühen CICS-Initialisierung können, wenn CISL noch nicht verfügbar ist, Nachrichten für dieses Ziel an die Konsole weitergeleitet werden.

**CISO (erforderlich für IPIC-Nachrichten)**

CISO ist die Warteschlange mit transienten Daten für IPIC-Verbindungsnachrichten (IPIC = IP Interconnectivity; IP-Interkonnektivität) der systemübergreifenden Domäne (IS).

**CJRM (erforderlich für Nachrichten der JRas-Protokollierungs- und -Tracefunktion)**

CJRM ist die Warteschlange mit transienten Daten für Nachrichten der JRas-Protokollierungs- und -Tracefunktion. Hierbei handelt es sich um Nachrichten, die von einer Java-Klasse abgesetzt werden, die in einer CICS-JVM ausgeführt wird, und die keine CICS-Nachrichten sind.



**CKQQ (erforderlich für CICS-MQ Connection-Nachrichten)**

CKQQ ist die Warteschlange mit transienten Daten, die die CICS-MQ-Nachrichten protokolliert, die beim Start und bei der Beendigung des CICS-MQ-Adapters abgesetzt werden. Sie wird als partitionsinterne Warteschlange definiert.

**CMIG (erforderlich für das Migrationsprotokoll)**

CMIG ist ein **Migrationsprotokoll**, das Nachrichten mit Meldungen zur Verwendung der Funktionen empfängt, die in CICS nicht mehr unterstützt werden (z. B. Befehl EXEC CICS ADDRESS CSA). Sie können CMIG als partitionsinterne, partitionsübergreifende oder indirekte Zielwarteschlange definieren.

**CML0 (erforderlich für Nachrichten zur Markup-Sprache)**

CML0 ist die Warteschlange mit transienten Daten für Nachrichten der Domäne für die Markup-Sprache.

**CMPO (erforderlich für Nachrichten der verwalteten Plattform)**

CMPO ist die Warteschlange mit transienten Daten für Nachrichten der Domäne für die verwaltete Plattform.

**CMQM (erforderlich für CICS-MQ-Nachrichten)**

CMQM ist die Warteschlange, in der alle CICS-MQ-Nachrichten protokolliert werden, die vom CICS-MQ-Adapter, vom CICS-MQ-Auslösemonitor und der CICS-MQ-Brücke abgesetzt werden.

**CPIO (erforderlich für CICS-SOAP-Nachrichten)**

CPIO ist die Warteschlange mit transienten Daten, die die CICS-SOAP-Nachrichten protokolliert.

**CRDI (erforderlich zur Protokollierung von Programmressourcendefinitionen)**

Dieses Ziel stellt ein Protokoll der installierten Ressourcendefinitionen bereit, z. B. für Programme, Transaktionen, Zuordnungen und Maskengruppen.

**CRLO (erforderlich für Nachrichten zum Lebenszyklus von Ressourcen)**

CRLO ist die Warteschlange mit transienten Daten für Nachrichten des Managers für den Lebenszyklus von Ressourcen.

**CRPO (erforderlich für ONC RPC-Nachrichten (ONC RPC = Open Network Computing Remote Procedure Call))**

CRPO ist als partitionsübergreifende Warteschlange definiert, kann jedoch auch als partitionsinterne oder indirekte Warteschlange definiert werden.

**CSBA (erforderlich bei Verwendung von BAM (Business Application Manager))**

CSBA ist die Warteschlange mit transienten Daten für Nachrichten der BAM-Domäne.

**CSBR (erforderlich bei Verwendung der Brückeneinrichtung)**

CSBR ist die Warteschlange mit transienten Daten, die die Nachrichten der Brückeneinrichtung protokolliert.

**CSCC (erforderlich für Fehlernachrichten des CICS-Clients)**

CSCC ist die Warteschlange mit transienten Daten, die die Fehlernachrichten des CICS-Clients protokolliert.

**CSCS (erforderlich für Anmeldetransaktionen)**

CSCS empfängt eine Nachricht, die Details zu jedem Anmelde- und Abmeldevorgang enthält. Außerdem empfängt diese Warteschlange eine Nachricht zu jedem zurückgewiesenen Anmeldeversuch und zu jedem Fehler bei der Ressourcenberechtigung. Diese Zielwarteschlange kann einen beliebigen Typ aufweisen.

**CSDH (erforderlich bei Verwendung der Dokumenthandlerschablone)**

CSDH ist die Warteschlange mit transienten Daten, die Erläuterungsnachrichten eines Exitprogramms der Dokumenthandlerschablone empfängt.

**CSDL (erforderlich zur Protokollierung von RDO-Befehlen)**

Die RDO-Transaktionen (RDO = Resource Definition Online; Onlinere Ressourcen- definition) schreiben alle Befehle in diese Zielwarteschlange, die zu Änderungen an der CSD-Datei des CICS-Systems oder am aktiven CICS-System führen.

Sie benötigen CSDL nur, wenn Sie RDO verwenden und ein Protokoll der Befehle aufbewahren wollen.

Die maximale Länge der Datensätze, die in CSDL geschrieben werden können, beträgt 128 Byte. Wenn Sie CSDL als partitionsübergreifend definieren, dann müssen in der zugeordneten SDSCI-Anweisung oder Datendefinitionsanweisung Datensätze im V-Format mit einer Mindestblockgröße von 136 Byte angegeben werden.

**CSFL (erforderlich zur Protokollierung von Dateiressourcendefinitionen)**

CSFL ist ein Protokoll, in dem alle Dateiressourcendefinitionen aufgezeichnet werden, die auf dem aktiven CICS-System installiert sind. Löschungen von Einträgen für Dateiressourcen werden ebenfalls hier protokolliert.

**CSJE und CSJO (erforderlich für umgeleitete Ausgabedaten aus Java-Programmen, die in einer JVM ausgeführt werden)**

CSJE und CSJO empfangen Ausgabedaten aus JVMs, die von der von CICS bereitgestellten Beispielklasse 'com.ibm.cics.samples.SJMergedStream' abgefangen wurden. Diese Beispielklasse kann optional in der Option USEROUTPUT-CLASS eines JVM-Profiles angegeben werden. CSJE wird für die Standardfehlerausgabe, interne Nachrichten und die nicht zurücksetzbare Ereignisprotokollierung verwendet, CSJO für die Standardausgabe. Die Klasse 'com.ibm.cics.samples.SJMergedStream' dient zur Verarbeitung beider Ausgabedatentypen und überträgt diese an die korrekte Warteschlange mit transienten Daten. CSJE und CSJO werden als indirekte Warteschlangen definiert, die auf die CSSL-Warteschlange verweisen.

Die Länge der Nachrichten, die von der JVM abgesetzt werden, kann variieren. Die maximale Datensatzlänge für die Warteschlange CSSL (133 Byte) ist möglicherweise für den Empfang bestimmter Nachrichten nicht ausreichend. In diesem Fall gibt die Beispielumleitungsklasse für die Ausgabe eine Fehlernachricht aus, die möglicherweise auch den Text der Nachricht betrifft. Wenn Sie feststellen, dass Nachrichten von der JVM empfangen werden, deren Länge 133 Byte übersteigt, dann müssen Sie CSJO und CSJE als separate Warteschlangen mit transienten Daten neu definieren. Definieren Sie sie als partitionsübergreifende Zielwarteschlangen und erhöhen Sie die Datensatzlänge der Warteschlangen. Sie können die Warteschlange einer physischen Datei oder einer Systemausgabedatei zuordnen. In diesem Fall eignet sich eine Systemausgabedatei möglicherweise besser, weil Sie dann die Warteschlange nicht schließen müssen, um die Ausgabe anzuzeigen. Wenn Sie CSJO und CSJE neu definieren, müssen Sie sicherstellen, dass die Warteschlangen so bald wie möglich während eines Kaltstarts auf dieselbe Weise wie Warteschlangen mit transienten Daten installiert werden, die in der Gruppe DFHDCTG definiert werden.

**CSKL (erforderlich zur Protokollierung von Transaktions- und Profilressourcendefinitionen)**

CSKL ist ein Protokoll, in dem alle Transaktions- und Profilressourcendefinitionen aufgezeichnet werden, die auf dem aktiven CICS-System installiert sind. Hier werden auch Löschungen protokolliert.

**CSLB (erforderlich zur Protokollierung von LIBRARY-Ressourcendefinitionen)**

CSLB ist ein Protokoll, in dem alle Änderungen an dynamischen LIBRARY-Ressourcen aufgezeichnet werden, die auf dem aktiven CICS-System installiert sind. Aufgezeichnet werden alle Installations- und Löschvorgänge für LIBRARY-Ressourcen, alle Änderungen an der Konfiguration einer LIBRARY-Ressource (einschließlich aller Änderungen an der Aktivierung) sowie an der Rangfolge oder am kritischen Status einer LIBRARY-Ressource. Wenn keine dynamischen LIBRARY-Ressourcen installiert sind, dann findet keine Auditprotokollierung in CSLB statt.

**CSML (erforderlich für Abmeldetransaktionen)**

Bei der CICS-Abmeldung werden Daten in diese Zielwarteschlange geschrieben.

**CSMT (erforderlich für Terminalfehler- und Abbruchnachrichten)**

Das Programm für abnormale Bedingungen bei Terminals (DFHTACP) und das Programm für abnormale Bedingungen (DFHACP) schreiben Terminalfehlernachrichten und ABEND-Nachrichten in diese Zielwarteschlange. Sie können diese Zielwarteschlange als partitionsübergreifend, partitionsintern oder indirekt definieren.

**CSNE (erforderlich für Knotenfehlernachrichten)**

Das Programm für abnormale Bedingungen bei Knoten (DFHZNAC) und das Knotenfehlerprogramm (DFHZNEP) schreiben Terminalfehlernachrichten und die entsprechenden Daten in diese Zielwarteschlange. Sie können diese Zielwarteschlange als partitionsübergreifend, partitionsintern oder indirekt definieren.

**CSOO (erforderlich für Socketdomänennachrichten)**

CSOO ist die Warteschlange mit transienten Daten für Nachrichten der Socketdomäne.

**CSPL (erforderlich zur Protokollierung von Programmressourcendefinitionen)**

CSPL ist ein Protokoll, in dem alle Programmressourcendefinitionen aufgezeichnet werden, die auf dem aktiven CICS-System installiert sind. Hier werden auch Löschungen protokolliert.

**CSQL (erforderlich für Nachrichten von Warteschlangen mit transienten Daten)**

CSQL ist die Warteschlange mit transienten Daten, die Auditprotokollnachrichten vom Manager für Warteschlangen mit transienten Daten selbst empfängt.

**CSRL (erforderlich zur Protokollierung von Partnerressourcendefinitionen)**

CSRL ist ein Protokoll, in dem alle PARTNER-Ressourcen protokolliert werden, die auf dem aktiven CICS-System installiert sind. Hier werden auch Löschungen aufgezeichnet.

**CSSH (erforderlich für Scheduler-Services)**

CSSH ist die Warteschlange mit transienten Daten, die Nachrichten von den Scheduler-Services empfängt.

**CSSL (erforderlich für Statistikdaten des Wiederherstellungsdienstprogramms)**

Das Programm für das Wiederherstellungsdienstprogramm (DFHRUP) schreibt Statistikdaten in diese Zielwarteschlange. Diese Zielwarteschlange benötigt eine Mindestlänge für logische Datensätze von 132 Byte und eine Mindestblockgröße von 136 Byte.

**CS TL (erforderlich für Nachrichten zu Terminal-E/A-Fehlern)**

Das Programm für abnormale Bedingungen bei Terminals (DFHTACP) schreibt

Nachrichten zu Terminal-E/A-Fehlern in diese Zielwarteschlange. Sie können diese Zielwarteschlange als partitionsübergreifend, partitionsintern oder indirekt definieren.

**CSZL (erforderlich für die Front-End-Programmierschnittstelle)**

CSZL ist die Warteschlange mit transienten Daten, in der FEPI-Nachrichten protokolliert werden.

**CSZX (erforderlich für die Front-End-Programmierschnittstelle)**

CSZX ist die Warteschlange mit transienten Daten, in der nicht erwartete Ereignisse für FEPI protokolliert werden.

**CWBO (erforderlich für Nachrichten der CICS-Webunterstützung)**

CWBO ist die Warteschlange mit transienten Daten, die die Nachrichten der CICS-Webunterstützung protokolliert, die von der CICS-Webschnittstelle abgesetzt werden.

**CWBW (erforderlich für CICS-Warnnachrichten im Web)**

CWBW ist die Warteschlange mit transienten Daten, die die HTTP-Warnungsholder von Nachrichten protokolliert, die von der CICS-Webunterstützung empfangen werden.

## **Pseudowarteschlangen mit transienten Daten verwenden**

Sie können Ausgabedaten in eine Pseudowarteschlange mit transienten Daten übertragen, wenn die Daten nicht gespeichert werden müssen.

### **About this task**

Eine indirekte Warteschlange mit transienten Daten (TDQ = Transient Data Queue) kann während der Laufzeit nicht aktiviert oder inaktiviert werden. Wenn die Daten nicht benötigt werden, können sie an eine Pseudo-TDQ umgeleitet werden. Eine Pseudo-TDQ kann erstellt werden, wenn beim Definieren einer Warteschlange mit transienten Daten **DSNAME(DUMMY)** angegeben wird. Weitere Informationen hierzu finden Sie in „TDQUEUE-Attribute“ auf Seite 189. Sie können die Warteschlange dauerhaft oder vorübergehend umleiten.

Dies kann z. B. sinnvoll sein, wenn Sie die Prüfnachrichten, die an die CADS-Warteschlange weitergeleitet werden, nicht benötigen.

Um die Warteschlange dauerhaft umzuleiten, müssen Sie die Definition der CADS-Warteschlange in der Gruppe DFHDCTG so ändern, dass eine Umleitung in die Pseudowarteschlange ausgeführt wird. Die Änderung wird wirksam, wenn CICS neu gestartet wird.

Gehen Sie wie folgt vor, um die Umleitung ohne Neustart zu steuern:

1. Verschieben Sie die Definition der CADS-Warteschlange aus der Gruppe DFHDCTG in eine benutzerdefinierte Gruppe.
2. Ändern Sie die Definition der CADS-Warteschlange, um eine Umleitung in die Pseudowarteschlange durchzuführen.
3. Installieren Sie die angepassten Warteschlangen.
4. Wenn die Warteschlange benötigt wird, verwenden Sie CEDA, um CADS in der Gruppe DFHDCTG zu installieren.

---

## TERMINAL-Ressourcen

Eine TERMINAL-Ressource definiert die Merkmale einer Terminaleinheit, die mit CICS kommuniziert. Terminaleinheiten umfassen grafische Bildschirmseinheiten, Drucker, Betriebssystemkonsolen und spezielle Einheiten wie beispielsweise Faxgeräte (FAX).

Die eindeutigen und eventuell dynamischen Eigenschaften von Terminals werden in der TERMINAL-Definition der CSD-Datei definiert.

Allerdings verfügen zahlreiche Terminals über identische Eigenschaften und Sie müssen nicht jede separat und vollständig in CICS definieren. Es gibt zwei Möglichkeiten zum Reduzieren des Zeit- und Arbeitsaufwands für die Definition der einzelnen Terminals. Diese lauten wie folgt:

1. **TYPETERM-Definitionen** mit und ohne die Funktion QUERY. Jede TERMINAL-Definition muss auf eine TYPETERM-Definition verweisen, in der die Eigenschaften definiert sind, die allgemein (und häufig komplexer) und normalerweise statisch sind. Zusammen bilden die Informationen in TERMINAL- und TYPETERM-Definitionen einen Terminaleintrag in einer Terminalsteuertabelle (TCTTE = Terminal Control Table Terminal Entry).

Ein TYPETERM-Element kann für eine Reihe von Eigenschaften zahlreicher Terminals stehen. Einige dieser Eigenschaften können zum Zeitpunkt der Erstellung der TYPETERM-Definition noch nicht definiert sein. Diese Eigenschaften können bei der Anmeldung für jedes Terminal anhand des strukturierten Felds QUERY ermittelt werden.

Allerdings existieren weitere Eigenschaften, die viele Terminals gemeinsam haben. Dies kann soweit gehen, dass die TERMINAL-Definitionen dieser Terminals identisch sind. CICS stellt eine Funktion bereit, die bewirkt, dass nicht für jedes Terminal eine eigene Ressourcendefinition in der Terminalsteuertabelle installiert werden muss, wenn CICS aktiv ist.

2. **Automatische Installation** mit einer TERMINAL-Definition zur Darstellung zahlreicher Terminals. Sie können von CICS die Ressourcen dynamisch erstellen und installieren lassen, wenn das Terminal bei der Anmeldung benötigt wird. Hierzu verwendet CICS eine **TERMINAL-Modelldefinition** aus der CSD-Datei. Dieser Prozess wird als automatische Installation oder Autoinstall bezeichnet.

Die automatische Installation reduziert das Volumen an virtuellem Speicher, das für die Terminalsteuertabelle (TCT) erforderlich ist, wenn einige Ihrer Terminals nicht angemeldet sind, während CICS aktiv ist.

Wenn Sie für die Planung und Verwaltung der CICS-Datenübertragungsressourcen wie z. B. Terminals verantwortlich sind, dann sollten Sie die weiterführenden Informationen in Autoinstalling z/OS Communications Server terminals lesen.

Informationen zur Installation von TERMINAL-Ressourcendefinitionen mit CEMT und CEDA finden Sie in Installing TERMINAL resource definitions. Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with BAS terminal resource definitions.

## Terminals zum Drucken

Eine TERMINAL-Definition für eine Bildschirmseinheit kann TERMINAL-Definitionen für Drucker angeben. Dazu werden die Attribute PRINTER und ALTPRINTER verwendet. Ein derartiger Verweis wird nicht aufgelöst, wenn die TERMINAL-Definitionen installiert werden. Stattdessen wird der Verweis aufgelöst, wenn der Drucker von der Bildschirmseinheit benötigt wird.

Zur Erstellung und zum Senden der Druckausgabe an einen Drucker stehen mehrere Möglichkeiten zur Verfügung.

- BMS-Seitenerstellung
- Erstellen von Anzeigekopien mithilfe der folgenden Komponenten:
  - Hardwarekopierschlüssel
  - Lokaler Kopierschlüssel
  - Befehl ISSUE PRINT

Die TYPETERM- und TERMINAL-Definitionen werden für Drucker und für Bildschirmseinheiten verwendet. Einige der Attribute gelten hierbei nur für Drucker oder haben für Drucker eine spezielle Bedeutung. Des Weiteren existieren einige Attribute, die Sie für eine Bildschirmseinheit angeben müssen, die für die Erstellung von Anzeigekopien verwendet werden soll.

## Drucker

Geben Sie für jeden Drucker eine TERMINAL-Definition an.

Geben Sie für AUTINSTMODEL die Einstellung NO an, es sei denn, Sie verwenden für Drucker die automatische Installation. (Weitere Informationen hierzu finden Sie in Autoinstall and output-only devices.)

### ALTPAGE

Für BMS (Basic Mapping Support) gibt das Attribut PAGESIZE die standardmäßige Seitengröße und außerdem die Größe des Druckpuffers an. Sie geben die Anzahl der Zeilen auf der Seite (Länge) und die Anzahl der Zeichen pro Zeile (Breite) an.

Das Attribut ALTPAGE gibt außerdem die Seitengröße an, die verwendet werden soll, wenn die alternative Anzeigegröße (ALTSCREEN) ausgewählt wird. Die von Ihnen in ALTPAGE angegebene Breite muss mit der Breite übereinstimmen, die im Attribut ALTSCREEN angegeben ist. Die Länge, die in ALTPAGE und ALTSCREEN angegeben wird, kann jedoch unterschiedlich sein. Dies kann sinnvoll sein, wenn Sie die gleiche BMS-Maske zum Anzeigen und zum Drucken verwenden. Sie können z. B. für die Anzeige eine Zeile mehr angeben als für die Seite, um so die letzte Zeile der Anzeige für Fehlermeldungen zu reservieren.

Die Attribute ALTPAGE, DEFSCREEN und ALTSCREEN gelten für Drucker normalerweise nicht.

### AUTOPAGE

AUTOPAGE muss für Drucker auf YES gesetzt werden. Dazu ist jedoch keine Aktion erforderlich, da RDO diese Angabe für alle DEVICE-Typen der Drucker eingibt. Als Autopaging wird die Funktion bezeichnet, mit der mehrere Seiten umfassende BMS-Nachrichten fortlaufend ausgegeben werden, ohne dass hierzu ein Bedienereingriff erforderlich ist. Diese Einstellung wird normalerweise für Drucker benötigt. (Gegensatz hierzu ist die Anforderung für Nachrichten mit mehreren Seiten, die auf einer 3270-Bildschirmseinheit angezeigt werden, wenn der Bediener das Lesen einer Seite abschließen möchte, bevor die nächste Seite angefordert wird.)

Nur BMS SEND-Befehle mit der Option PAGING verwenden das Autopaging. BMS SEND-Befehle mit TERMINAL oder SET verwenden das Autopaging hingegen nicht.

Sie benötigen für die verschiedenen verwendeten Druckertypen jeweils mindestens eine TYPETERM-Definition. Möglicherweise sind mehr Definitionen erforderlich, wenn Sie möchten, dass Drucker nur für bestimmte Funktionen, für andere jedoch nicht verwendet werden können.

## DEVICE

Die TERMINAL-Definition für jeden Drucker muss auf ein TYPETERM-Element mit einem geeigneten DEVICE-Typ verweisen. Das Attribut DEVICE und (in einem Fall) das Attribut SESSIONTYPE legen fest, ob mit TYPETERM Drucker oder Bildschirmeinheiten definiert werden. Für Drucker können die folgenden Werte angegeben werden:

DEVICE	SESSIONTYPE	Drucker
3270P	-	Alle Drucker, die den 3270-Datenstrom unterstützen (ohne SNA-Verbindung).
LUTYPE3	-	Alle Drucker, die den 3270-Datenstrom unterstützen (mit SNA-Verbindung).
SCSPRINT	-	Alle Drucker, die den SNA-Zeichensatz unterstützen (mit SNA-Verbindung).
3790	SCSPRINT	IBM 3793-Tastatordrucker, die den SNA-Zeichensatz unterstützen (mit SNA-Verbindung).

## FORMFEED

Definieren Sie für die BMS-Seitenerstellung für FORMFEED die Einstellung YES.

## PAGESIZE

Gibt die Standardseitengröße für dieses Terminal an. Das Produkt aus Zeilen und Spalten darf den Wert von 32767 nicht überschreiten. Hierbei entspricht Zeilen der Anzahl von Zeilen auf der Seite und Spalten der Anzahl der Zeichen pro Zeile.

Wenn PAGESIZE nicht angegeben wird, dann werden die folgenden Standardwerte benutzt:

TW33, TW35	(12,80)
3270-Bildschirmeinheit - Modell 1	(12,40)
3270-Bildschirmeinheit - Modell 2	(24,80)
3270-Drucker	(12,80)

## TERMINAL

Der Name des Druckers gibt den TERMINAL-Namen in der Ressourcendefinition für diesen Drucker an.

**Anmerkung:** Das Attribut PRINTER wird in der Definition einer Bildschirmeinheit verwendet, um auf eine Druckereinheit zu verweisen, die für die Ausgabe über den Bildschirm verwendet wird. Geben Sie PRINTER nicht in der Druckerdefinition an. Weitere Informationen hierzu finden Sie in „Bildschirmeinheiten einen bestimmten Drucker zuordnen“.

## Bildschirmeinheiten einen bestimmten Drucker zuordnen

Wenn Sie den Inhalt einer Anzeige direkt auf einen Drucker kopieren möchten, ordnen Sie der Bildschirmeinheit in der TERMINAL-Definition einen bestimmten Drucker zu. Hierzu können Sie die Attribute PRINTER und ALTPRINTER verwenden.

## ALTPRINTCOPY

Geben Sie für ALTPRINTCOPY die Option YES an, wenn Sie die Hardwarefunktion COPY auf dem alternativen Drucker verwenden wollen.

## ALTPRINTER

Der alternative Drucker wird verwendet, wenn der primäre Drucker nicht ver-

füßbar ist. ALTPRINTER und ALTPRINTCOPY können mit dem Steuerprogramm für die automatische Installation dynamisch festgelegt werden, wenn die Bildschirmdefinition automatisch installiert wird.

#### **PRINTER**

Der primäre Drucker soll verwendet werden. PRINTER und PRINTERCOPY können mit dem Steuerprogramm für die automatische Installation dynamisch festgelegt werden, wenn die Bildschirmdefinition automatisch installiert wird.

#### **PRINTERCOPY**

Geben Sie für PRINTERCOPY die Option YES an, wenn Sie die Hardwarefunktion COPY auf dem primären Drucker verwenden wollen.

## **Pipeline-Terminal-Definitionen**

Wenn Sie eine logische 3600-Pipelineeinheit definieren, dann generieren Sie einen Terminaleintrag der Terminalsteuertabelle (TCTTE = Terminal Control Table Terminal Entry), der einem Pool von TCTTEs zugeordnet ist.

Wenn Nachrichten der logischen 3600-Pipelineeinheit in CICS empfangen werden, dann wird eine Task zur Verarbeitung dieser Nachricht zugeordnet. Dazu wird als Ankerblock einer der TCTTEs aus dem Pool verwendet. Auf diese Weise können Nachrichten, die aufeinanderfolgend über die logische Pipelineeinheit gesendet werden, gleichzeitig verarbeitet werden. Die Anzahl der gleichzeitigen Transaktionen wird anhand der TCTTEs im Pool begrenzt. Die Anzahl der TCTTEs im Pool sollte der Obergrenze der Anfrageaktivität entsprechen. Auf diese Weise ermöglicht die Pipelinefunktion die Definition einer geringeren Anzahl von TCTTEs in CICS als die Gesamtzahl der Pipelineanfrageterminals.

Alle TERMINAL-Definitionen in einem angegebenen POOL müssen sich in der CSD-Datei in derselben Gruppe befinden. Für mindestens eine der Definitionen in der Gruppe muss TASKLIMIT angegeben werden. Wenn dieses Attribut für mehrere Definitionen angegeben wird, dann gibt der verwendete Wert den Maximalwert für TASKLIMIT innerhalb der Definitionen in der Gruppe an.

TERMINAL- und TYPETERM-Definitionen werden wie bei normalen Terminals aufgelöst.

Wenn Sie PIPELINE-Terminals nicht auf Basis der GROUP-Angabe installieren, dann kann es zu unvorhersehbaren Ergebnissen kommen.

**Anmerkung:** CICS bietet keine Unterstützung für die automatische Transaktionsinitialisierung (ATI = Automatic Transaction Initiation) auf Pipeline-Terminals.

### **Pipeline-Terminals definieren**

Wenn Sie ein Pipeline-Terminal definieren, dann müssen Sie spezielle Attribute angeben.

Definieren Sie Pipeline-Terminals mit den folgenden Attributen:

#### **NETNAME**

Die z/OS Communications Server-Sitzung, die verwendet wird.

#### **POOL**

Alle TERMINAL-Definitionen, die denselben POOL-Namen haben, gehören zu demselben Pipeline-Pool. Das Vorhandensein eines Werts für das Attribut POOL unterscheidet diese von gewöhnlichen TERMINAL-Definitionen.



### SESSIONTYPE

Verwenden Sie dieses Attribut in der TYPETERM-Definition, um TYPETERM als Attribut zu identifizieren, das Pipeline-Terminals darstellt. Geben Sie PIPELINE als Wert an.

### TASKLIMIT

Geben Sie die maximal zulässige Anzahl gleichzeitig ablaufender Tasks, die für den Terminal-Pool aktiv sein können, in mindestens einer der TERMINAL-Definitionen an.

Normalerweise eignet sich ein TYPETERM-Wert für alle Definitionen. Dabei kann sich das TYPETERM-Element in einer anderen Gruppe befinden.

Pipelinetransaktionen werden einer PROFILE-Definition zugeordnet, in der das Attribut ONEWTE angegeben ist. Für ein Programm, das diesen Transaktionen zugeordnet ist, ist nur eine Schreiboperation oder EXEC CICS SEND-Operation zulässig. Andernfalls wird die Verarbeitung mit dem Abbruchcode ATCC beendet. Dies bedeutet, dass CICS-Tasks innerhalb des Sitzungspools schnell auftreten und verschwinden.

In „TERMINAL-Attribute“ auf Seite 222 finden Sie ein Beispiel für Definitionen für einen Pool mit Pipeline-Terminals in der Beschreibung des Attributs POOL.

### Anmerkung:

1. Wenn Sie ein Pipeline-Terminal installieren, in dem eine Pipeline angegeben ist, die in CICS bereits vorhanden ist, dann werden sowohl die alte Pipeline als auch alle zugehörigen Terminals gelöscht, bevor die neuen Definitionen installiert werden.
2. Wenn Sie das Terminal löschen, das als Eigner der vorhandenen Pipeline definiert ist, dann werden die vorhandene Pipeline und alle zugehörigen Terminals gelöscht.
3. Wenn Sie ein Terminal löschen, das nicht der Pipelineeigner ist, oder eine andere Pipeline oder ein Nicht-Pipeline-Terminal angeben, bleibt die restliche Pipelinedefinition unverändert. (Das Eignerterminal einer Pipeline ist das Terminal, dessen Namen in der alphanumerischen Reihenfolge der erste ist, der der Pipeline in der Gruppe zugeordnet ist, über die die Pipeline installiert wurde.)
4. Sie können ein Terminal in einer vorhandenen Pipeline nicht in ein Nicht-Pipeline-Terminal ändern oder eine neue Pipeline angeben, wenn die alte Pipeline ebenfalls in derselben Gruppe ebenfalls neu installiert wird. Hierzu müssen Sie die Installation in zwei Phasen aufteilen. Wenn Sie die Gruppe zweimal installieren, dann müssen Sie die relevanten Terminals in der Zwischenzeit inaktivieren.

## Einheiten mit LDC-Listen

Für logische 3600- und 3770-Stapelverarbeitungseinheiten, logische 3770- und 3790-Stapeldatenaustauscheinheiten sowie für logische LUTYPE 4-Einheiten können Sie den Namen einer LDC-Liste (LDC = Logical Device Code; Code für logische Einheiten) angeben.

In der Liste ist angegeben, welche LDCs für diese logische Einheit gültig sind und (optional) welche Einheitenmerkmale für die einzelnen LDCs gültig sind. Der erste LDC in dieser Liste wird als standardmäßiger LDC verwendet, wenn CICS für eine logische Einheit einen standardmäßigen Code für logische Einheiten auswählen muss.

Alle TERMINAL-Definitionen für Einheiten, die auf eine spezielle LDC-Liste verweisen, müssen die gleiche Angabe für TYPETERM enthalten, weil das Attribut LDCLIST sich in der TYPETERM-Definition befindet.

Die LDC-Liste selbst muss mit dem Makro DFHTCT TYPE=LDCLIST definiert werden. Dabei kann es sich um eine lokale LDC-Liste oder eine erweiterte lokale LDC-Liste handeln. In „TYPETERM-Attribute“ auf Seite 279 finden Sie ein Beispiel für die Codierung dieser beiden Komponenten und weitere Anleitungen in „DFHTCT-Codes für logische Einheiten: z/OS Communications Server (nicht 3270)“ auf Seite 456. Sie verwenden die Bezeichnung, die in der Makroinstruktion angegeben ist, um die LDC-Liste in der TYPETERM-Definition zu identifizieren. Die Angabe erfolgt im Attribut LDCLIST.

## APPC-Einzelsitzungsterminal (LUTYPE6.2)

Ein APPC-Einzelsitzungsterminal (LUTYPE6.2) kann als TERMINAL und mit einem Verweis auf ein TYPETERM-Element mit DEVICE(APPC) definiert werden. Wenn diese Definitionen installiert werden, dann sind die Ressourcen, die durch diese Definitionen definiert werden, für CICS als Verbindung und Modusset bekannt, wie dies auch bei der Definition in RDO (Resource Definition Online; Onlineressourcendefinition) als CONNECTION und SESSIONS der Fall ist. Der Name der Verbindung ist der TERMINAL-Name und der Name des Modussets entspricht der Angabe für MODENAME in der TERMINAL-Definition.

Bei einem APPC-Terminal kann es sich um eine PS/2-, eine Application System/400- (AS/400), eine System/38-Einheit oder eine ähnliche Einheit handeln. Sie können ein APPC-Terminal entweder mit einer TERMINAL-TYPETERM-Definition oder mit einer CONNECTION-SESSIONS-Definition definieren. Beide Definitionsarten können automatisch installiert werden (siehe hierzu die Informationen zur automatischen Installation in Autoinstalling z/OS Communications Server terminals und Autoinstalling APPC connections). Wenn Sie die TERMINAL-TYPETERM-Methode verwenden wollen, dann sind die folgenden Attribute von Bedeutung:

### DEVICE

Die TERMINAL-Definition verweist auf ein TYPETERM-Element mit APPC (Advanced Program-to-Program Communications), das in DEVICE angegeben wird. Eine solche TYPETERM-Definition kann für zahlreiche Terminals verwendet werden.

### MODENAME

Dies ist der Name, den CICS verwendet, um die Sitzung bei der Installation der Definition auf dem aktiven System zu identifizieren.

## Terminals für Transaktionsrouting

Das Transaktionsrouting ermöglicht Terminals auf einem CICS-System das Starten von Transaktionen auf einem anderen CICS-System.

In diesen Abschnitten wird das Wort 'System' verwendet, um ein CICS-System anzugeben, das mit anderen Systemen kommuniziert, die IPIC, MRO oder ISC verwenden. Sie können das Transaktionsrouting zwischen Systemen verwenden, die über IPIC, MRO oder über eine APPC-Verbindung verbunden sind. Weitere Informationen hierzu finden Sie in CICS transaction routing.

Sie können ein Terminal zur Verwendung beim Transaktionsrouting mit einem der folgenden Typen definieren:

- Lokales Terminal

- Fernes Terminal
- Terminal mit Doppelfunktion

Der Terminaltyp, der von Ihnen definiert werden muss, hängt davon ab, wie die Definition für das AOR-CICS-System (AOR = Application-Owning Region; Anwendungsverwaltungsregion) zur Verfügung gestellt wird. Sie können die Definition mit einer der folgenden Methoden bereitstellen:

- Duplizieren von Terminaldefinitionen
- Gemeinsame Nutzung von Terminaldefinitionen
- Versenden von Terminaldefinitionen

### Beim Transaktionsrouting verwendete TERMINAL-Definitionen

Sie können Ihre Terminals zur Verwendung beim Transaktionsrouting auf zwei Arten definieren. Sie können eine *lokale Terminaldefinition* in der Terminalverwaltungsregion (TOR = Terminal-Owning Region) und eine *ferne Terminaldefinition* in der Anwendungsverwaltungsregion (AOR = Application-Owning Region) installieren. Alternativ dazu können Sie in beiden Regionen eine *Terminaldefinition mit Doppelfunktion* installieren.

#### Lokale Terminaldefinitionen:

Eine *lokale Terminaldefinition* ist eine vollständige Definition des Terminals, die auf dem Terminalverwaltungssystem installiert wird.

Eine lokale Terminaldefinition wird bei der **Duplizierungsmethode** und bei der **Versandmethode** verwendet.

- In der TERMINAL- oder der CONNECTION-Definition werden die Attribute REMOTESYSTEM, REMOTESYSNET oder REMOTENAME nicht benötigt.  
Wenn das Attribut REMOTESYSTEM, das in der TERMINAL-Definition angegeben ist, mit dem Namen des (im Systeminitialisierungsoperanden SYSIDNT angegebenen) lokalen Systems übereinstimmt, dann wird die Definition als lokales Terminal behandelt, wenn es installiert wird, und nicht als fernes Terminal.
- Für SHIPPABLE in TYPETERM wird für das Duplizieren von Definitionen NO angegeben.
- Für SHIPPABLE in TYPETERM wird für das Versenden von Definitionen YES angegeben.
- Die Angabe von SHIPPABLE in TYPETERM ist für APPC-Einheiten zwingend erforderlich.
- Alle anderen erforderlichen Attribute in den TERMINAL- und TYPETERM-Definitionen müssen angegeben werden.

#### Ferne Terminaldefinitionen:

Wenn ein Terminal, das (lokal und vollständig definiert) zu einem System gehört, eine Transaktion aufruft, die zu einem anderen System gehört, dann ist es in der Anwendungsverwaltungsregion (AOR = Application-Owning Region) als *fernes Terminal* definiert.

Das Anwendungsverwaltungssystem muss über Zugriff auf mindestens eine Teildefinition des fernen Terminals verfügen. Diese Teildefinition wird auch als *ferne Definition* bezeichnet. Dies ist eine Teildefinition des Terminals, die in der Anwendungsverwaltungsregion und in den Transitregionen installiert wird. Sie enthält die mindestens erforderlichen Informationen, die das Terminal für den Zugriff auf eine

Transaktion auf diesem System benötigt. Sie erstellen ferne Definitionen nur dann, wenn Sie die **Duplizierungsmethode** verwenden.

- Der REMOTESYSTEM-Name muss mit dem Namen der CONNECTION-Definition für die nächste Region im Pfad für das Transaktionsrouting zur Terminalverwaltungsregion identisch sein.
- REMOTESYSNET muss der Netzname (generische Anwendungs-ID) der Terminalverwaltungsregion sein. Wenn in REMOTESYSTEM eine direkte Verbindung (CONNECTION) zur Terminalverwaltungsregion angegeben ist, dann ist die Angabe von REMOTESYSNET nur dann erforderlich, wenn die Terminalverwaltungsregion Mitglied einer generischen z/OS Communications Server-Ressourcengruppe ist und die direkte Verbindung eine APPC-Verbindung ist.
- Der in REMOTENAME angegebene Wert ist der Name, unter dem das Terminal oder die APPC-Einheit in der Terminalverwaltungsregion definiert ist. Er kann mit dem TERMINAL- oder CONNECTION-Namen übereinstimmen oder sich von diesem unterscheiden. Dabei muss es sich um den Namen handeln, unter dem das Terminal auf dem Anwendungsverwaltungssystem definiert ist. Für REMOTENAME wird standardmäßig der TERMINAL-Name verwendet, wenn keine Angabe gemacht wird.
- Für SHIPPABLE in TYPETERM wird für Nicht-APPC-Einheiten NO angegeben.
- Die Angabe von SHIPPABLE in TYPETERM ist für APPC-Einheiten zwingend erforderlich.
- Einige der Attribute in den TERMINAL- und TYPETERM-Definitionen können weggelassen werden. Weitere Anleitungen zu diesen Definitionen finden Sie in *Defining intercommunication resources in Configuring*.

Die folgenden Terminals und logischen Einheiten können das Transaktionsrouting nicht verwenden und können daher nicht als fern definiert werden:

- Logische 3600- oder 3650-Pipelineeinheiten im Pool
- Terminals vom Typ IBM 2260
- MVS-Konsole

### **Terminaldefinitionen mit Doppelfunktion:**

Eine *Terminaldefinition mit Doppelfunktion* ist eine vollständige Definition des Terminals, die auf dem Terminalverwaltungssystem und in den Anwendungsverwaltungsregionen sowie den Transitregionen installiert wird.

Sie wird nur bei der Methode mit **gemeinsamer Nutzung** verwendet.

- Der REMOTESYSTEM-Name muss der Angabe für SYSIDNT der Terminalverwaltungsregion (TOR = Terminal-Owning Region) entsprechen. Dieser Wert muss außerdem als Name für die Verbindung (CONNECTION) verwendet werden, den Sie von der Anwendungsverwaltungsregion (AOR = Application-Owning Region) zur Transitregion (sofern vorhanden) und von der Transitregion zur Terminalverwaltungsregion definieren. Allen CONNECTION-Definitionen im Pfad von der Anwendungsverwaltungsregion zur Terminalverwaltungsregion muss dieser Name zugeordnet werden.
- REMOTESYSNET muss der Netzname (generische Anwendungs-ID) der Terminalverwaltungsregion sein.
- Der Wert für REMOTENAME ist identisch mit dem TERMINAL- oder CONNECTION-Namen.
- Für SHIPPABLE in TYPETERM wird für Nicht-APPC-Einheiten NO angegeben.
- Die Angabe von SHIPPABLE in TYPETERM ist für APPC-Einheiten zwingend erforderlich.

- Alle anderen erforderlichen Attribute in den TERMINAL- und TYPETERM-Definitionen müssen angegeben werden.

## Teildefinitionen für das Transaktionsrouting zur Verfügung stellen

Sie können eine Teildefinition eines Terminals zur CICS-Anwendungsverwaltungsregion auf unterschiedliche Weise zur Verfügung stellen.

### Terminaldefinitionen versenden:

Sie können in der CSD-Datei, die für die Terminalverwaltungsregion (TOR = Terminal-Ownning Region) verfügbar ist, eine lokale Terminaldefinition definieren, die bei Bedarf an andere Regionen versendet werden kann.

1. Erstellen Sie eine lokale Definition für das Terminal in der CSD-Datei der Terminalverwaltungsregion. (Hierbei kann es sich um eine gemeinsam genutzte CSD-Datei handeln.)
2. Installieren Sie die lokale Definition in der Terminalverwaltungsregion. (Diese Definition kann bei der Systeminitialisierung, mit dem Befehl CEDAS INSTALL oder mithilfe der automatischen Installation installiert werden.)
3. Wenn das Terminal eine Transaktion aufruft, die zu einer anderen Region gehört, dann werden die Informationen, die zum Erstellen einer fernen Definition erforderlich sind, an die betreffende Region gesendet und eine temporäre Definition wird dort automatisch installiert.

Wenn die lokale Definition automatisch installiert wurde, dann ist die versendete Definition so lange gültig, bis das Terminal abgemeldet wird. Andernfalls ist die versendete Definition gültig, bis die lokale Definition erneut installiert wird oder bis die Verbindung zwischen den Regionen unterbrochen wird.

### Terminaldefinitionen duplizieren:

Sie können eine separate Terminaldefinition für jede verwendete Region erstellen. Diese Terminaldefinitionen werden in einer CSD-Datei gespeichert, auf die von der betreffenden Region zugegriffen werden kann.

Eine Terminaldefinition wird dabei als lokale Definition und eine oder mehrere als ferne Definition erstellt. Dieser Vorgang wird als das **Duplizieren von Terminaldefinitionen** bezeichnet, weil mehrere Ressourcendefinitionen für dasselbe Terminal vorhanden sind. (Die Definitionen stellen nicht zwingend exakte Duplikate dar.)

Gehen Sie wie folgt vor, um Terminaldefinitionen zu duplizieren:

1. Erstellen Sie eine lokale Definition für das Terminal in der CSD-Datei der Terminalverwaltungsregion oder in einer gemeinsam genutzten CSD-Datei.
2. Erstellen Sie eine ferne Definition für das Terminal in der CSD-Datei der Anwendungsverwaltungsregion oder in einer gemeinsam genutzten CSD-Datei. Wenn mehrere Anwendungsverwaltungsregionen vorhanden sind, dann benötigen Sie möglicherweise mehrere ferne Definitionen. Wenn die Regionen jedoch gemeinsam eine CSD-Datei nutzen, dann können Sie eventuell dieselbe ferne Definition für alle verwenden.
3. Installieren Sie die lokale Definition in der Terminalverwaltungsregion. Diese Definition kann automatisch installiert werden.
4. Installieren Sie die ferne Definition in den Anwendungsverwaltungsregionen.

**Anmerkung:** Wenn Ihre Regionen eine CSD-Datei gemeinsam nutzen, dann vergewissern Sie sich, dass die Definitionen sich in unterschiedlichen Gruppen befinden. Dies hat die folgenden Gründe:

- Sie wollen sie in unterschiedlichen Regionen installieren.
- Die TERMINAL-Namen sind voraussichtlich identisch, sodass sich die Definitionen nicht in derselben Gruppe befinden dürfen.

*Vorteile:*

Das Duplizieren von Terminaldefinitionen bietet bestimmte Vorteile.

- Sie können diese Methode unabhängig davon verwenden, ob Ihre Systeme eine CSD-Datei gemeinsam nutzen. Sie können sie also für das Transaktionsrouting zwischen unterschiedlichen MVS-Images verwenden.
- Hierbei handelt es sich um die **einzige** Methode, die Sie anwenden können, wenn Ihre Terminals auf verschiedenen Systemen unter unterschiedlichen Namen angegeben sind.
- Sie können die automatische Transaktionsinitialisierung (ATI = Automatic Transaction Initiation) auf einem Anwendungsverwaltungssystem für ein fernes Terminal verwenden, ohne dass die Exits XALTENF und XICTENF definiert werden müssen. Dies ist insbesondere nützlich für Drucker, weil sie **vor** der Ausführung von ATI angefordert werden müssen, damit die automatische Transaktionsinitialisierung erfolgreich verläuft.

*Nachteile:*

Das Duplizieren von Terminaldefinitionen bietet allerdings auch bestimmte Nachteile.

- Die CSD-Datei verwendet mindestens das Doppelte des erforderlichen Platten-speicherplatzes für Ihre Definitionen.
- Die Terminalsteuertabelle (TCT = Terminal Control Table) verwendet mehr als die erforderliche Kapazität an virtuellem Speicher für Ihre Definitionen.
- Für die Verwaltung der Definitionen fallen unnötige Arbeitsaufwände an.
- Sie können das Terminal auf dem Terminalverwaltungssystem automatisch installieren. Auf den Anwendungsverwaltungssystemen ist dies jedoch nicht möglich.

*Beispiel:*

In diesem Beispiel wird eine APPC-Einheit (Netzname APPC1) mit einer Terminalverwaltungsregion (Anwendungs-ID CICA) verbunden, die ihrerseits mit einer Anwendungsverwaltungsregion (Anwendungs-ID CICB) verbunden ist.

Ressourcen, die in der Terminalverwaltungsregion installiert sind, werden in CSD1 verwaltet. Die Ressourcen, die in der Anwendungsverwaltungsregion installiert sind, werden hingegen in CDS2 verwaltet. Die Konfiguration ist in Abb. 4 auf Seite 221 dargestellt.

Diese Tabelle enthält die Ressourcendefinitionen, die in den einzelnen Regionen erforderlich sind:

Zweck der Definitionen	Definitionen in Terminalverwaltungsregion	Definitionen in Anwendungsverwaltungsregion
Definiert die Verbindung zwischen den Systemen	CONNECTION(AOR) NETNAME(CICB)	CONNECTION(TOR) NETNAME(CICA)

Zweck der Definitionen	Definitionen in Terminalverwaltungsregion	Definitionen in Anwendungsverwaltungsregion
Definiert die Verbindung zur APPC-Einheit unter Verwendung einer TERMINAL- und einer TYPETERM-Definition	TERMINAL (APC1) TYPETERM (APPC0001) NETNAME (APPC1)  TYPETERM (APPC0001) DEVICE (APPC)	TERMINAL (BPC1) TYPETERM (APPC0001) NETNAME (APPC1) REMOTESYSTEM (TOR) REMOTENAME (APC1)  TYPETERM (APPC0001) DEVICE (APPC)
Definiert die Verbindung zur APPC-Einheit unter Verwendung einer CONNECTION- und einer SESSIONS-Definition	CONNECTION (APC1) NETNAME (APPC1)  SESSIONS (APPC0001) CONNECTION (APC1)	CONNECTION (BPC1) NETNAME (APPC1) REMOTESYSTEM (TOR) REMOTENAME (APC1)

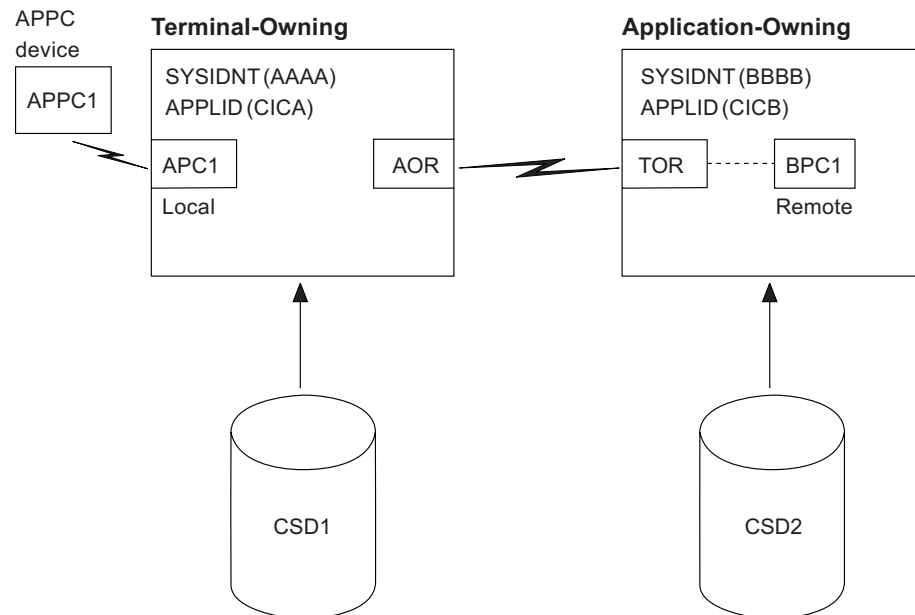


Abbildung 4. Lokale und ferne Definitionen separat verwalten.

## APPC-Einheiten für Transaktionsrouting

Das APPC-Transaktionsrouting ermöglicht einer APPC-Einheit, die zu einem CICS-System gehört, das Aufrufen von Transaktionen auf einem anderen CICS-System.

Die in „Terminals für Transaktionsrouting“ auf Seite 216 beschriebenen drei Methoden zum Definieren eines Terminals für das Transaktionsrouting gelten auch für APPC-Einheiten. Diese Methoden können die Nutzung von TERMINAL-TYPE-TERM- oder CONNECTION-SESSION-Definitionen umfassen. Für APPC-Einzelsitzungsterminals wird die Methode mit der TERMINAL-TYPETERM-Definition empfohlen (siehe hierzu „APPC-Einzelsitzungsterminal (LUTYPE6.2)“ auf Seite 216.)

## Zusammenfassung zum Transaktionsrouting

Bevor Sie Terminals für das Transaktionsrouting definieren, müssen Sie prüfen, welche der verschiedenen Methoden zum Definieren von Terminals sich am besten für Ihre CICS-Konfiguration eignet.

- Verwenden Sie die **Versandmethode**, es sei denn, Sie benutzen Terminals, die auf verschiedenen Systemen mit unterschiedlichen Namen angegeben wurden. Damit ATI (Automatic Transaction Initiation; automatische Transaktionsinitialisierung) mit der Versandmethode in einem Transaktionsverwaltungssystem verwendet werden kann, müssen Sie ggf. die globalen Exits XALTENF und XICTENF verwenden. Weitere Informationen zur Programmierung für diese Exits hierzu finden Sie in Terminal not known condition exits XALTENF and XICTENF.
- Verwenden Sie die **Methode mit gemeinsamer Nutzung** für Systeme, die eine CSD-Datei gemeinsam nutzen, wenn Sie die gleichen Namen auf unterschiedlichen Systemen benutzen und wenn Sie keine globalen Exits verwenden wollen, um sicherzustellen, dass ATI funktioniert.
- Verwenden Sie die **Duplizierungsmethode**, wenn Sie Terminals benutzen, die auf verschiedenen Systemen unter unterschiedlichen Namen angegeben sind oder wenn Sie ATI zum Anfordern von Terminals verwenden, jedoch nicht über eine gemeinsam genutzte CSD-Datei verfügen und die globalen Benutzerexits XALTENF und XICTENF nicht verwenden wollen.

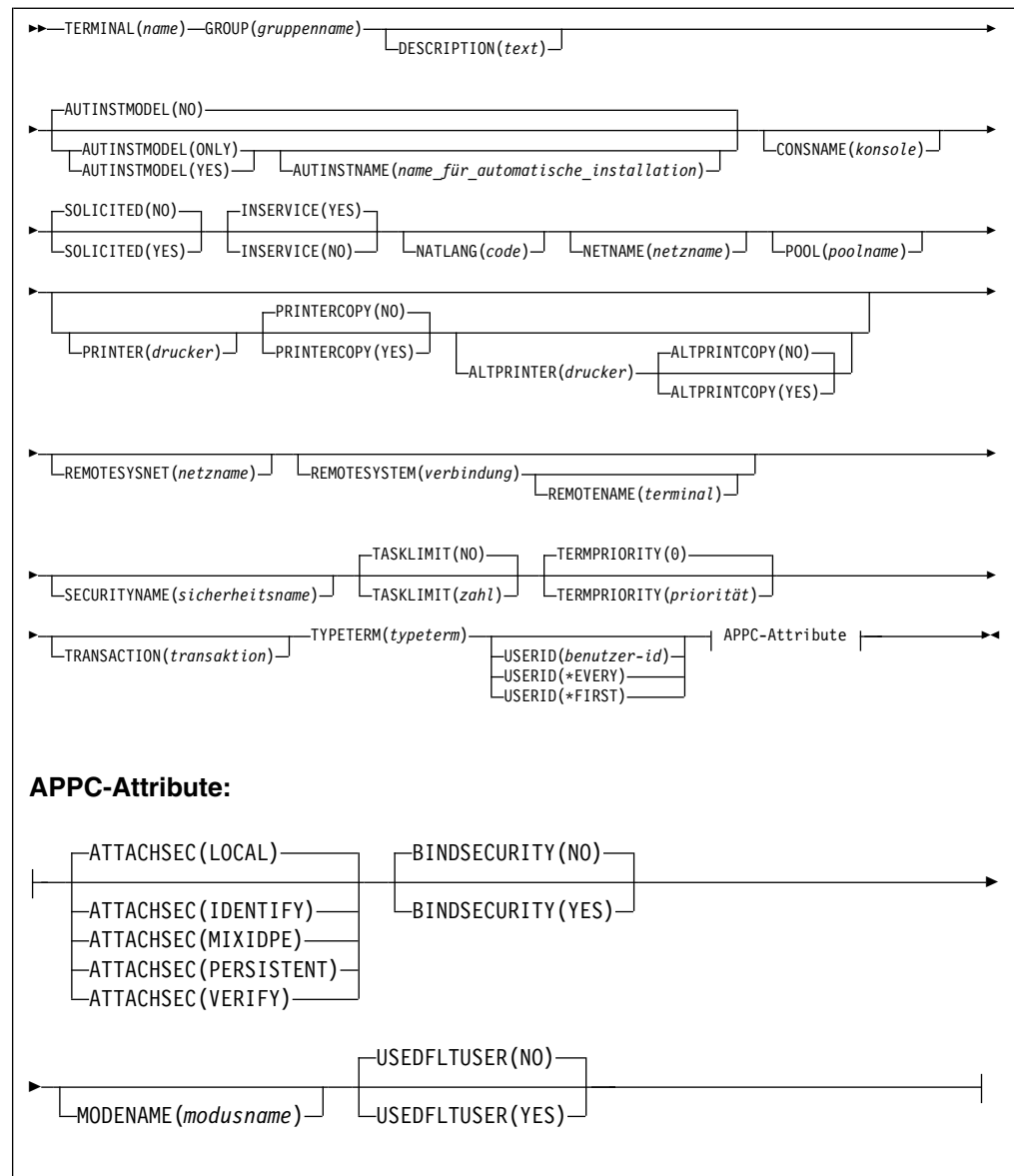
Sie können die Methoden kombinieren und z. B. die Versandmethode für Bildschirmanzeigen und die Duplizierungsmethode für Drucker benutzen, zu deren Anforderung ATI benötigt wird, ohne aber die globalen Benutzerexits XALTENF und XICTENF zu verwenden.

Bevor Sie mit der Erstellung von Definitionen für Ressourcen für die übergreifende Kommunikation beginnen, sollten Sie sich mit den weiterführenden Informationen in Defining connections to remote systems vertraut machen. Dort finden Sie nützliche Beispiele für die Attribute, die Sie für die unterschiedlichen Typen von Verbindungen und Sitzungen angeben müssen.

## TERMINAL-Attribute

Beschreibt die Syntax und die Attribute der TERMINAL-Ressource.





#### ALTPRINTERCOPY({NO|YES})

Gibt an, ob die Hardwarefunktion COPY verwendet werden soll, um eine Druckeranforderung auf dem im Attribut ALTPRINTER angegebenen Drucker auszuführen. Weitere Einzelheiten hierzu finden in der Beschreibung zum Attribut PRINTERCOPY.

**NO** CICS verwendet die Hardwarefunktion COPY.

**YES** CICS verwendet die Hardwarefunktion COPY nicht.

#### ALTPRINTER(*drucker*)

Gibt den Namen eines 3270-Druckers an, der verwendet werden soll, wenn der im Attribut PRINTER angegebene Drucker dieser Terminaldefinition nicht verfügbar ist. Der Name kann bis zu vier Zeichen lang sein. Weitere Einzelheiten hierzu finden in der Beschreibung zum Attribut PRINTER. Wenn Sie ALTPRINTER ohne einen Wert für PRINTER angeben, wird ALTPRINTER ignoriert.

Der angegebene Drucker muss dem CICS-System zugeordnet sein, das auch Eigner dieser Terminaldefinition ist.

Um die Hardwarefunktion COPY für den alternativen Drucker anzugeben, müssen Sie für ALTPRINTCOPY in dieser Terminaldefinition YES angeben.

**ATTACHSEC({LOCAL|IDENTIFY|VERIFY| PERSISTENT|MIXIDPE})) (nur APPC)**

Gibt die erforderliche Stufe der Benutzersicherheit an, die für die Herstellung der Verbindung erforderlich ist. PERSISTENT und MIXIDPE sind nur gültig, wenn als Zugriffsmethode z/OS Communications Server festgelegt ist und wenn APPC verwendet wird.

**LOCAL**

Die Berechtigung des Benutzers wird mit der Berechtigung der Verbindung selbst gleichgesetzt und die Verbindungssicherheit schützt nur die Ressource.

**IDENTIFY**

Eingehende Verbindungsanforderungen müssen eine Angabe zur Benutzer-ID enthalten. Geben Sie IDENTIFY an, wenn das Terminal, das die Verbindung herstellt, über einen Sicherheitsmanager verfügt.

**MIXIDPE**

Eine Verbindung kann die Zuordnung über den Sicherheitstyp IDENTIFY und/oder PERSISTENT unterstützen. Der verwendete Sicherheitstyp hängt von der eingehenden Verbindungsanforderung ab.

Wie in früheren Releases impliziert IDENTIFY ein gewisses Maß an Vertrauen zwischen den beiden Systemen, die es diesem System ermöglicht, die Anmeldelogik des anderen Systems zu akzeptieren. Tatsächlich haben Sie einen verteilten Sicherheitsmanager, bei dem ein System die Anmeldefunktion und das andere System die Sicherheitsprüfung ausführt.

**PERSISTENT**

Diese Option umfasst die Anmeldung eines Benutzers an einem fernen System, die über mehrere Dialoge hinweg solange erhalten bleibt, bis sich der Benutzer vom fernen System wieder abmeldet. Auf diese Weise werden die Benutzer-ID und das Kennwort nur bei der ersten Verbindungsherstellung (Anmeldung) übergeben. Für nachfolgende Verbindungsanforderungen ist lediglich die Benutzer-ID erforderlich.

**VERIFY**

Eingehende Verbindungsanforderungen müssen eine Angabe zur Benutzer-ID und ein Benutzerkennwort enthalten. Geben Sie VERIFY an, wenn das Terminal, das die Verbindung herstellt, keinen Sicherheitsmanager hat und daher verifiziert werden muss.

**AUTINSTMODEL({NO|YES|ONLY})**

Gibt an, ob diese Terminaldefinition als Modellterminaldefinition für die automatische Installation verwendet werden kann. Weitere Informationen zur automatischen Installation und zu den Modellterminaldefinitionen finden Sie in Autoinstalling z/OS Communications Server terminals.

**NO**

Diese Definition wird nicht als Modell für die automatische Installation verwendet. Sie wird nur als Definition für eine bestimmte Einheit verwendet, die nicht automatisch installiert wird.

**ONLY**

Diese Definition wird nur als Modell für die automatische Installation verwendet. Sie wird nicht als Definition für eine bestimmte Einheit verwendet.

**YES**

Diese Definition wird für eine bestimmte Einheit verwendet, die nicht

automatisch installiert wird. Die Definition wird auch als Modell für die automatische Installation verwendet.

**AUTINSTNAME**(*name\_für\_automatische\_installation*)

Gibt den Namen an, unter dem diese Modelldefinition im Steuerprogramm für die automatische Installation bekannt ist. Der Name kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Dieser Name wird nur dann angegeben, wenn für AUTINSTMODEL die Einstellung YES oder ONLY angegeben ist. Sie können standardmäßig den Terminalnamen gefolgt von vier Leerzeichen verwenden, wenn dieser Name für das Steuerprogramm für die automatische Installation zulässig ist. Weitere Informationen zu Modellen für die automatische Installation, zu Namen für die automatische Installation und zum Steuerprogramm für die automatische Installation finden Sie in Autoinstall control program.

**BINDPASSWORD**

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt.

**BINDSECURITY**(**{NO|YES}**) (nur APPC)

Gibt an, ob für die Bindungssicherheit ein externer Sicherheitsmanager (ESM) verwendet wird.

**NO** Es ist keine externe Bindungssicherheit erforderlich.

**YES** Wenn die Sicherheitsfunktion aktiv ist und der Systeminitialisierungsparameter **XAPPC** auf YES gesetzt ist, dann versucht CICS, den Sitzungsschlüssel aus RACF zu extrahieren, um die Bindungssicherheit zu erreichen. Wenn kein RACF-Profil verfügbar ist, dann schlägt die Bindung fehl.

**CONSOLE**

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt.

**CONSNAME**(*konsole*)

Mit CONSNAME können Sie eine CICS-Konsolendefinition installieren, ohne dass hierzu bereits eine Konsole vorhanden sein muss und ohne dass die Konsole zuvor im Member CONSOL $nn$  der MVS-Bibliothek SYS1.PARMLIB definiert werden muss. Bevor Sie die Konsole verwenden können, müssen Sie allerdings den Namen in MVS entweder im Member CONSOL $nn$  von SYS1.PARMLIB oder durch dynamische Zuordnung definieren. Die Länge von CONSNAME muss zwischen zwei und acht Zeichen betragen und der Name muss mit einem alphabetischen Zeichen oder einem der Zeichen #, @ oder \$ beginnen. Er dient zur eindeutigen Identifikation der Konsoleneinheit in einer CICS-Region unabhängig von dem MVS-Image, zu dem eine Verbindung besteht. Dies bedeutet, dass Sie nicht zwei Konsolendefinitionen mit derselben Angabe für CONSNAME installieren können. Das Attribut CONSNAME entspricht dem Namen, der für die Konsole im Member der MVS-Bibliothek SYS1.PARMLIB (CONSOL $nn$ ) definiert ist.

Um einen TSO-Benutzer als Konsoleneinheit zu definieren, müssen Sie CONSNAME(*name*) angeben. Dabei steht *name* für die TSO-Benutzer-ID. Diese Defi-

nition aktiviert einen TSO-Benutzer, der zur Verwendung des Befehls TSO CONSOLE zur Einleitung von CICS-Transaktionen berechtigt ist. Die TSO-Benutzer-ID muss nicht im Member CONSOLnn von SYS1.PARMLIB definiert sein.

Die Entsprechung von CONSOLE(00) ist CONSNAME(INTERNAL) oder CONSNAME(INSTREAM), und zwar abhängig von der Servicestufe von CICS und dem verwendeten Release von MVS. Geben Sie diese Option an, wenn Sie eine CICS-Transaktion einleiten und einen Befehl für sie in einer JCL-Anweisung absetzen wollen. Anweisungen zur Verwendung von JCL zum Absetzen von CICS-Befehlen finden Sie in Using JCL to initiate CICS commands.

#### **DESCRIPTION**(*text*)

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

#### **GROUP**(*gruppenname*)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

#### **INSERVICE**(**{YES|NO}**)

Gibt den Status des Terminals an, das definiert wird.

**YES** Transaktionen können initialisiert werden und Nachrichten werden automatisch an das Terminal gesendet.

**NO** Das Terminal kann keine Nachrichten empfangen oder Eingabedaten übertragen.

#### **MODENAME**(*modusname*) (**nur APPC-Einzelsitzungsterminals**)

Gibt den Namen an, der als LOGMODE-Name an z/OS Communications Server übergeben wird. Der Name kann bis zu acht Zeichen lang sein, der reservierte Name SNASVCMG darf jedoch nicht verwendet werden. Der Name wird auf Basis der Regeln der Assemblersprache gebildet und muss mit einem alphabetischen Zeichen beginnen.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Weitere Anweisungen zum LOGMODE-Namen finden Sie in Coding entries in the z/OS Communications Server LOGON mode table.

### **NATLANG(*code*)**

Gibt die Sprache an, in der alle globalisierungsfähigen Nachrichten für dieses Terminal angezeigt werden. Geben Sie einen der folgenden NATLANG-Codes an. Der Standardwert ist der in NATLANG system initialization parameter angegebene Wert.

- E** Englisch. Dies ist die Standardeinstellung des *Systems* (die auch dann verwendet wird, wenn Sie den Code E nicht explizit angeben).
- C** Vereinfachtes Chinesisch. Hierbei handelt es sich um eine DBCS-Sprache (DBCS = Double-Byte Character Set; Doppelbytezeichensatz). Die Umsetzung wird von IBM durchgeführt.
- K** Japanisch. Hierbei handelt es sich um eine DBCS-Sprache (DBCS = Double-Byte Character Set; Doppelbytezeichensatz). Die Umsetzung wird von IBM durchgeführt.

### **NETNAME(*netzname*)**

Gibt den Netznamen an, der das Terminal für ACF/Communications Server identifiziert. Der Name kann bis zu acht Zeichen lang sein. Der Name wird auf Basis der Regeln der Assemblersprache gebildet und muss mit einem alphabetischen Zeichen beginnen.

#### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Wenn Sie keinen Namen angeben, dann wird für NETNAME standardmäßig der TERMINAL-Name verwendet.

Der Wert für NETNAME muss eindeutig sein, es sei denn, es handelt sich um ein fernes Terminal. Das heißt, Sie können nicht zwei lokale Terminals mit demselben Wert für NETNAME oder ein lokales Terminal und eine beliebige Verbindung mit demselben Wert für NETNAME installieren. Der Wert für NETNAME für ein fernes Terminal kann jedoch mit dem Wert für NETNAME für jedes andere Terminal oder dem Wert für NETNAME für eine beliebige Verbindung identisch sein.

Wenn die CICS-Region die dynamische LU-Alias-Funktion von z/OS Communications Server unterstützt und in der Anweisung APPL der CICS-Region LUAPFX=*xx* angegeben ist, dann wird angenommen, dass sich das Terminal mit diesem Netznamen (NETNAME) in demselben Netz wie die CICS-Region befindet. Wenn das Terminal sich in einem anderen Netz befindet, dann muss es für z/OS Communications Server in einer CDRSC-Definition mit einem vordefinierten Wert für LUALIAS (LUALIAS=*netzname*) definiert sein, um die dynamische Zuordnung von z/OS Communications Server zu überschreiben. In diesem Fall muss *netzname* im Parameter LUALIAS mit dem Wert für NETNAME übereinstimmen, der in dieser Terminalressourcendefinition definiert ist.

### **OPERID**

### **OPERPRIORITY**

### **OPERRSL**

### **OPERSECURITY**

Diese Attribute sind veraltet, werden jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt.

### **POOL**(*poolname*)

Gibt den Poolnamen für ein 3600- oder 3650-Pipeline-Terminal an, das mit anderen Pipeline-Terminals in einem Pool zusammengefasst ist.

Wenn Sie eine logische 3600-Pipelineeinheit definieren, generieren Sie einen Terminaleintrag der Terminalsteuertabelle (TCTTE = Terminal Control Table Terminal Entry), der einem Pool von TCTTEs zugeordnet ist. Ein Pool mit Pipeline-TCTTEs kann von einer logischen Pipelineeinheit verwendet werden, oder er kann von einer Reihe von logischen Pipelineneinheiten gemeinsam genutzt werden.

Der Poolname wird nur als Methode zum Identifizieren der zugehörigen TERMINAL-Definitionen in der CSD-Datei verwendet. Er wird nicht auf dem aktiven CICS-System verwendet. Der Name kann bis zu acht Zeichen lang sein.

#### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Für ein Pipeline-Terminal müssen Sie einen TYPETERM-Wert mit der Angabe SESSIONTYPE(PIPELINE) definieren. Sie müssen einen Wert für TASKLIMIT in mindestens einem Pool von Pipeline-Terminals angeben. Sie müssen für jedes der Pipeline-Terminals in einem Pool dieselbe Gruppe angeben.

Im Folgenden finden Sie ein Beispiel für die Definitionen eines Pools von Pipeline-Terminals:

```
DEFINE TERMINAL(ttt1) GROUP(g) POOL(poolid)
    TYPETERM(xxxxxxx) NETNAME(nnnnnnn1)...
DEFINE TERMINAL(ttt2) GROUP(g) POOL(poolid)
    TYPETERM(xxxxxxx) NETNAME(nnnnnnn2)...
DEFINE TERMINAL(ttt3) GROUP(g) POOL(poolid)
    TYPETERM(xxxxxxx) NETNAME(nnnnnnn3)...
DEFINE TERMINAL(ttt4) GROUP(g) POOL(poolid)
    TASKLIMIT(nn) TYPETERM(xxxxxxx)
    NETNAME(nnnnnnn4)...
DEFINE TYPETERM(xxxxxxx) GROUP(g)
    DEVICE(3650) SESSIONTYPE(PIPELINE)
```

### **PRINTER**(*drucker*)

Gibt den Namen des primären 3270-Druckers an, der verwendet werden soll, um auf einen Befehl ISSUE PRINT oder eine PRINT-Anforderung von einem Bediener zu reagieren, der eine Programmmzugriffstaste (PA-Taste) drückt. Der Name kann bis zu vier Zeichen lang sein. Der Name gibt den TERMINAL-Namen in der Definition für den Drucker an. Wenn Sie hier einen Wert für PRINTER angeben, muss das TYPETERM-Element, auf das durch diese TERMINAL-Definition verwiesen wird, die Angabe PRINTADAPTER(NO) enthalten.

Der angegebene Drucker muss demselben CICS-System zugeordnet sein, das auch Eigner dieser Terminaldefinition ist.

Sie können einen Wert für PRINTER angeben, wenn diese TERMINAL-Definition für eine der folgenden Einheiten gilt:

- 3270-Bildschirmeinheit ohne Druckeradapterfunktion
- 3270-Bildschirmeinheit, die an eine 3274-Steuereinheit angeschlossen ist
- Datensichtgerät eines 3276-Controllers
- 3790-Einheit im 3270-Kompatibilitätsmodus

Wenn Sie die Hardwarefunktion COPY angeben wollen, müssen Sie in dieser TERMINAL-Definition PRINTERCOPY(YES) angeben.

Beachten Sie, dass SCS-Drucker (SCS = SNA Character String) nur 3790-Datenströme akzeptieren; 3270-Datenströme werden hingegen nicht akzeptiert. Sie können daher nicht dazu verwendet werden, den Pufferinhalt einer Bildschirm-  
einheit zu drucken.

Wenn Sie eine Programmabrufaste (z. B. PA1) verwenden, um die Inhalte der Anzeige auf einem zugeordneten z/OS Communications Server-Drucker auszugeben, dann wird die Anzeigegröße des Druckers auf Basis des Werts im Operanden SCRNSIZE, der in dem von CICS bereitgestellten Standardprofil DFHCICST angegeben wurde, ausgewählt. Dieses Profil ist mit SCRNSIZE(DEFAULT) definiert und wenn Sie die alternative Anzeigegröße des Druckers verwenden wollen, dann müssen Sie den Profileintrag für die CSPP-Transaktion ändern.

#### **PRINTERCOPY({NO|YES})**

Gibt an, ob die Hardwarefunktion COPY verwendet werden soll, um eine Druckanforderung auf dem im Attribut PRINTER angegebenen Drucker dieser Terminaldefinition zu verarbeiten.

CICS verwendet die Hardwarefunktion COPY der 3270-Einheiten, um die Druckoperation auszuführen, es sei denn, der Anzeige ist momentan eine Task zugeordnet.

In der TYPETERM-Definition müssen Sie COPY(YES) nicht angeben, weil diese Einstellung durch die Angabe von PRINTERCOPY(YES) in der Terminaldefinition impliziert wird.

Wenn Sie einen Wert für ALTPRINTER sowie für PRINTER angegeben haben, dann können Sie ALTPRINTCOPY(YES) festlegen.

Zur Verwendung der COPY-Funktion müssen sich sowohl der Drucker als auch die Bildschirmanzeige unter der Steuerung desselben 3270-Controllers befinden. Andernfalls schlägt entweder die COPY-Operation fehl und führt zu einer Fehlerbedingung oder (wenn die Adresse der Bildschirmeinheit für den Druckercontroller gültig ist) die Kopieroperation wird von einem anderen Bildschirm aus ausgeführt.

Geben Sie PRINTERCOPY(YES) nicht an, wenn in einer Netzumgebung der 3270-Controller mit einem TCAM-System in einer Domäne verbunden ist und ein CICS-System in einer anderen Domäne über z/OS Communications Server Zugriff auf den Controller hat. Die Hardware-COPY-Adresse steht in CICS nicht zur Verfügung und kann deshalb von Terminals, die an einen solchen Controller angeschlossen sind, nicht verwendet werden.

Der Befehl COPY ist für einen Bildschirm mit 3270-Kompatibilitätsmodus nicht zulässig.

#### **REMOTENAME(*terminal*)**

Gibt den Namen an, unter dem das Terminal auf dem System oder in der Region bekannt ist, das bzw. die Eigner des Terminals ist. Der Name kann bis zu vier Zeichen lang sein.

##### **Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

#### **REMOTESYSNET(*netzname*)**

Gibt den Netznamen (APPLID) der Region an, die Eigner des Terminals ist. Der Name kann bis zu acht Zeichen lang sein. Er wird auf Basis der Regeln der Assemblersprache gebildet und muss mit einem alphabetischen Zeichen beginnen.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Das Attribut REMOTESYSNET wird verwendet, wenn keine direkte Verbindung zwischen der Region, in der diese Definition installiert ist, und der Terminalverwaltungsregion (TOR = Terminal-Owning Region) besteht. REMOTESYSNET muss nicht angegeben werden, wenn eine der folgenden Bedingungen zutrifft:

- Sie definieren ein lokales Terminal. Dies bedeutet, dass REMOTESYSTEM nicht angegeben ist oder die System-ID des lokalen Systems angibt.
- REMOTESYSTEM gibt eine direkte Verbindung zur Terminalverwaltungsregion (TOR = Terminal-Owning Region) an. Wenn die Terminalverwaltungsregion Mitglied einer Gruppe für generische z/OS Communications Server-Ressourcen ist und die direkte Verbindung zu ihr eine APPC-Verbindung ist, dann muss REMOTESYSNET möglicherweise angegeben werden. REMOTESYSNET wird in diesem Fall benötigt, wenn der in der CONNECTION-Definition angegebene NETNAME-Wert für die direkte Verbindung mit dem generischen Ressourcennamen (und nicht mit der Anwendungs-ID) der Terminalverwaltungsregion (TOR = Terminal-Owning Region) übereinstimmt.

**REMOTESYSTEM(verbindung)**

Gibt den Namen an, der die Verbindung für die übergreifende Kommunikation zu dem System identifiziert, das Eigner des Terminals ist. Der Name kann bis zu vier Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

In REMOTESYSTEM ist eine der folgenden Kennungen enthalten:

- Für IPIC-Verbindungen gibt REMOTESYSTEM die ersten vier Zeichen des IPCONN-Namens in der IPCONN-Definition an, die zur Verfügung steht und angefordert werden kann.
- Für MRO- und APPC-Verbindungen gibt REMOTESYSTEM den CONNECTION-Namen in der CONNECTION-Definition an. Erfolgt keine Angabe für den CONNECTION-Namen oder wird der Name als die System-ID des lokalen Systems angegeben, dann handelt es sich um ein lokales Terminal dieses Systems. Wird der Name eines anderen Systems angegeben, dann handelt es sich um ein fernes Terminal. Aus diesem Grund können Sie dieselbe Definition für das Terminal auf lokalen und auch fernen Systemen verwenden.

Sind zwischen diesem CICS-System und der Terminalverwaltungsregion Transitsysteme vorhanden, dann muss in REMOTESYSTEM die erste Verbindung im Pfad zu der Terminalverwaltungsregion angegeben werden. Wenn mehr als ein Pfad möglich ist, gibt REMOTESYSTEM die erste Verbindung im bevorzugten Pfad an.

REMOTESYSTEM wird ignoriert, wenn Sie AUTINSTMODEL(YES) oder AUTINSTMODEL(ONLY) angeben.



**SECURITYNAME(*sicherheitsname*)**

Gibt den Sicherheitsnamen des fernen Systems an.

Auf einem CICS-System mit initialisierter Sicherheitsfunktion (SEC=YES oder MIGRATE) wird der Sicherheitsname verwendet, um die Berechtigung des fernen Systems einzurichten.

Wenn in der Sitzungsdefinition (Befehl DEFINE SESSIONS), die der Verbindungsdefinition zugeordnet ist, die Option USERID angegeben wird, dann überschreibt ihr Wert die Benutzer-ID, die im Attribut SECURITYNAME angegeben ist. Dieser Wert wird dann für die Verbindungssicherheit verwendet.

Der Sicherheitsname oder der Wert für USERID in der Sitzungsdefinition muss eine gültige RACF-Benutzer-ID auf Ihrem System darstellen. Der Zugriff auf geschützte Ressourcen auf Ihrem System basiert auf dem RACF-Benutzerprofil und seiner Gruppenzugehörigkeit.

Weitere Informationen zum Definieren der Sicherheit für MRO-, LUTYPE6.1- und APPC-Verbindungen finden Sie in Intercommunication security.

**SOLICITED(NO|YES)**

Gibt an, ob CICS-Nachrichten, die an einer Konsole abgesetzt werden, von NetView als angefordert oder nicht angefordert eingestuft werden.

**NO** CICS-Nachrichten werden als nicht angefordert eingestuft.

**YES** CICS-Nachrichten werden als angefordert eingestuft. Wenn SOLICITED(YES) für eine Konsole angegeben wird, dann fügt CICS den Konsolennamen oder die Identifikationsnummer der Konsole sowie ein Befehls- und Antworttoken zu jeder Konsolennachricht hinzu.

Das Attribut SOLICITED gilt nur für Konsolen; es wird für andere TERMINAL-Definitionen ignoriert.

**TASKLIMIT({NO|zahl})**

Gibt die Anzahl der gleichzeitig ablaufenden Tasks an, die in einer Pipelinesitzung oder in einem Pool von Pipelinesitzungen ausgeführt werden dürfen.

**NO** Es sind keine gleichzeitig ablaufenden Tasks zulässig.

**zahl** Die Anzahl der Tasks, die gleichzeitig ausgeführt werden dürfen, im Bereich zwischen 1 und 32767.

Wenn Sie eine logische 3600-Pipelineeinheit definieren, generieren Sie einen Terminaleintrag der Terminalsteuertabelle (TCTTE = Terminal Control Table Terminal Entry), der einem Pool von TCTTEs zugeordnet ist. Wenn Nachrichten in CICS von der logischen 3600-Pipelineeinheit empfangen werden, dann wird eine Task zur Verarbeitung dieser Nachricht zugeordnet. Dazu wird als Ankerblock einer der TCTTEs aus dem Pool verwendet. Auf diese Weise können aufeinanderfolgende Nachrichten, die über die logische Pipelineeinheit gesendet werden, gleichzeitig verarbeitet werden. Die Anzahl der gleichzeitig ablaufenden Transaktionen wird anhand der TCTTEs im Pool begrenzt. Die Anzahl der TCTTEs im Pool sollte der Obergrenze der Anfrageaktivität entsprechen. Auf diese Weise ermöglicht die Pipelinefunktion die Definition einer geringeren Anzahl von TCTTEs in CICS als die Gesamtzahl der Pipelineanfrageterminals.

**TERMINAL(*name*)**

Gibt die Terminal-ID an. Für ein Terminal kann der Name bis zu vier Zeichen lang sein.

**Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Für ein APPC-LU6.2-Einzelsitzungsterminal sind die folgenden Zeichen zulässig:

**Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ #

Wenn der angegebene Name weniger als vier Zeichen umfasst, dann wird er linksbündig ausgerichtet und mit Leerzeichen auf eine Länge von vier Zeichen aufgefüllt. Es sollten keine TERMINAL- und CONNECTION-Definitionen mit demselben Namen vorhanden sein.

**Anmerkung:** Wenn Sie in einem Namen ein Komma (,) verwenden, dann können folgende Befehle nicht verwendet werden:

```
CEMT INQUIRE TERMINAL(wert1,wert2)
CEMT SET      TERMINAL(wert1,wert2)
```

Hierbei dient das Komma als Listenbegrenzer. Informationen zur Verwendung von Listen mit Ressourcen-IDs finden Sie in List of resource identifiers.

Wenn Sie zulassen, dass Sitzungsnamen oder Terminal-IDs für Konsolen von CICS automatisch generiert werden, dann sollten Sie die Verwendung der folgenden Symbole als erstes oder viertes Zeichen im Terminalnamen vermeiden:

- Der Bindestrich wird von CICS für automatisch generierte Terminalnamen für APPC-Sitzungen verwendet.
- > und < Das Symbol > (größer als) und das Symbol < (kleiner als) werden von CICS für automatisch generierte Terminalnamen für IRC-Sitzungen verwendet.
- ¬ Das Symbol ¬ (logisches Nicht) wird von CICS für automatisch generierte Terminalnamen für MVS-Konsolen verwendet.

Der angegebene Name wird als Name des TCT-Eintrags verwendet, wenn diese TERMINAL-Definition installiert wird. Aus diesem Grund muss der TERMINAL-Name eindeutig sein. Beachten Sie, dass der Wert CERR für die Kennung reserviert ist, die für die Fehlerkonsole generiert wird. Wenn Sie AUTINSTMODEL(ONLY) angeben, dann benötigen Sie keinen eindeutigen TERMINAL-Namen, weil er nicht als Name eines TCT-Eintrags verwendet wird. Wenn Sie AUTINSTMODEL(YES) angeben, dann wird der TERMINAL-Name als Name des TCT-Eintrags verwendet, der in der Terminalsteuertabelle (TCT) installiert wird, wenn die TERMINAL-Definition installiert wird. Die Namen der TCT-Einträge für automatisch installierte Terminals werden mithilfe des Benutzerprogramms für die automatische Installation ermittelt.

Wenn das Terminal einem Ziel für transiente Daten zugeordnet werden soll, müssen der Terminalname und der Name der Warteschlange mit transienten Daten in der TDQUEUE-Ressourcendefinition identisch sein.

**TERMPRIORITY({0|priorität})**

Gibt die Terminalpriorität an. Dieser Dezimalwert (0 bis 255) wird verwendet, um die übergreifende Priorität für die Transaktionsverarbeitung einzurichten. Die Priorität für die Transaktionsverarbeitung entspricht der Summe der Terminalpriorität, Transaktionspriorität und Bedienerpriorität und darf den Wert 255 nicht überschreiten.

### TRANSACTION(*transaktion*)

Gibt einen aus einem bis vier Zeichen bestehenden Namen der Transaktion an, die immer dann eingeleitet werden soll, wenn Eingabedaten von dem Terminal empfangen werden und keine aktive Task vorhanden ist.

#### Zulässige Zeichen:

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Für Nicht-3270-Einheiten von z/OS Communications Server wird für einen TRANSACTION-Namen mit weniger als vier Zeichen ein Begrenzungszeichen benötigt.

Informationen zu den Vorgängen, die bei der Einleitung einer Transaktion ausgeführt werden, finden Sie in Terminal control.

Wenn Sie diesen Operanden für eine 3270-Bildschirmeinheit angeben, dann kann der Bediener außer dieser Transaktion nur die CICS-Funktionen für Blätterbefehle und Druckanforderungen starten.

Die Angabe des Attributs TRANSACTION für eine 3270-Bildschirmeinheit oder einen SCS-Drucker ist eher die Ausnahme. Für logische 3601-Einheiten ist das Attribut optional, für logische 3614-Einheiten jedoch zwingend erforderlich.

Wenn dieser Operand für ein 3790 Communication System angegeben wird und mehrere Sitzungen verwendet werden, um eine Verbindung zu derselben 3791-Einheit herzustellen, dann geben Sie den gleichen TRANSACTION-Namen für alle Sitzungen an.

### TYPETERM(*typeterm*)

Gibt den Namen der TYPETERM-Definition an, die dieser TERMINAL-Definition zugeordnet werden soll. Der Name kann bis zu acht Zeichen lang sein.

#### Zulässige Zeichen:

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Die TYPETERM-Definition gibt viele Attribute für eine Reihe von ähnlichen Terminals an. Weitere Informationen hierzu finden Sie in „TYPETERM-Attribute“ auf Seite 279 und Ways of defining CICS resources. Das Attribut TYPETERM ist für alle TERMINAL-Definitionen zwingend erforderlich.

Die TYPETERM-Definition muss bereits installiert sein, wenn Sie diese TERMINAL-Definition installieren.

### USEDFTUSER ({**NO**|**YES**}) (nur APPC)

Gibt die Art der Sicherheitsprüfung an, die für jeden eingehenden FMH (Function Management Header; Funktionsverwaltungsheader) für die Verbindungsherstellung ausgeführt werden soll.

**NO** Gibt an, dass jeder eingehende FMH für die Verbindungsherstellung auf das Vorhandensein der Felder überprüft wird, die für die Option ATTACHSEC erforderlich sind. Wenn die erforderlichen Felder nicht vorhanden sind, wird eine Nachricht über einen Protokollverstoß ausgegeben und die Herstellung der Verbindung schlägt fehl. Der Standardwert ist NO.

**YES** Gibt an, dass bestimmte Prüfungen auf die Gültigkeit des FMH für die

Verbindungsherstellung übergangen werden. Weitere Informationen hierzu finden Sie in Implementing LU6.2 security.

#### **USERID({name|\*EVERY|\*FIRST})**

Gibt eine Benutzer-ID an, die für die Anmeldung verwendet wird und auf die in Sicherheitsfehlernachrichten, Sicherheitsverstoßnachrichten und im Prüfprotokoll verwiesen wird. Hierbei muss es sich um eine gültige Benutzer-ID handeln, die für den Sicherheitsmanager definiert wurde.

Das Attribut USERID bietet die einzige Möglichkeit, um eine Benutzer-ID für Einheiten wie beispielsweise Drucker anzugeben, die keine Anmeldung mit CESN durchführen können. Sie können USERID auch für eine Bildschirm Einheit angeben. In diesem Fall bleibt der Bildschirm permanent angemeldet. Bediener können sich nicht anmelden. Der gesamte Zugriff auf geschützte Ressourcen hängt von den Berechtigungen ab, die von RACF für die angegebene Benutzer-ID (USERID) erteilt wurden.

Für ein APPC-Einzelsitzungsterminal überschreibt der Wert von USERID alle Angaben für SECURITYNAME, die Sie für die Verbindung angegeben haben.

**name** Dieser Name kann bis zu acht Zeichen lang sein.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

#### **\*EVERY**

Dieser Sonderoperand gilt nur für automatisch installierte Konsolen. Dies bedeutet, dass CICS die mit dem MVS-Befehl MODIFY übergebene Benutzer-ID immer dann verwendet, wenn ein Befehl MODIFY empfangen wird. Die Konsole wird mit der MVS-Benutzer-ID als voreingestellter Benutzer-ID für die automatisch installierte Konsole angemeldet. Die Konsole bleibt mit dieser Benutzer-ID angemeldet, bis sie gelöscht wird oder ein anderer Befehl MODIFY mit einer anderen Benutzer-ID empfangen wird. Wenn ein Befehl MODIFY ohne eine Benutzer-ID empfangen wird, dann führt CICS die Anmeldung mit der CICS-Standard-Benutzer-ID durch, bis ein Befehl MODIFY empfangen wird, der über eine gültige Benutzer-ID verfügt. Für Nicht-Konsolenterminals oder wenn die Sicherheitsfunktion nicht aktiviert wurde, wird dieser Wert ignoriert.

#### **\*FIRST**

Dieser Sonderoperand gilt nur für automatisch installierte Konsolen. Dies bedeutet, dass CICS die Benutzer-ID verwendet, die mit dem ersten MVS-Befehl MODIFY übergeben wird, in dem die automatische Installation der Konsole angefordert wird. Die Konsole wird mit der MVS-Benutzer-ID als voreingestellter Benutzer-ID angemeldet. Die Konsole bleibt mit dieser Benutzer-ID angemeldet, bis die Konsole gelöscht wird. Wenn ein Befehl MODIFY ohne eine Benutzer-ID empfangen wird, dann führt CICS die Anmeldung mit der CICS-Standard-Benutzer-ID durch. Für Nicht-Konsolenterminals oder wenn die Sicherheitsfunktion nicht aktiviert wurde, wird dieser Wert ignoriert.

Wenn diese Terminaldefinition eine Konsole definiert, muss die Benutzer-ID für das korrekte Profil in der allgemeinen CONSOLE-Ressourcenklasse berechtigt sein. Informationen zur voreingestellten Sicherheit für Konsolen und Terminals finden Sie in Defining port of entry profiles.

---

## TRANCLASS-Ressourcen

Eine TRANCLASS-Ressource definiert die Merkmale einer Transaktionsklasse.

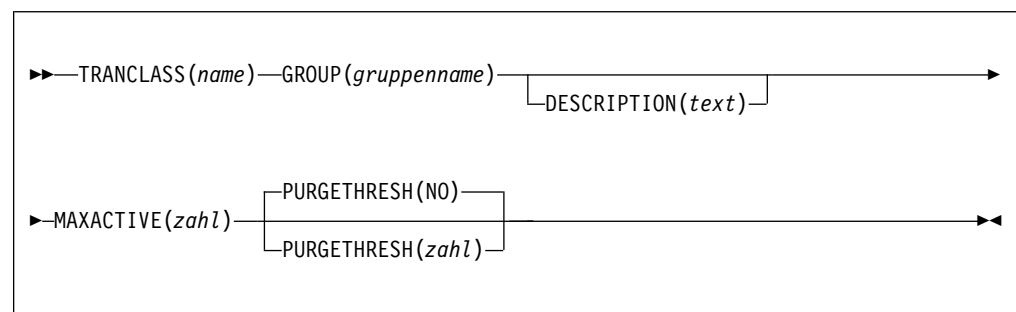
Transaktionen, die als zu einer Transaktionsklasse gehörig definiert wurden, müssen die geltenden Planvorgaben erfüllen, bevor sie ausgeführt werden können. Wenn Transaktionen, die zu einer aktiven Transaktionsklasse gehören, bereits ausgeführt werden, dann werden alle neuen Transaktionen in die Warteschlange eingestellt. Verwenden Sie das Attribut MAXACTIVE, um die maximale Anzahl von Transaktionen anzugeben, die Sie ausführen möchten. Zur Beschränkung der Warteschlangengröße können Sie das Attribut PURGETHRESH verwenden.

Indem Sie Ihre Transaktionen in Transaktionsklassen einordnen, können Sie steuern, wie CICS Tasks zuteilt. Sie können Transaktionen z. B. in ressourcenintensive Transaktionen und in solche Transaktionen aufteilen, die von nachgeordneter Bedeutung sind (z. B. Broadcastnachrichten mit Willkommensgrüßen). Anschließend können Sie die Attribute in der TRANCLASS-Definition verwenden, um die Anzahl der aktiven Tasks zu steuern, die aus jeder Transaktionsklasse zulässig sind.

Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with BAS transaction class resource definitions.

## TRANCLASS-Attribute

Beschreibt die Syntax und die Attribute der TRANCLASS-Ressource.



Die Beschreibungen der Attribute für die Transaktionsklassendefinition lauten wie folgt:

### DESCRIPTION(text)

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

### GROUP(gruppenname)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

**MAXACTIVE(*zahl*)**

Gibt die maximale Anzahl der Transaktionen in dieser Transaktionsklasse an, die aktiv sein dürfen. Sie müssen einen Wert für MAXACTIVE angeben, wenn Sie eine Transaktionsklasse definieren. Dieser Wert muss im Bereich zwischen 0 und 999 liegen.

Neue Transaktionen, die zugeordnet werden, wenn die Anzahl der aktiven Transaktionen den in MAXACTIVE angegebenen Grenzwert erreicht hat, werden unter Berücksichtigung des PURGETHRESH-Grenzwerts auf die Möglichkeit zur Einstellung in die Warteschlange geprüft.

Das Definieren einer Transaktionsklasse mit einem Nullwert für MAXACTIVE bedeutet, dass **alle** Tasks in die Warteschlange eingestellt werden.

**PURGETHRESH({NO|*zahl*})**

Dieses Attribut gibt einen optionalen Bereinigungsschwellenwert für die Transaktionsklasse an. Es definiert einen numerischen Schwellenwert, bei dessen Erreichung Transaktionen in der Warteschlange für die Zugehörigkeit zur Transaktionsklasse bereinigt werden. Geben Sie an, wenn Sie die Anzahl der Transaktionen beschränken möchten, die in der Warteschlange für diese Transaktionsklasse stehen können. Folgende Werte sind zulässig:

**NO** Die Größe der Warteschlange wird nur durch den Speicher begrenzt, der für die Zuordnung von Tasks verfügbar ist.

*zahl* Der numerische Wert für den Bereinigungsschwellenwert im Bereich zwischen 1 und 1 000 000.

Wenn Sie hier 1 angeben, dürfen keine Transaktionen in die Warteschlange eingestellt werden. Wenn Sie eine beliebige andere Nummer angeben (*n*), dann wird die Größe der Warteschlange auf den Wert *zahl*-1 beschränkt. Alle neuen Transaktionen, die zugeordnet werden, nachdem der Grenzwert von *n*-1 erreicht wurde, werden gelöscht.

**Beispiel für PURGETHRESH:** Im Falle einer Transaktionsklasse, bei der die maximale Anzahl aktiver Tasks (MAXACTIVE) auf den Wert 50 und der Bereinigungsschwellenwert (PURGETHRESH) auf 10 gesetzt ist, um die Einstellung von Transaktionen in die Warteschlange zu begrenzen, beginnt CICS damit, neue Transaktionen für die Klasse abzuberechnen, wenn die beiden folgenden Bedingungen eintreten:

- Die Anzahl der aktiven Transaktionen erreicht den Wert 50.
- Die Anzahl der Transaktionen, die sich in der Warteschlange für die Zugehörigkeit zur Transaktionsklasse befinden, erreicht den Wert 9.

CICS akzeptiert neue Transaktionen für diese Transaktionsklassenwarteschlange erst dann wieder, wenn die Anzahl der bereits eingestellten Elemente unter die Maximalgröße der Warteschlange (im Beispiel also 9) fällt.

**TRANCLASS(*name*)**

Gibt den Namen der Transaktionsklasse an. Transaktionen, die zu einer Transaktionsklasse gehören, müssen die geltenden Planvorgaben erfüllen, bevor sie

ausgeführt werden können. Der reservierte TRANCLASS-Name DFHTCL00 wird verwendet, um anzugeben, dass die Transaktion zu keiner Transaktionsklasse gehört.

Aus Gründen der Kompatibilität mit Releases, die das Attribut TCLASS unterstützen, stellt CICS die folgenden TRANCLASS-Äquivalente bereit:

TCLASS	TRANCLASS
NO	DFHTCL00
1	DFHTCL01
2	DFHTCL02
3	DFHTCL03
4	DFHTCL04
5	DFHTCL05
6	DFHTCL06
7	DFHTCL07
8	DFHTCL08
9	DFHTCL09
10	DFHTCL10

Beispieldefinitionen für diese Transaktionsklassen befinden sich in der Gruppe DFHTCL, die als Teil von DFHLIST bereitgestellt wird.

**Anmerkung:** Wenn eine Transaktion ausgeführt wird und die zugehörige TRANCLASS-Definition nicht installiert ist, wird die Transaktion ohne die in TRANCLASS angegebenen Planvorgaben ausgeführt. Das System gibt die Warnung DFHXM0212 aus.

Der Wert für TRANCLASS kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

## TRANSACTION-Ressourcen

Eine TRANSACTION-Ressource definiert Transaktionsattribute, die sich auf Funktionen beziehen, die von CICS bereitgestellt werden.

Eine CICS-Anwendung besteht aus mindestens einem Programm, das zur Ausführung einer bestimmten Funktion geschrieben wurde. Ein bestimmter Aufruf einer solchen Anwendung wird als *Transaktion* bezeichnet. Hierbei identifiziert der CICS-Transaktionsmanager die Transaktion anhand ihrer Transaktions-ID (TRANSID). Die TRANSACTION-Ressource kann auch als Anwendungseinstiegspunkt verwendet werden. Einstiegspunkte werden in Application entry points beschrieben.

Sie weisen CICS an, wie Ihre Transaktion ausgeführt werden soll. Dies geschieht primär über eine TRANSACTION-Definition, in der Sie Informationen wie beispielsweise die Transaktionspriorität, den Sicherheitsschlüssel sowie die Länge des Transaktionsarbeitsbereichs (TWA = Transaction Work Area) angeben. Der Name dieser Definition (sog. TRANSACTION-Name) ist identisch mit der Transaktions-ID (TRANSID). Außerdem müssen Sie die Transaktion mit anderen Ressourcen verbinden, indem Sie die Namen der zugehörigen Definitionen in der TRANSACTION-Definition codieren. Diese anderen Ressourcen haben die Bezeichnung PROGRAM, PROFILE, PARTITIONSET, REMOTESYSTEM und TRANCLASS.

### PROGRAM

In der PROGRAM-Definition geben Sie Optionen an, die sich auf die Softwareimplementierung Ihrer Anwendung beziehen. Anschließend wird das Pro-

gramm definiert, an das die Steuerung zur Verarbeitung der Transaktion übergeben wird. Die TRANSACTION-Definition verweist auf die PROGRAM-Definition.

## PROFILE

Sie müssen nicht für jede Transaktion die Attribute angeben, die die Interaktion mit einem Terminal oder einer logischen Einheit steuern. Stattdessen verweist die TRANSACTION-Definition auf eine PROFILE-Definition, in der diese Angaben für eine Reihe von Transaktionen festgelegt sind.

## REMOTESYSTEM

Für das Transaktionsrouting geben Sie anstelle eines PROGRAM-Namens in der TRANSACTION-Definition den Namen einer REMOTESYSTEM-Ressource an. Als Wert kann der Name eines anderen CICS-Systems angegeben werden, das seinerseits in diesem CICS-System in einer CONNECTION- oder IPCONN-Definition mit dem gleichen Namen definiert ist.

Wenn Sie in einer CONNECTION-Definition eine REMOTESYSTEM-Ressource angeben, dann können Sie auch einen Wert für REMOTESYSTEM angeben, bei dem es sich um den Namen der Transaktion handelt, die auf dem fernen System ausgeführt werden soll. Das ferne System entscheidet, welches Programm die Steuerung erhalten soll.

Wenn Sie einen REMOTESYSTEM-Namen angeben, der dem System entspricht, auf dem die Definition installiert ist, dann installiert CICS eine lokale Transaktionsressource. Wenn der REMOTESYSTEM-Name nicht dem System entspricht, auf dem die Definition installiert ist, dann installiert CICS eine ferne Transaktionsressource. Wird kein REMOTESYSTEM-Name angegeben, dann installiert CICS eine lokale Transaktionsressource.

## TRANCLASS

Dieses Attribut gibt den Namen der Transaktionsklasse an, zu der die Transaktion gehört. Transaktionen, die zu einer Transaktionsklasse gehören, müssen die geltenden Planvorgaben erfüllen, bevor sie ausgeführt werden können. Die Einschränkungen werden in der zugehörigen TRANCLASS-Definition angegeben. Die TRANCLASS-Definition wird in TRANCLASS resources beschrieben.

Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with BAS transaction resource definitions.

## TRANSACTION-Ressourcen in CICS-Bundles

Sie können ein CICS-Bundle verwenden, um eine TRANSACTION-Ressourcendefinition zu erstellen, zu bearbeiten und zu installieren. Wenn Sie eine TRANSACTION-Ressource auf diese Weise erstellen, dann müssen Sie zum Verwalten des Lebenszyklus dieser Ressource das CICS-Bundle verwenden. Die unabhängige Verwaltung der Ressource ist nicht möglich. Sie können eine TRANSACTION-Ressource abfragen, die von einem CICS-Bundle dynamisch generiert wird. Verwenden Sie hierzu den Befehl **EXEC CICS INQUIRE TRANSACTION** oder **CEMT INQUIRE TRANSACTION**. Der Befehl SET oder DISCARD kann jedoch nicht für eine TRANSACTION-Ressource ausgeführt werden, die von einem CICS-Bundle dynamisch generiert wird. Sie müssen die Befehle für die BUNDLE-Ressource absetzen und CICS wendet sie dann auf die TRANSACTION-Ressource an. Sie können eine TRANSACTION-Ressource über ein CICS-Bundle als Anwendungseinstiegspunkt deklarieren.

Einer Transaktion können Arbeiten zugeordnet werden, die zu einem geplanten Zeitpunkt mit dem Befehl **EXEC CICS START** gestartet werden sollen. Wenn der Be-

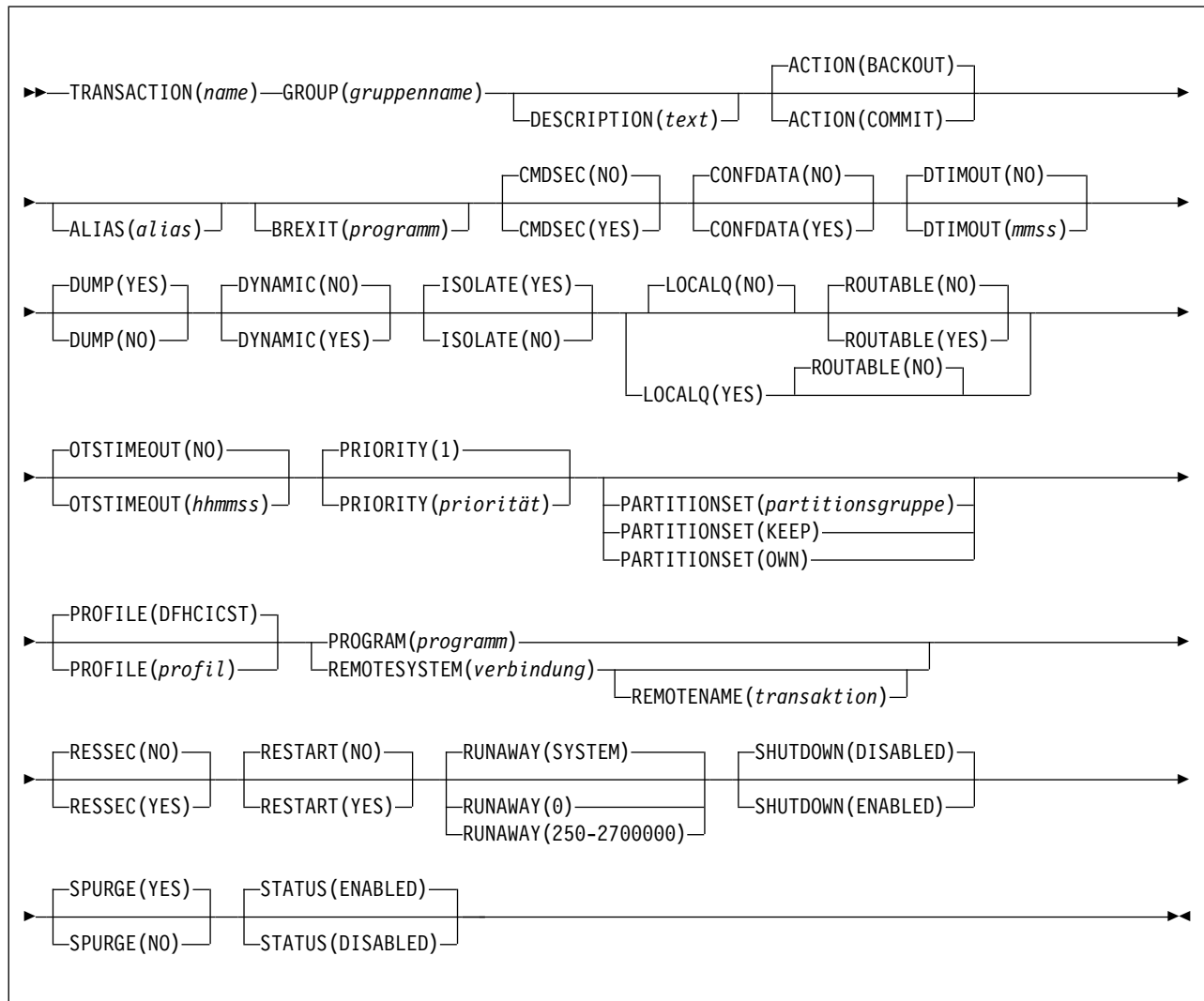


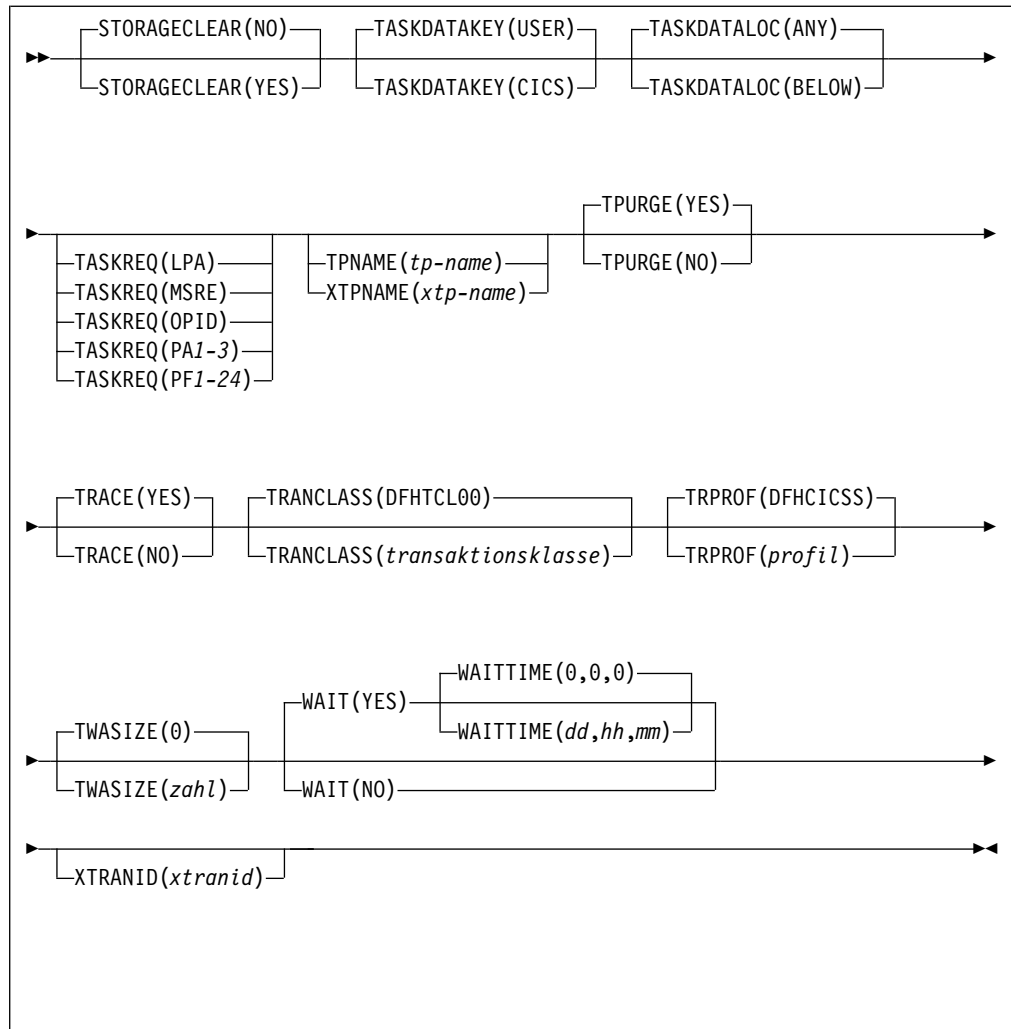
ginn dieser Arbeiten für einen Zeitpunkt nach der Inaktivierung des CICS-Bundles geplant ist, dann werden die geplanten Arbeiten storniert und für diese Transaktion wird das Starten neuer Arbeiten nicht mehr zugelassen. Wird das CICS-Bundle anschließend erneut aktiviert, dann werden die stornierten Arbeiten nicht erneut geplant, sondern verbleiben weiterhin im Status 'Storniert'.

Weitere Informationen zum Definieren von Ressourcen in CICS-Bundles finden Sie in Defining CICS bundles.

## TRANSACTION-Attribute

Beschreibt die Syntax und die Attribute der TRANSACTION-Ressource.





#### **ACTION({BACKOUT|COMMIT})**

Gibt die Aktion an, die ausgeführt werden soll, wenn die Ausführung einer CICS-Region fehlschlägt oder die Konnektivität zum zugehörigen Koordinator während der zweiphasigen Festschreibung verloren geht, nachdem die Arbeitseinheit in den unbestätigten Zeitraum eingetreten ist. Die Aktion richtet sich nach der Angabe im Attribut WAIT. Wird in WAIT die Option YES angegeben, hat ACTION keine Auswirkung, es sei denn, der in WAITTIME angegebene Zeitraum wird überschritten, bevor die Wiederherstellung nach dem Fehler ausgeführt werden kann.

Wenn in WAIT die Option NO angegeben wird, dann wird eine der folgenden Aktionen ausgeführt:

#### **BACKOUT**

Alle Änderungen, die an den wiederherstellbaren Ressourcen ausgeführt wurden, werden zurückgesetzt und die Ressourcen werden in den Status zurückversetzt, in dem sie sich vor dem Beginn der Arbeitseinheit befanden.

#### **COMMIT**

Alle Änderungen, die an den wiederherstellbaren Ressourcen ausgeführt wurden, werden festgeschrieben, und die Arbeitseinheit wird als abgeschlossen markiert.

### **ALIAS**(*alias*)

Gibt den Namen für eine Aliastransaktion für diese Transaktion an. Der Name kann bis zu vier Zeichen lang sein. Diese Option ist nützlich, wenn Sie die Ausführung auf einem Terminal durchführen wollen, für das UCTRAN(NO) definiert wurde, oder wenn es sich um eine Transaktion handelt, die gemischte Groß-/Kleinschreibung bei der Eingabe zulässt (PROFILE UCTRAN(NO)). Sie können z. B. mit ALIAS(abcd) die gleiche Transaktion als ABCD starten.

Wenn Sie eine TRANSACTION-Definition installieren, die das Attribut ALIAS enthält, dann ist das Ergebnis davon abhängig, ob der Aliasname auf dem System bereits verwendet wird:

- Wenn der Aliasname als primäre Transaktions-ID benutzt wird, dann wird das Attribut ALIAS ignoriert.
- Wenn der Aliasname als Alias für eine andere Transaktion benutzt wird, dann wird der ursprüngliche Alias durch den neuen Alias ersetzt. Dies bedeutet, dass der Aliasname nach der Installation der TRANSACTION-Definition auf die neue Transaktion und nicht die ursprüngliche Transaktion verweist.

### **BREXIT**(*programm*)

Definiert den Namen des standardmäßigen Brückenexits, der dieser Transaktion zugeordnet werden soll, wenn sie in der 3270-Brückenumgebung mit einem Befehl **START BREXIT** gestartet wird, in dem kein Name für die Option BREXIT angegeben wurde. Der Name kann bis zu acht Zeichen lang sein.

#### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Wird BREXIT definiert, dann sollten Sie die Attribute REMOTESYSTEM, REMOTENAME, DYNAMIC(YES) und RESTART(YES) nicht angeben, da sie ignoriert werden.

**Anmerkung:** Der Link3270-Mechanismus stellt nun den empfohlenen Weg zur Verwendung der 3270-Brücke dar.

### **CMDSEC**({NO|YES})

Gibt an, ob die Sicherheitsprüfung für die Systemprogrammierungsbefehle angewendet werden soll.

**NO** Es wird keine Prüfung durchgeführt. Die Befehle werden immer ausgeführt.

**YES** Ein Aufruf an den externen Sicherheitsmanager (ESM) wird abgesetzt. CICS erteilt oder verweigert den Zugriff. Wenn der ESM die Ressource oder den Ressourcentyp nicht identifizieren kann, dann wird der Zugriff verweigert.

### **CONFDATA**({NO|YES})

Gibt an, ob CICS Benutzerdaten aus CICS-Traceeinträgen unterdrücken soll, wenn im Systeminitialisierungsparameter CONFDATA die Option HIDEITC angegeben ist. Wenn im Systeminitialisierungsparameter CONFDATA=SHOW angegeben ist, wird die CONFDATA-Angabe in der Transaktionsdefinition ignoriert.

Wenn im Systeminitialisierungsparameter CONFDATA=HIDEITC angegeben ist, sind die folgenden Optionen wirksam:

**NO** CICS unterdrückt die Benutzerdaten nicht. Die Erstbenutzerdaten für z/OS Communications Server und MRO werden am Tracepunkt AP FC92 überwacht. FEPI-Benutzerdaten werden an den normalen CICS-FEPI-Tracepunkten überwacht. IPIC-Benutzerdaten werden an den normalen CICS-IS-Tracepunkten überwacht.

**YES** CICS unterdrückt Benutzerdaten der CICS-Tracepunkte.

**DESCRIPTION**(*text*)

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

**DTIMOUT**(**{NO|*mmss*}**)

Gibt an, ob ein Deadlock-Zeitlimit für die Task angewendet werden soll. Wenn die Task ausgesetzt wird (z. B. aufgrund von Speichermangel), dann wird die Bereinigung der Task eingeleitet, wenn die Task für mehr als die in DTIMOUT angegebene Zeitdauer ausgesetzt bleibt. Wenn die Bereinigung zum Abbruch einer Transaktion führt, dann hängt der verwendete Abbruchcode davon ab, welche Komponente von CICS die Task ausgesetzt hat. Bei Verwendung von CEDF gibt die Benutzertask (sofern möglich) DTIMOUT(NO) oder einen hohen Wert an. Dieser Wert wird auch als Zeitlimitwert in allen RLS-Dateianforderungen verwendet, wenn für DTIMOUT ein Wert ungleich null angegeben ist. Andernfalls erhält die Anforderung den Wert für SIT FTIMEOUT. FTIMEOUT gilt für Transaktionen, die über kein aktives Deadlock-Zeitlimitintervall verfügen. Wenn für das Schlüsselwort DTIMOUT der TRANSACTION-Definition ein Zeitwert angegeben wurde, dann wird dieser Wert für das Dateizeitlimit dieser Transaktion verwendet.

Bei Verwendung von CEDF wird, wenn ein Wert für DTIMOUT für die Benutzertask angegeben wurde, der Wert für DTIMOUT ignoriert, während die Benutzertask ausgesetzt wird und eine CEDF-Task aktiv ist. Aus diesem Grund kann die ausgesetzte Benutzertask nicht mit einer Deadlock-Zeitlimitüberschreitung (Abbruch AKCS) enden, während eine CEDF-Task auf eine Benutzerantwort wartet.

Damit DTIMOUT bei Nichtverwendung von RLS wirksam bleibt, müssen Sie für SPURGE die Option YES setzen.

CICS unterdrückt an bestimmten Punkten Deadlock-Zeitlimitüberschreitungen.

DTIMOUT wird für Terminal-E/A-Wartestatus nicht ausgelöst. Da die Relay-Transaktion nach dem Abrufen einer Sitzung nicht auf Ressourcen zugreift, benötigt sie DTIMOUT nur zum Abfangen ausgesetzter Zuordnungsanforderungen. Für E/A-Wartestatus in einer Sitzung kann allerdings das Attribut RTIMOUT in PROFILE-Definitionen für das Transaktionsrouting in IPIC- oder MRO-Sitzungen und zugeordneten APPC-Verbindungen angegeben werden.

Sie müssen bestimmte Transaktionen mit einem Wert für DTIMOUT definieren, weil das Deadlock-Zeitlimit der Mechanismus ist, den CICS zur Verarbeitung von SOS-Situationen (SOS = Short-on-storage) verwendet.

**NO** Die Funktion für Deadlock-Zeitlimits ist nicht erforderlich.

*mmss* Die Zeitdauer (MMSS für Minuten und Sekunden), nach deren Ablauf

das Deadlock-Zeitlimit eine ausgesetzte Task beendet. Der Maximalwert, den Sie angeben können, sind 68 Minuten. Dieser Wert ist auf eine Sekunde genau.

#### **DUMP({YES|NO})**

Gibt an, ob ein Aufruf an die Speicherauszugsdomäne abgesetzt werden soll, um einen Transaktionsspeicherauszug zu generieren, wenn die Transaktion abnormal beendet wird.

Dieser Operand hat bei den folgenden Speicherauszugsoperationen keine Auswirkungen:

- Beim Befehl **EXEC CICS DUMP**, mit dem immer ein Speicherauszug generiert wird.
- Bei Systemspeicherauszügen für die Speicherauszugscode AP0001 und SR0001, die von CICS für einen Abbruch des Typs ASRA, ARSB oder ASRD generiert werden. Wenn Sie für das Transaktionsattribut DUMP die Einstellung NO angeben, dann unterdrückt CICS den Transaktionsspeicherauszug, nicht jedoch den Systemspeicherauszug.

**YES** CICS ruft die Speicherauszugsdomäne auf, um einen Transaktionsspeicherauszug zu generieren. Beachten Sie hierbei, dass die abschließende Entscheidung zum Generieren oder Unterdrücken des Transaktionsspeicherauszugs von der Transaktionsspeicherauszugstabelle gesteuert wird. Weitere Informationen zur Speicherauszugstabelle finden Sie in *The dump code options you can specify*.

Wenn für den angegebenen Speicherauszugscode kein Eintrag in der Transaktionsspeicherauszugstabelle existiert, wenn eine Transaktion abgebrochen wird, dann erstellt CICS einen temporären Eintrag, für den standardmäßig ein Transaktionsspeicherauszug erstellt wird.

Die Einträge der Transaktionsspeicherauszugstabelle werden mit dem Befehl **CEMT SET TRDUMPCODE** oder dem SPI-Befehl **SET TRANDUMPCODE** gesteuert.

**NO** Das System setzt keinen Aufruf an die Speicherauszugsdomäne ab und unterdrückt alle potenziellen Transaktionsspeicherauszüge.

#### **DYNAMIC({NO|YES})**

Gibt an, ob die Transaktion mit der CICS-Funktion für das dynamische Transaktionsrouting dynamisch an eine ferne Region weitergeleitet werden kann.

**NO** Erstellt eine lokale oder ferne Definition entsprechend der Angabe im Attribut REMOTESYSTEM.

**YES** Erlaubt dem Programm für das dynamische Transaktionsrouting die dynamische Ermittlung des lokalen oder fernen Status während des Aufrufs. Programmierinformationen zum Programm für das dynamische Transaktionsrouting finden Sie in *Writing a dynamic routing program*.

#### **EXTSEC**

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt.

#### **GROUP(gruppenname)**

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

**INDOUBT**

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt.

**ISOLATE({YES|NO})**

Gibt an, ob CICS den Benutzerschlüsselspeicher für die Lebensdauer der Task für die Transaktion isolieren soll, um einen Schutz zwischen Transaktionen sicherzustellen. (Eine Beschreibung des Benutzerschlüsselspeichers finden Sie unter dem Attribut TASKDATAKEY.) Isolation bedeutet, dass der Benutzerschlüsselspeicher für die Lebensdauer der Task sowohl gegen Lese- als auch Schreibzugriffe durch die Benutzerschlüsselprogramme anderer Transaktionen geschützt wird. Dies sind die Programme, die mit der Einstellung EXECKEY(USER) definiert wurden. In der Abbildung Auswirkung des Attributs ISOLATE auf den Speicherzugriff wird dargestellt, wie sich die Angabe des Attributs ISOLATE auswirkt.

**Anmerkung:**

1. Das Attribut ISOLATE bietet keinen Schutz gegen Anwendungsprogramme, die im CICS-Schlüssel ausgeführt werden. Dies sind Programme, die mit der Einstellung EXECKEY(CICS) definiert werden.
2. Nicht gemeinsam genutzte VSAM-Ressourcen (NSR = Nonshared Resources) werden für Transaktionen, die die Transaktionsisolation verwenden, nicht unterstützt. ISOLATE(NO) muss angegeben werden, wenn Sie Transaktionen definieren wollen, die über NSR auf VSAM-Dateien zugreifen. Sie können die Dateianforderung auch über eine Funktionsverlagerung an eine ferne Region senden. Das Programm DFHMIRS, das die Anforderung ausführt, wird mit der EXECKEY-Angabe CICS definiert. Ein CICS-Schlüsselprogramm hat Lese- und Schreibzugriff auf den CICS-Schlüsselspeicher und den Benutzerschlüsselspeicher der eigenen Task und aller anderen Tasks. Dies gilt unabhängig davon, ob die Transaktionsisolation aktiv ist.
3. Die Transaktionsisolation gilt nicht für 64-Bit-Speicher.

**YES** Der Benutzerschlüsselspeicher für die Lebensdauer der Task für die Transaktion wird gegen die Benutzerschlüsselprogramme aller anderen Transaktionen geschützt. Dies sind die Programme, die mit der Einstellung EXECKEY(USER) definiert wurden. Für Programme, die mit EXECKEY(CICS) definiert wurden, erfolgt keine Isolation.

Darüber hinaus wird der Benutzerschlüsselspeicher für die Lebensdauer der Task für *alle* anderen Transaktionen *gegen* die Benutzerschlüsselprogramme geschützt, deren Transaktionen mit der Einstellung ISOLATE(YES) definiert wurden.

**NO** Wenn Sie ISOLATE(NO) angeben, wird der Speicher für die Lebensdauer der Task der Transaktion gegen die Benutzerschlüsselprogramme der Transaktionen isoliert, die mit der Einstellung ISOLATE(YES) definiert wurden. Der Transaktionsspeicher wird allerdings nicht gegenüber Benutzerschlüsselprogrammen anderer Transaktionen isoliert, für

die ebenfalls ISOLATE(NO) angegeben wurde, weil die Transaktionen mit dieser Option alle dem allgemeinen Unterbereich zugeordnet werden.

Darüber hinaus ist zu beachten, dass der Benutzerschlüsselspeicher für die Lebensdauer der Task für alle Transaktionen, die mit ISOLATE(YES) definiert wurden, *gegen* die Benutzerschlüsselprogramme von Transaktionen geschützt wird, die mit der Einstellung ISOLATE(NO) definiert wurden.

Geben Sie ISOLATE(NO) für diejenigen Transaktionen an, die bestimmte Komponenten ihres Benutzerschlüsselspeichers für die Lebensdauer der Task gemeinsam nutzen.

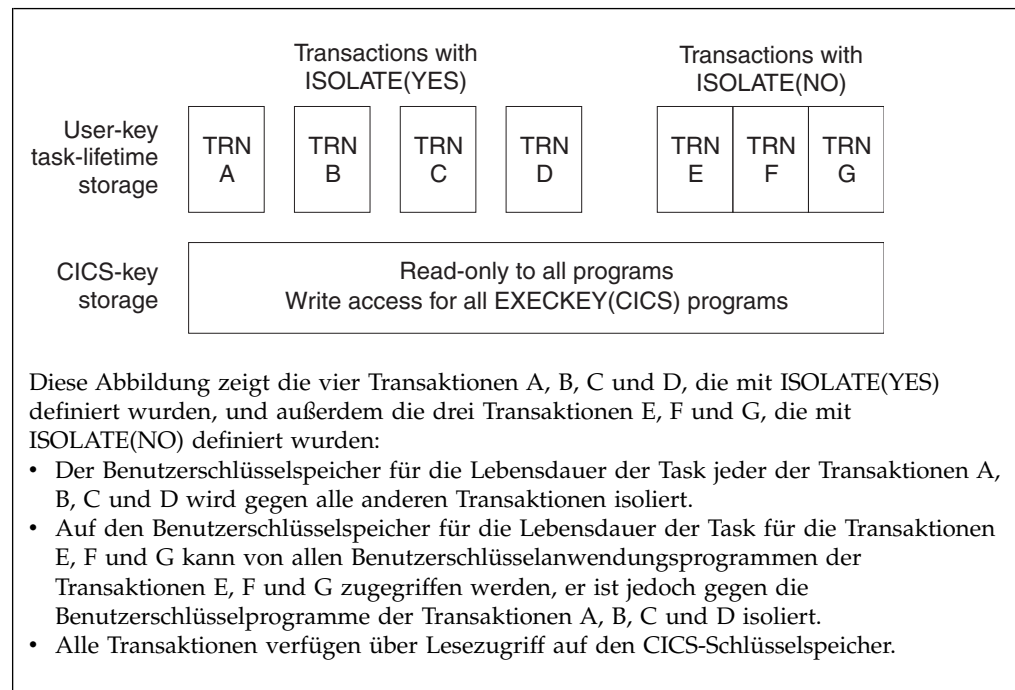


Abbildung 5. Auswirkung des Attributs ISOLATE auf den Speicherzugriff

#### LOCALQ({NO|YES})

Gibt an, ob die Einstellung in die Warteschlange auf dem lokalen System (lokale Warteschlangensteuerung) ausgeführt werden soll.

**NO** Die lokale Warteschlangensteuerung wird nicht ausgeführt.

**YES** Die lokale Warteschlangensteuerung kann für eine Anforderung mit einem Befehl START mit der Option NOCHECK versucht werden, wenn das System oder die IPIC-Verbindung nicht verfügbar ist und der Systemname gültig ist.

Ein System wird als nicht verfügbar definiert, wenn die folgenden Bedingungen zutreffen:

- Das System ist im Status OUT OF SERVICE (Außer Betrieb), wenn die Anforderung eingeleitet wird.
- Der Versuch zum Initialisieren einer Sitzung auf dem fernen System schlägt fehl und die Korrekturaktion, die vom Programm für abnormale Bedingungen (DFHZNAC) oder vom Knotenfehlerprogramm

(DFHZNEP) ausgeführt wird, besteht darin, das System in den Status OUT OF SERVICE zu versetzen.

- Für das ferne System stehen nicht sofort Sitzungen zur Verfügung und das Programm für den globalen Benutzerexit XISCONA gibt an, dass die Anforderung nicht in der absetzenden Region in die Warteschlange eingestellt werden soll.

Eine IPIC-Verbindung wird als nicht verfügbar definiert, wenn die folgenden Bedingungen zutreffen:

- Die IPIC-Verbindung wird nicht angefordert.
- Es ist keine Sitzung verfügbar und CICS stellt die Anforderung für eine neue Sitzung nicht in die Warteschlange ein.

Verwenden Sie die lokale Warteschlangensteuerung nur für die START-Befehle, die für zeitunabhängige Anforderungen stehen. Die Verzögerung, die durch die lokale Warteschlangensteuerung impliziert wird, wirkt sich auf den Zeitpunkt aus, zu dem die Anforderung gestartet wird. Sie sind dafür verantwortlich sicherzustellen, dass diese Bedingung erfüllt wird.

Wenn Sie LOCALQ(YES) angeben, dann können Sie nicht ROUTABLE(YES) festlegen.

Sie können den globalen Benutzerexit XISLCLQ oder XISQLCL verwenden, um die Einstellung des Attributs LOCALQ zu überschreiben. Programmierinformationen zu den Benutzerexits im Programm für die systemübergreifende Kommunikation finden Sie in Global user exit points.

#### **OTSTIMEOUT** ({NO|*hhmmss*})

Gibt in Stunden, Minuten und Sekunden die Zeitdauer an, für die eine OTS-Transaktion (OTS = Object Transaction Service) die in einer Enterprise-Bean-Umgebung erstellt wurde und als Task unter dieser CICS-Transaktion ausgeführt wird, aktiv sein darf, bevor der Initiator der OTS-Transaktion einen Synchronisationspunkt verwenden oder für die Transaktion eine Rollback-Operation ausführen muss. Wenn die angegebene Zeitdauer abgelaufen ist, dann bereinigt CICS die Task.

Der Initiator der OTS-Transaktion kann wie folgt lauten:

- Der Client der Enterprise-Bean.
- Der EJB-Container. Der Container gibt einen Synchronisationspunkt zum Abschluss der Bean-Methode aus.
- Eine Session-Bean, die ihre eigenen OTS-Transaktionen verwaltet.

Methoden von Session-Beans, die eigene OTS-Transaktionen verwalten, können den standardmäßigen Zeitlimitwert mithilfe der Methode `setTransactionTimeout` der Schnittstelle `javax.Transaction.UserTransaction` überschreiben.

**NO** Für OTS-Transaktionen treten keine Zeitlimitüberschreitungen auf. Der Standardwert ist NO.

*hhmmss*

Die Zeitdauer (im Format HHMMSS), bevor die Task bereinigt wird. Die maximale Zeitdauer beträgt 24 Stunden (240000).

#### **PARTITIONSET** ({*partitionsgruppe*|**KEEP**|**OWN**})

Gibt den Namen der Partitionsgruppe an, die als standardmäßige Partitionsgruppe der Anwendung benutzt werden soll. Der Name kann bis zu acht Zeichen lang sein.



**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Wenn Sie keinen Namen für die Partitionsgruppe angeben oder aber einen der reservierten Namen, dann macht CICS die vorhandenen Partitionen unbrauchbar, bevor die erste BMS-Ausgabe an das Terminal durch die Transaktion abgesetzt wird.

*partitionsgruppe*

CICS macht vorhandene Partitionen unbrauchbar und lädt die angegebene Partitionsgruppe, bevor die erste BMS-Ausgabe an das Terminal von der Transaktion abgesetzt wird. (Vorhandene Partitionen werden nicht unbrauchbar gemacht, wenn die Terminalpartitionsgruppe mit der Partitionsgruppe der Anwendung übereinstimmt.)

Dieser Name darf nicht mit dem Wert identisch sein, der in PROGRAM(*name*) angegeben wurde.

**KEEP** Die Transaktion verwendet die Anwendungspartitionsgruppe für dieses Terminal. Diese Option wird in der Regel für nachfolgende Transaktionen in einer Kette von Pseudodialogtransaktionen verwendet.

**OWN** Die Transaktion führt eine eigene Partitionsverwaltung aus.

**PRIMESIZE**

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt.

**PRIORITY**(({1|*priorität*})

Gibt die Transaktionspriorität an. Dieser ein- bis dreistellige Dezimalwert zwischen 0 und 255 wird verwendet, um die Gesamtpriorität für die Transaktionsverarbeitung festzulegen. Die Priorität für die Transaktionsverarbeitung entspricht der Summe der Terminalpriorität, Transaktionspriorität und Bedienerpriorität und darf den Wert 255 nicht überschreiten. Je höher die Nummer ist, um so höher ist die Priorität.

**PROFILE**(({DFHCICST|*profil*})

Der Name der PROFILE-Definition, die die Verarbeitungsoptionen angibt, die mit dem Terminal verwendet werden, das die Transaktion eingeleitet hat.

**Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % &amp; ? ! : | " = ~ , ; &lt; &gt;

Der Standardwert ist DFHCICST.

Die mit dem Standardwert DFHCICST bereitgestellten Verarbeitungsoptionen werden in PROFILE definitions in group DFHISC dargestellt. DFHCICST eignet sich nicht für DPL (Distributed Program Link). Stattdessen sollten Sie dort DFHCICSA mit der Einstellung INBFMH=ALL angeben.

**PROGRAM**(*programm*)

Gibt den Namen des Programms an, dem CICS die Steuerung zur Verarbeitung dieser Transaktion übergibt. Der Name kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Vergewissern Sie sich, dass dieser Name nicht mit dem Namen übereinstimmt, der in PARTITIONSET(name) angegeben wurde.

Wenn für REMOTESYSTEM ein Name angegeben wurde und wenn dieser Name vom Namen für das aktuelle System abweicht, dann müssen Sie keinen Namen für PROGRAM angeben. Wenn Sie in diesem Fall einen Namen für PROGRAM angeben, dann wird dieser Name möglicherweise ignoriert.

Wenn diese Transaktionsdefinition zur Verwendung in einer Verknüpfungsanforderung eines fernen Programms verwendet wird, dann muss der von Ihnen in diesem Attribut angegebene Name mit dem Namen des CICS-Spiegelprogramms DFHMIRS identisch sein. Weitere Informationen hierzu finden Sie unter dem Attribut TRANSID in der PROGRAM-Definition in PROGRAM attributes.

**REMOTENAME(transaktion)**

Gibt den Namen dieser Transaktion in dem auf dem fernen System bekannten Format an, wenn die Ausführung auf einem fernen System oder in einer entsprechenden Region über die systemübergreifende Kommunikation stattfinden soll. Das ferne System kann hierbei eine andere CICS-Region oder ein IMS-System sein. Der Wert für REMOTENAME kann zwischen einem und vier Zeichen lang sein, wenn im Attribut REMOTESYSTEM eine weitere CICS-Region angegeben ist, oder zwischen einem und acht Zeichen, wenn in REMOTESYSTEM ein IMS-System angegeben ist. IMS verwendet aus acht Zeichen bestehende Namen. Wenn in REMOTENAME weniger als acht Zeichen angegeben werden, setzt IMS diesen Wert in ein verwendbares Format um.

**Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Wenn Sie REMOTESYSTEM angeben, REMOTENAME jedoch nicht, dann wird für das Attribut REMOTENAME standardmäßig der lokale Name verwendet, also der TRANSACTION-Name in dieser Definition. Beachten Sie hierbei, dass die Transaktion sich nicht auf dem fernen System bzw. in der fernen Region befinden muss.

**REMOTESYSTEM(verbindung)**

Gibt den Namen an, der die Verbindung für die übergreifende Kommunikation identifiziert, über die die Verbindungsanforderung der Transaktion gesendet wird.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

In REMOTESYSTEM ist eine der folgenden Kennungen definiert:

- Für IPIC-Verbindungen gibt REMOTESYSTEM die ersten vier Zeichen des IPCONN-Namens in der IPCONN-Definition an, die zur Verfügung steht und angefordert werden kann.

- Für MRO- und APPC-Verbindungen gibt REMOTESYSTEM den CONNECTION-Namen in der CONNECTION-Definition an.

REMOTESYSTEM wird für die CICS-Funktionsanforderungsverlagerung (asynchrone Verarbeitung und Transaktionsrouting) verwendet.

Vergewissern Sie sich, dass die REMOTESYSTEM-Optionen, die für eine Gruppe von Transaktionen angegeben werden, nicht auf andere Transaktionen in dieser Gruppe verweisen. Dies bedeutet, dass diese nicht auf sich selbst verweisen dürfen, was zum Starten von Schleifen für mehrere Spiegeltransaktionen führen kann.

#### **RESSEC({NO|YES})**

Gibt an, ob die Ressourcensicherheitsprüfung für Ressourcen verwendet werden soll, auf die durch diese Transaktion zugegriffen wird.

**NO** Alle Ressourcen stehen allen Benutzern zur Verfügung, die über die Berechtigung zur Verwendung dieser Transaktion verfügen.

**YES** Es wird ein externer Sicherheitsmanager verwendet. Weitere Einzelheiten zur externen Sicherheitsprüfung finden Sie in Security of resource definitions.

#### **RESTART({NO|YES})**

Gibt an, ob die Funktion für den Neustart von Transaktionen verwendet werden soll, um die Tasks erneut zu starten, die abnormal beendet wurden und später von der Funktion für das dynamische Zurücksetzen von Transaktionen zurückgesetzt werden.

Wird RESTART(YES) angegeben, dann wird die fehlgeschlagene Task ab dem Anfang des Startprogramms neu gestartet. Wenn das dynamische Zurücksetzen von Transaktionen fehlschlägt oder wenn der Neustart dynamisch unterdrückt wird, dann wird DFHPEP normal aufgerufen. Die Funktion für den Neustart von Transaktionen ist beispielsweise in Situationen besonders nützlich, in denen ein Deadlock bei der Programmisolation auftritt und die Task automatisch neu gestartet werden kann, anstatt manuell erneut übergeben werden zu müssen. Eine terminalgesteuerte Transaktion kann während des CICS-Systemabschlusses neu gestartet werden. Dies gilt auch dann, wenn SHUTDOWN(DISABLED) angegeben wurde. Weitere Einzelheiten zum automatischen Neustart von Transaktionen finden Sie in Splitting the application into transactions.

**NO** Die Neustartfunktion ist nicht erforderlich.

**YES** Die Neustartfunktion wird verwendet.

#### **ROUTABLE({NO|YES})**

Gibt an, ob eine Transaktion bei Angabe in einem auswählbaren Befehl **EXEC CICS START** mit der erweiterten Routing-Methode weitergeleitet werden soll.

**NO** Wenn die Transaktion in einem Befehl START aufgeführt wird, dann wird sie mithilfe der traditionellen Methode weitergeleitet.

**YES** Wenn die Transaktion in einem auswählbaren Befehl START aufgeführt wird, dann wird sie mithilfe der erweiterten Methode weitergeleitet.

Wenn Sie ROUTABLE(YES) angeben, dann können Sie nicht LOCALQ(YES) festlegen.

Detaillierte Informationen zur erweiterten und zur traditionellen Methode beim Routing von Transaktionen, die mit dem Befehl **EXEC CICS START** aufgerufen werden können, finden Sie in Routing transactions invoked by START commands.

## RSL

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt.

## **RUNAWAY**(**{SYSTEM|0|250-2700000}**)

Die Zeitdauer (in Millisekunden), für die eine unter dieser Transaktionsdefinition ausgeführte Task die Steuerung des Prozessors übernehmen kann, bevor das System davon ausgeht, dass sie in einen nicht mehr steuerbaren Zustand (logische Schleife) eingetreten ist. Nach Ablauf dieses Zeitintervalls kann CICS die Task abnormal beenden.

### **SYSTEM**

CICS verwendet zur Angabe des Zeitlimits für die Einstufung einer Transaktion als nicht mehr steuerbar den Wert des Systeminitialisierungsparameters ICVR.

**0** Es gilt kein Zeitlimit und für die Transaktion ist keine Erkennung von nicht mehr steuerbaren Tasks erforderlich.

### **250-2700000**

Das Zeitlimit für nicht mehr steuerbare Tasks liegt im Bereich zwischen 250 und 2700000. Bei der Überprüfung einer Task auf einen nicht mehr steuerbaren Zustand rundet CICS den von Ihnen angegebenen Wert auf ein Vielfaches von 250 nach unten ab.

## **SHUTDOWN**(**{DISABLED|ENABLED}**)

Gilt für alle Transaktionen und gibt an, ob die Transaktion während des Systemabschlusses von CICS ausgeführt werden kann. Diese Option ergänzt die Option XLT des Befehls **PERFORM SHUTDOWN**. Damit eine Transaktion während des Systemabschlusses verbunden werden kann, muss für sie entweder SHUTDOWN(ENABLED) definiert werden oder sie muss (bei terminalbasierten Transaktionen) in der Transaktionslistentabelle (XLT) aufgeführt werden, die im Befehl **PERFORM SHUTDOWN** angegeben ist.

### **DISABLED**

Die Transaktion wird während des Systemabschlusses von CICS inaktiviert.

### **ENABLED**

Die Transaktion wird zur Ausführung während des Systemabschlusses von CICS aktiviert.

## **SPURGE**(**{YES|NO}**)

Gibt an, ob die Transaktion zu Beginn als **vom System bereinigbar** eingestuft wird.

Mit der Angabe SPURGE=NO wird verhindert, dass eine Transaktion wie folgt bereinigt werden kann:

- Deadlock-Zeitlimit (DTIMOUT)
- Befehl **EXEC CICS ... PURGE**
- Definieren von TWAOC (Abbrechen einer Task) im Knotenfehlerprogramm (NEP = Node Error Program)
- Befehl **CEMT SET ... PURGE**

Mit der Angabe SPURGE=YES können solche Bereinigungen in Bezug auf den Benutzer fortgesetzt werden. CICS kann die Bereinigung allerdings verhindern, wenn die Bereinigung in der aktuellen Phase der Transaktion mit Risiken verbunden ist.

Beachten Sie hierbei, dass mit der Einstellung SPURGE=NO die Bereinigung einer Transaktion mit der Funktion für Lesezeitlimits (RTIMOUT), mit dem Befehl EXEC CICS SET ... FORCEPURGE oder dem Befehl CEMT SET TRANSACTION(tranid) FORCEPURGE nicht verhindert wird. Mit SPURGE wird lediglich der Anfangswert ermittelt, der von der Transaktion während der Ausführung geändert werden kann.

**YES** Die Transaktion wird zu Beginn als vom System bereinigbar eingestuft.

**NO** Die Transaktion wird zu Beginn nicht als vom System bereinigbar eingestuft.

#### **STATUS({ENABLED|DISABLED})**

Gibt den Transaktionsstatus an.

##### **ENABLED**

Ermöglicht die normale Ausführung der Transaktion.

##### **DISABLED**

Verhindert, dass die Transaktion ausgeführt wird.

Dieses Attribut wird für TRANSACTION-Ressourcen ignoriert, die dynamisch von einem CICS-Bundle generiert werden. Der Anfangsstatus einer TRANSACTION-Ressource wird aus dem Anfangsstatus des Bundles abgeleitet, das die Ressource definiert.

#### **STORAGECLEAR({NO|YES})**

Gibt an, ob der Speicher für die Lebensdauer der Task für diese Transaktion bei seiner Freigabe bereinigt werden soll. Verwenden Sie das Attribut STORAGECLEAR, um zu verhindern, dass andere Tasks versehentlich vertrauliche oder sensible Daten anzeigen können, die von dieser Transaktion im Speicher für die Lebensdauer der Task gespeichert wurden.

#### **TASKDATAKEY({USER|CICS})**

Gibt den Speicherschlüssel des Speichers an, den CICS bei der Initialisierung einer Task für die Laufzeit der Task (Speicher für die Lebensdauer der Task) zuordnet und auf den von der Anwendung zugegriffen werden kann. Diese Speicherbereiche sind der EIB (EXEC Interface Block; EXEC-Schnittstellenblock) und der TWA (Transaction Work Area; Transaktionsarbeitsbereich).

In TASKDATAKEY wird außerdem der Schlüssel des Speichers angegeben, den CICS für alle Programme anfordert, die unter der Transaktion ausgeführt werden. Der programmbezogene Speicher, der von CICS im angegebenen Schlüssel zugeordnet wird, umfasst die folgenden Elemente:

- Die Kopien des Arbeitsspeichers, die CICS für jede Ausführung eines Anwendungsprogramms anfordert.
- Den Speicher, den CICS für das Programm in Beantwortung impliziter und expliziter GETMAIN-Anforderungen anfordert. Das Programm kann beispielsweise Speicherplatz mit dem Befehl GETMAIN oder GETMAIN64 anfordern oder als Ergebnis der Angabe der Option SET in anderen CICS-Befehlen.

Sie müssen TASKDATAKEY(USER) angeben, wenn Programme innerhalb der Transaktion mit EXECKEY(USER) definiert wurden. Wenn Sie für eine Transaktion TASKDATAKEY(CICS) angeben, dann führt der Versuch, ein Programm im Benutzerschlüssel unter dieser Transaktion auszuführen, zum Abbruch der Task mit dem Abbruchcode AEZD.

**USER** CICS fordert den Benutzerschlüsselspeicher für diese Transaktion an.

Anwendungsprogramme, die in einem beliebigen Schlüssel ausgeführt werden, können in diesen Speicherbereichen sowohl Lese- als auch Änderungsoperationen ausführen.

Benutzerschlüsselprogramme von Transaktionen, die mit der Angabe ISOLATE(YES) definiert wurden, verfügen nur für die eigenen Tasks über Zugriff auf den Benutzerschlüsselspeicher für die Lebensdauer der Task.

Benutzerschlüsselprogramme für Transaktionen, die mit ISOLATE(NO) definiert wurden, verfügen außerdem auch für andere Tasks, die mit ISOLATE(NO) definiert wurden, über Zugriff auf den Benutzerschlüsselspeicher für die Lebensdauer der Task.

Weitere Informationen zum Schutz des Taskspeichers finden Sie in der Beschreibung zum Attribut EXECKEY in der PROGRAM-Definition.

**CICS** CICS fordert den CICS-Schlüsselspeicher für diese Transaktion an. Anwendungsprogramme, die im CICS-Schlüssel ausgeführt werden, können in diesen Speicherbereichen sowohl Lese- als auch Änderungsoperationen ausführen. Anwendungsprogramme, die im Benutzerschlüssel ausgeführt werden, können in diesen Speicherbereichen nur Leseoperationen ausführen.

#### **TASKDATALOC({ANY|BELOW})**

Gibt an, ob der Speicher für die Lebensdauer der Task, der von CICS für die Dauer der Transaktion angefordert wird, oberhalb der 16-MB-Grenze im virtuellen Speicher vorhanden sein darf. Diese Bereiche, die speziellen CICS-Tasks zugeordnet sind, umfassen den EIB (EXEC Interface Block; EXEC-Schnittstellenblock) und den TWA (Transaction Work Area; Transaktionsarbeitsbereich).

Sie müssen TASKDATALOC(BELOW) angeben, wenn Programme innerhalb der Transaktion im 24-Bit-Adressierungsmodus ausgeführt werden. Diese Einschränkung gilt auch für taskbezogene Benutzerexits, die für die Transaktion ausgeführt werden.

Für Transaktionen, die diese Bedingungen nicht erfüllen, können Sie ANY angeben, um die entsprechende Entlastung des virtuellen Speichers zu erreichen.

CICS überwacht die Verwendung von TASKDATALOC(ANY) und hierbei insbesondere die folgenden Aktionen:

- Ein Versuch zum Aufrufen eines AMODE 24-Programms (AMODE = Addressing Mode; Adressierungsmodus), das unter einer Transaktion ausgeführt wird, die mit TASKDATALOC(ANY) definiert wurde, führt zu einem Abbruch des Typs AEZC.
- Der Versuch, einen Befehl EXEC CICS abzusetzen oder einen taskbezogenen Benutzerexit während der Ausführung von AMODE(24) mit TASKDATALOC(ANY) aufzurufen, führt zu einem Abbruch des Typs AEZA.
- Ein AMODE 31-Programm, das als Transaktion mit TASKDATALOC(ANY) ausgeführt wird, und versucht, einen taskbezogenen Benutzerexit aufzurufen, bei dem die Ausführung von AMODE(24) erzwungen wird, führt zu einem Abbruch des Typs AEZB.
- Wenn ein taskbezogener Benutzerexit, dessen Ausführung in AMODE 24 erzwungen wird, für den Taskstart aktiviert wird, dann erzwingt CICS für alle Transaktionen die Verwendung von TASKDATALOC(BELOW) für die verbleibende Zeit der CICS-Ausführung.

**ANY** Speicherbereiche, die von CICS für die Transaktion angefordert werden, können im virtuellen Speicher oberhalb der 16-MB-Grenze liegen.

## BELOW

Speicherbereiche, die von CICS für die Transaktion angefordert werden, müssen unterhalb der 16-MB-Grenze liegen.

### TASKREQ(*wert*)

Gibt an, ob eine Transaktion durch Drücken einer Funktionstaste (F oder PF), durch Verwendung eines Lichtstifts oder durch Verwendung einer Karte eingeleitet werden soll. Folgende Werte sind zulässig:

- Programmabruftaste PA1, PA2 oder PA3.
- Funktionstasten PF1 bis PF24.
- OPID für den Leser für Operator-ID-Karten.
- LPA für ein Feld mit Lichtstifterkennung auf einer 3270-Einheit.
- MSRE für den 10/63-Zeichen-Magnetcodeleser.

Bei PA- und PF-Tasten muss Folgendes beachtet werden:

- Wenn eine PA- oder PF-Taste im Systeminitialisierungsparameter PRINT angegeben wird, dann darf dieselbe PF-Taste nicht für TASKREQ zur Einleitung einer Transaktion verwendet werden.
- PA- oder PF-Tasten, die im Systeminitialisierungsparameter SKRxxxx als Seitenabrufstasten angegeben werden, werden während einer Seitenabrufsitung als solche eingestuft. Zu einem anderen Zeitpunkt können Sie die gleichen Tasten verwenden, um Transaktionen einzuleiten. Definieren Sie die Schlüssel mit den folgenden Werten:

```
TASKREQ=KEY-ID  
PROGRAM=DFHTPR  
TWSIZE=1024  
TPURGE=NO  
SPURGE=NO
```

- Wenn Sie eine Transaktion mit PROGRAM(DFHTPR) definieren und eine Taste für TASKREQ festlegen, dann wird mit der Taste die Transaktion eingeleitet und gleichzeitig die Seitenabrufsitung geöffnet.

### TCLASS

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt.

### TPNAME(*name*)

Gibt den Namen der Transaktion an, die von einem APPC-Partner verwendet werden kann, wenn die Längenbegrenzung von vier Zeichen für das Attribut TRANSACTION eine zu große Einschränkung bedeutet. Dieser Name kann bis zu 64 Zeichen lang sein.

#### Zulässige Zeichen:

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Wenn dieser Zeichenbereich für einen Namen, den Sie angeben wollen, nicht ausreicht, dann können Sie anstelle von TPNAME das Attribut XTPNAME verwenden.

### TPURGE({YES|NO})

Gibt nur für Nicht-z/OS Communications Server-Terminals an, ob die Transaktion aufgrund eines Terminalfehlers bereinigt werden kann.

**YES** Die Task kann bereinigt werden, wenn ein Terminalfehler auftritt.

**NO** Die Task kann nicht bereinigt werden, wenn ein Terminalfehler auftritt. In diesem Fall ist ein manueller Eingriff des Hauptterminalbedieners erforderlich.

### **TRACE({YES|NO})**

Gibt an, ob die Aktivität dieser Transaktion mit Trace protokolliert werden soll.

**YES** Die Aktivität für diese Transaktion soll mit Trace protokolliert werden.

**NO** Die Aktivität dieser Transaktion soll nicht mit Trace protokolliert werden.

Die von CICS bereitgestellten Transaktionsdefinitionen für CEDF und CSGM geben TRACE(NO) an.

### **TRANCLASS(DFHTCL00|transaktionsklasse)**

Gibt den Namen der Transaktionsklasse an, zu der die Transaktion gehört. Transaktionen, die zu einer Transaktionsklasse gehören, müssen die geltenden Planvorgaben erfüllen, bevor sie ausgeführt werden können. Der reservierte TRANCLASS-Name DFHTCL00 wird verwendet, um anzugeben, dass die Transaktion zu keiner Transaktionsklasse gehört.

Wenn eine Transaktion ausgeführt wird und die zugehörige TRANCLASS-Definition nicht installiert ist, wird die Transaktion ohne die in TRANCLASS angegebenen Planvorgaben ausgeführt. Die Nachricht DFHXM0212 wird als Warnung ausgegeben.

Der Wert für TRANCLASS kann bis zu acht Zeichen lang sein.

#### **Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

### **TRANSACTION(name)**

Gibt den Namen der Transaktion oder die Transaktions-ID (TRANSID) an. Der Name kann bis zu vier Zeichen lang sein.

#### **Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Sie sollten keine Namen verwenden, die mit dem Buchstaben C beginnen, weil dieses Zeichen zur Verwendung durch CICS reserviert ist. CICS kann solche Ressourcen löschen oder verwerfen und durch CICS-Ressourcen ersetzen, wenn Sie ein Upgrade durchführen oder neue Funktionen oder Fixes bereitstellen.

#### **Anmerkung:**

1. Wenn Sie in einem Namen ein Komma (,) verwenden, dann können die folgenden Befehle nicht verwendet werden:

```
CEMT INQUIRE TRANSACTION(wert1,wert2)
CEMT SET      TRANSACTION(wert1,wert2)
```

Hierbei dient das Komma als Listenbegrenzer. Informationen zur Verwendung von Listen mit Ressourcen-IDs finden Sie in List of resource identifiers.

2. Wenn Sie die Warteschlangen mit transienten Daten mithilfe von RACF schützen, dann vermeiden Sie im Namen die Verwendung der Zeichen % und &. In den RACF-Befehlen haben diese Zeichen eine Sonderbedeutung, wenn Sie in Profilnamen verwendet werden. Weitere Informationen hierzu finden Sie in Transaction security.

Wenn Sie andere Sonderzeichen in einer Transaktions-ID verwenden wollen, dann benutzen Sie das Attribut XTRANID zur Angabe eines anderen Namens,



der zum Einleiten der Transaktion verwendet wird. Sie müssen auch einen TRANSACTION-Namen angeben, weil dieser Name verwendet wird, um die TRANSACTION-Definition in der CSD-Datei zu identifizieren.

Beim Definieren einer Transaktion müssen Sie auch einen Wert für PROGRAM oder REMOTESYSTEM angeben.

#### TRANSEC

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt.

#### TRPROF({DFHCICSS|profil})

Gibt den Namen für PROFILE für die Sitzung an, die während des ISC-Transaktionsroutings die Abläufe zwischen den beteiligten Systemen ausführt. Der Name kann bis zu acht Zeichen lang sein.

##### Zulässige Zeichen:

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Sie können diese Option nur für ferne Transaktionen angeben.

#### TWASIZE({0|zahl})

Gibt die Größe in Byte des Transaktionsarbeitsbereichs an, der für diese Transaktion angefordert werden soll. Geben Sie einen ein- bis fünfstelligen Dezimalwert im Bereich zwischen 0 und 32767 an.

#### Anmerkung:

1. Wird für TWASIZE ein zu geringer Wert angegeben, dann führt dies möglicherweise zu Beschädigungen des Speichers.
2. Führen Sie für die von CICS bereitgestellten Transaktionen keine Änderungen am Wert für TWASIZE durch.

#### WAIT({YES|NO})

Gibt an, ob eine unbestätigte Arbeitseinheit (UOW) in den Wartestatus versetzt werden soll, wenn eine Wiederherstellung nach einem Fehler ansteht, der aufgetreten ist, nachdem die Arbeitseinheit in den unbestätigten Status versetzt wurde.

Traditionelle Transaktionsdefinitionen, die mit der Einstellung INDOUBT(WAIT) arbeiten, werden von CICS akzeptiert und als WAIT(YES) ACTION(BACKOUT) interpretiert.

**YES** Die Arbeitseinheit (UOW) muss bei einer anstehenden Wiederherstellung nach einem Fehler im Wartestatus verbleiben, bis der unbestätigte Status behoben und ermittelt werden kann, ob wiederherstellbare Ressourcen entweder zurückgesetzt oder festgeschrieben werden sollen. Dies bedeutet, dass die UOW *verzögert* wird.

Folgende wiederherstellbaren Ressourcen sind möglich:

- DBCTL-Datenbanken
- Db2-Datenbanken
- Warteschlangen für temporären Speicher
- Logisch wiederherstellbare partitionsinterne Warteschlangen mit transienten Daten, für die in der TDQUEUE-Definition WAIT(YES) angegeben wurde
- VSAM-Dateien
- BDAM-Dateien

Die Option WAIT(YES) wird wirksam, *wenn keine der folgenden Bedingungen vorliegt*:

- Die Transaktion verfügt über untergeordnete MRO-Sitzungen (MRO = Multiregion Operation; Mehrregionenbetrieb) zu Systemen einer früheren Version.
- Die Transaktion verfügt über untergeordnete LU6.1-Sitzungen. Beachten Sie hierbei, dass in diesem Kontext LU6.1 IMS-Sitzungen nicht als untergeordnet eingestuft werden.
- Die Transaktion umfasst mehrere Sitzungen und die zugehörige Koordinatorsitzung besteht zu einem System einer früheren Version oder zu einem LU6.1-System.
- Die taskbezogenen Benutzerexits, die der Transaktion zugeordnet sind, bieten keine Unterstützung für die CICS-Protokolle für unbestätigte Elemente.

Wenn keine der hier aufgeführten Ausnahmerebedingungen gilt, jedoch untergeordnete LU6.2-Sitzungen zu Systemen vorhanden sind, bei denen es sich nicht um CICS Transaction Server for z/OS-Systeme handelt, die nicht mit der CICS Transaction Server for z/OS-Architektur für unbestätigte Elemente arbeiten, dann kann CICS angeben, dass das untergeordnete Element in den Wartemodus versetzt werden muss, indem eine Sitzungsunterbrechung erzwungen wird.

Wenn bestimmte Ressourcen nicht auf die Auflösung unbestätigter Elemente durch den Koordinator warten können, dann wird in Übereinstimmung mit dem Attribut ACTION eine Entscheidung für die Transaktion getroffen. In der Praxis wird die Herbeiführung einer Entscheidung auf diese Weise nur bei Aktualisierungen an Warteschlangen mit transienten Daten mit der Angabe WAIT(NO) in der TDQUEUE-Definition sowie bei Installationen für terminalbezogene Ressourcen erzwungen. Im zweiten Fall erfolgt die Installation normalerweise mit einem Befehl INSTALL.

In Tabelle 4 wird gezeigt, wie das Attribut WAIT, das in einer TRANSACTION-Definition definiert wurde, und eine logisch wiederherstellbare TDQUEUE-Definition bei Auftreten eines Konflikts aufgelöst werden.

**NO** Die Arbeitseinheit soll nicht warten. CICS führt unverzüglich die im Attribut ACTION angegebene Aktion aus.

*Tabelle 4. Auflösung von WAIT-Attributen in TRANSACTION- und TDQUEUE-Definitionen*

Attribut WAIT der TDQUEUE-Definition	Attribut WAITACTION der TDQUEUE-Definition	Attribut WAIT der TRANSACTION-Definition	Aktion
NO	Nicht zutreffend	YES	TD WAIT(NO) überschreibt in der TRANSACTION-Definition die Angabe von WAIT(YES). Für die Arbeitseinheit (UOW) wird in Übereinstimmung mit den ACTION-Attributen der Transaktion entweder die Festschreibung oder Zurücksetzung erzwungen.

Tabelle 4. Auflösung von WAIT-Attributen in TRANSACTION- und TDQUEUE-Definitionen (Forts.)

Attribut WAIT der TDQUEUE-Definition	Attribut WAITACTION der TDQUEUE-Definition	Attribut WAIT der TRANSACTION-Definition	Aktion
NO	Nicht zutreffend	NO	Für die Arbeitseinheit (UOW) wird in Übereinstimmung mit den ACTION-Attributen der Transaktion entweder die Festschreibung oder Zurücksetzung erzwungen.
YES	QUEUE	YES	Die Arbeitseinheit (UOW) wartet, wird also verzögert. Eine Anforderung von einer anderen Task für eine Sperre der Warteschlange für transiente Daten muss warten und wird von CICS in die Warteschlange eingestellt.
YES	QUEUE	NO	Die Angabe WAIT(NO) für die Transaktion überschreibt die TDQUEUE-Definition. Für die Arbeitseinheit (UOW) wird in Übereinstimmung mit den ACTION-Attributen der Transaktion entweder die Festschreibung oder Zurücksetzung erzwungen.
YES	REJECT	YES	Die Arbeitseinheit (UOW) wartet, wird also verzögert. Eine Anforderung von einer anderen Task für eine Sperre für die Warteschlange für transiente Daten wird mit der Bedingung LOCKED zurückgewiesen.
YES	REJECT	NO	Die Angabe WAIT(NO) für die Transaktion überschreibt die TDQUEUE-Definition. Für die Arbeitseinheit (UOW) wird in Übereinstimmung mit den ACTION-Attributen der Transaktion entweder die Festschreibung oder Zurücksetzung erzwungen.

Wenn die Arbeitseinheit (UOW) auf mehrere Warteschlangen mit transienten Daten verweist und die Warteschlangen über nicht einheitliche WAIT-Optionen verfügen, dann hat WAIT(NO) immer Vorrang und überschreibt die Angabe WAIT(YES). Aus diesem Grund erzwingt die Angabe WAIT(NO) in einer TDQUEUE-Definition für eine fehlgeschlagene unbestätigte Arbeitseinheit (UOW) entweder die Übernahme des Attributs BACKOUT oder des Attributs COMMIT, das in der TRANSACTION-Definition der Arbeitseinheit (UOW) festgelegt wurde.

**WAITTIME**({00,00,00}|*tt, hh, mm*)

Gibt an, wie lange eine Transaktion warten soll, bevor eine Entscheidung zu einer unbestätigten Arbeitseinheit auf Basis der Angabe im Attribut ACTION getroffen wird.

00,00,00

Die Transaktion wartet unbegrenzt.

*tt, hh, mm*

Die Zeitdauer in Tagen (tt), Stunden (hh) und Minuten (mm), die die Transaktion warten muss. Der Maximalwert ist 99,23,59.

Das Attribut WAITTIME wird nur dann wirksam, wenn WAIT(YES) angegeben wurde.

#### **XTPNAME(wert)**

Sie können dieses Attribut als Alternative für TPNAME verwenden. Geben Sie eine hexadezimale Zeichenfolge mit einer Länge von bis zu 128 Zeichen ein, die für den Namen der Transaktion steht, die von einem APPC-Partner verwendet werden kann. Alle hexadezimalen Kombinationen *mit Ausnahme von* X'40' sind zulässig. Um einen XTPNAME-Wert mit mehr als 72 Zeichen in DFHCSDUP anzugeben, geben Sie in Spalte 72 einen Stern (\*) ein. Der Stern bewirkt, dass die folgende Zeile mit der aktuellen Zeile verknüpft wird.

#### **XTRANID(xtranid)**

Gibt einen anderen Namen an, der anstelle des TRANSACTION-Namens für die Einleitung von Transaktionen verwendet werden soll. Der Name kann bis zu acht Hexadezimalziffern lang sein. Da XTRANID im Hexadezimalformat angegeben wird, können Sie einen Namen verwenden, der Zeichen enthält, die Sie im Attribut TRANSACTION nicht angeben können.

Weitere Informationen finden Sie im Abschnitt zu TASKREQ, einem weiteren Transaktionsalias, der angegeben werden kann.

*wert* Eine aus vier Byte bestehende Transaktions-ID in Hexadezimalschreibweise. Die ID verwendet deshalb bis zu acht Hexadezimalziffern. Wenn Sie weniger als acht Hexadezimalziffern angeben, dann wird die ID auf der rechten Seite mit Leerzeichen aufgefüllt.

Bestimmte Werte sind für die Verwendung durch CICS reserviert, so dass die Werte, die Sie angeben können, folgenden Beschränkungen unterliegen:

- Das erste Byte darf nicht X'C3' sein.
- Das erste Byte darf nicht kleiner-gleich X'40' sein.
- Der Wert darf nicht X'00000000' sein.
- Die letzten drei Byte dürfen nicht X'FFFFFF' sein.

Vermeiden Sie die Verwendung von Werten im Bereich zwischen X'00' und X'3F' im zweiten, dritten und vierten Byte, wenn die Transaktion nicht erwarteten Daten zugeordnet werden soll, die von einem Terminal empfangen werden, das als 3270-Einheit definiert wurde, weil CICS diese Werte als Steuerzeichen interpretiert und nicht als Teil der Transaktions-ID. Wenn Sie z. B. den Befehl EXEC CICS RETURN oder EXEC CICS START absetzen und dabei TRANSID(X'41303238') angeben, wird die korrekte Transaktion zugeordnet. Wenn Sie allerdings den Befehl EXEC CICS RETURN ohne einen Wert für TRANSID absetzen, und wenn die 3270-Einheit Daten überträgt, die mit dem Wert X'41303238' beginnen, dann versucht CICS, eine Transaktion zuzuordnen, bei deren Auswahl davon ausgegangen wird, dass X'41404040' übertragen wurde.

## TSMODEL-Ressourcen

Ein Modell für temporären Speicher gibt die Eigenschaften einer Gruppe von Warteschlangen für temporären Speicher an. Einzelne Warteschlangen für temporären Speicher werden einer TSMODEL-Ressource mithilfe des *Präfix* zugeordnet. Hierbei handelt es sich um eine Zeichenfolge, die mit den führenden Zeichen des Warteschlangennamens übereinstimmt.

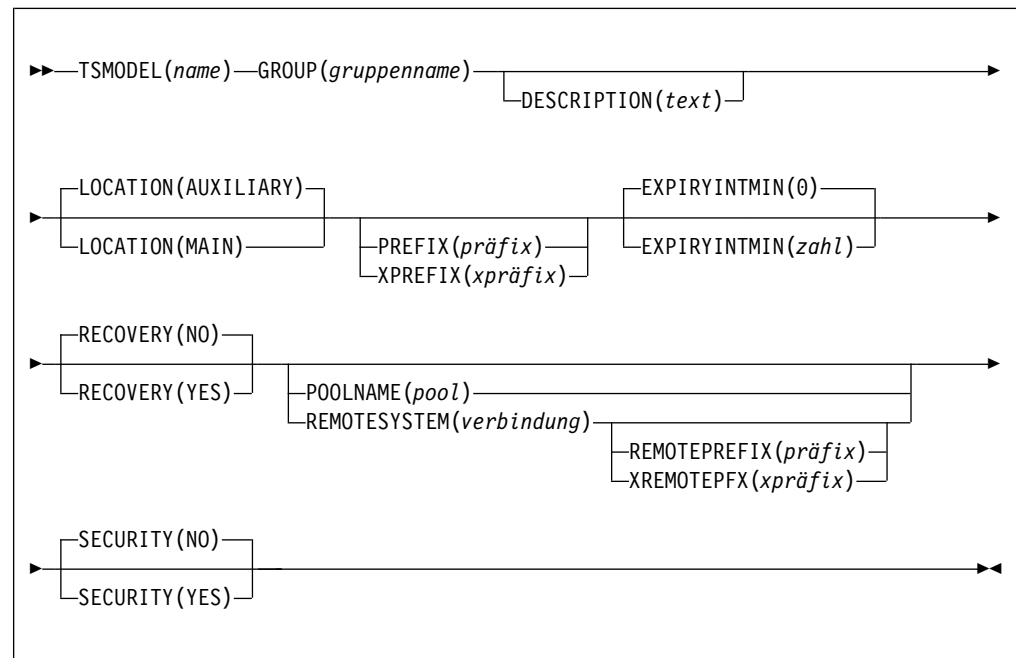
Sie können Namen auch direkt einem Pool für gemeinsam genutzten temporären Speicher (ohne Angabe einer gemeinsam genutzten System-ID) zuordnen.

**Anmerkung:** CICS führt die Standardaktionen für eine Region aus, in der keine TSMODEL-Ressource definiert ist. Wenn Sie also über eine AOR (Application-Owning Region; Anwendungsverwaltungsregion) und eine QOR (Queue-Owning Region; Warteschlangenverwaltungsregion) verfügen und wenn eine in der AOR definierte TSMODEL-Ressource Anforderungen an die QOR weiterleitet, dann werden für bestimmte Warteschlangenattribute die Standardwerte übernommen. Eine Ausnahme bildet dabei der Fall, dass in der QOR eine entsprechende TSMODEL-Ressource existiert. Die Position einer Warteschlange (MAIN oder AUX) wird beispielsweise anhand der Standardeinstellungen in CICS ermittelt. Wenn kein übereinstimmendes Modell vorhanden ist, dann wird die im Befehl EXEC CICS angegebene Position verwendet. Existiert eine Übereinstimmung mit einem Modell, dann wird die in diesem Modell angegebene Position verwendet.

Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with BAS temporary storage model resource definitions.

## TSMODEL-Attribute

Beschreibt die Syntax und die Attribute der TSMODEL-Ressource.



### DESCRIPTION(text)

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem

Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

#### **EXPIRYINTMIN({0|zahl})**

Gibt das Ablaufintervall (in Minuten) für eine Warteschlange für temporären Speicher an, die mit diesem Modell übereinstimmt. CICS verwendet den Wert, der auf das nächste Vielfache von zehn Minuten aufgerundet wird. Die Intervallzählung beginnt nach jeder Verwendung der Warteschlange für temporären Speicher. Wenn die Warteschlange nicht erneut verwendet wird, bevor das Intervall abläuft, dann wird die Warteschlange für die automatische Löschung durch CICS auswählbar.

**0** Für Warteschlangen für temporären Speicher, die mit diesem Modell übereinstimmen, gilt kein Ablaufintervall. Sie werden niemals als für die automatische Löschung auswählbar eingestuft. Dies ist die Standardeinstellung.

**zahl** Gibt ein Ablaufintervall (in Minuten) im Bereich zwischen 1 und 900000 an. Nach Ablauf dieses Intervalls wird eine Warteschlange für temporären Speicher, die mit diesem Modell übereinstimmt, für die automatische Lösung auswählbar, wenn sie nicht zwischenzeitlich erneut verwendet wurde.

Ablaufintervalle gelten für Warteschlangen für temporären Speicher, die sich an den folgenden Positionen befinden:

- Temporärer Hauptspeicher in der lokalen CICS-Region.
- Nicht wiederherstellbarer temporärer Hilfsspeicher (DFHTEMP-Datei), der der lokalen CICS-Region zugeordnet ist.
- Warteschlangen für temporären Speicher in gemeinsam genutzten Pools für temporären Speicher (CICS TS 5.2 und höher).

Ablaufintervalle gelten nicht für die folgenden Typen von Warteschlangen für temporären Speicher, die aus diesem Grund von CICS niemals automatisch gelöscht werden:

- Warteschlangen im temporären Hilfsspeicher, die als wiederherstellbar definiert sind.
- Warteschlangen in einer fernen CICS-Region. Damit CICS ferne Warteschlangen für temporären Speicher löschen kann, müssen Sie ein Ablaufintervall in einer geeigneten TSMODEL-Ressourcendefinition in der Region angeben, die die Warteschlangen verwaltet.
- Warteschlangen, die von CICS zur eigenen Verwendung erstellt werden.

Wenn Sie das Ablaufintervall in einer TSMODEL-Ressourcendefinition ändern, dann sind bereits vorhandene Warteschlangen für temporären Speicher, die mit diesem Modell übereinstimmen, nicht von der Änderung betroffen. Diese Warteschlangen benutzen weiterhin das Ablaufintervall, das bei deren Erstellung angewendet wurde. Wenn alle TSMODEL-Ressourcendefinitionen mit einem Ablaufintervall ungleich null aus einer CICS-Region gelöscht werden, dann stoppt CICS die Suche nach abgelaufenen Warteschlangen für temporären Speicher.

### **GROUP**(*gruppenname*)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

#### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

### **LOCATION**(**{AUXILIARY|MAIN}**)

Gibt an, ob die Warteschlange im Hilfsspeicher oder im Hauptspeicher gehalten werden soll:

#### **AUXILIARY**

Warteschlangen, die mit diesem Modell übereinstimmen, werden im temporären Hilfsspeicher gehalten. Alle Angaben in der API-Anforderung werden ignoriert.

**MAIN** Warteschlangen, die mit diesem Modell übereinstimmen, werden im temporären Hauptspeicher gehalten. Alle Angaben in der API-Anforderung werden ignoriert.

Die Angabe für LOCATION wird für Modelle für den temporären Speicher ignoriert, die sich auf ferne Warteschlangen und auf Warteschlangen beziehen, die sich in Pools für temporären Speicher befinden. Wird LOCATION in einer TSMODEL-Ressourcendefinition für eine ferne Warteschlange verwendet, dann kann dieselbe Definition sowohl in der lokalen als auch in der fernen Region installiert werden. Weitere Informationen hierzu finden Sie in Shared resources for intercommunication.

### **POOLNAME**(*pool*)

Gibt den Namen für die Definition des Pools für gemeinsam genutzten temporären Speicher an, die mit dieser TSMODEL-Ressourcendefinition verwendet werden soll. Der Name kann bis zu acht Zeichen lang sein.

#### **Zulässige Zeichen:**

A-Z 0-9 \$ @ # \_

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Die Verwendung eingebetteter Leerzeichen ist nicht zulässig und Namen, die vollständig aus Leerzeichen zusammengesetzt sind, werden so interpretiert, als wäre kein Poolname angegeben worden.

Wenn REMOTESYSTEM angegeben wurde, dann ist die gleichzeitige Angabe von POOLNAME nicht zulässig.

CICS sucht nicht nach einer übereinstimmenden TSMODEL-Ressourcendefinition, wenn ein Anwendungsprogramm im Befehl für temporären Speicher EXEC CICS einen Wert für SYSID angibt oder wenn eine solche SYSID durch ein globales XTSEREQ-Benutzerexitprogramm hinzugefügt wird. Um CICS für die Suche des Namens eines Pools für die gemeinsame Datennutzung im temporären Speicher zu aktivieren, wenn im Anwendungsprogramm explizit ein Wert für SYSID angegeben wurde, dann müssen Sie eine Tabelle für temporären Speicher (TST = Temporary Storage Table) mit einem passenden TYPE=SHARED-Eintrag verwenden.

### **PREFIX(*präfix*)**

Gibt die Zeichenfolge an, die von CICS verwendet wird, um übereinstimmende Warteschlangen für temporären Speicher zu ermitteln. Das Präfix kann bis zu 16 Zeichen lang sein.

#### **Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Im einfachsten Fall stimmt ein Präfix mit den Namen der Warteschlangen für temporären Speicher überein, die mit dem Präfix beginnen. Beispiel: Das Präfix ABCD stimmt mit den Warteschlangennamen ABCD, ABCDE und ABCD99 überein.

Sie können das generische Zeichen (+) einmal oder mehrmals innerhalb des Präfix für ein beliebiges Zeichen im Namen einer Warteschlange für temporären Speicher verwenden. Beispiel: Das Präfix A++D stimmt mit den Warteschlangennamen ABCD, A99D und ABCD99 überein. Es stimmt hingegen nicht mit ABD überein. Sie müssen am Ende eines Präfix keine Pluszeichen (+) angeben. Das Präfix wird rechts bis zur maximalen Länge eines Namens für eine Warteschlange für temporären Speicher (16 Zeichen) mit Pluszeichen (+) aufgefüllt. Beispiel: Das Präfix ABC entspricht exakt dem Präfix ABC+ oder ABC++++.

Wenn Sie das Pluszeichen (+) in einem Präfix verwenden, dann werden Sie möglicherweise feststellen, dass bei separater Betrachtung mehrere Präfixe mit dem Namen einer Warteschlange für temporären Speicher übereinstimmen. In diesem Fall verwendet CICS die folgenden Regeln, um das übereinstimmende TS-Modell auszuwählen:

- Zeichen, die im Präfix exakt angegeben wurden, stellen eine stärkere Übereinstimmung dar als das Pluszeichen (+).
- Zeichen werden jeweils einzeln von links nach rechts verglichen und übereinstimmende Zeichen ergeben eine stärkere Übereinstimmung, je weiter links sie aufgeführt sind.

Beispiel:

- Der Warteschlangenname ABCD99 stimmt mit dem Präfix ABCD besser überein als mit dem Präfix AB+D: Im ersten Präfix sind die vier übereinstimmenden Zeichen exakt angegeben, wohingegen im zweiten Präfix nur drei Zeichen exakt angegeben sind.
- Der Warteschlangenname ABCD99 stimmt mit dem Präfix ABC+ besser überein als mit dem Präfix AB+D: In beiden Präfixen wird die gleiche Anzahl von übereinstimmenden Zeichen exakt angegeben. Allerdings befindet sich das dritte übereinstimmende Zeichen im ersten Präfix weiter links.

Die Erstellung einer TSMODEL-Definition mit einem Präfix aus Leerzeichen kann zu nicht erwarteten Ergebnissen führen. Eine TSMODEL-Definition mit einem Präfix aus Leerzeichen stimmt mit allen Warteschlangennamen überein, die nicht mit einer anderen TSMODEL-Definition übereinstimmen.

Um CICS für die Suche des Namens eines Pools für die gemeinsame Datennutzung im temporären Speicher zu aktivieren, wenn im Anwendungsprogramm explizit ein Wert für SYSID angegeben wurde, dann müssen Sie eine Tabelle für temporären Speicher (TST = Temporary Storage Table) mit einem passenden TYPE=SHARED-Eintrag verwenden.

### **RECOVERY({NO|YES})**

Gibt an, ob Warteschlangen, die mit diesem Modell übereinstimmen, wiederherstellbar sein sollen.



**NO** Warteschlangen, die mit diesem Modell übereinstimmen, sind nicht wiederherstellbar.

**YES** Warteschlangen, die mit diesem Modell übereinstimmen, sind wiederherstellbar.

RECOVERY(YES) darf nicht zusammen mit LOCATION(MAIN) angegeben werden.

#### **REMOTEPREFIX**(*präfix*)

Gibt die Zeichenfolge an, die von CICS verwendet wird, um übereinstimmende Warteschlangen für temporären Speicher auf dem fernen System zu ermitteln. Das Präfix kann bis zu 16 Zeichen lang sein.

##### **Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

REMOTEPREFIX ist nur dann zulässig, wenn auch REMOTESYSTEM angegeben wird. Die Länge des Attributs REMOTEPREFIX muss mit der Länge des Attributs PREFIX übereinstimmen.

Die Regeln für den Zeichenabgleich im Attribut REMOTEPREFIX sind identisch mit den entsprechenden Regeln für das Attribut PREFIX. Wenn Sie Pluszeichen (+) für den generischen Abgleich verwenden, dann müssen sich diese in PREFIX und REMOTEPREFIX an derselben Position befinden. Beispiel:

PREFIX: A++D

REMOTEPREFIX: X++Y

#### **REMOTESYSTEM**(*verbindung*)

Gibt den Namen der Verbindung an, über die das lokale System mit dem fernen System verbunden wird, auf dem sich die Warteschlange für temporären Speicher befindet.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Die Verwendung eingebetteter Leerzeichen ist nicht zulässig und Namen, die vollständig aus Leerzeichen zusammengesetzt sind, werden so interpretiert, als wäre kein fernes System angegeben worden.

Die Angabe von REMOTESYSTEM und POOLNAME schließt sich gegenseitig aus. Wird REMOTESYSTEM angegeben, dann wird die Angabe für POOLNAME ignoriert.

#### **SECURITY**({**NO**|**YES**})

Gibt an, ob die Sicherheitsprüfung für Warteschlangen ausgeführt wird, die mit diesem Modell übereinstimmen.

**NO** Die Sicherheitsprüfung wird für Warteschlangen, die mit diesem Modell übereinstimmen, nicht ausgeführt.

**YES** Die Sicherheitsprüfung wird für Warteschlangen, die mit diesem Modell übereinstimmen, ausgeführt.

Weitere Informationen hierzu finden Sie in Security for temporary storage.

**TSMODEL**(*name*)

Gibt den Namen dieser TSMODEL-Ressourcendefinition an. Der Name kann bis zu acht Zeichen lang sein.

**Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Eingebettete Leerzeichen sind nicht zulässig.

Dieser Name wird verwendet, um die TSMODEL-Ressourcendefinition in der CSD-Datei zu identifizieren. Innerhalb des aktiven CICS-Systems wird er nicht verwendet.

**XPREFIX**(*xpräfix*)

Kann als Alternative zu PREFIX verwendet werden. Geben Sie eine hexadezimale Zeichenfolge mit einer Länge von bis zu 32 Zeichen ein, die als Präfix für dieses Modell verwendet werden soll. Da XPREFIX im Hexadezimalformat angegeben wird, können Sie einen Namen verwenden, der Zeichen enthält, die Sie im Attribut PREFIX nicht eingeben können.

Zur Angabe generischer Zeichen im Attribut XPREFIX müssen Sie X'4E' angeben.

Die Regeln für den Abgleich des Werts für XPREFIX mit den Namen der Warteschlangen für temporären Speicher sind identisch mit den entsprechenden Regeln für das Attribut PREFIX.

**XREMOTEPFX**(*xpräfix*)

Kann als Alternative zu REMOTEPREFIX verwendet werden. Geben Sie eine hexadezimale Zeichenfolge mit einer Länge von bis zu 32 Zeichen ein, die als Präfix auf dem fernen System verwendet werden soll. Da XREMOTEPFX im Hexadezimalformat angegeben wird, können Sie einen Namen verwenden, der Zeichen enthält, die Sie im Attribut REMOTEPREFIX nicht eingeben können.

Zur Angabe generischer Zeichen im Attribut XPREFIX müssen Sie X'4E' angeben.

Die Regeln für den Abgleich des Werts für XREMOTEPFX mit den Namen der Warteschlangen für temporären Speicher sind identisch mit den entsprechenden Regeln für das Attribut REMOTEPREFIX.

---

## TYPETERM-Ressourcen

Eine TYPETERM-Ressource definiert eine Gruppe von Attributen, die von einer Gruppe von Terminals gemeinsam verwendet werden.

Die Ressource ist eine logische Erweiterung der TERMINAL-Ressource. Wenn Sie über eine Reihe von Terminals mit den gleichen Eigenschaften verfügen, dann können Sie eine TYPETERM-Komponente mit den erforderlichen Werten definieren und diese TYPETERM-Komponente dann in allen TERMINAL-Definitionen (oder bei Verwendung der automatischen Installation in der Definition des Modells für die automatische Installation) angeben.

Jede TERMINAL-Definition muss eine Angabe zu einer TYPETERM-Definition enthalten. Dieses einzelne Attribut repräsentiert zahlreiche andere Merkmale und kann dadurch den Arbeitsaufwand sowie das Fehlerrisiko deutlich reduzieren. TYPETERM-Komponenten vereinfachen das Definieren Ihrer Terminals, wenn Sie mit einer großen Zahl von Terminals derselben Art arbeiten.

Zwei TYPETERM-Attribute sollen hier gesondert betrachtet werden, da sie den Terminaldefinitionsprozess weiter vereinfachen:

DEVICE

QUERY

#### DEVICE

Gibt den **Einheitentyp** an, für den die TYPETERM-Komponente steht. Hierbei handelt es sich um ein Schlüsselattribut, weil die Standardwerte für eine Reihe anderer Attribute von dem Wert abhängig sind, den Sie für dieses Attribut angeben:

- Bestimmte Attribute sind für jede Einheit desselben Typs immer gleich. Sie müssen nicht alle diese Attribute selbst definieren, weil RDO über die entsprechenden Informationen zu ihnen verfügt. Sie müssen für RDO lediglich den Einheitentyp Ihrer Terminals angeben, wenn Sie die TYPETERM-Komponente für sie definieren. Werte für feste Attribute werden automatisch bereitgestellt.
- Andere Attribute erhalten abhängig vom Einheitentyp entsprechende Standardwerte. Allerdings ist hierbei zu berücksichtigen, dass Sie die von CICS bereitgestellten Werte nicht verwenden müssen und stattdessen bei Bedarf andere Werte angeben können. Wenn Sie den Einheitentyp in einer TYPETERM-Definition ändern, dann werden die Standardwerte nicht zurückgesetzt.

Sie müssen einen Wert für das Attribut DEVICE angeben, weil dieses Attribut keinen Standardwert hat.

Informationen zu den von RDO unterstützten Terminals finden Sie in Unterstützte Einheiten. In Standardwerte für TYPETERM-Attribute finden Sie außerdem eine Liste mit den für das Attribut DEVICE von TYPETERM zulässigen Werten. Dort erhalten Sie einen Überblick zu den anderen Attributwerten, die für die unterschiedlichen Einheitentypen bereitgestellt werden. In bestimmten Fällen hängen diese Werte auch von den Werten ab, die für SESSIONTYPE und TERMMODEL angegeben werden, diese Attribute verfügen jedoch ebenfalls über Standardwerte, die von dem für DEVICE angegebenen Wert abhängig sind.

Neben den normalen Bildschirmeinheiten, Druckern und anderen Ein- und Ausgabeeinheiten für speziellere Einsatzgebiete können Sie eine TYPETERM-Definition auch für Ihre CICS-Konsolen erstellen.

#### QUERY

Durch Angabe des Attributs QUERY erhalten Sie die Möglichkeit, bestimmte Funktionen Ihrer Terminals erst dann zu definieren, wenn eine Verbindung hergestellt wird. Informationen zu diesen Attributen können dann von CICS selbst abgefragt werden, indem Sie das strukturierte Feld QUERY verwenden.

Alle Attribute, für die sie QUERY verwenden können, sind ebenfalls TYPETERM-Attribute. Hierbei handelt es sich um die folgenden Attribute:

ALTPAGE  
ALTSCREEN  
APLTEXT  
BACKTRANS  
CGCSGID  
COLOR  
EXTENDEDDES  
HILIGHT  
MSRCONTROL  
OBFORMAT

OUTLINE  
PARTITIONS  
PROGSYMBOLS  
SOSI  
VALIDATION

Die Verwendung von QUERY führt zur Überschreibung aller Werte, die für eines der TYPETERM-Attribute, die hier aufgelistet werden, explizit definiert wurden. **Eine Ausnahme bildet hier ALTSCREEN.** Die mit QUERY bereitgestellten Werte für ALTSCREEN werden nur dann verwendet, wenn in der TYPETERM-Definition kein expliziter ALTSCREEN-Wert definiert wurde.

Sie können QUERY für 3270-Einheiten mit erweitertem 3270-Datenstrom verwenden. QUERY kann für die folgenden DEVICE-Typen verwendet werden:

3270  
3270P  
LUTYPE2  
LUTYPE3  
SCSPRINT

Sie können angeben, dass QUERY auf eine der beiden folgenden Arten verwendet wird:

- QUERY(COLD) gibt an, dass QUERY nur dann abgesetzt werden soll, wenn das Terminal nach einem Erst- oder Kaltstart zum ersten Mal verbunden wird.
- QUERY(ALL) gibt an, dass QUERY jedes Mal abgesetzt werden soll, wenn das Terminal verbunden wird.

Die Funktion QUERY ist besonders bei konfigurierbaren Einheiten wie z. B. IBM Personal System/2 (PS/2) und bei der Einheit IBM 3290 nützlich. Sie ermöglicht Ihnen die Neukonfiguration der Einheit zwischen der Ab- und der Anmeldung bei CICS, ohne dass Sie Ressourcendefinitionen ändern müssen.

Die Funktion QUERY ist auch dann besonders nützlich, wenn Sie bei der automatischen Installation verwendet wird.

Beachten Sie hierbei, dass die Funktion QUERY nur die Informationen abrufen, die von CICS benötigt werden. Wenn ein Anwendungsprogramm weitere Einheitenmerkmale ermitteln muss, dann muss dennoch ein strukturiertes Feld QUERY gesendet und die Antwort analysiert werden.

Zusammenfassend ist festzuhalten, dass Sie lediglich eine TYPETERM-Definition für jeden Einheitentyp benötigen. Wenn sich die Attribute, die mithilfe von QUERY ermittelt werden können, bei den verschiedenen Terminals unterscheiden, dann benötigen Sie dennoch nur eine TYPETERM-Definition für jeden Einheitentyp. Wenn andere Attribute Ihrer Terminals variieren, dann benötigen Sie möglicherweise mehrere TYPETERM-Definitionen für einen Einheitentyp.

Für die häufiger verwendeten Terminals stellt CICS verschiedene TYPETERM-Definitionen bereit. Diese werden in TYPETERM definitions in group DFHTYPE beschrieben.

Wenn alle Terminals grundlegend gleich sind, dann können Sie mit einer TYPETERM-Definition und mit einer TERMINAL-Definition mit der Angabe AUTINST-MODEL(YES) arbeiten. Möglicherweise ist es sinnvoll, QUERY zur Verarbeitung verschiedener Funktionen zu verwenden, die von Ihren Terminals verwendet werden.

Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with BAS typeterm resource definitions.

## Standardwerte für TYPETERM-Attribute

Wenn Sie in einer TYPETERM-Definition die Attribute DEVICE, SESSIONTYPE und TERMMODEL angeben, dann stellt CICS Standardwerte für viele der anderen Attribute bereit.

Die Standardwerte werden in Tabelle 5 dargestellt. Beachten Sie hierbei, dass die bereitgestellten Werte für einige der Attribute zwingend verwendet werden müssen, sodass Sie sie nicht ändern können.

Tabelle 5. Standardwerte für TYPETERM-Attribute

DEVICE	SESSIONTYPE	TERMMODEL	Standardwerte
3270 3277 L3277  Siehe hierzu Anmerkung 1 auf Seite 274		1	DEFSCREEN(12,40) PAGESIZE(12,40) AUTOPAGE(N) BRACKET(Y) BUILDCHAIN(N) ROUTEDMSGGS(ALL)
3270		2 (Standardwert)	DEFSCREEN(24,80) PAGESIZE(24,80) AUTOPAGE(N) BRACKET(Y) BUILDCHAIN(N) ROUTEDMSGGS(ALL)
3275		1	DEFSCREEN(12,40) PAGESIZE(12,40) AUTOPAGE(N) BRACKET(Y) BUILDCHAIN(N) ROUTEDMSGGS(ALL)
3275		2 (Standardwert)	DEFSCREEN(24,80) PAGESIZE(24,80) AUTOPAGE(N) BRACKET(Y) BUILDCHAIN(N) ROUTEDMSGGS(ALL)
3270P 3284 L3284 3286 L3286  Siehe hierzu Anmerkung 2 auf Seite 274		1	DEFSCREEN(12,40) PAGESIZE(12,40) AUTOPAGE(Y) BRACKET(Y) BUILDCHAIN(N) ROUTEDMSGGS(ALL)

Tabelle 5. Standardwerte für TYPETERM-Attribute (Forts.)

DEVICE	SESSIONTYPE	TERMMODEL	Standardwerte
3270P		2 (Standardwert)	DEFSCREEN(24,80) PAGESIZE(24,80) AUTOPAGE(Y) BRACKET(Y) BUILDCHAIN(N) ROUTEDMSGs(ALL)
APPC			DEFSCREEN(0,0) PAGESIZE(1,40) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(Y) (zwingend erforderlich) ROUTEDMSGs(NONE) (zwingend erforderlich)
CONSOLE			DEFSCREEN(0,0) (zwingend erforderlich) PAGESIZE(1,124) (zwingend erforderlich) AUTOPAGE(N) BRACKET(N) (zwingend erforderlich) BUILDCHAIN(N) (zwingend erforderlich) ROUTEDMSGs(NONE)
LUTYPE2		1	DEFSCREEN(12,40) PAGESIZE(12,40) AUTOPAGE(N) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(Y) (zwingend erforderlich) ROUTEDMSGs(ALL)
LUTYPE2		2 (Standardwert)	DEFSCREEN(24,80) PAGESIZE(24,80) AUTOPAGE(N) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(Y) (zwingend erforderlich) ROUTEDMSGs(ALL)
LUTYPE3		1	DEFSCREEN(12,40) PAGESIZE(12,40) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)
LUTYPE3		2 (Standardwert)	DEFSCREEN(24,80) PAGESIZE(24,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)

Tabelle 5. Standardwerte für TYPETERM-Attribute (Forts.)

DEVICE	SESSIONTYPE	TERMMODEL	Standardwerte
LUTYPE4			DEFSCREEN(0,0) PAGESIZE(50,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)
BCHLU	(Standardwert)		DEFSCREEN(0,0) PAGESIZE(12,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)
BCHLU	BATCHDI		DEFSCREEN(0,0) PAGESIZE(12,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)
BCHLU	USERPROG		DEFSCREEN(0,0) PAGESIZE(12,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)
INTLU			DEFSCREEN(0,0) PAGESIZE(12,80) AUTOPAGE(Y) BRACKET(Y) BUILDCHAIN(N) ROUTEDMSGs(ALL)
SCSPRINT			DEFSCREEN(0,0) (zwingend erforderlich) PAGESIZE(24,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)
TLX oder TWX	CONTLU (Standardwert)		DEFSCREEN(0,0) PAGESIZE(1,40) AUTOPAGE(Y) BRACKET(Y) BUILDCHAIN(N) ROUTEDMSGs(ALL)

Tabelle 5. Standardwerte für TYPETERM-Attribute (Forts.)

DEVICE	SESSIONTYPE	TERMMODEL	Standardwerte
TLX oder TWX	INTLU		DEFSCREEN(0,0) PAGESIZE(1,40) AUTOPAGE(Y) BRACKET(Y) BUILDCHAIN(N) ROUTEDMSGs(ALL)
3600	(Standardwert)		DEFSCREEN(0,0) PAGESIZE(1,40) AUTOPAGE(Y) BRACKET(Y) BUILDCHAIN(N) ROUTEDMSGs(ALL)
3600	PIPELINE		DEFSCREEN(0,0) PAGESIZE(6,30) AUTOPAGE(Y) BRACKET(Y) BUILDCHAIN(N) ROUTEDMSGs(ALL)
3614			DEFSCREEN(0,0) PAGESIZE(1,40) AUTOPAGE(Y) BRACKET(Y) BUILDCHAIN(N) ROUTEDMSGs(ALL)
3650	USERPROG (Standardwert)		DEFSCREEN(0,0) PAGESIZE(3,80) AUTOPAGE(Y) BRACKET(Y) BUILDCHAIN(N) ROUTEDMSGs(ALL)
3650	3270		DEFSCREEN(12,40) PAGESIZE(23,80) AUTOPAGE(Y) BRACKET(Y) BUILDCHAIN(N) ROUTEDMSGs(ALL)
3650	3653		DEFSCREEN(0,0) PAGESIZE(6,30) AUTOPAGE(Y) BRACKET(Y) BUILDCHAIN(N) ROUTEDMSGs(ALL)



Tabelle 5. Standardwerte für TYPETERM-Attribute (Forts.)

DEVICE	SESSIONTYPE	TERMMODEL	Standardwerte
3650	PIPELINE		DEFSCREEN(0,0) PAGESIZE(6,30) AUTOPAGE(Y) BRACKET(Y) BUILDCHAIN(N) ROUTEDMSGs(ALL)
3767			DEFSCREEN(0,0) PAGESIZE(12,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)
3767C			DEFSCREEN(0,0) PAGESIZE(12,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)
3767I			DEFSCREEN(0,0) PAGESIZE(12,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)
3770	(Standardwert)		DEFSCREEN(0,0) PAGESIZE(12,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)
3770	USERPROG		DEFSCREEN(0,0) PAGESIZE(12,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)
3770	BATCHDI		DEFSCREEN(0,0) PAGESIZE(12,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)

Tabelle 5. Standardwerte für TYPETERM-Attribute (Forts.)

DEVICE	SESSIONTYPE	TERMMODEL	Standardwerte
3770B	(Standardwert)		DEFSCREEN(0,0) PAGESIZE(12,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)
3770B	BATCHDI		DEFSCREEN(0,0) PAGESIZE(12,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)
3770B	USERPROG		DEFSCREEN(0,0) PAGESIZE(12,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)
3770C			DEFSCREEN(0,0) PAGESIZE(12,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)
3770I			DEFSCREEN(0,0) PAGESIZE(12,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)
3790	(Standardwert)		DEFSCREEN(0,0) PAGESIZE(1,40) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(NONE) (zwingend erforderlich)
3790	SCSPRINT		DEFSCREEN(0,0) PAGESIZE(24,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)

Tabelle 5. Standardwerte für TYPETERM-Attribute (Forts.)

DEVICE	SESSIONTYPE	TERMMODEL	Standardwerte
3790	USERPROG		DEFSCREEN(0,0) PAGESIZE(12,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)
3790	BATCHDI		DEFSCREEN(0,0) PAGESIZE(12,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)
3790	3277CM  Siehe hierzu An- merkung 3 auf Seite 274	1	DEFSCREEN(12,40) PAGESIZE(12,40) AUTOPAGE(N) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(Y) (zwingend erforderlich) ROUTEDMSGs(ALL)
3790	3277CM  Siehe hierzu An- merkung 3 auf Seite 274	2 (Standardwert)	DEFSCREEN(24,80) PAGESIZE(24,80) AUTOPAGE(N) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(Y) (zwingend erforderlich) ROUTEDMSGs(ALL)
3790	3284CM  Siehe hierzu An- merkung 4 auf Seite 274	1	DEFSCREEN(12,40) PAGESIZE(12,40) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)
3790	3284CM  Siehe hierzu An- merkung 4 auf Seite 274	2 (Standardwert)	DEFSCREEN(24,80) PAGESIZE(24,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)
3790	3286CM  Siehe hierzu An- merkung 4 auf Seite 274	1	DEFSCREEN(12,40) PAGESIZE(12,40) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGs(ALL)

Tabelle 5. Standardwerte für TYPETERM-Attribute (Forts.)

DEVICE	SESSIONTYPE	TERMMODEL	Standardwerte
3790	3286CM Siehe hierzu An- merkung 4	2 (Standardwert)	DEFSCREEN(24,80) PAGESIZE(24,80) AUTOPAGE(Y) BRACKET(Y) (zwingend erforderlich) BUILDCHAIN(N) ROUTEDMSGS(ALL)
<b>Anmerkung:</b> 1. Wenn Sie DEVICE(3277) oder DEVICE(L3277) angeben, dann ersetzt CICS den von Ihnen angegebenen Wert durch DEVICE(3270). 2. Wenn Sie DEVICE(3284), DEVICE(3286), DEVICE(L3284) oder DEVICE(L3286) angeben, dann ersetzt CICS den von Ihnen angegebenen Wert durch DEVICE(3270P). 3. Wenn Sie DEVICE(3790) und SESSIONTYPE(3277CM) angeben, dann ersetzt CICS die von Ihnen angegebenen Werte durch DEVICE(LUTYPE2). Es existiert kein SESSIONTYPE-Wert für DEVICE(LUTYPE2). 4. Wenn Sie DEVICE(3790) und SESSIONTYPE(3284CM) oder SESSIONTYPE(3286CM) angeben, dann ersetzt CICS die von Ihnen angegebenen Werte durch DEVICE(LUTYPE3). Es existiert kein SESSIONTYPE-Wert für DEVICE(LUTYPE3).			

## Unterstützte Einheiten

In diesem Abschnitt werden die Einheitennamen aufgelistet, die Sie in einer TYPE-TERM-Definition verwenden können.

In Tabelle 6 ist eine Liste der von RDO unterstützten Terminaltypen aufgeführt. Um die Informationen in dieser Tabelle zu nutzen, müssen Sie die Nummer der Produktfamilie Ihrer Einheit in der Spalte ganz links (Überschrift **Terminal- oder Systemtyp**) suchen. Anschließend müssen Sie in der zweiten Spalte (Überschrift **Einheiten**) prüfen, ob Ihr Einheitentyp hier explizit aufgeführt wird. Als Nächstes müssen Sie den Inhalt der dritten Spalte (Überschrift **Anschluss**) prüfen, um festzustellen, welchen Einheitentyp Sie in Ihrer TYPETERM-Definition verwenden müssen.

Die Anmerkungen, die im Anschluss an die Tabelle aufgeführt sind, enthalten weitere erforderliche Erläuterungen. Folgende Abkürzungen werden verwendet:

**lokal** Über Kanal oder Adapter angeschlossen

**s/s** Start-Stopp-Übertragung

**SDLC** Synchronous Data Link Control (synchrone Datenübertragungssteuerung)

**sw** Kommunikation über Wählleitung (switched)

**BSC** Binärsynchron (Binary Synchronous)

**nonsw**

Kommunikation über Standleitung (nonswitched)

Tabelle 6. Von RDO unterstützte z/OS Communications Server-Terminals und -Subsysteme

Terminal- oder Systemtyp	Einheiten	Anschluss	Anmerkungen
3101		Unterstützung als TWX 33/35	9 auf Seite 278
3230		Unterstützung als INTLU (z/OS Communications Server)	

Tabelle 6. Von RDO unterstützte z/OS Communications Server-Terminals und -Subsysteme (Forts.)

Terminal- oder Systemtyp	Einheiten	Anschluss	Anmerkungen
3270	3178, 3179, 3180, 3262, 3271, 3272, 3274, 3276, 3290	lokal, SDLC, BSC nonsw	1 auf Seite 277, 2 auf Seite 277
	3275, 3277, 3278, 3279, 3284, 3286, 3287, 3288, 3289	BSC sw oder nonsw	2 auf Seite 277
3270PC	3270PC, 3270PC/G, 3270PC/GX	Unterstützung als 3270	
3287	Modelle 11, 12	SDLC-Unterstützung als SCSPT	12 auf Seite 278
3600	3601, 3602, 3690, 3604, 3610, 3612, 3618, 3614, 3624	SDLC, BSC nonsw	3 auf Seite 277, 4 auf Seite 277, 13 auf Seite 278
3630	3631, 3632, 3643, 3604	Anschluss als 3600	3 auf Seite 277, 10 auf Seite 278
3640	3641, 3644, 3646, 3647	SDLC-Anschluss als INTLU	12 auf Seite 278
	3642	SDLC-Anschluss als SCSPT	12 auf Seite 278
	3643	SDLC-Unterstützung als LUTYPE2	12 auf Seite 278
	3645	SDLC-Unterstützung als SCSPT	12 auf Seite 278
3650	3651, 3653, 3275, 3284	SDLC	3 auf Seite 277
3680	3684	Unterstützung als 3790/3650	3 auf Seite 277
3730	3791	Unterstützung als 3790	3 auf Seite 277
3767		SDLC s/s-Unterstützung als 2740/2741	
3770	3771, 3773, 3774	SDLC	3 auf Seite 277, 5 auf Seite 278
	3775, 3776, 3777	BSC-Unterstützung als 2770	
3790	3791	SDLC oder lokal	3 auf Seite 277, 6 auf Seite 278
4300	4331, 4341, 4361, 4381	BSC oder SDLC	3 auf Seite 277, 7 auf Seite 278
4700	4701-1	Unterstützung als 3600	3 auf Seite 277, 4 auf Seite 277
5280		Unterstützung als 3270 (z/OS Communications Server)	
5520		SDLC-Unterstützung als logische 3790-Einheit mit vollem Funktionsumfang BSC-Unterstützung als 2770 SDLC-Anschluss als APPC	3 auf Seite 277
5550		Unterstützung als 3270	
5937		SDLC/BSC-Anschluss als 3270	2 auf Seite 277
6670		SDLC BSC-Unterstützung als 2770	

Tabelle 6. Von RDO unterstützte z/OS Communications Server-Terminals und -Subsysteme (Forts.)

Terminal- oder Systemtyp	Einheiten	Anschluss	Anmerkungen
8100	8130/8140-Prozessoren mit DPCX	Unterstützung als 3790	3 auf Seite 277
	DPPX/BASE mit Host Presentation Services oder Host Transaction Facility	Anschluss als 3790	3 auf Seite 277
	DPPX/DSC oder DPCX/DSC (einschließlich 8775-Anschluss)	Unterstützung als 3270	3 auf Seite 277,11 auf Seite 278
8775		SDLC-Unterstützung als LUTYPE2	
8815		Unterstützung als APPC	
Displaywriter		Unterstützung als APPC SNA for EDDS Unterstützung als 3270; Anschluss als 2741 (s/s) oder 3780 (BSC) SDLC-Anschluss als APPC	
Personal Computer		Unterstützung als 3270 und als APPC	13 auf Seite 278
PS/2		Unterstützung als 3270 und als APPC	13 auf Seite 278
Scanmaster		Unterstützung als APPC	
Series/1		Anschluss als System/3; Unterstützung als 3650 Pipeline (z/OS Communications Server) oder 3790 (logische Einheit mit vollem Funktionsumfang)	3 auf Seite 277
System/32	5320	SDLC-Unterstützung als 3770 BSC-Unterstützung als 2770	3 auf Seite 277,8 auf Seite 278
System/34	5340	SDLC-Unterstützung als 3770 BSC-Anschluss als System/3	3 auf Seite 277,8 auf Seite 278
System/36		Unterstützung als System/34 SDLC-Anschluss als APPC	
System/38	5381	SDLC-Anschluss als 3770 SDLC-Anschluss als APPC BSC-Anschluss als System/3	3 auf Seite 277,8 auf Seite 278
AS/400	5381	SDLC-Anschluss als 3770 SDLC-Anschluss als APPC BSC-Anschluss als System/3	3 auf Seite 277,8 auf Seite 278
System/370		SDLC-Anschluss als APPC	3 auf Seite 277
System/390		SDLC-Anschluss als APPC	3 auf Seite 277
z Systems		SDLC-Anschluss als APPC	3 auf Seite 277
TWX 33/35		z/OS Communications Server (via NTO) ss sw	9 auf Seite 278

Tabelle 6. Von RDO unterstützte z/OS Communications Server-Terminals und -Subsysteme (Forts.)

Terminal- oder Systemtyp	Einheiten	Anschluss	Anmerkungen
WTYY		z/OS Communications Server (via NTO) ss nonsw	9 auf Seite 278

**Anmerkung:**

1. CICS unterstützt 3290-Einheiten über die Terminalsteuerung und über die BMS-Schnittstellen. Die 3290-Einheit kann sich in einem der folgenden drei Statuszustände befinden: Standard, Alternativ oder Partitioniert. Auf dem System können bis zu 16 Partitionen definiert werden. Die 3290-Einheit verfügt außerdem über die Funktionen für programmierte Symbole und für die zusätzliche Hervorhebung sowie über zwei Arten von Datenvalidierungsfunktionen (MANDATORY FILL und MANDATORY ENTER). Ein 3290-Terminal kann mit 1 - 5 logischen Einheiten konfiguriert werden. Sie definieren die Größe der einzelnen logischen Einheiten beim Einrichten der 3290-Einheit. Sie müssen sicherstellen, dass die Ressourcendefinitionen der einzelnen logischen Einheiten mit der konfigurierten Größe übereinstimmen, um unvorhersehbare Ergebnisse zu vermeiden. In allen Konfigurationen können bis zu vier interaktive Anzeigen gleichzeitig aktiv sein, allerdings kann jeweils nur eine interaktive Anzeige für die Verwendung programmierter Symbole definiert werden. Dies bedeutet, dass alle Gruppen für programmierte Symbole der interaktiven Anzeige zugeordnet sein müssen.

Um lange Datenzeilen (z. B. Zeilen mit 132 Zeichen für die CEMT-Ausgabe) anzuzeigen, müssen Sie eine Standardanzeigenbreite von 132 Zeichen angeben.

Wenn Sie den Einsatz eines großen Puffers planen, dann müssen Sie für IOAREALEN (siehe „TYPETERM-Attribute“ auf Seite 279) möglicherweise einen erheblich höheren Wert angeben. Ob dies erforderlich ist, hängt davon ab, wie wahrscheinlich es ist, dass die Bediener große Datenmengen ändern oder eingeben. Wenn ein Terminal nur zu Abfragezwecken oder zur Dateneingabe in begrenztem Umfang verwendet wird, dann muss der Wert für IOAREALEN nicht hoch sein.

2. 3270-SDLC-Einheiten werden nur über z/OS Communications Server unterstützt. Lokal angeschlossene Drucker oder 3274- und 3276-SDLC-Einheiten werden über z/OS Communications Server entweder als LU Type 3-Einheiten (bei Verwendung des 3270-Druckerdatenstroms) oder als LU Type 1-Einheiten (bei Verwendung des SCS-Datenstroms) unterstützt. Beim SCS-Datenstrom handelt es sich um eine Untergruppe des Datenstroms, der für 3767-SDLC- sowie für 3770- und 3790-Drucker verwendet wird. Die 3288-Einheit wird als 3286 Modell 2 unterstützt. CICS unterstützt die 3270-Kopierfunktion (#1550).
3. Einheiten und Funktionen, die von einem System oder einem programmierbaren Controller unterstützt werden, sind im Allgemeinen für CICS transparent. In bestimmten Fällen stellt CICS eine spezielle Einheitenunterstützung zur Verfügung. Ist dies der Fall, werden die betreffenden Einheiten aufgelistet.
4. SDLC wird über z/OS Communications Server unterstützt. Die 3614-Einheit wird sowohl für den Schleifenanschluss an die 3601-Einheit als auch für den SDLC-Anschluss an den Host über einen 3704/3705-DFV-Controller unterstützt.

Die 3614-Einheit wird von CICS als BSC-Einheit nur dann unterstützt, wenn ein Schleifenanschluss zu den 3601/3602-Controllern verwendet wird. Die 3624-Einheit wird als 3614-Einheit unterstützt.

Die 3690-Einheit wird als 3602-Einheit unterstützt.

5. CICS unterstützt die Datenübertragungsfunktion der programmierbaren SDLC-Modelle für das 3770 Data Communication System. Bei Verwendung dieser Funktion sind Sie für die Zuordnung von Dateien und die Verwaltung der Programmbibliothek verantwortlich.
6. CICS bietet keine Unterstützung für 3790/Data Entry Configuration bei Verwendung von 3760-Einheiten. Die Konfiguration #9165 oder #9169 wird benötigt, um die CICS-Erweiterung zu unterstützen, die zuerst in Version 1 Release 3.0 bereitgestellt wurde.

Drucker auf 3790-Systemen werden auf eine der drei folgenden Arten unterstützt:

- Mit einem von Ihnen bereitgestellten Funktionsprogramm
- Über die 3270-Datenstromkompatibilität mit einem 3270-Druckerdatenstrom (LU3)
- Über einen SCS-Datenstrom, der eine Untergruppe des Datenstroms für 3767-SDLC-Einheiten (LU1) unterstützt

Beim Betrieb des Systems im 3270-Modus wird 3288 Modell 2 als 3286 Modell 2 unterstützt.

7. Der 4300-Anschluss via BSC erfordert ein geeignetes Telekommunikationsprogramm (z. B. VSE/3270 Bisync Pass Through Program) auf dem System, das über eine Verbindung zu CICS verfügt. CICS stellt **kein** geeignetes Programm für die ferne CPU dar. Der SDLC-Anschluss wird über die systemübergreifende Kommunikation von CICS unterstützt.
8. System/32 mit dem Dienstprogramm für SNA/SDLC-Workstationsysteme sowie System/34 und System/38 werden als kompatible Versionen eines mit den entsprechenden Funktionen ausgestatteten 3770 Communication System unterstützt, das als logische Stapelverarbeitungseinheit ausgeführt wird. Das vom Benutzer geschriebene System/34- oder System/38-Programm ist für die Unterstützung der korrekten SNA-Sequenzen des angeschlossenen Subsystems verantwortlich.
9. TWX und WTTY werden über z/OS Communications Server durch das Network Terminal Option-Lizenzprogramm (5735-XX7) mit den von NTO definierten Anschlüssen unterstützt.

WTTY wird mit 50 Bit/s über Netze mit Fernmeldewählleitungen angeschlossen. Hier werden Terminals mit einer Schnittstelle über IBM World Trade Corporation Telegraph Terminal Control with Telegraph Line Adapter unterstützt.

Als Übertragungscode wird International Telegraph Alphabet No. 2 (CCITT No. 2) verwendet. CICS bietet keine Unterstützung für automatische Anrufe oder die automatische Hostverbindungsunterbrechung mithilfe von WRITE-Breaks.

10. Die 3643-Einheit wird als 3604-Einheit unterstützt.
11. Die 8775-Unterstützung umfasst die Validierung der Feldattribute MANDATORY FILL und MANDATORY ENTER.
12. Der Anschluss erfolgt über den Leitungsschleifenadapter #4830, #4831 und den Datenverbindungsadapter #4840 des 4331-Prozessors.
13. Für die 3270-Unterstützung ist es erforderlich, dass der 3278/3279-Emulationsadapter auf dem Personal Computer oder PS/2 installiert wird.
14. Die logische 3600-Pipelineeinheit dient zur Bereitstellung hoher Durchsätze für bestimmte Transaktionstypen wie z. B. die Kreditkartenberechtigung oder allgemeine Abfrageanwendungen. Zur Erzielung einer hohen Durchsatzrate für Abfragenachrichten und die zugehörigen Antworten verwendet die CICS-



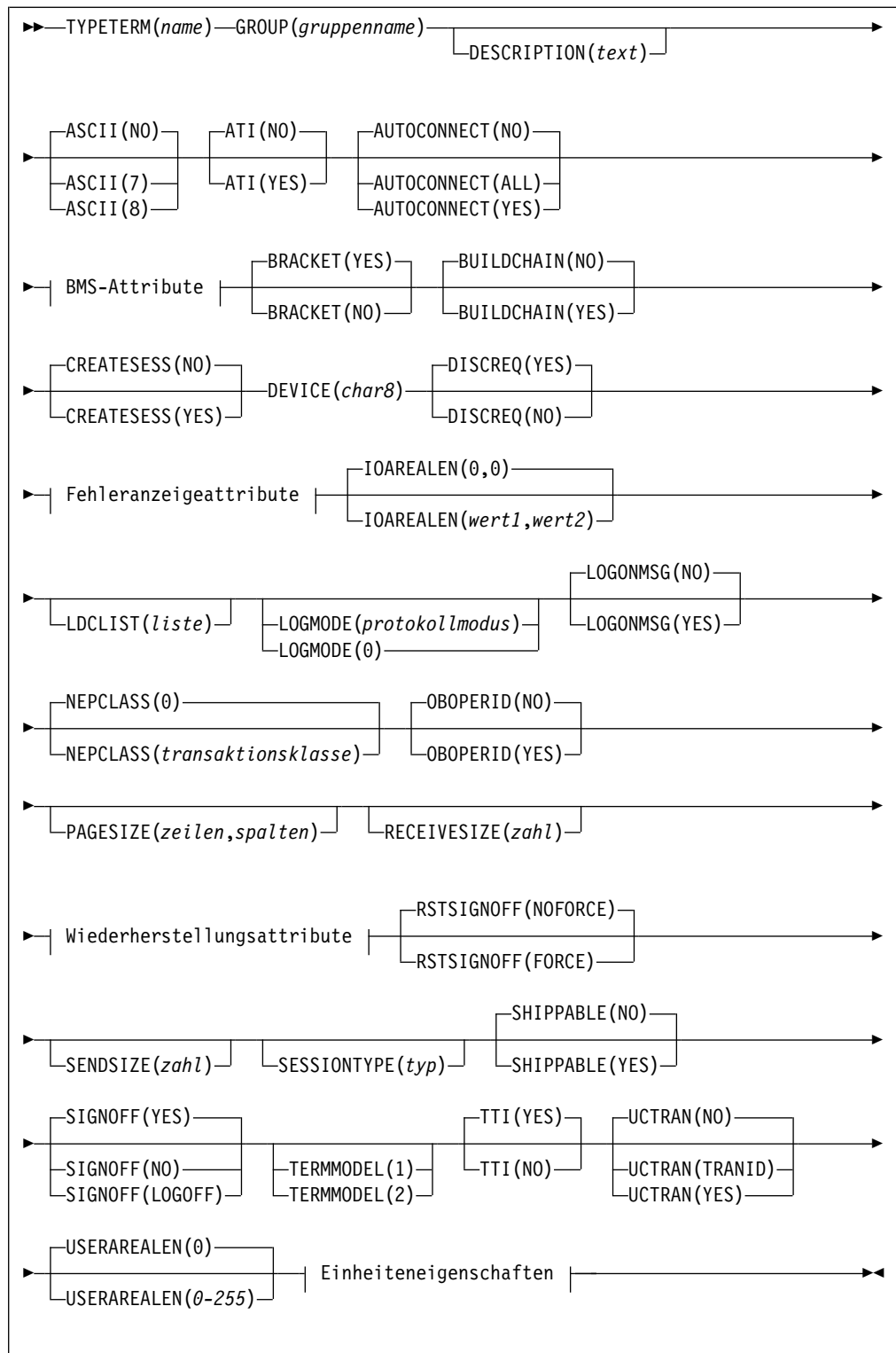
Pipelinesitzung eine eingeschränkte Gruppe von Kommunikationsprotokollen, die für die logische 3601-Einheit verwendet werden.

Diese Einschränkungen führen zu einem Vollduplexnachrichtenfluss, wodurch zu einem bestimmten Zeitpunkt zahlreiche Abfragennachrichten ausstehend sind und wodurch der Rückfluss der Antworten in einer Reihenfolge erfolgen kann, die sich von der Reihenfolge der ursprünglichen Abfragen unterscheidet. Das 4700/3600-Anwendungsprogramm, das zur Steuerung der Abfrage-terminals eingesetzt wird, ist für die Verwaltung der Protokolle sowie die Korrelation der Antworten mit den Abfragen und auch die Steuerung des Nachrichtenflusses zu der Gruppe von Terminals verantwortlich, die der logischen Pipelineinheit zugeordnet ist.

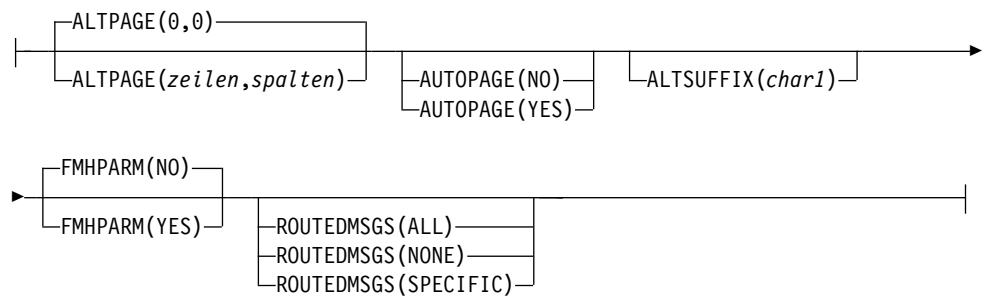
CICS bietet keine Unterstützung für die automatische Transaktionsinitialisierung (ATI = Automatic Transaction Initiation) auf Pipeline-Terminals.

## **TYPETERM-Attribute**

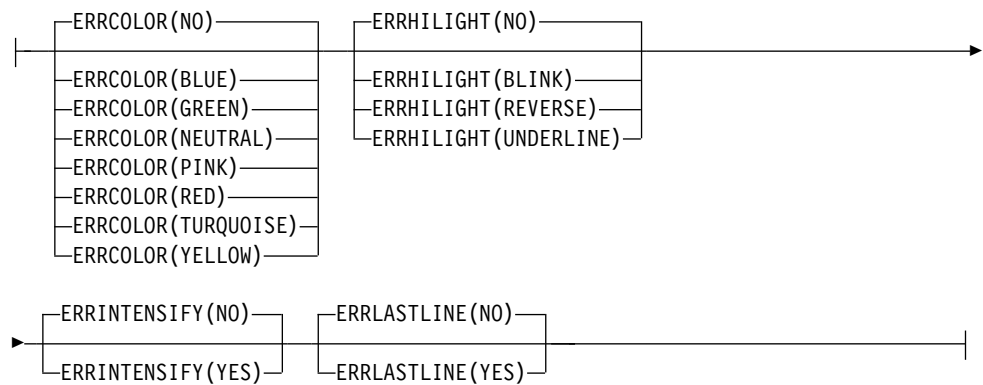
Beschreibt die Syntax und die Attribute der TYPETERM-Ressource.



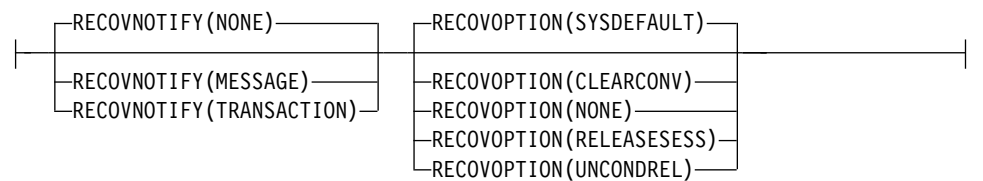
### BMS-Attribute:



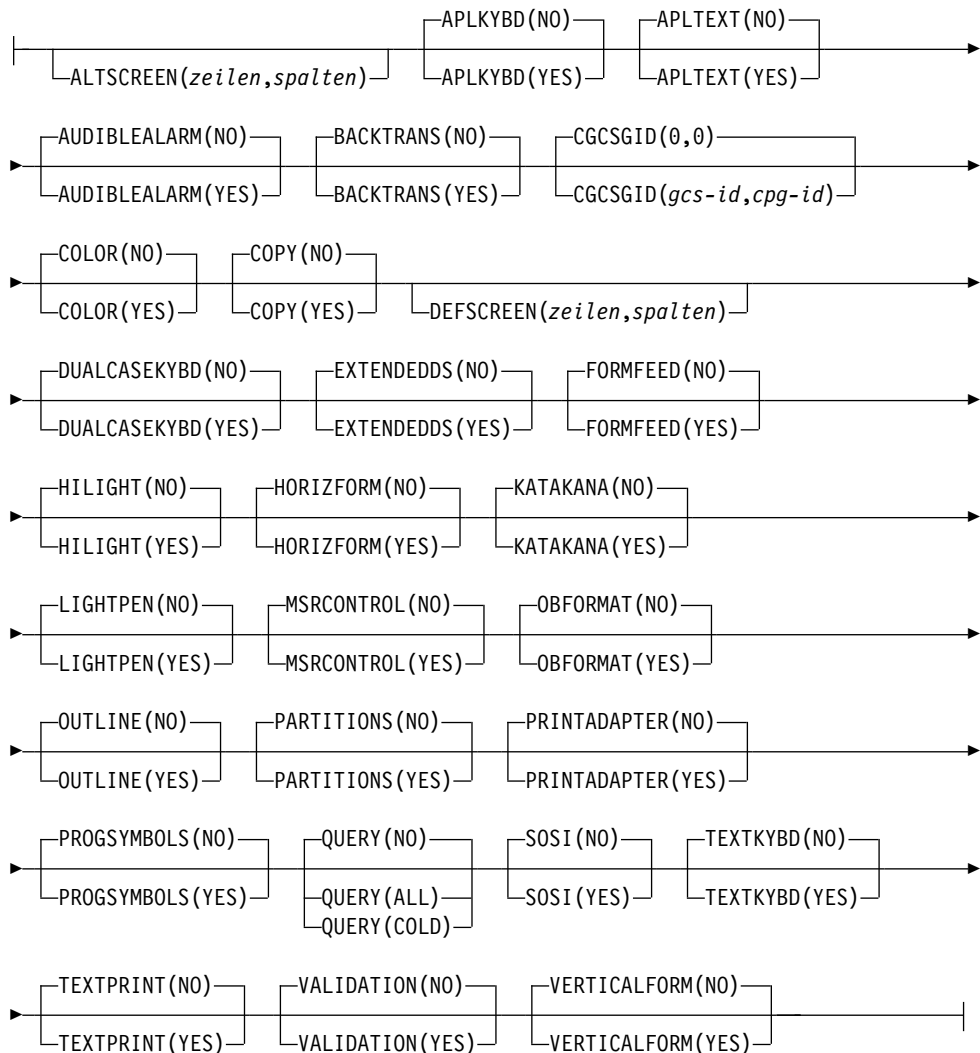
### Fehleranzeigeattribute:



### Wiederherstellungsattribute:



### Einheiteneigenschaften:



Die Beschreibungen der Attribute für die TYPETERM-Definition lauten wie folgt:

#### **ALTPAGE({0|zeilen},{0|spalten})**

Gibt die Seitengröße an, die von BMS für diesen Terminaleintrag verwendet werden soll, wenn für die Anzeigegröße ALTSCREEN ausgewählt wurde. Der Standardwert ist PAGESIZE. Die Werte für *zeilen* und *spalten* müssen im Bereich zwischen 0 und 255 liegen. Das Produkt von *zeilen* und *spalten* darf maximal 32767 betragen.

Sie erhalten nicht erwartete Ergebnisse, wenn der Wert für *spalten* in ALTPAGE sich vom entsprechenden Wert für ALTSCREEN unterscheidet. Der Wert für *zeilen* von ALTPAGE kann kleiner sein als der entsprechende Wert für ALTSCREEN, um ggf. die unterste Zeile der Anzeige für Fehlernachrichten zu reservieren.

Wenn Sie das strukturierte Feld QUERY verwenden, dann entspricht die verwendete alternative Seitengröße der Größe, die als alternative Anzeigegröße festgelegt wurde. Für Terminals, die abgefragt werden können, können Sie den

Wert für ALTPAGE auf null setzen und den Wert für ALTSCREEN explizit mit CINIT BIND definieren. Wird für ALTPAGE ein Wert ungleich null angegeben, ist es möglich, für ALTPAGE und ALTSCREEN unterschiedliche Werte zu verwenden.

#### **ALTSCREEN**(*zeilen,spalten*)

Gibt die 3270-Anzeigegröße an, die für eine Transaktion verwendet werden soll, für die eine alternative Anzeigegröße in der Profildefinition angegeben ist. Die Werte für *zeilen* und *spalten* müssen im Bereich zwischen 0 und 255 liegen. Folgende Werte können angegeben werden:

Einheit	Alternative Anzeigegröße
3276-1, 3278-1	(12,80)
3276-2, 3278-2	(24,80)
3276-3, 3278-3	(32,80)
3276-4, 3278-4	(43,80)
3278-5	(27,132)
3279-2A, 3279-2B	(24,80)
3279-3A, 3279-3B	(32,80)

Für die ausgewählte Anzeigegröße wird keine Gültigkeitsprüfung durchgeführt und fehlerhafte Größen führen möglicherweise zu unvorhersehbaren Ergebnissen.

Für BSC-Einheiten werden sowohl die alternative als auch die standardmäßige Anzeigegröße über die Einheitenhardware ermittelt. Die alternative Anzeigegröße stellt hierbei die maximal zulässige Anzeigegröße dar. Für die 3290-Anzeige werden sowohl die standardmäßige als auch die alternative Anzeigegröße anhand der Installationsprozedur des Kunden bestimmt.

Für SNA-Einheiten (LUTYPE2 und LUTYPE3) können Sie einen beliebigen Wert für die alternative und die standardmäßige Anzeigegröße angeben, deren Maximalwert der physischen Anzeigegröße entspricht. Sowohl die alternative als auch die standardmäßige Anzeigegröße können der maximal zulässigen Anzeigegröße entsprechen. Alternativ hierzu kann die standardmäßige Anzeigegröße auch der maximalen Anzeigegröße entsprechen, wobei keine alternative Anzeigegröße angegeben wird. Die SNA-Bindung wird von CICS anhand dieser Informationen generiert. Sie müssen keine Protokollmodustabelleneinträge angeben oder die Einheit anpassen.

Für Nicht-SNA-3270-Einheiten und -LUTYPE2-Einheiten können Sie das strukturierte Feld QUERY verwenden, um die alternative Anzeigegröße zu ermitteln, die für die Anzeige festgelegt wurde. Zur Verwendung von QUERY müssen Sie für DEFSCREEN den Standardwert (24,80) übernehmen und für ALTSCREEN keinen Wert angeben. Die alternative Anzeigegröße stellt die Größe dar, die vom Terminalbenutzer festgelegt wurde. Andernfalls hat die Angabe QUERY(COLD) oder QUERY(ALL) keine Auswirkung auf die alternative Anzeigegröße. Wird ALTSCREEN nicht angegeben, ohne dass QUERY verwendet wird, dann führt dies in den beschriebenen Situationen zur alternativen Anzeigegröße (00,00).

Wenn Sie zwei Anzeigegrößen verwenden, dann können Sie festlegen, dass eine CICS-Transaktion die alternative Anzeigegröße verwenden soll, indem Sie im zugehörigen Profil SCRNSIZE(ALTERNATE) angeben. Wenn eine Anwendung aus mehreren Transaktionen besteht, die im Pseudodialogmodus verbun-

den sind, dann geben Sie im Profil dieser Transaktionen SCRNSIZE(ALTERNATE) an, wenn die Anwendung die alternative Anzeigegröße verwendet.

Für 3287- und 3289-Drucker muss der angegebene Wert gleich der Puffergröße der betreffenden Einheit sein. Für Nicht-SNA-3287- und -3289-Drucker hängen die Größen vom bestellten Feature und nicht von der Modellnummer ab. Bei SNA-Druckern sind keine Features vorhanden und Sie können zwei beliebige Größen aus der Liste der zulässigen Größenangaben angeben. Beim Drucken auf einem Drucker, in dessen TERMINAL-Definition PRINTERCOPY(YES) angegeben ist, muss der Wert für ALTSCREEN mit der Anzeigegröße des Terminals übereinstimmen, dessen Anzeige gedruckt werden soll. Wenn die Werte abweichen, dann können unvorhersehbare Ergebnisse die Folge sein.

#### **ALTSUFFIX({leer|zahl})**

Ein aus einem Zeichen bestehendes numerisches Suffix, das von BMS an Maskengruppennamen (Angabe im Operanden SUFFIX des Makros DFHMSD TYPE={DSECT|MAP}) angefügt wird.

**leer** Geben Sie für dieses Attribut keinen Wert an, wenn die Maskengruppe kein Suffix erhalten soll.

**zahl** BMS fügt dieses Suffix zu den Maskengruppennamen hinzu, wenn die benutzte Anzeigegröße denselben Wert wie die alternative Anzeigegröße hat, d. h., wenn die Transaktion über eine alternative Anzeigegröße verfügt, die in der PROFILE-Definition angegeben ist, oder wenn die standardmäßige und die alternative Anzeigegröße identisch sind. In diesem Fall versuchen die BMS-Maskenauswahlroutinen, die Maskengruppe mit dem im Operanden ALTSUFFIX angegebenen Suffix zu laden.

Wenn keine solche Maskengruppe existiert, dann versucht BMS, eine Maskengruppe mit dem Suffix M oder L zu laden. Schlägt dieser Ladeversuch fehl, dann versucht BMS, eine Maskengruppenversion ohne Suffix zu laden.

Wenn die Transaktion die Standardanzeigegröße verwendet, dann versucht BMS zuerst, eine Maskengruppe mit dem Suffix M oder L zu laden. Schlägt diese Operation fehl, dann versucht BMS, eine Maskengruppenversion ohne Suffix zu laden.

Zur Verwendung einer Maskengruppe mit Suffix müssen Sie den Systeminitialisierungsparameter BMS=(„DDS) angeben.

#### **APLKYBD({NO|YES})**

Gibt an, ob die 3270-Einheit über die APL-Tastaturfunktion verfügt:

**YES** Die 3270-Einheit verfügt über die APL-Tastaturfunktion.

**NO** Die 3270-Einheit verfügt nicht über die APL-Tastaturfunktion.

#### **APLTEXT({NO|YES})**

Gibt an, ob die 3270-Einheit über die APL-Textfunktion verfügt:

**YES** Die 3270-Einheit verfügt über die APL-Textfunktion.

**NO** Die 3270-Einheit verfügt nicht über die APL-Textfunktion.

Geben Sie für einen 3288-Drucker mit oder ohne TEXTPRINT(YES) nicht YES an. Die APLTEXT-Funktion wird zusammen mit den Operanden TEXTKYBD und APLKYBD verwendet.

Sie können das strukturierte Feld QUERY verwenden, um festzustellen, ob die Einheit für die Verwendung der APL-Textfunktion eingerichtet wurde.

## ASCII({NO|7|8})

Gibt an, ob das Terminal über eine ASCII-Funktion verfügt.

**NO** Dieses Terminal verfügt nicht über eine ASCII-Funktion.

**7** Geben Sie diesen Wert an, um mit ASCII-7-Terminals zu kommunizieren. Einheiten, die mit der ASCII-7-Funktion konfiguriert wurden, müssen LUTYPE2- oder LUTYPE3-Einheiten ohne erweiterte 3270-Funktionen sein. Nur die folgenden Einheiten werden unterstützt:

3274 Modell 1C und 51C

3276 Modell 12

3278

3287

Für alle mit der ASCII-7-Option konfigurierten Terminals werden sämtliche von CICS abgehenden FM-Daten ins ASCII-7-Format und alle in CICS eingehenden FM-Daten ins EBCDIC-Format konvertiert. Nur FM-Anforderungsdaten werden umgesetzt. Bei allen anderen Daten in der Anforderungseinheit (RU = Request Unit) wie z. B. den Daten zum LU-Status oder den Prüfdaten wird davon ausgegangen, dass sie sich bei der Ausgabe im EBCDIC-Format befinden. ASCII-7 bietet **keine** Unterstützung für Datenströme, die erweiterte Attribute enthalten, wie dies z. B. für strukturierte Felder oder Funktionsverwaltungsheader der Fall ist.

Die ASCII-7-Unterstützung ist für 3274-1C als Option in der Konfiguration des Standardmikrocodes verfügbar. Die Verwendung der ASCII-7-Option wird bei der Sitzungsinitialisierung mit den BIND-Parametern festgelegt, die von CICS als Ergebnis der beschriebenen TCT-Definition definiert werden.

**8** Geben Sie diesen Wert an, um mit ASCII-8-Terminals zu kommunizieren. Bei den mit der ASCII-8-Funktion konfigurierten Einheiten kann es sich um LUTYPE1-, LUTYPE2- oder LUTYPE3-Einheiten mit oder ohne die erweiterten 3270- und SCS-Datenstromfunktionen handeln.

Für alle mit der ASCII-8-Option konfigurierten Terminals werden sämtliche von CICS abgehenden FM-Daten ins ASCII-8-Format und alle in CICS eingehenden FM-Daten ins EBCDIC-Format konvertiert. Alle FM-Anforderungsdaten werden umgesetzt. Dieser Vorgang umfasst die AID, die Cursoradresse, die FM-Header und die strukturierten Felder. Für alle anderen RU-Formen wie z. B. für die Daten zum LU-Status oder die Prüfdaten wird angenommen, dass sie sich bei der Eingabe im EBCDIC-Format befinden und dass sie bei der Ausgabe im EBCDIC-Format übertragen werden.

Diese ASCII-8-Unterstützung ist nur für Einheiten vorgesehen, die in EBCDIC arbeiten, den Datenstrom aber in oder aus ASCII-8 umsetzen bzw. neu umsetzen, wie dies bei der vorliegenden CICS-Unterstützung der Fall ist. Dies ist darauf zurückzuführen, dass der Datenstrom als Zeichenfolge behandelt wird und dass alle binären numerischen Felder byteweise umgesetzt werden, wie dies auch bei Grafikzeichen der Fall ist. Aus diesem Grund stellen sie möglicherweise nicht den wahren Wert dar, während sie im ASCII-Format angegeben sind.

Die ASCII-8-Unterstützung steht als Mikrocode-RPQ auf 3274-Einheiten zur Verfügung und kann nicht zusammen mit der ASCII-7-Option angegeben werden. Die Verwendung der ASCII-8-Option wird bei der

Sitzungsinitialisierung mit den BIND-Parametern festgelegt, die von CICS als Ergebnis der beschriebenen TCT-Definitionen festgelegt werden.

**ATI({NO|YES})**

Gibt an, ob Transaktionen am Terminal durch die automatische Transaktionsinitialisierung gestartet werden können:

**YES** Gibt an, dass Transaktionen am Terminal durch die automatische Transaktionsinitialisierung gestartet werden können.

**NO** Gibt an, dass Transaktionen am Terminal nicht durch die automatische Transaktionsinitialisierung gestartet werden können.

Die Angabe von ATI(YES) ermöglicht das Starten von Transaktionen am Terminal über die Steuerfunktion für transiente Daten oder mithilfe des Befehls EXEC CICS START, der von einer anderen Transaktion abgesetzt wird. Wenn auf dem Terminal bereits eine Transaktion definiert ist, dann wird die ATI-Transaktion angehalten, bis diese abgeschlossen ist. Wenn Sie ATI(YES) angeben, dann müssen Sie einen Wert für IOAREALEN mit mindestens einem Byte angeben. Eine Ausnahme bildet die Verwendung von DEVICE(APPC), wenn für ATI und IOAREALEN die erzwungenen Standardwerte YES und 0 verwendet werden.

Wenn für ATI die Option YES und für CREATESESS die Option YES angegeben ist und wenn eine Transaktion eingeleitet wird, wenn für das Terminal nicht ACQUIRED angegeben ist, dann erfolgt die Anforderung automatisch.

Weitere Informationen hierzu finden Sie in der Beschreibung zum Attribut TTI.

**AUDIBLEALARM({NO|YES})**

Gibt an, ob die Funktion für akustische Signale für einen 3270-Bildschirm oder für einen 3270-Drucker installiert wurde, der an einen 3651-Controller angeschlossen ist.

**YES** Die Funktion für akustische Signale wird installiert.

**NO** Die Funktion für akustische Signale wird nicht installiert.

**AUTOCONNECT({NO|YES|ALL})**

Gibt an, ob die Verarbeitung für die automatische Verbindung für das Terminal ausgeführt werden soll. AUTOCONNECT(YES) oder (ALL) gibt an, dass die Sitzung mit dem Terminal während der CICS-Initialisierung oder beim Start der Kommunikation mit z/OS Communications Server mit dem Befehl **SET VTAM OPEN** eingerichtet werden soll (d. h., dass BIND ausgeführt werden muss). Wenn zu diesem Zeitpunkt keine Verbindung hergestellt werden kann, weil das Terminal nicht verfügbar ist, dann muss die Verbindung nachfolgend mit dem Befehl CEMT SET TERMINAL(*termid*) INSERVICE ACQUIRED angefordert werden, es sei denn, das Terminal wird in der Zwischenzeit wieder verfügbar und kann die Kommunikation selbst einleiten.

**Anmerkung:** Wenn Sie die LOGAPPL-Funktion von z/OS Communications Server verwenden, dann dürfen Sie AUTOCONNECT(YES) nicht angeben, weil dies zu Racebedingungen führen kann, die Fehler oder das Blockieren logischer Einheiten zur Folge haben können.

**NO** CICS versucht nicht, bei der Verbindungsherstellung Sitzungsbindungen durchzuführen.

**YES** CICS versucht bei der Verbindungsherstellung, die Bindung als Konfliktgewinnersitzung herzustellen.



**ALL** Nicht zutreffend.

Hintergrundinformationen zu AUTOCONNECT finden Sie in Defining inter-communication resources in Configuring.

**AUTOPAGE({NO|YES})**

Gibt an, ob das BMS-Autopaging verwendet werden soll. Geben Sie für Drucker YES und für Bildschirmeinheiten NO an. Der Standardwert ist von dem Wert abhängig, den Sie für das Attribut DEVICE angegeben haben. Die Standardwerte werden in „Standardwerte für TYPETERM-Attribute“ auf Seite 267 angegeben.

**BACKTRANS({NO|YES})**

Gibt an, ob die Einheit über die Hintergrundtransparenzfunktion verfügt:

**NO** Die Einheit verfügt nicht über die Hintergrundtransparenzfunktion.

**YES** Die Einheit verfügt über die Hintergrundtransparenzfunktion.

Sie können das strukturierte Feld QUERY verwenden, um festzustellen, ob die Einheit für die Verwendung der Hintergrundtransparenzfunktion eingerichtet wurde.

**BRACKET({YES|NO})**

Gibt an, ob die Verwendung des Klammerprotokolls für diese logische Einheit erzwungen werden soll. Der Standardwert ist von dem Wert abhängig, den Sie für das Attribut DEVICE (siehe „Standardwerte für TYPETERM-Attribute“ auf Seite 267) angegeben haben.

**YES** Das Klammerprotokoll soll verwendet werden. Diese Option ist für die 3790-Abfrage und für logische Einheiten mit vollem Funktionsumfang erforderlich. Die Verwendung von BRACKET(YES) wird für zahlreiche DEVICE-Typen erzwungen.

**NO** Das Klammerprotokoll soll nicht verwendet werden. Sie müssen BRACKET(NO) für eine logische 3614-Einheit und für Sitzungen des 3650 Host Command Processor (HCP) angeben.

Beim Klammerprotokoll handelt es sich um eine SNA-Funktion. Wenn Sie für Nicht-SNA-Einheiten BRACKET(YES) angeben, dann wird die strikte Befolgung des Klammerprotokolls von CICS weder durchgeführt noch erzwungen.

**BUILDCHAIN({NO|YES})**

Gibt an, ob CICS die Kettenassemblierung ausführen soll, bevor die Eingabedaten an das Anwendungsprogramm übergeben werden.

Der Standardwert ist von dem Wert abhängig, den Sie für das Attribut DEVICE angegeben haben.

**NO** Jeder Terminal-E/A-Bereich (TIOA = Terminal Input/Output Area), der durch ein Anwendungsprogramm von dieser logischen Einheit empfangen wird, enthält eine Anforderungseinheit (RU = Request Unit).

**YES** Jeder Terminal-E/A-Bereich (TIOA), der durch ein Anwendungsprogramm von dieser logischen Einheit empfangen wird, enthält eine komplette Kette.

**CGCSGID({0,0|wert1,wert2})**

Die globale ID des codierten Grafikzeichensatzes (CGCSGID) ermöglicht Anwendungsprogrammen die Feststellung des von einer Einheit unterstützten Zeichensatzes.

Sie können diese Informationen für bestimmte Einheiten aus dem strukturierten Feld QUERY ablesen. Für andere Einheiten müssen Sie diese Informationen hier angeben, sodass die Anwendungsprogramme sie mit dem Befehl **EXEC CICS ASSIGN** abrufen können.

**0,0** Es wird kein Wert für CGCSGID angegeben.

*gcs-id,cpg-id*

Der Wert für CGCSGID besteht aus zwei fünfstelligen Dezimalzahlen, die im Bereich zwischen 1 und 65535 liegen können. *gcs-id* steht für die globale ID des Grafikzeichensatzes (GCSGID) und *cpg-id* gibt die Codepunkte des Zeichensatzes (globale Codepage-ID; CPGID) an.

**COLOR({NO|YES})**

Gibt an, ob die 3270-Einheit oder der SCS-Drucker über die erweiterte Farbfunktion verfügt, die die Auswahl von Farben für jedes Feld oder Zeichen ermöglicht.

**NO** Die Einheit verfügt nicht über die erweiterte Farbfunktion.

**YES** Die Einheit verfügt über die erweiterte Farbfunktion.

Sie können das strukturierte Feld QUERY verwenden, um festzustellen, ob die Einheit für die Verwendung der Farbfunktion eingerichtet wurde.

**COPY({NO|YES})**

Gibt an, ob die Kopierfunktion für eine 3270-Bildschirmeinheit oder einen 3270-Drucker in den 3270-Controller integriert ist:

**NO** Die Kopierfunktion ist eingeschlossen.

**YES** Die Kopierfunktion ist nicht eingeschlossen.

Behalten Sie für logische 3270-Einheiten mit Kompatibilitätsmodus den Standardwert COPY(NO) bei, weil COPY(YES) ignoriert wird.

Weitere Informationen hierzu finden Sie in den Beschreibungen der Attribute PRINTERCOPY und ALTPRINTCOPY der TERMINAL-Definition.

Weitere Einzelheiten zum Erstellen von Anzeigenkopien finden Sie im Handbuch *CICS 3270 Data Stream Device Guide*.

**CREATESESS({NO|YES})**

Gibt an, ob Sitzungen erstellt werden sollen.

**NO** Geben Sie diese Option an, um zu verhindern, dass mit intern generierten Sitzungsanforderungen eine Sitzung erstellt wird. Während der Ausführung von CICS kann diese Einstellung nur mit einem CEMT-Befehl geändert werden.

Die Angabe von CREATESESS(NO) verhindert, dass mithilfe von EXEC START-Anforderungen und ATI-Anforderungen (ATI = Automatic Transaction Initiation; automatische Transaktionsinitialisierung) Sitzungen für dieses Terminal erstellt werden können. Dies bedeutet, dass die Anforderungen entweder in die Warteschlange eingestellt oder zurückgewiesen werden, wenn momentan keine Sitzung eingerichtet ist.

**YES** Geben Sie diese Option für einen Status an, in dem mit intern generierten Sitzungsanforderungen eine Sitzung erstellt werden kann. Während der Ausführung von CICS kann dieser Status nur mit einem CEMT-Befehl generiert werden.

Die Angabe von CREATSESS(YES) lässt zu, dass mithilfe von EXEC START-Anforderungen und ATI-Anforderungen für dieses Terminal automatisch eine Sitzung erstellt werden kann.

#### **DEFSCREEN**(*zeilen , spalten*)

Gibt die 3270-Anzeigegröße oder die 3270-Druckerseitengröße an, die auf dieser Einheit verwendet werden soll, wenn eine Verbindung zu einer Transaktion besteht. Bei Verwendung durch BMS ist die Standardanzeigegröße in der Profildefinition angegeben. Die Werte für *zeilen* und *spalten* müssen im Bereich zwischen 0 und 255 liegen. Der Standardwert ist von dem Wert abhängig, den Sie für das Attribut DEVICE (siehe „Standardwerte für TYPETERM-Attribute“ auf Seite 267) angegeben haben. Folgende Werte können für eine BSC-3270-Einheit angegeben werden:

Einheit	Anzeigegröße
3278-1	(12,40)
3278-2	(24,80)
3276-3, 3278-3	(24,80)
3276-4, 3278-4	(24,80)
3278-5	(24,80)
3279-2A, 3279-2B	(24,80)
3279-3A, 3279-3B	(24,80)

Für BSC-Einheiten werden sowohl die standardmäßige als auch die alternative Anzeigegröße über die Terminalhardware ermittelt. Die Standardanzeigegröße ist (24,80). Eine Ausnahme bildet die 3278-1-Einheit, deren Standardanzeigegröße (12,40) ist.

Für SNA-Einheiten (LUTYPE2 und LUTYPE3) können Sie für die standardmäßige und die alternative Anzeigegröße einen beliebigen Wert auswählen, sofern dieser nicht die maximale physische Anzeigegröße (siehe ALTSCREEN) überschreitet. Sowohl die standardmäßige als auch die alternative Anzeigegröße können der maximal zulässigen Anzeigegröße entsprechen. Alternativ hierzu kann die standardmäßige Anzeigegröße der maximalen Anzeigegröße entsprechen, wobei keine alternative Anzeigegröße angegeben wird. Die SNA-Bindung wird von CICS anhand dieser TCT-Informationen generiert. Sie müssen keine Protokollmodustabelleneinträge angeben oder die Einheit anpassen.

#### **DESCRIPTION**(*text*)

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

#### **DEVICE**(*einheit*)

Gibt den Einheitentyp an, der in diesem TYPETERM-Wert definiert ist. Dieses Attribut muss in allen TYPETERM-Definitionen zwingend angegeben werden.

Wenn Sie in der Befehlszeile DEVICE(*xxxx*) zusammen mit den Attributen SESSIONTYPE und TERMMODEL angeben, dann werden bei korrekter Eingabe den anderen Attributen die geeigneten Standardwerte zugeordnet. (Die Variable *xxxx* steht hierbei für einen gültigen Einheitentyp.) Weitere Anleitungen zu diesem Thema finden Sie in „Standardwerte für TYPETERM-Attribute“ auf Seite 267-

te 267. Wenn Sie die Werte für DEVICE, SESSIONTYPE oder TERMMODEL in der Anzeige zum Ändern durch Überschreibung (Overtyping) eingeben oder überschreiben, dann werden diese Standardwerte **nicht** angegeben.

Die gültigen Attribute und die Standardwerte für die einzelnen Einheitentypen werden in „Standardwerte für TYPETERM-Attribute“ auf Seite 267 aufgelistet. Die empfohlenen Attribute für Nicht-SNA-3270-Einheiten für z/OS Communications Server sind 3270 und 3270P für Bildschirme bzw. Drucker. Die folgenden Attribute können ebenfalls angegeben werden und werden zur Erhaltung der Kompatibilität mit Vorgängerreleases beibehalten:

- Bildschirme: 3277 und L3277
- Drucker: 3284 und L3284, 3286 und L3286

Für SNA-3270-Einheiten für z/OS Communications Server verwenden Sie das Attribut LUTYPE2 bzw. LUTYPE3. Logische LUTYPE2-Einheiten werden von SNA definiert und akzeptieren einen 3270-Anzeigedatenstrom. Logische LUTYPE3-Einheiten werden von SNA definiert und akzeptieren einen Datenstrom ähnlich dem Datenstrom für einen 3270-Drucker.

Eine Liste der Einheitentypen, die von CICS unterstützt werden, finden Sie in „DFHTCT: CICS-Terminalliste“ auf Seite 474. Eine Liste mit den zulässigen Einheitenamen und den Standardattributen, die mit diesen generiert werden, finden Sie in „Standardwerte für TYPETERM-Attribute“ auf Seite 267.

#### **DISCREQ=({YES|NO})**

Gibt an, ob Unterbrechungsanforderungen berücksichtigt werden sollen.

**YES** CICS berücksichtigt eine Unterbrechungsanforderung für eine z/OS Communications Server-Einheit und setzt eine z/OS Communications Server-Makroinstruktion CLSDST ab, um die z/OS Communications Server-Sitzung mit dieser logischen Einheit zu beenden.

Darüber hinaus wird durch Angabe von CESF LOGOFF oder GOOD-NIGHT über das Terminal die Verbindung unterbrochen, wenn Sie YES angeben.

YES ist von zentraler Bedeutung, wenn auf die TYPETERM-Definition von AUTINSTMODEL TERMINAL-Definitionen verwiesen wird, so dass die automatisch installierten Terminaleinträge automatisch gelöscht werden können.

YES ist der Standardwert, in bestimmten Situationen, in denen Ihre Ressourcendefinition dieses Attribut nicht enthält, erzwingt CICS jedoch die Verwendung des Werts NO für das Attribut, wenn dieser Wert zur Wahrung der Kompatibilität mit anderen Optionen in Ihrer Ressourcendefinition erforderlich ist.

**NO** CICS berücksichtigt eine Unterbrechungsanforderung für eine z/OS Communications Server-Einheit nicht.

#### **DUALCASEKYBD({NO|YES})**

Gibt an, ob ein 3270-Bildschirmeinheit über eine Schreibmaschinentastatur oder eine Bedienerkonsolentastatur verfügt. Mit beiden Tastaturen können sowohl Daten in Groß- als auch in Kleinschreibung übertragen werden.

**NO** Die Einheit verfügt nicht über eine Groß-/Kleinschreibungstastatur.

**YES** Die Einheit verfügt über eine Groß-/Kleinschreibungstastatur.

#### **ERRCOLOR({NO|farbe})**

Gibt an, ob die Fehlnachricht in Farbe angezeigt werden soll. Die Angabe ERRCOLOR(farbe) impliziert die Angabe von ERRLASTLINE(YES).

Folgende Farben können angegeben werden:

BLUE  
RED  
PINK  
GREEN  
TURQUOISE  
YELLOW  
NEUTRAL

**ERRHIGHLIGHT**( { NO | BLINK | REVERSE | UNDERLINE } )

Gibt ggf. die Hervorhebung an, mit der Fehlernachrichten angezeigt werden sollen.

**ERRINTENSIFY**( { NO | YES } )

Gibt an, ob die Fehlernachricht in einem Feld mit intensiver Hervorhebung angezeigt werden soll. Die Angabe von ERRINTENSIFY(YES) impliziert die Angabe von ERRLASTLINE(YES).

**ERRLASTLINE**( { NO | YES } )

Gibt an, wo Fehlernachrichten angezeigt werden sollen.

**NO** Eine Fehlernachricht wird an der aktuellen Cursorposition und ohne zusätzliche Attribute angezeigt.

**YES** Eine Fehlernachricht wird beginnend am Anfang der untersten Zeile des Bildschirms so angezeigt, dass die gesamte Nachricht in die Anzeige passt.

Da alle Fehlernachrichten in derselben Zeile angezeigt werden, können sich diese Nachrichten beim Empfang in schneller Folge überlagern, so dass früher angezeigte Nachrichten möglicherweise bereits wieder ausgeblendet werden, bevor Sie sie lesen konnten.

**EXTENDEDDES**( { NO | YES } )

Gibt an, ob die 3270-Einheit oder der SCS-Drucker Unterstützung für Erweiterungen zum 3270-Datenstrom bieten:

**NO** Die Einheit bietet keine Unterstützung für 3270-Datenstromerweiterungen.

**YES** Die Einheit bietet Unterstützung für 3270-Datenstromerweiterungen.

Die Angabe von EXTENDEDDES(YES) wird impliziert, wenn Sie für eines der Attribute COLOR, HIGHLIGHT, PROGSYMBOLS, QUERY oder VALIDATION (nur 3270) YES angeben.

Wenn für den erweiterten Datenstrom (EXTENDEDDES) YES angegeben ist, dann unterstützt die Einheit das strukturierte Schreibfeld COMMAND und das strukturierte Feld "Outbound Query".

Sie können das strukturierte Feld QUERY verwenden, um festzustellen, ob die Einheit für die Verwendung des erweiterten Datenstroms eingerichtet wurde. Mithilfe des strukturierten Felds QUERY können Sie für EXTENDEDDES die Einstellung YES festlegen, wenn die Abfrage gültig ist.

**FMHPARM**( { NO | YES } )

Gibt an, ob BMS vom Benutzer bereitgestellte Parameter für die Einbindung in den Funktionsverwaltungsheader akzeptieren soll, der von BMS erstellt wird:

**NO** Es werden keine vom Benutzer bereitgestellten Parameter für die Einbindung in den Funktionsverwaltungsheader akzeptiert, der von BMS erstellt wird.

**YES** Es werden vom Benutzer bereitgestellten Parameter für die Einbindung in den Funktionsverwaltungsheader akzeptiert, der von BMS erstellt wird.

Geben Sie YES nur dann an, wenn der DEVICE-Typ 3650 lautet.

**FORMFEED({NO|YES})**

Gibt an, ob die Einheit über die Formularvorschubfunktion verfügt, was bedeutet, dass BMS das Formularvorschubzeichen bei der Formatierung von Ausgabedokumenten verwendet:

**NO** Die Einheit verfügt nicht über die Formularvorschubfunktion.

**YES** Die Einheit verfügt über die Formularvorschubfunktion.

Wenn DEVICE(SCSPRINT) angegeben wird, dann fügt BMS am Anfang des Datenstroms ein Formularvorschubzeichen ein. Dadurch springt die Einheit zum oberen Rand einer neuen Seite, bevor der Druckvorgang gestartet wird.

Der obere Rand wird von einem SVF-Datenstrom (SVF = Set Vertical Format; festgelegtes vertikales Format) definiert und kann als Zeilennummer angegeben werden, die größer-gleich 1 ist. Wenn kein SVF-Datenstrom an den Drucker gesendet wurde, dann befindet sich der obere Rand in Zeile 1. Der Zeilenzähler der Einheit ist auf den Wert 1 gesetzt, wenn der Bediener das Papier einrichtet.

Beachten Sie hierbei, dass die Einheit auch einen automatischen Formularvorschub ausführen kann, wenn Sie versuchen, über den unteren Rand hinaus zu drucken. Der untere Rand wird ebenfalls vom SVF-Datenstrom festgelegt. Standardmäßig wird hier der MPL-Wert (MPL = Maximum Presentation Line; maximale Präsentationszeile) verwendet. Die maximale Präsentationszeile gibt die letzte Zeile auf der Seite an und ihr Wert stellt die Seiten- oder Formularlänge als eine Anzahl von Zeilen (physische Seitengröße mal Zeilendichte) dar. Sowohl die maximale Präsentationszeile als auch die Zeilendichte können anhand des SVF-Datenstroms ermittelt werden. Andernfalls kann die maximale Präsentationszeile (Anzahl der Zeilen) auf der Einheit auch vom Bediener eingerichtet werden.

Wenn DEVICE(3270), DEVICE(3270P), DEVICE(LUTYPE2) oder DEVICE(LUTYPE3) angegeben wird, dann verwenden Sie FORMFEED(YES) zusammen mit der Option FORMFEED in den BMS-Befehlen SEND. Durch die Verwendung des Formularvorschubs auf Bildschirmeneinheiten kann auf eine neue Seite gesprungen werden, wenn die Anzeigedaten auf einen Drucker kopiert werden.

Die hier erläuterten Optionen für die Operation SCSPRINT gelten nicht, wenn die Einheiten als 3270P- oder LUTYPE3-Einheiten betrieben werden. In diesem Fall gilt nur das Konzept der Formularlänge. Diese Formularlänge kann auf der Einheit nur vom Bediener festgelegt werden.

**GROUP(*gruppenname*)**

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

**HIGHLIGHT({NO|YES})**

Gibt an, ob die 3270-Einheit oder der SCS-Drucker über die Funktion für die zusätzliche Hervorhebung verfügt. Mit dieser Funktion können Felder oder Zeichen in Umkehranzeige, im Unterstreichungsmodus oder blinkend (nur 3270) dargestellt werden:

**NO** Die Einheit verfügt nicht über die Funktion für die zusätzliche Hervorhebung.

**YES** Die Einheit verfügt über die Funktion für die zusätzliche Hervorhebung.

Sie können das strukturierte Feld QUERY verwenden, um festzustellen, ob die Einheit für die Verwendung der Funktion für die zusätzliche Hervorhebung eingerichtet wurde.

**HORIZFORM({NO|YES})**

Gibt an, ob die Einheit über die Funktion für horizontale Formulare verfügt, was bedeutet, dass BMS bei der Formatierung von Ausgabedokumenten die horizontale Tabulierung verwenden muss:

**NO** Die Einheit verfügt nicht über die Funktion für horizontale Formulare.

**YES** Die Einheit verfügt über die Funktion für horizontale Formulare.

Folgende Einheiten können diese Funktion verwenden: logische Stapelverarbeitungseinheiten, logische Stapeldatenaustauscheinheiten, interaktive logische Einheiten sowie logische SCSPT- oder LUTYPE4-Einheiten.

**NO** Die Option HTAB in der BMS-Maskendefinition wird ignoriert.

**YES** BMS verwendet die horizontale Tabulierung bei der Formatierung von Ausgabedokumenten.

**IOAREALEN({0|wert1},{0|wert2})**

Gibt die Länge (in Byte) eines Terminal-E/A-Bereichs an, die an eine Transaktion übergeben werden soll.

Wenn Sie ATI(YES) angeben, dann müssen Sie für IOAREALEN einen Wert angeben, der mindestens aus einem Byte besteht.

*wert1* In *wert1* wird die minimale Größe eines Terminal-E/A-Bereichs angegeben, die an ein Anwendungsprogramm übergeben wird, wenn ein Befehl **RECEIVE** abgesetzt wird.

*wert2* Sie können für *wert2* einen Wert größer-gleich dem Wert von *wert1* angeben. In diesem Fall verwendet CICS, wenn die Größe einer Eingabemessage den Wert von *wert1* überschreitet, einen Terminal-E/A-Bereich, dessen Länge der in *wert2* angegebenen Anzahl von Byte entspricht. Übersteigt die Größe der Eingabemessage auch den Wert für *wert2*, sendet das Programm für abnormale Bedingungen des Knotens eine Ausnahmeantwort an das Terminal.

Wenn *wert2* nicht angegeben wird oder wenn der Wert niedriger als der Wert für *wert1* ist, dann wird standardmäßig der Wert von *wert1* verwendet.

Der Maximalwert, den Sie für IOAREALEN angeben können, lautet 32767 Byte.

#### **KATAKANA({NO|YES})**

Gibt an, ob Katakana-Unterstützung erforderlich ist. Auf Katakana-Terminals ist es nicht möglich, Ausgabedaten in gemischter Groß-/Kleinschreibung anzuzeigen. Großbuchstaben erscheinen als englische Zeichen in Großschreibung, wohingegen Kleinbuchstaben als Katakana-Zeichen dargestellt werden. Wenn an Ihrem CICS-System Katakana-Terminals angeschlossen sind, dann geben Sie im Systeminitialisierungsparameter MSGCASE die Option UPPER an.

**NO** Die Katakana-Unterstützung ist nicht erforderlich.

**YES** Die Katakana-Unterstützung ist erforderlich. Alle Kleinbuchstaben, die von den folgenden Transaktionen an das Terminal gesendet werden, werden in Großbuchstaben umgesetzt:

CBRC CDBC CDBI CEBR CECI CEDA CEDF CEMT CEOT CESN  
CEST CMSG CRTE CSPG CWTO

**Important:** Bei emulierten Katakana-Terminals hängt das Ergebnis von der Codepage ab, die momentan verwendet wird. In bestimmten Fällen werden englische Zeichen in Kleinschreibung nicht in Großbuchstaben umgesetzt.

#### **LDCLIST(*liste*)**

Gibt den Namen einer LDC-Liste (LDC = Logical Device Code; Code für logische Einheiten) an. Der Name kann bis zu acht Zeichen lang sein. Der Name wird auf Basis der Regeln der Assemblersprache gebildet. Er muss mit einem alphabetischen Zeichen beginnen.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Definieren Sie den Wert für LDCLIST und die zugehörigen Inhalte mit Makroinstruktionen.

Eine lokale LDC-Liste wird wie folgt definiert:

```
listname DFHTCT TYPE=LDCLIST,  
LDC(aa=nnn,bb=nnn,...)
```

Eine erweiterte lokale LDC-Liste wird wie folgt definiert:

```
listname DFHTCT TYPE=LDC,LOCAL=INITIAL  
DFHTCT TYPE=LDC=(aa=nnn)....  
DFHTCT TYPE=LDC=(bb=nnn)....  
DFHTCT TYPE=LDC,LOCAL=FINAL
```

In dieser Variablen *listenname* geben Sie den Wert für das Attribut LDCLIST in der TYPETERM-Definition an.

Dieses Attribut gilt nur für logische 3600-Einheiten, logische 3770-Stapelverarbeitungseinheiten, logische 3770- und 3790-Stapeldatenaustauscheinheiten sowie logische LUTYPE4-Einheiten. Die Liste gibt an, welche LDCs für diese logische Einheit gültig sind und (optional) welche Einheitenmerkmale für jeden LDC gültig sind. CICS verwendet den ersten in dieser Liste generierten LDC bei der Auswahl eines Standard-LDC für eine logische Einheit. Weitere Anlei-



tungen zu diesem Thema finden Sie in „DFHTCT-Codes für logische Einheiten: z/OS Communications Server (nicht 3270)“ auf Seite 456.

#### **LIGHTPEN({NO|YES})**

Gibt an, ob ein 3270-Bildschirm über die Lichtstiftfunktion verfügt:

**NO** Die 3270-Bildschirmeinheit verfügt nicht über die Lichtstiftfunktion.

**YES** Die 3270-Bildschirmeinheit verfügt über die Lichtstiftfunktion.

#### **LOGMODE({leer|0|protokollmodus})**

Gibt an, wie CICS die Bindung (BIND) erstellen soll, die an die logische Einheit gesendet wird.

**leer** Eine definierte Terminaldefinition verwendet das BIND-Image, das anhand der CICS-Definitionen für diese Einheit mithilfe dieser TYPE-TERM-Definition und der zugehörigen Terminaldefinitionen generiert wird. Ein automatisch installiertes Terminal verwendet die Felder, die im eingehenden CINIT-Element angegeben wurden.

**name** Hierbei handelt es sich um den LOGMODE-Namen aus einer z/OS Communications Server-Anmeldemodustabelle, die zur Verwendung durch diese logische Einheit eingerichtet wurde. Der Name kann bis zu acht Zeichen lang sein und muss auf Basis der Regeln der Assemblersprache gebildet werden. Der Name muss mit einem alphabetischen Zeichen beginnen.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Auf diese Weise ist es möglich, das von CICS für die logische Einheit bereitgestellte BIND-Image zu überschreiben. Weitere Informationen zu diesem Thema finden Sie im CICS-Handbuch für das jeweilige Subsystem.

Der LOGMODE-Name kann nicht angegeben werden, wenn das Terminal eine Fremddomänenressource ist.

Der TCT-Terminaleintrag (TCTTE) wird aktualisiert, um die Felder für das BIND-Image des Protokollmodus wiederzugeben. Dazu gehören die Größenangaben für SEND und RECEIVE sowie die standardmäßige und alternative Anzeigegröße. Wenn der Protokollmodus angibt, dass das Terminal nicht abgefragt werden kann, dann enthalten die Felder für den erweiterten Datenstrom alle den Wert null.

#### **0 (null)**

Diese Einstellung veranlasst CICS zur Verwendung bestimmter Informationen aus dem BIND-Image, die im CINIT-Element enthalten sind, das von der logischen Einheit stammt. Das BIND-Image im CINIT-Element wurde mit z/OS Communications Server auf Basis des LOGMODE-Eintrags erstellt, der für die logische Einheit definiert wurde, die die Anmeldung bei CICS angefordert hat. Der Knoteninitialisierungsblock (NIB = Node Initialization Block) wird mit LOGMODE=0 und BNDAREA=0 erstellt. Wenn für SENDSIZE und RECEIVESIZE in der TYPETERM-Definition null angegeben wurde, dann ersetzt CICS sie durch die RUSIZES-Werte für LOGMODE.

Der TCT-Terminaleintrag (TCTTE) wird aktualisiert, um die Felder für eingehende CINIT-Elemente wiederzugeben. Dazu gehören die Grö-

ßenangaben für SEND und RECEIVE sowie die standardmäßige und alternative Anzeigegröße. Wenn der Protokollmodus angibt, dass das Terminal nicht abgefragt werden kann, dann werden alle Felder für den erweiterten Datenstrom auf den Wert null gesetzt. Verwenden Sie LOGMODE(0) nur in Ausnahmesituationen. Obwohl die logische Einheit (LU) mit der z/OS Communications Server-Definition gebunden wird, behält CICS die wichtigsten Sitzungsmerkmale aus der CICS-Definition bei. Wenn z. B. ein Drucker für z/OS Communications Server als LUTYPE1 definiert wird, für CICS jedoch als LUTYPE3 mit LOGMODE(0), dann akzeptiert CICS die Bindung, sendet jedoch LUTYPE3-Steuerzeichen an den Drucker, was zu fehlerhaften Ergebnissen führen kann. Für Pipeline-Terminals gilt diese Einschränkung nicht.

**Anmerkung:**

1. Diesen Wert müssen Sie für das Protokollmodusattribut nur in Ausnahmesituationen verwenden.
2. Geben Sie für eine logische Einheit in einer domänenübergreifenden Umgebung LOGMODE(0) an und stellen Sie die Informationen zum Modus der logischen Einheit in den Operanden DLOGMOD und MODETAB der LU-Anweisung von z/OS Communications Server bereit. In einer domänenübergreifenden Umgebung führt die Verwendung von LOGMODE mit einem Namen zu einem z/OS Communications Server-Fehler.

**LOGMODECOM**

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt.

**LOGONMSG({NO|YES})**

Gibt an, ob die Transaktion 'good morning', die im Systeminitialisierungsparameter **GMTRAN** angegeben ist, wie folgt verarbeitet wird:

- Die Transaktion wird automatisch initialisiert, wenn die logische Einheit zum ersten Mal über z/OS Communications Server bei CICS angemeldet wird.
- Die Transaktion wird initialisiert, nachdem die TIMEOUT-Zeitdauer des Terminalbenutzers unter bestimmten Bedingungen abgelaufen ist.

Wenn Sie ERRLASTLINE(YES) angegeben haben, dann überschreiben die von der Transaktion geschriebenen Nachrichten die Zeile für Fehlermeldungen nicht.

**NO** CICS führt die Transaktion 'good morning' nicht aus.

**Anmerkung:** Wenn Sie ein Nicht-SNA-Terminal wie beispielsweise ein Telnet 3270-Terminal verwenden, dann wird bei Angabe von LOGONMSG(NO) nicht automatisch die Tastaturverriegelung freigegeben. Sie müssen die Grundstellungstaste drücken, um die Tastaturverriegelung aufzuheben.

**YES** CICS führt die Transaktion 'good morning' aus, wenn der OPNDST-Exit erfolgreich ausgeführt und eine Sitzung eingerichtet wurde. Die Transaktion wird über ATI (Automatic Transaction Initiation; automatische Transaktionsinitialisierung) initialisiert und konkurriert mit anderen ATI-Transaktionen um die Nutzung des Terminals. Geben Sie für diese TYPETERM-Definition ATI(YES) an.

**Anmerkung:** Wenn Sie ein Nicht-SNA-Terminal wie beispielsweise ein Telnet 3270-Terminal verwenden, dann wird bei Angabe von LOGONMSG(YES) auch automatisch die Tastaturverriegelung freigegeben.

**MSRCONTROL({NO|YES})**

Gibt an, ob das Terminal (8775 oder 3643) über einen Magnetcodeleser verfügt. Diese Option ist für SCS-Drucker nicht gültig.

Sie können das strukturierte Feld QUERY verwenden, um festzustellen, ob die Einheit für die Verwendung eines Magnetcodelesers eingerichtet wurde.

**NEPCCLASS({0|*transaktionsklasse*})**

Gibt die NEP-Transaktionsklasse (NEP = Node Error Program; Knotenfehlerprogramm) an.

0 Die Angabe dieses Werts führt dazu, dass eine Verbindung zum Standardmodul des Knotenfehlerprogramms (NEP = Node Error Program) hergestellt wird.

*transaktionsklasse*

Die Transaktionsklasse für das (nicht standardmäßige) NEP-Modul. Der Wert für *transaktionsklasse* kann im Bereich zwischen 1 und 255 liegen. Informationen zur Programmierung für das Knotenfehlerprogramm (NEP) finden Sie in Writing a node error program.

**OBFORMAT({NO|YES})**

Gibt an, ob die externe Formatierung verwendet wird. Wenn die Einheiten, für die Sie diese TYPETERM-Definition definieren, die externe Formatierung von BMS verwenden, dann geben Sie OBFORMAT(YES) an. OBFORMAT(YES) darf nur für zwei Einheitentypen angegeben werden:

- 3650, SESSIONTYPE(3270)
- LUTYPE2, für ein 8100 Information System, das mit dem Betriebssystem DPPX mit DPPX/DPS Version 2 für Nachrichtenübergaberoutinen arbeitet

Verwenden Sie das strukturierte Feld QUERY, um zu ermitteln, ob die Einheit für die Verwendung der externen Formatierung eingerichtet wurde.

**OBOPERID({NO|YES})**

Gibt an, ob CICS die externen Bediener-IDs verwendet, um die BMS-Routing-Funktionen zu unterstützen, die für dieses Terminal benötigt werden. Diese Option gilt nur für logische 3790- und 3770-Stapeldatenaustauscheinheiten.

NO CICS verwendet die externen Bediener-IDs nicht.

**YES** CICS verwendet die externen Bediener-IDs.

**OUTLINE({NO|YES})**

Gibt an, ob die Einheit die Feldumrandungsfunktion unterstützt:

NO Die Einheit bietet keine Unterstützung für die Feldumrandungsfunktion.

**YES** Die Einheit bietet Unterstützung für die Feldumrandungsfunktion.

Verwenden Sie das strukturierte Feld QUERY, um zu ermitteln, ob die Einheit für die Verwendung der Feldumrandungsfunktion eingerichtet wurde.

**PAGESIZE(*zeilen* , *spalten*)**

Gibt die Standardseitengröße für diesen Drucker an. Die Werte für *zeilen* und *spalten* müssen im Bereich zwischen 0 und 255 liegen. Die Standardseitengröße wird von BMS verwendet, wenn im Attribut DEFSCREEN die Standardanzei-gegröße ausgewählt wurde.

*zeilen* Gibt die Anzahl der Zeilen auf der Seite an. Der PAGESIZE-Wert für *zeilen* kann kleiner sein als der DEFSCREEN-Wert für *zeilen*, um ggf. die unterste Zeile der Anzeige für Fehlermeldungen zu reservieren (siehe Attribut ERRLASTLINE), wenn die gleiche BMS-Maske sowohl für das Drucken als auch für die Anzeige benutzt wird.

*spalten* Gibt die Anzahl der Zeichen in jeder Zeile an. Wenn der in PAGESIZE angegebene Wert für *spalten* von dem Wert für *spalten* in DEFSCREEN abweicht, kann es zu unerwarteten Ergebnissen kommen.

Das Produkt von *zeilen* und *spalten* darf maximal 32767 betragen.

Der Standardwert ist von dem Wert abhängig, den Sie für das Attribut DEVICE angegeben haben. Detaillierte Informationen hierzu finden Sie in „Standardwerte für TYPETERM-Attribute“ auf Seite 267.

BMS verwendet die Seitengrößenwerte bei der Vorbereitung von Ausgabedatenströmen. Die angegebene Anzahl von Zeichen pro Zeile auf der Seite darf die physische Zeilenbreite des Terminals nicht überschreiten. Bei Druckern, die bei Erreichen des Zeilenendes automatisch die Zeilenvorschubfunktion ausführen (z. B. 3270-Drucker), muss die hier angegebene Zeilenbreite geringer sein als die physische Zeilenbreite.

Dadurch wird sichergestellt, dass die Formatierung der Ausgabedaten vollständig durch die Zeilenvorschubzeichen (NL = New Line) reguliert wird, die von BMS oder von Ihnen angegeben werden, und nicht durch die Zeilenvorschubfunktionen, die von der Einheit selbst ausgeführt werden. Diese würden zu zusätzlichen Ausgabezeilen führen, was wiederum eine physische Seitentiefe zur Folge hätte, die den hier angegebenen Wert übersteigt.

Für 3270-Drucker beschränkt die Hardware die Datenmenge, die von BMS übertragen werden kann. Wenn in der Masken- oder Anwendungsprogrammanforderung L40, L64 oder L80 angegeben wird, oder wenn im Befehl SEND MAP die Option NLEOM nicht angegeben wird, dann darf das Produkt aus *zeilen* und *spalten*, das in PAGESIZE angegeben ist, die Puffergröße nicht übersteigen.

Wenn in der BMS-Anforderung eine Angabe für NLEOM definiert ist, dann kann die Seitenlänge eine beliebige Anzahl von Zeilen umfassen, das Produkt aus *zeilen* und *spalten*, das im Attribut DEFSCREEN oder ALTSCREEN angegeben ist, darf jedoch die Puffergröße der Einheit nicht überschreiten. Dies bedeutet, dass die Anzahl der Zeichen, die von BMS übertragen werden, die physische Puffergröße des Druckers nicht übersteigen darf.

**Anmerkung:** BMS teilt eine große Seite für die Übertragung in kleinere Segmente auf. Der Wert für PAGESIZE muss deshalb der erforderlichen **logischen** Seitengröße (Produkt aus *zeilen* und *spalten*) entsprechen und der Wert für DEFSCREEN der tatsächlichen Puffergröße.

Für eine 3600-Einheit von z/OS Communications Server wird der angegebene Wert für PAGESIZE verwendet, wenn eine BMS-Seitenerstellungsoperation versucht wird, ohne dass ein Code für logische Einheiten (LDC = Logical Device Code) angegeben wird. Das System nimmt den Standardeinheitentyp 3604 an.

Für logische 3770-, LUTYPE4- oder 3790-Stapeldatenaustauscheinheiten wird der angegebene Wert für PAGESIZE verwendet, wenn eine BMS-Seitenerstellungsoperation angefordert wird, ohne dass ein Code für logische Einheiten (LDC = Logical Device Code) angegeben wird. Der Standardeinheitentyp ist der Konsolldrucker.

Bei der Weiterleitung einer Nachricht an eine Liste von Terminals ist Vorsicht geboten. Wenn der Wert für PAGESIZE, den Sie definiert (oder als Standardwert übernommen) haben, zu niedrig für die Nachricht ist, wird die Transaktion abgebrochen.

Für die kumulative Textverarbeitung beträgt die maximal zulässige Puffergröße 32767. Wird dieser Wert überschritten, dann erzwingt BMS intern eine reduzierte Seitenlänge, um sicherzustellen, dass die Seitengröße (PAGESIZE) innerhalb des Grenzwerts bleibt.

#### **PARTITIONS({NO|YES})**

Gibt an, ob eine Einheit Partitionen verwenden soll. Diese Option ist für SCS-Drucker nicht gültig.

**NO** Die Einheit verwendet keine Partitionen.

**YES** Die Einheit verwendet Partitionen.

Sie können das strukturierte Feld QUERY verwenden, um festzustellen, ob die Einheit für die Verwendung von Partitionen eingerichtet wurde.

#### **PRINTADAPTER({NO|YES})**

**Für 3275-Einheiten:** Gibt an, ob die Druckeradapterfunktion und das entsprechende Modell 3 des 3284-Druckers auf dem 3275-Datensichtgerät vorhanden sind. Diese Funktion macht die 3284-Einheit über eine Programmabruf Taste des 3275-Hostsystems für Druckanforderungen auswählbar.

**NO** Die Druckeradapterfunktion und das entsprechende Modell 3 des 3284-Druckers sind nicht verfügbar.

**YES** Die Druckeradapterfunktion und das entsprechende Modell 3 des 3284-Druckers sind verfügbar.

**Für logische LUTYPE2-Einheiten:** Gibt an, ob für Druckanforderungen, die mit der Drucktaste (PRINT) oder mit dem Befehl ISSUE PRINT eingeleitet wurden, die Druckerzuordnung von der 3790-Einheit oder von der 3274- bzw. 3276-Einheit gemäß der Druckerberechtigungs matrix für z/OS Communications Server-Anschlüsse und Nicht-z/OS Communications Server-Anschlüsse gesteuert wird.

**NO** Druckanforderungen werden nicht auf Basis der Druckerberechtigungs matrix für z/OS Communications Server- und Nicht-z/OS Communications Server-Anschlüsse gesteuert.

**YES** Druckanforderungen werden auf Basis der Druckerberechtigungs matrix für z/OS Communications Server- und Nicht-z/OS Communications Server-Anschlüsse gesteuert.

Des Weiteren stehen 3270-Drucker, die mit derselben 3790-Einheit verbunden sind, für Druckanforderungen zur Verfügung, die mit einer Terminalsteuerungsdruckanforderungen oder nach Einleitung durch den Bediener an die logische 3270-Bildschirmereinheit gesendet wurden. Wenn für PRINTADAPTER die Einstellung NO angegeben wird, dann wird die Druckerzuordnung anhand der Attribute PRINTER und ALTPRINTER der TERMINAL-Definition festgelegt.

Wenn die Ausgabe in der Anzeige über BMS-Anforderungen mit der Option PRINT, über BMS-Anforderungen mit der Option NLEOM oder mit dem Befehl CMSG erstellt wird, dann werden die Inhalte der Anzeige automatisch auf einen 3270-Drucker kopiert, und zwar unabhängig davon, ob die in CICS definierte Drucktaste (PRINT), bei der es sich normalerweise um eine Programmabruf Taste handelt, gedrückt wurde.

**PROGSYMBOLS({NO|YES})**

Gibt an, ob die PS-Funktion (PS = Programmed Symbol; programmiertes Symbol) auf dieser 3270-Einheit bzw. auf diesem SCS-Drucker verwendet werden kann. Die Funktion ermöglicht die Speicherung von und den Zugriff auf bis zu sechs aus jeweils 191 Zeichen bestehenden Zeichensätzen mit kundendefinierten und vom Programm geladenen Schriftarten und Codes.

**NO** Die PS-Funktion kann nicht verwendet werden.

**YES** Die PS-Funktion kann verwendet werden.

Sie können das strukturierte Feld QUERY verwenden, um festzustellen, ob die Einheit für die Verwendung programmierter Symbole eingerichtet wurde.

**QUERY({NO|COLD|ALL})**

Gibt an, ob CICS das strukturierte Feld QUERY zur Ermittlung der Merkmale der Einheit verwenden soll.

**NO** CICS verwendet die Funktion QUERY nicht.

**COLD** CICS verwendet die Funktion QUERY, um die Merkmale der Einheit zu ermitteln, nur dann, wenn die Einheit zum ersten Mal nach einem Erst- oder Kaltstart von CICS verbunden wird. Die Einheitenmerkmale werden im globalen Katalog von CICS zur Verwendung bei nachfolgenden Warm- und Notfallstarts gespeichert.

**ALL** CICS verwendet die Funktion QUERY, um die Merkmale der Einheit zu ermitteln, jedes Mal, wenn die Einheit verbunden wird.

**RECEIVESIZE(*zahl*)**

Geben Sie für ein **definiertes, nicht automatisch installiertes Terminal** die maximale Größe einer Anforderungseinheit an, die zur Verarbeitung einer z/OS Communications Server-RECEIVE-Anforderung ausreicht. Der Wert für RECEIVESIZE wird an die verbundene logische Einheit übertragen und muss im Bereich zwischen 0 und 30720 liegen. Er wird von CICS möglicherweise abgerundet, weil er in einer architekturbasierten Form übertragen werden muss.

Die Auswirkung von RECEIVESIZE hängt davon ab, ob RECEIVE RUSIZE in der Protokollmodustabelle (LOGMODE) von z/OS Communications Server angegeben wurde. In Tabelle 7 wird dargestellt, wie RECEIVE RUSIZE zum Binden einer Sitzung für jede mögliche Kombination von TYPETERM- und LOGMODE-Werten verwendet werden kann.

*Tabelle 7. RECEIVE RUSIZE für definierte Terminals (ohne automatische Installation)*

RECEIVE RUSIZE (VTAM)	TYPETERM RECEIVESIZE	RUSIZE - Verwendung zur Bindung
0	0	0
0	Angegeben	TYPETERM RECEIVESIZE-Größe
Angegeben	0	Diese Kombination ist ungültig und führt zu einem Bindefehler mit der Nachricht DFHZC2403
Angegeben	Angegeben	TYPETERM RECEIVESIZE-Größe

**Anmerkung:** Die Ausnahme zu dieser Tabelle stellt die Angabe von LOGMODE(0) dar. Wenn Sie diese Einstellung in Ihrer TYPETERM-Definition angeben, dann werden VTAM-Werte verwendet, und zwar unabhängig davon, welche anderen Optionen angegeben wurden.

**APPC-Terminal:** Für ein APPC-Einzelsitzungsterminal (LUTYPE6.2) stellt 256 einen passenden Wert dar.

**Automatisch installiertes Terminal:** Für ein automatisch installiertes Terminal gibt ein Wert ungleich null für RECEIVESIZE entweder den maximal zulässigen oder den tatsächlichen Wert für RECEIVE RUSIZE an, der zum Binden einer Sitzung für eine logische Einheit verwendet wird, die mit dieser TYPE-TERM-Angabe definiert wird.

Die Auswirkung von RECEIVESIZE hängt davon ab, ob RECEIVE RUSIZE in der Protokollmodustabelle (LOGMODE) von z/OS Communications Server angegeben wurde. In Tabelle 8 wird dargestellt, wie RECEIVE RUSIZE zum Binden einer Sitzung für jede mögliche Kombination von TYPETERM- und LOGMODE-Werten verwendet werden kann.

*Tabelle 8. RECEIVE RUSIZE für automatisch installierte Terminals*

RECEIVE RUSIZE (VTAM)	TYPETERM RECEIVESIZE	RUSIZE - Verwendung zur Bindung
0	0 und BUILDCHAIN(YES)	256
0	0 und BUILDCHAIN(NO)	0
0	Angegeben	TYPETERM RECEIVESIZE-Größe
Angegeben	0	VTAM RECEIVE RUSIZE-Größe
Angabe kleiner-gleich TYPETERM RECEIVESIZE	Angegeben	VTAM RECEIVE RUSIZE-Größe
Angabe größer als TYPETERM RECEIVESIZE	Angegeben	Diese Kombination ist ungültig und führt zur Ausgabe der Nachricht DFHZC5963

#### **RECOVNOTIFY({NONE|MESSAGE|TRANSACTION})**

In einer CICS-Region, die mit der Unterstützung für persistente Sitzungen ausgeführt wird, gibt diese Option an, wie ein Terminalendbenutzer darüber benachrichtigt wird, dass die Terminalsitzung wiederhergestellt wurde. In einer CICS-Region, die mit XRF-Unterstützung ausgeführt wird, wird auf diese Weise angegeben, wie der Terminalbenutzer darüber informiert wird, dass eine XRF-Übernahme stattgefunden hat.

Diese Option ist bei APPC-Sitzungen nicht anwendbar.

##### **NONE**

Das System gibt keine Benachrichtigung aus, in der Sie darüber informiert werden, dass eine Terminalsitzung wiederhergestellt wurde oder dass eine XRF-Übernahme stattgefunden hat.

##### **MESSAGE**

Am Bildschirm wird eine Nachricht angezeigt, in der Sie darüber informiert werden, dass das System wiederhergestellt wurde. Die Nachricht wird in zwei BMS-Masken (DFHXRC1 und DFHXRC2) für XRF und in einer BMS-Maske (DFHXRC3) für persistente Sitzungen von z/OS Communications Server angegeben. Diese Masken befinden sich in der Maskengruppe DFHXMSG. Wenn eine reduzierte Übernahmezeitdauer wichtig ist, dann verwenden Sie anstelle von TRANSACTION besser MESSAGE.

Das Terminal muss mit der Option ATI(YES) definiert werden und es muss in der Lage sein, eine BMS-Maske anzuzeigen.

## TRANSACTION

Auf dem Terminal wird eine Transaktion initialisiert. Der Name der Transaktion wird im Systeminitialisierungsparameter **RMTRAN** angegeben. Für persistente Sitzungen von z/OS Communications Server wird nur die erste Transaktion verwendet, die im Systeminitialisierungsparameter **RMTRAN** aufgeführt ist.

**Tip:** Die Standardtransaktion für RMTRAN ist die im Systeminitialisierungsparameter **GMTRAN** angegebene Transaktion: Transaktion 'good-morning'.

Für die Option TRANSACTION muss das Terminal mit der Option ATI(YES) definiert werden. Wenn eine reduzierte Übernahmezeitdauer wichtig ist, dann verwenden Sie anstelle von TRANSACTION besser MESSAGE.

## RECOVOPTION({SYSDEFAULT|CLEARCONV|RELEASESESS| UNCONDREL|NONE})

Diese Option gilt für die Wiederherstellung von Sitzungen in einer CICS-Region, die mit persistenten z/OS Communications Server-Sitzungen oder mit XRF ausgeführt wird.

In einer CICS-Region, die mit Unterstützung für persistente Sitzungen ausgeführt wird, gibt diese Option an, wie CICS bei der Wiederherstellung der Sitzung vorgehen und das Terminal beim Neustart des Systems innerhalb des Verzögerungsintervalls für persistente Sitzungen wieder aktivieren soll.

Für alle Wiederherstellungsoptionen außer NONE: Wenn die ausgeführte Aktion eine z/OS Communications Server-Aktion zum Auflösen der Bindung (UNBIND) ist, dann folgt auf UNBIND eine z/OS Communications Server-Aktion SIMLOGON.

SYSDEFAULT ist der Standardwert, in bestimmten Situationen, in denen Ihre Ressourcendefinition dieses Attribut nicht enthält, erzwingt CICS jedoch die Verwendung des Werts NONE für das Attribut, wenn dieser Wert zur Wahrung der Kompatibilität mit anderen Optionen in Ihrer Ressourcendefinition erforderlich ist.

## SYSDEFAULT

In einer CICS-Region, die mit der Unterstützung für persistente Sitzungen ausgeführt wird, gibt diese Option an, dass CICS die optimale Prozedur zur Wiederherstellung einer Sitzung beim Neustart des Systems innerhalb des Verzögerungsintervalls für persistente Sitzungen abhängig von der Sitzungsaktivität und den Merkmalen des Terminals auswählen soll.

Obwohl die Sitzungen wiederhergestellt werden, werden zum Fehlerzeitpunkt ausgeführte Operationen abgebrochen und nicht wiederhergestellt. Transaktionen werden ebenfalls abgebrochen, wenn die wiederhergestellte Sitzung momentan von einer anderen CICS-Region über eine APPC-Verbindung benutzt wird.

CICS stellt die Sitzung mit den geringstmöglichen Auswirkungen wie folgt wieder her:

- Wenn das Terminal bei Auftreten des CICS-Fehlers keine Transaktion ausführt, dann ist keine Wiederherstellungsaktion erforderlich und CICS führt die geeignete Aktion zur Wiederherstellungsbenachrichtigung aus, die im Attribut RECOVNOTIFY definiert ist.
- Wenn das Terminal bei Auftreten des CICS-Fehlers aktiv war (d. h. eine Transaktion ausgeführt hat), dann versucht CICS zuerst, die Sit-



zung wiederherzustellen, indem ein z/OS Communications Server-Endklammerindikator gesendet wird. Wenn die Sitzungswiederherstellung mit der Endklammer nicht möglich ist (wenn CICS z. B. im RECEIVE-Modus arbeitet), dann wird von CICS der Befehl CLEAR ausgegeben. Wenn das Terminal die Ausführung des Befehls CLEAR nicht unterstützt, dann wird als Wiederherstellungsaktion eine z/OS Communications Server-UNBIND-Aktion gefolgt von einer SIMLOGON-Aktion ausgeführt.

Weitere Informationen zu persistenten Sitzungen finden Sie in Application design considerations .

#### **CLEARCONV**

Verhindert, dass CICS einen Endklammerindikator sendet, um eine In-Bracket-Sitzung zu schließen. Stattdessen sendet CICS eine CLEAR-Anforderung, um die Dialogstatus zurückzusetzen. Wenn die Sitzung die CLEAR-Anforderung nicht unterstützt, sendet CICS eine UNBIND-Anforderung. Die CLEAR- oder UNBIND-Anforderung wird nur gesendet, wenn die Sitzung während des Neustarts des Systems belegt war (bei persistenten Sitzungen) oder wenn eine Übernahme stattgefunden hat (bei XRF).

#### **RELEASESESS**

Fordert CICS zum Senden einer UNBIND-Anforderung für die Freigabe der aktiven Sitzung auf. Die UNBIND-Anforderung wird nur gesendet, wenn die Sitzung während des Neustarts des Systems belegt war (bei persistenten Sitzungen) oder wenn eine Übernahme stattgefunden hat (bei XRF). Nach der UNBIND-Anforderung wird die Sitzung in die Warteschlange für SIMLOGON eingestellt. Wenn die Sitzung nicht belegt ist, wird die angeforderte Wiederherstellungsbenachrichtigung durchgeführt.

#### **UNCONDREL**

Fordert CICS zum Senden einer UNBIND-Anforderung für die Freigabe der aktiven Sitzung auf. Die UNBIND-Anforderung wird unabhängig davon gesendet, ob die Sitzung während des Neustarts des Systems belegt war (bei Unterstützung für persistente Sitzungen) oder ob eine Übernahme stattgefunden hat (bei XRF). Nach der UNBIND-Anforderung wird die Sitzung in die Warteschlange für SIMLOGON eingestellt.

#### **NONE**

In einer CICS-Region, die mit der Unterstützung für persistente Sitzungen ausgeführt wird, gibt diese Option an, dass die Terminalsitzung beim Neustart des Systems nicht innerhalb des Verzögerungsintervalls für persistente Sitzungen wiederhergestellt werden soll. Das Terminal verfügt also praktisch nicht über Unterstützung für persistente Sitzungen. Für LU6.2-Sitzungen wird die Bindung aufgehoben, der zuletzt vereinbarte CNOS-Wert wird jedoch nach dem Neustart an das CICS-System zurückgegeben. Wenn Sie AUTOCONNECT(YES) angeben, wird das Terminal nach dem Neustart des Systems automatisch, jedoch abhängig von der Angabe für den Systeminitialisierungsparameter **AIRDELAY** wieder verbunden. (Bei Angabe von AIRDELAY=0 wird AUTOCONNECT(YES) allerdings überschrieben, sodass das Terminal nicht wieder verbunden wird.)

Verwenden Sie RECOVOPTION(NONE), wenn dieses Terminal oder Modell für die automatische Installation mit persistenten Sitzungen verwendet

werden soll (**PSDINT** = nnn in der Systeminitialisierungstabelle (SIT)), das Terminal jedoch in einem Befehl **EXEC CICS ISSUE PASS LUNAME() LOGONLOGMODE** angegeben wird.

**RELREQ=({NO|YES})**

Gibt an, ob CICS die logische Einheit bei einer entsprechenden Anforderung eines anderen z/OS Communications Server-Anwendungsprogramms freigeben soll.

**NO** CICS gibt die logische Einheit nicht frei.

**YES** CICS gibt die logische Einheit frei, wenn sie momentan nicht Teil einer Transaktion ist.

**ROUTEDMSG({ALL|NONE|SPECIFIC})**

Gibt an, welche Nachrichten mit dem Befehl **EXEC CICS ROUTE** an dieses Terminal weitergeleitet werden sollen. Der Standardwert ist von dem Wert abhängig, den Sie für das Attribut **DEVICE** angeben. Detaillierte Informationen hierzu finden Sie in „Standardwerte für TYPETERM-Attribute“ auf Seite 267.

**ALL** BMS leitet an dieses Terminal Nachrichten weiter, die an **alle** Terminals gerichtet sind, und außerdem Nachrichten, die speziell an **dieses** Terminal gerichtet sind.

**NONE**

BMS leitet keine Nachrichten an dieses Terminal weiter, und zwar unabhängig davon, ob diese Nachrichten an alle Terminals oder nur an dieses Terminal gerichtet sind.

**SPECIFIC**

BMS leitet Nachrichten an dieses Terminal weiter, wenn sie speziell an dieses Terminal gerichtet sind, nicht jedoch, wenn die Nachrichten an **alle** Terminals gerichtet sind.

**RSTSIGNOFF({NOFORCE|FORCE})**

Gibt an, ob der Terminalbenutzer abgemeldet werden soll, wenn für eine persistente Sitzung ein Neustart oder eine XRF-Übernahme ausgeführt wird.

**FORCE**

Das Terminal wird nach einem Neustart einer persistenten Sitzung oder einer XRF-Übernahme abgemeldet.

**NOFORCE**

Das Terminal bleibt nach dem Neustart einer persistenten Sitzung oder einer XRF-Übernahme angemeldet, sofern der Systeminitialisierungsparameter **RSTSIGNOFF** und der XRFSOFF-Eintrag im CICS-Segment des RACF-Benutzerprofils beide auf **NOFORCE** gesetzt sind.

**SENDSIZE(zahl)**

**Definiertes Terminal (nicht automatisch installiert:** Für ein nicht automatisch installiertes Terminal ist dies die maximale Größe (in Byte) einer Anforderungseinheit, die zur Verarbeitung einer z/OS Communications Server-Anforderung **VTAM SEND** ausreicht. Der Wert für **SENDSIZE** wird an die verbundene logische Einheit übertragen und muss im Bereich zwischen 0 und 30720 liegen. Er wird von CICS möglicherweise abgerundet, weil er in einer architekturbasierten Form übertragen werden muss.

Die Auswirkung von **SENDSIZE** hängt davon ab, ob **SEND RUSIZE** in der Protokollmodustabelle (**LOGMODE**) von z/OS Communications Server angegeben wurde. In Tabelle 9 auf Seite 305 wird dargestellt, wie **SEND RUSIZE** zum Binden einer Sitzung für jede mögliche Kombination von TYPETERM- und LOGMODE-Werten verwendet werden kann.

Tabelle 9. SEND RUSIZE für definierte Terminals (ohne automatische Installation)

SEND RUSIZE (VTAM)	TYPETERM SENDSIZE	RUSIZE - Verwendung zur Bindung
0	0	0
0	Angegeben	TYPETERM SENDSIZE-Größe
Angegeben	0	0
Angegeben	Angegeben	TYPETERM SENDSIZE-Größe
<b>Anmerkung:</b> Die Ausnahme zu dieser Tabelle stellt die Angabe von LOGMODE(0) dar. Wenn Sie diese Einstellung in Ihrer TYPETERM-Definition angeben, dann werden VTAM-Werte verwendet, und zwar unabhängig davon, welche anderen Optionen angegeben wurden. <b>Anmerkung:</b> VTAM wird jetzt als z/OS Communications Server bezeichnet.		

**APPC-Terminal:** Für ein APPC-Einzelsitzungsterminal (LUTYPE6.2) stellt 256 einen passenden Wert dar.

**Automatisch installiertes Terminal:** Für ein automatisch installiertes Terminal gibt ein Wert ungleich null für SENDSIZE entweder den maximal zulässigen oder den tatsächlichen Wert für SEND RUSIZE an, der zum Binden einer Sitzung für eine logische Einheit verwendet wird, die mit dieser TYPETERM-Angabe definiert wird.

Die Auswirkung von SENDSIZE hängt davon ab, ob SEND RUSIZE in der Protokollmodustabelle (LOGMODE) von z/OS Communications Server angegeben wurde. In Tabelle 10 wird dargestellt, wie SEND RUSIZE zum Binden einer Sitzung für jede mögliche Kombination von TYPETERM- und LOGMODE-Werten verwendet werden kann.

Tabelle 10. SEND RUSIZE für automatisch installierte Terminals

SEND RUSIZE (VTAM)	TYPETERM SENDSIZE	RUSIZE - Verwendung zur Bindung
0	0	0
0	Angegeben	TYPETERM SENDSIZE-Größe
Angegeben	0	VTAM SEND RUSIZE-Größe
Angabe kleiner-gleich TYPETERM SENDSIZE	Angegeben	VTAM SEND RUSIZE-Größe
Angabe größer als TYPETERM SENDSIZE	Angegeben	Diese Kombination ist ungültig und führt zur Ausgabe der Nachricht DFHXC5963

### SESSIONTYPE(*typ*)

Gibt den Typ der Sitzung an, die für eine logische z/OS Communications Server-SNA-Einheit benutzt werden kann. Detaillierte Informationen hierzu finden Sie in „Standardwerte für TYPETERM-Attribute“ auf Seite 267.

### SHIPPABLE({NO|YES})

Gibt an, ob die Definition an ein fernes System gesendet werden darf, wenn diese Einheit versucht, eine ferne Transaktion einzuleiten.

**NO** Diese Definition kann nicht an ein fernes System gesendet werden.

**YES** Diese Definition kann an ein fernes System gesendet werden.

Diese Funktion kann für jedes Terminal verwendet werden, und zwar unabhängig davon, ob eine automatische Installation durchgeführt oder die Installation mit einer eigenen TERMINAL-Definition durchgeführt wurde. Der Versand funktioniert nur dann, wenn das Terminal über eine Definition verfügt, die mit einer der folgenden Methoden auf dem lokalen System installiert wurde.

Bei Verwendung von SHIPPABLE(YES) müssen Sie nicht sicherstellen, dass eine Definition des Terminals auf dem fernen System für ein lokal definiertes Terminal vorhanden ist, um eine Transaktion auf diesem System einzuleiten. Diese Methode ist nützlich, wenn das ferne System die CSD-Datei nicht gemeinsam mit dem lokalen System verwenden kann.

Eine Definition für das Terminal muss bereits auf dem fernen System installiert (oder an dieses System gesendet) worden sein.

Anweisungen zur Entscheidung über die Verwendung von SHIPPABLE(YES) finden Sie in „Terminals für Transaktionsrouting“ auf Seite 216.

#### **SIGNOFF({YES|NO|LOGOFF})**

Gibt die Aktionen an, die ausgeführt werden, wenn GNTRAN (CESF oder benutzerdefinierte Transaktion) zugeordnet wird und eine Abmeldung des Terminals versucht wird. Wenn Sie mit RACF arbeiten, dann müssen Sie den TIME-OUT-Grenzwert im RACF-Segment angeben.

**YES** Wenn seit der letzten Eingabe des Bedieners der angegebene Zeitraum verstrichen ist, wird das Terminal automatisch bei CICS abgemeldet.

**NO** Für das Terminal gilt kein Zeitlimit.

#### **LOGOFF**

Wenn seit der letzten Eingabe des Bedieners der angegebene Zeitraum verstrichen ist, wird das Terminal automatisch bei CICS und dann bei z/OS Communications Server abgemeldet. Die Verwendung von LOGOFF ist nützlich bei einem Modell für die automatische Installation, weil so nicht unnötig virtueller Speicher für Einträge für Terminals verbraucht wird, bei denen eine Zeitlimitüberschreitung aufgetreten ist.

Wenn die Zuordnung mit GNTRAN fehlschlägt, weil nicht verarbeitete Daten im Terminalpuffer vorhanden sind (und zu einem BID-Fehler führen), dann wird das Terminal freigegeben und abgemeldet. GNTRAN wird nicht ausgeführt und hat keine Auswirkungen.

**Anmerkung:** Der Wert dieses Attributs kann nicht geändert werden, wenn DEVICE(APPC) angegeben wurde. Der Standardwert lautet in diesem Fall SIGNOFF(NO).

#### **SOSI({NO|YES})**

Gibt an, ob die Einheit Felder mit gemischtem EBCDIC-Zeichensatz und Doppelbytezeichensatz (DBCS) unterstützt.

**NO** Die Einheit unterstützt keine Felder mit gemischtem EBCDIC-Zeichensatz und Doppelbytezeichensatz (DBCS).

**YES** Die Einheit unterstützt Felder mit gemischtem EBCDIC-Zeichensatz und Doppelbytezeichensatz (DBCS).

Sie können das strukturierte Feld QUERY verwenden, um festzustellen, ob die Einheit für die Verwendung von Feldern mit gemischtem EBCDIC-Zeichensatz und Doppelbytezeichensatz (DBCS) eingerichtet wurde.

### **TERMMODEL({1|2})**

Gibt die Modellnummer des Terminals an. Wenn die Einheit eine Komponente des 3270 Information Display System ist, dann geben Sie die Modellnummer des Terminals an:

- 1** Geben Sie 1 für 3270 Modell 1-Bildschirmeinheiten und -Drucker (z. B. 3277 Modell 1) mit einer standardmäßigen Anzeige- oder Puffergröße von 12x40 (480 Byte/Zeichen) an. TERMMODEL(1) ist der Standardwert für 3270 Modell 1-Drucker und -Bildschirmeinheiten.

Geben Sie 1 für 3275 Display Station Modell 11 an. Die angeforderte CICS-Unterstützung ist identisch mit der Unterstützung, die durch Angabe von TERMMODEL(1) für 3275 Display Station Modell 1 angefordert wird.

- 2** Geben Sie 2 für 3270-Bildschirmeinheiten und -Drucker (z. B. 3278 Modell 4) mit einer standardmäßigen Anzeige- oder Puffergröße von 24x80 (1920 Byte/Zeichen) an. TERMMODEL(2) ist der Standardwert für 3286-Drucker im 3270-Kompatibilitätsmodus.

Geben Sie 2 für 3275 Display Station Modell 12 an. Die angeforderte CICS-Unterstützung ist identisch mit der Unterstützung, die durch Angabe von TERMMODEL(2) für 3275 Display Station Modell 2 angefordert wird.

### **TEXTKYBD({NO|YES})**

Gibt an, ob die 3270-Einheit über die Text-Tastatur-Funktion verfügt.

**NO** Die 3270-Einheit verfügt nicht über die Text-Tastatur-Funktion.

**YES** Die 3270-Einheit verfügt über die Text-Tastatur-Funktion.

### **TEXTPRINT({NO|YES})**

Gibt an, ob der 3288-Drucker über die Text-Druck-Funktion verfügt.

**NO** Der 3288-Drucker verfügt nicht über die Text-Druck-Funktion.

**YES** Der 3288-Drucker verfügt über die Text-Druck-Funktion.

### **TTI({YES|NO})**

Gibt an, ob Transaktionen vom Benutzer am Terminal eingeleitet werden können.

**YES** Transaktionen können vom Benutzer am Terminal eingeleitet werden. Wenn Sie außerdem ATI(YES) angeben, dann können Transaktionen auch automatisch eingeleitet werden. In diesem Fall setzt die automatische Transaktionsinitialisierung anhand der Steuerung transienter Daten oder der Intervallsteuerung eine Bedingung in einem entsprechenden Terminaleintrag einer Terminalsteuertabelle. Wenn sowohl für ATI als auch für TTI die Einstellung YES angegeben wurde und wenn keine Transaktion in dem Terminal stattfindet, dann leitet die Terminalsteuerung die benutzerdefinierte Task ein. Diese Task sendet erwartungsgemäß Nachrichten an das Terminal.

Geben Sie für ein Terminal, das zur Verarbeitung von Transaktionen (z. B. Abfragen oder Auftragserfassungen) eingesetzt wird, TTI(YES) und ATI(NO) an. Dies gilt auch für Datensichtgeräte oder Hardcopy-Terminals, an die ohne eine Terminalanforderung keine Nachrichten gesendet werden und über die Transaktionen eingegeben werden. Beachten Sie, dass es sich hierbei um die einzige Spezifikation handelt, die für logische 3790-Abfrageeinheiten zulässig ist.

**NO** Transaktionen können vom Benutzer am Terminal nicht eingeleitet wer-

den. Wenn Sie für dieses Attribut die Einstellung NO angeben, dann müssen Sie ATI(YES) festlegen, damit Transaktionen automatisch eingeleitet werden können. Ein Beispiel für diesen Terminaltyp stellt ein Terminal dar, das sich an einem fernen Standort befindet (z. B. in einem Warehouse) und das unbeaufsichtigt arbeitet, aber Nachrichten empfangen kann.

#### **TYPETERM**(*name*)

Gibt den Namen dieser Erweiterung für eine TERMINAL-Definition an. Der Name kann bis zu acht Zeichen lang sein.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Auf diesen Namen wird in allen TERMINAL-Definitionen verwiesen, in denen diese Angabe für TYPETERM verwendet wird. Beachten Sie hierbei, dass diese TYPETERM-Definition vor der Installation der TERMINAL-Definitionen oder zeitgleich mit den TERMINAL-Definitionen installiert werden muss, die auf sie verweisen.

#### **UCTRAN**(**{NO|YES|TRANID}**)

Gibt an, ob der Eingabedatenstrom von einem Terminal in Großbuchstaben umgesetzt werden soll. Der Eingabedatenstrom kann eine Transaktions-ID sowie Programmdateien enthalten. CICS unterstützt die Definition von Transaktions-IDs in gemischter Groß-/Kleinschreibung. Das Attribut UCTRAN kann verwendet werden, um sicherzustellen, dass die korrekte Transaktion lokalisiert wird. Die Umsetzung in Großbuchstaben wird sowohl für 3270-Datenströme als auch für Datenströme eines anderen Typs durchgeführt.

**NO** Es wird keine Umsetzung in Großbuchstaben durchgeführt.

**YES** Die gesamte Dateneingabe vom Terminal, also sowohl die Transaktions-ID (falls vorhanden) als auch die Programmdateien, werden in Großbuchstaben umgesetzt, bevor die Verarbeitung ausgeführt wird.

#### **TRANID**

Wenn der Eingabedatenstrom eine Transaktions-ID umfasst, dann setzt CICS diese ID in Großbuchstaben um, bevor versucht wird, die zugehörige Definition zu lokalisieren. Allerdings werden die gesamten Eingabedaten (sowohl die Transaktions-ID als auch die Programmdateien) ohne Umsetzung an das Programm übergeben.

Aus diesem Grund ist es sowohl bei Auswahl von YES als auch der TRANID-Optionen zulässig, Transaktions-IDs in Großbuchstaben zu definieren und über das Terminal sowohl in Groß- als auch in Kleinbuchstaben einzugeben. Bei Angabe der Option TRANID werden Transaktions-IDs sowie Programmdateien allerdings ohne Umsetzung an das Programm übergeben.

Für PROFILE-Ressourcen können Sie außerdem eine Umsetzung in Großbuchstaben auf Transaktionsebene anfordern (siehe hierzu „PROFILE-Attribute“ auf Seite 137), sollten dabei allerdings berücksichtigen, dass eine Definition mit TYPETERM UCTRAN(YES) eine Definition mit PROFILE UCTRAN(NO) überschreibt. Wenn Sie also TYPETERM UCTRAN(YES) angeben, dann bleibt die Angabe von PROFILE UCTRAN(NO) ohne Wirkung. Die Umsetzung kann

vom Anwendungsprogramm für alle RECEIVE-Anforderungen außer der ersten mithilfe der Option ASIS überschrieben werden.

In Tabelle 11 wird gezeigt, welcher Teil der Terminaleingabe (Transaktions-ID und/oder Daten) entsprechend der Einstellung von UCTRAN in den PROFILE- und TYPETERM-Ressourcendefinitionen umgesetzt wird.

*Tabelle 11. Auswirkung von UCTRAN-Attributen auf Umsetzung von Transaktions-IDs (TRANID) und Daten*

UCTRAN in PROFILE	UCTRAN in TYPETERM	TRANID umgesetzt?	Daten umgesetzt?
YES	YES	Ja	Ja
YES	NO	Nein	Ja
YES	TRANID	Ja	Ja
NO	YES	Ja	Ja
NO	NO	Nein	Nein
NO	TRANID	Ja	Nein

Bestimmte nationale Sonderzeichen werden nicht automatisch umgesetzt, wenn UCTRAN(YES) oder UCTRAN(TRANID) angegeben wird. In diesem Fall können Sie eine der in Uppercase translation beschriebenen Methoden verwenden.

#### **USERAREALEN({0|zahl})**

Gibt die Länge in Byte (0 bis 255) des Benutzerbereichs für dieses Terminal an. Der Wert sollte möglichst niedrig sein. Der Terminalbenutzerbereich wird bei der Systeminitialisierung mit Nullen initialisiert.

Der Terminalbenutzerbereich kann sich im 31-Bit-Speicher oder im 24-Bit-Speicher befinden. Die Speicherposition ist von dem Wert abhängig, der im Systeminitialisierungsparameter **TCTUALOC** angegeben ist. Der Standardwert legt fest, dass sich der Terminalbenutzerbereich sowohl im 31-Bit-Speicher als auch im 24-Bit-Speicher befinden kann, CICS verwendet jedoch nach Möglichkeit immer den 31-Bit-Speicher. Wenn der Terminalbenutzerbereich unbedingt im 24-Bit-Speicher angelegt werden muss, weil Sie mit Anwendungsprogrammen arbeiten, die die 31-Bit-Adressierung nicht unterstützen, dann müssen Sie für die CICS-Region den Systeminitialisierungsparameter **TCTUALOC=BELOW** angeben.

#### **VALIDATION({NO|YES})**

**Für 8775-Einheiten** gibt dieses Attribut an, ob die 8775-Einheit über die Funktion für die erweiterte Validierung verfügt, die das Definieren der Angabe TRIGGER, MANDATORY FILL oder MANDATORY ENTER für Felder erlaubt.

**Für 3290-Einheiten** gibt dieses Attribut an, ob die 3290-Einheit über die Validierungsfunktion verfügt, die das Definieren der Angabe MANDATORY FILL oder MANDATORY ENTER für Felder erlaubt.

Diese Option ist für SCS-Drucker nicht gültig. Wenn für einen SCS-Drucker VALIDATION(YES) angegeben wird, dann gibt das System eine Fehlermeldung aus und die Angabe dieser Option wird ignoriert.

Sie können das strukturierte Feld QUERY verwenden, um festzustellen, ob die Einheit für die Verwendung der Validierungsfunktion eingerichtet wurde.

#### **VERTICALFORM({NO|YES})**

Gibt an, ob die Einheit über die Funktion für vertikale Formulare verfügt. Folgende Einheiten können diese Funktion verwenden: logische Stapelverarbeitungseinheiten, logische Stapeldatenaustauscheinheiten, interaktive logische Einheiten sowie logische SCSPT- oder LUTYPE4-Einheiten.

**NO** Die Einheit verfügt nicht über die Funktion für vertikale Formulare.

**YES** Die Einheit verfügt über die Funktion für vertikale Formulare.

**XRFSIGNOFF**

Dieses Attribut ist veraltet, wird jedoch zur Sicherstellung der Kompatibilität mit früheren Releases von CICS weiterhin unterstützt.

---

## URIMAP-Ressourcen

Bei URIMAP-Definitionen handelt es sich um Ressourcen, die zum Abgleich der Anforderungen für HTTP-URIs, Atom-Feeds oder Web-Services dienen und Informationen zur Vorgehensweise bei der Verarbeitung der Anforderungen bereitstellen.

### Anforderungen eines Web-Clients an CICS als HTTP-Server

In URIMAP-Definitionen für Anforderungen für CICS als HTTP-Server ist im Attribut USAGE die Option SERVER angegeben. Diese URIMAP-Definitionen gleichen die URLs von HTTP-Anforderungen ab, deren Empfang CICS von einem Web-Cli-ent erwartet. Außerdem definieren sie, wie CICS eine Antwort auf die jeweilige Anforderung bereitstellen soll. Sie können eine URIMAP-Definition verwenden, um CICS die folgenden Informationen zu liefern:

- Bereitstellung einer statischen Antwort auf die HTTP-Anforderung mithilfe einer Dokumentschablone oder einer z/OS UNIX System Services-Datei.
- Bereitstellung einer anwendungsgenerierten Antwort auf die HTTP-Anforderung mithilfe eines Anwendungsprogramms.

Web-Client-Anforderungen kommen möglicherweise für die Verarbeitung durch direkt angehängte Benutzertransaktionen in Betracht, wobei die Webanhängetask umgangen wird. Weitere Informationen hierzu finden Sie in HTTP requests are processed by directly attached user transactions.

Informationen zum Konfigurieren von CICS als HTTP-Server finden Sie in Enabling CICS web support for CICS as a HTTP server. Informationen zu den CICS-Ressourcen, die für den HTTP-Server erforderlich sind, finden Sie in Defining resources for CICS as a HTTP server.

Informationen zur Installation von URIMAP-Ressourcendefinitionen mit CEMT und CEDA finden Sie in Installing URIMAP resource definitions. Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in Working with BAS URIMAP resource definitions.

### Anforderungen an einen Server von CICS als HTTP-Client

In URIMAP-Definitionen für Anforderungen von CICS als HTTP-Client ist im Attribut USAGE die Option CLIENT angegeben. Diese URIMAP-Definitionen geben URLs an, die verwendet werden, wenn eine Benutzeranwendung, die als Web-Client verwendet wird, über die CICS-Webunterstützung eine Anforderung an einen HTTP-Server absetzt. Wenn Sie zu diesem Zweck eine URIMAP-Definition einrichten, dann können Sie die Angabe der URL in Ihrem Anwendungsprogramm vermeiden. Sie können außerdem geöffnete Verbindungen zur Wiederverwendung durch dieselbe Anwendung oder andere Anwendungen in einen Pool einstellen. Weitere Informationen hierzu finden Sie in Creating a URIMAP resource for CICS as a HTTP client.



## Anforderungen eines SOAP-Clients an CICS als Web-Service-Provider

URIMAP-Definitionen für Anforderungen eingehender Web-Services verfügen über ein Attribut **USAGE**, für das **PIPELINE** angegeben ist. Diese URIMAP-Definitionen ordnen einen URI für eine Anforderung eines eingehenden Web-Service zu (d. h., eine Anforderung, mit der ein Client einen Web-Service in CICS startet). Dabei werden die **PIPELINE**- und **WEBSERVICE**-Ressourcen verwendet, die die auszuführenden Verarbeitungsoperationen angeben.

Für Service-Provider, die mit dem CICS-Web-Service-Assistenten bereitgestellt werden, werden die URIMAP-Ressourcen automatisch erstellt, wenn das Abholverzeichnis durchsucht wird. Dieses Scanning wird ausgeführt, wenn die **PIPELINE**-Ressource installiert wird oder wenn der Befehl **PERFORM PIPELINE SCAN** abgesetzt wird. Die URIMAP-Ressource, die für CICS die Informationen für die Zuordnung der **WEBSERVICE**-Ressource zu einem bestimmten URI bereitstellt, ist eine erforderliche Ressource. Die Attribute für diese Ressource werden durch eine Datei für Web-Service-Bindungen im Abholverzeichnis angegeben. Die URIMAP-Ressource, die für CICS die Informationen für die Zuordnung der WSDL-Archivdatei oder des WSDL-Dokuments zu einem bestimmten URI bereitstellt, ist eine optionale Ressource. Sie wird erstellt, wenn entweder eine WSDL-Datei oder eine WSDL-Archivdatei im Abholverzeichnis vorhanden ist.

In *Web services overview* finden Sie weiterführende Informationen zu den Web-Services in CICS.

SOAP-Client-Anforderungen kommen möglicherweise für die Verarbeitung durch direkt angehängte Benutzertransaktionen in Betracht, wobei die Webanhängetask umgangen wird. Weitere Informationen hierzu finden Sie in *HTTP requests are processed by directly attached user transactions*.

## Anforderungen an einen Web-Service-Provider von CICS als Web-Service-Anforderer

Die URIMAP-Definition für Anforderungen abgehender Web-Services, die mit dem Befehl **INVOKE WEBSERVICE** ausgeführt werden, sind optional und verfügen über ein Attribut **USAGE** mit der Option **CLIENT**. Sie können die URIMAP-Definition z. B. verwenden, um die Cipher-Suites oder die Zertifikatsbezeichnung anzugeben, die verwendet werden soll(en), wenn Sie eine Socketverbindung einrichten, die mit TLS (Transport Layer Security) arbeitet. Sie können außerdem geöffnete Verbindungen zur Wiederverwendung für weitere **INVOKE WEBSERVICE**-Befehle durch die Anwendung in einen Pool einstellen, anstatt jedes Mal eine neue Verbindung zu öffnen. URIMAP-Definitionen ermöglichen Administratoren die Steuerung der Verbindung zu einem fernen Web-Service. Sie können den URI bei Bedarf ändern oder die URIMAP aktivieren bzw. inaktivieren, um die Herstellung der Verbindung zuzulassen oder nicht zuzulassen.

Für Serviceanforderer erstellt CICS nicht automatisch URIMAP-Ressourcen, wenn die **PIPELINE**-Ressource installiert wird oder wenn der Befehl **PERFORM PIPELINE SCAN** abgesetzt wird.

## Anforderungen eines Web-Clients an die Java-Webanwendung

URIMAP-Definitionen für die Java-Webanwendungen verfügen über ein Attribut **USAGE** mit der Option **VMSERVER**. Diese URIMAP-Definitionen gleichen eine eingehende Anforderung eines Web-Clients mit einem Servlet oder einer JSP ab,

das bzw. die auf einem JVM-Server ausgeführt wird. Der URIMAP-Abgleich für Webanwendungen wird während der ersten HTTP-Anforderung und nicht für weitergeleitete Anforderungen ausgeführt. Die URIMAP kann eine CICS-Transaktion, die zur Ausführung der Anforderung auf dem JVM-Server dient, und optional eine Benutzer-ID angeben.

Weitere Informationen hierzu finden Sie in *Deploying a Java EE application in a CICS bundle to a Liberty JVM server*.

## **Atom-Feed-Anforderungen**

URIMAP-Definitionen für Atom-Feed-Anforderungen verfügen über ein Attribut `USAGE` mit der Option `ATOM`. Diese URIMAP-Definitionen gleichen eine eingehende Anforderung eines Web-Clients für ein Atom-Dokument ab. Sie verweisen auf eine `ATOMSERVICE`-Ressourcendefinition, die das Atom-Dokument angibt, das zurückgegeben werden soll.

Weitere Informationen hierzu finden Sie in *Setting up CICS definitions for an Atom feed*.

Atom-Feed-Anforderungen kommen möglicherweise für die Verarbeitung durch direkt angehängte Benutzertransaktionen in Betracht, wobei die Webanhangetask umgangen wird. Weitere Informationen hierzu finden Sie in *HTTP requests are processed by directly attached user transactions*.

## **Geschäftsereignisoutputs eines HTTP-Ereignisverarbeitungsadapters**

URIMAP-Definitionen für die CICS-Ereignisverarbeitung verfügen über ein Attribut `USAGE` mit der Option `CLIENT`. Diese URIMAP-Definitionen geben eine URL an, die vom HTTP-Ereignisverarbeitungsadapter zur Ausgabe von Ereignissen an einen HTTP 1.1-konformen Server verwendet wird, der mit HTTP POST arbeitet. Sie können geöffnete Verbindungen zur Wiederverwendung für weitere Ereignisoutputs in einen Pool einstellen. In *Event processing overview* finden Sie weiterführende Informationen zur Ereignisverarbeitung von CICS.

Für CICS als HTTP-Server umfassen die URIMAP-Definitionen die Mehrzahl der Funktionen, die zuvor vom Analyseprogramm bereitgestellt wurden, das der `TCPIPSERVICE`-Definition zugeordnet ist. Sie können im Verarbeitungspfad weiterhin ein Analyseprogramm verwenden, wenn dies erforderlich ist.

## **URIMAP-Suchreihenfolge**

Wenn mehrere URIMAPs verfügbar sind, dann werden URIMAPs mit demselben URI und einer Angabe für `TCPIPSERVICE` zuerst durchsucht. URIMAPs, die Platzhalterzeichen enthalten, werden zuerst auf übereinstimmende URIs und zuletzt auf die `TCPIPSERVICE`-Werte durchsucht. Diese Suchreihenfolge stellt sicher, dass die URIMAP mit dem höchsten Übereinstimmungsgrad verwendet wird.

## **URIMAP-Ressourcen in CICS-Bundles**

Sie können ein CICS-Bundle verwenden, um eine URIMAP-Ressourcendefinition zu erstellen, zu bearbeiten und zu installieren. Wenn Sie eine URIMAP-Ressource auf diese Weise erstellen, dann müssen Sie zum Verwalten des Lebenszyklus dieser Ressource das CICS-Bundle verwenden. Die unabhängige Verwaltung der Ressource ist nicht möglich. Sie können eine URIMAP-Ressource abfragen, die von einem

CICS-Bundle dynamisch generiert wird. Verwenden Sie hierzu den Befehl **EXEC CICS INQUIRE URIMAP** oder **CEMT INQUIRE URIMAP**. Der Befehl SET oder DISCARD kann jedoch nicht für eine URIMAP-Ressource ausgeführt werden, die von einem CICS-Bundle dynamisch generiert wird. Sie müssen die Befehle für die BUNDLE-Ressource absetzen und CICS wendet sie dann auf die URIMAP-Ressource an. Weitere Informationen zum Definieren von Ressourcen in CICS-Bundles finden Sie in Defining CICS bundles.

## **URIMAP-Ressourcen als Anwendungseinstiegspunkte**

Die folgenden Einsatzszenarios werden unterstützt, wenn Sie die URIMAP-Ressource als Anwendungseinstiegspunkt verwenden:

### **URIMAP-Ressource mit USAGE(JVMSEVER) und Verwendung von Liberty**

Wenn die URIMAP-Ressource, die vom Liberty-JVM-Server lokalisiert wird, mit einem Anwendungseinstiegspunkt definiert wird, dann werden die Anwendungskontextdaten für die Anforderungsprozessortask des JVM-Servers oder seinen Aliasnamen gemäß der Angabe im Transaktionsattribut festgelegt.

### **URIMAP-Ressource mit USAGE(SERVER) oder USAGE(PIPELINE) ohne Verwendung von WebSphere MQ**

Wenn die URIMAP-Ressource, die von CWXN lokalisiert wurde, mit einem Anwendungseinstiegspunkt definiert wird, dann werden die Anwendungskontextdaten für die Task CWXN festgelegt. Anschließend können die Anwendungskontextdaten (da es sich bei CWXN um eine Benutzertask handelt) automatisch an die Aliastransaktionen weitergegeben werden, die mit CWXN gestartet wurden. Beispiele sind CPIH für USAGE(PIPELINE), CWBA für USAGE(SERVER) oder die Transaktion, die im Feld **TRANSACTION** angegeben wird.

Die folgenden Einsatzszenarios werden nicht unterstützt, wenn Sie die URIMAP-Ressource als Anwendungseinstiegspunkt verwenden:

### **URIMAP-Ressource mit USAGE(PIPELINE) und Verwendung von WebSphere MQ**

Dieses Szenario wird nicht unterstützt.


### **URIMAP-Ressource mit USAGE(CLIENT) oder USAGE(ATOM)**

Diese Szenarios werden nicht unterstützt. Ein Bundle mit URIMAP und USAGE(CLIENT) oder ein Bundle mit URIMAP und USAGE(ATOM) weisen den Status DISABLED auf, wenn als Anwendungseinstiegspunkt URIMAP definiert ist. Wenn das Bundle installiert wird und wenn Sie versuchen, das Bundle zu aktivieren, dann weist die ENTRYPOINT-Bundlekomponente innerhalb des Bundles den Status DISABLED auf. Demzufolge hat auch das Bundle den Status DISABLED.

Verwenden Sie die Befehle INQUIRE URIMAP und CEMT INQUIRE URIMAP, um die Details der Anwendung zurückzugeben, für die die URIMAP-Ressource als Einstiegspunkt definiert ist.

Weitere Informationen zu Anwendungseinstiegspunkten finden Sie in Application entry points.

#### **Related tasks:**

 [Encrypting data sets](#)

## URIMAP: Zugehörige Ressourcen

Die Attribute und Werte, die Sie für eine URIMAP-Ressource angeben, müssen mit den Angaben übereinstimmen, die für andere Ressourcen angegeben wurden.

Allerdings führt CICS keine Konsistenzprüfung für alle zugehörigen Ressourcen durch, wenn die URIMAP-Definition installiert wird und meldet aus diesem Grund auch die Mehrzahl der während der Installation auftretenden Inkonsistenzen nicht. Die Ausnahme stellt die in Installing URIMAP resource definitions erläuterte Prüfung dar.

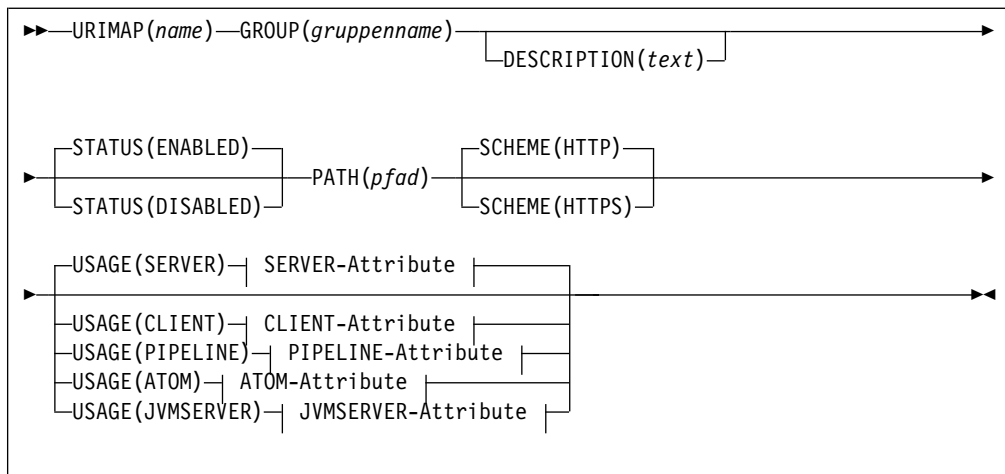
- In URIMAP-Definitionen für CICS als HTTP-Server und Web-Services gibt das Attribut TCPIPSERVICE den Namen einer TCPIPSERVICE-Ressourcendefinition an, die einen Port für eingehende Daten definiert, dem diese URIMAP-Definition zugeordnet ist. Das Attribut ist optional. Wird es nicht angegeben, dann gilt die URIMAP-Definition für alle eingehenden Anforderungen für alle TCPIPSERVICE-Definitionen. Wenn das Attribut TCPIPSERVICE angegeben wird, dann gilt Folgendes:
  - Die ausgewählte TCPIPSERVICE-Ressourcendefinition muss die Angabe PROTOCOL(HTTP) enthalten.
  - Wenn in der TCPIPSERVICE-Ressourcendefinition SSL(YES) oder SSL(CLIENTAUTH) angegeben ist, dann muss im Attribut SCHEME der URIMAP-Definition die Option HTTPS angegeben sein. Wenn eine URIMAP-Definition mit dem HTTPS-Schema mit einer Anforderung übereinstimmt, die von einem Web-Client ausgeführt wird, dann überprüft CICS, ob der Port für eingehende Daten, der von der Anforderung verwendet wird, mit SSL arbeitet. Wenn für den Port nicht SSL angegeben ist, wird die Anforderung mit dem Statuscode 403 (Forbidden) zurückgewiesen.

Sie müssen die ausgewählte TCPIPSERVICE-Definition installieren, bevor die URIMAP-Definition verwendet werden kann.

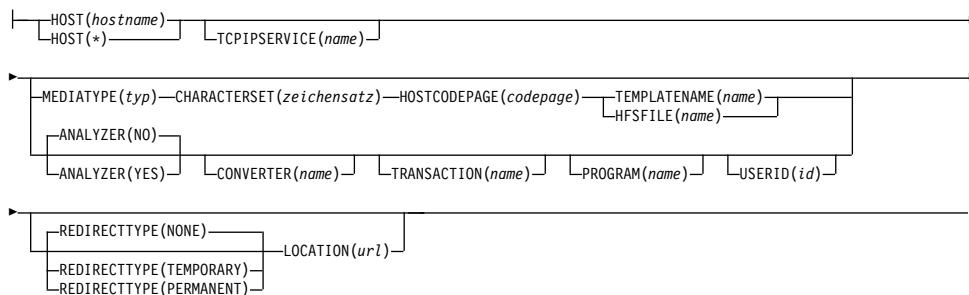
- In URIMAP-Definitionen für CICS als HTTP-Server, in denen die Bereitstellung einer statischen Antwort vorgesehen ist, gibt das Attribut TEMPLATENAME den aus 1 - 48 Zeichen bestehenden Namen einer CICS-Dokumentschablone an, die zur Erstellung der statischen Antwort benutzt wird. Die Dokumentschablone muss mithilfe einer DOCTEMPLATE-Ressourcendefinition definiert werden und das in dieser Definition festgelegte Attribut TEMPLATENAME gibt den Namen an, der in der URIMAP-Definition verwendet wird. Die DOCTEMPLATE-Ressourcendefinition muss verfügbar sein, bevor die URIMAP-Definition verwendet werden kann.

## URIMAP-Attribute

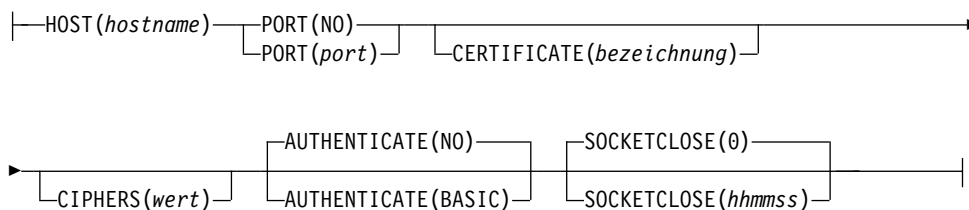
Beschreibt die Syntax und die Attribute der URIMAP-Ressource.



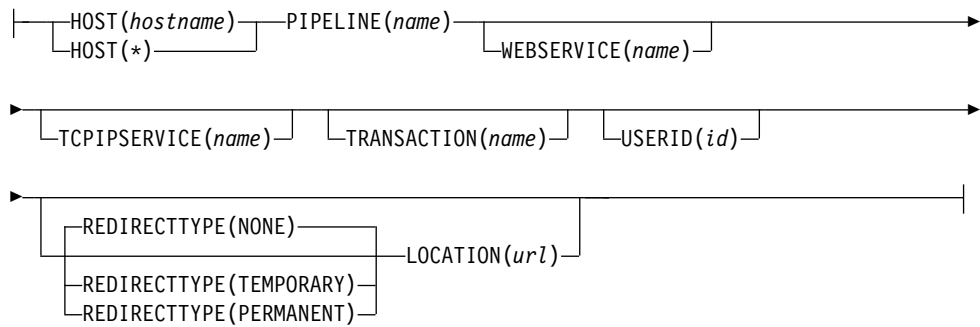
### SERVER-Attribute:



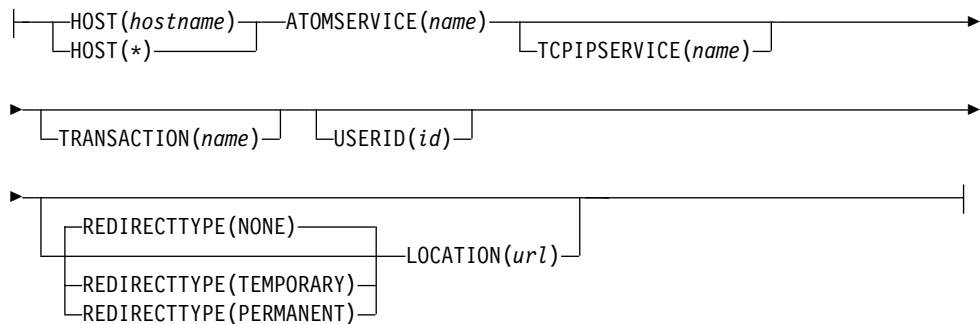
### CLIENT-Attribute:



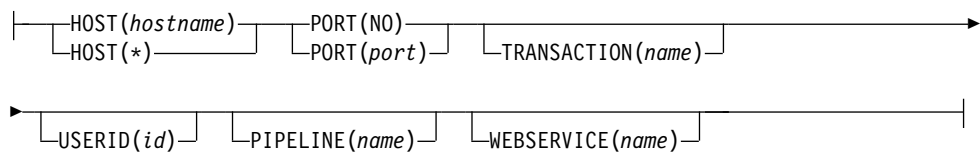
### PIPELINE-Attribute:



### ATOM-Attribute:



### JVMSERVER-Attribute:



### ANALYZER({NO|YES})

Dieses Attribut gilt für USAGE(SERVER), wenn eine anwendungsgenerierte Antwort bereitgestellt werden muss. Für andere Verwendungstypen wird für das Attribut die Einstellung NO erzwungen.

Für ANALYZER muss für HTTP-Anforderungen die Einstellung NO festgelegt werden, damit diese Anforderungen durch eine direkte Zuordnung zur Benutzertransaktion verarbeitet werden können. Weitere Informationen hierzu finden Sie in HTTP requests are processed by directly attached user transactions. ANALYZER gibt an, ob zur Verarbeitung von HTTP-Anforderungen ein Analyseprogramm verwendet werden soll. Das Analyseprogramm muss den TCPIP-

SERVICE-Definitionen zugeordnet werden, zu denen diese URIMAP-Definition gehört. (Ein Analyseprogramm muss sich in der lokalen CICS-Region befinden.) Bei Angabe von YES wird das Analyseprogramm ausgeführt. Bei Angabe von NO wird das Analyseprogramm hingegen nicht verwendet.

Das Standardanalyseprogramm DFHWBAAX sowie das Beispielanalyseprogramm DFHWBADX analysieren eine Anforderung nicht, wenn für die Anforderung eine übereinstimmende URIMAP-Definition gefunden wurde. Dies gilt auch dann, wenn für URIMAP die Einstellung ANALYZER(YES) angegeben wurde.

Wenn ein Analyseprogramm verwendet wird, dann können Sie dennoch die Attribute CONVERTER, TRANSACTION, USERID und PROGRAM weiterhin verwenden. Die Werte, die Sie in diesen Attributen angeben, werden als Eingabe für das Analyseprogramm verwendet, können jedoch von diesem überschrieben werden. Alternativ hierzu können Sie für diese Attribute auch keine Werte angeben und sie durch das Analyseprogramm festlegen lassen.

#### **ATOMSERVICE**(*name*)

Dieses Attribut gilt für USAGE(ATOM).

Wenn ein Client eine Anforderung für einen Atom-Feed an CICS absetzt, und dabei den URI verwendet, der in dieser URIMAP-Definition angegeben ist, dann wird in ATOMSERVICE der aus einem bis acht Zeichen bestehende Name der ATOMSERVICE-Ressourcendefinition für den Atom-Feed angegeben. Die ATOMSERVICE-Ressourcendefinition definiert einen Atom-Service, einen Atom-Feed, eine Atom-Erfassung oder ein Atom-Kategoriedokument und identifiziert die Atom-Konfigurationsdatei, die CICS-Ressource oder das Anwendungsprogramm sowie die XML-Bindung, die verwendet werden, um die Daten für den Feed bereitzustellen.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

#### **AUTHENTICATE**(**{NO|BASIC}**)

Dieses Attribut gilt für USAGE(CLIENT).

In AUTHENTICATE wird angegeben, ob Informationen für die HTTP-Basisauthentifizierung an den HTTP-Server gesendet werden sollen. Für AUTHENTICATE(BASIC) ist eine Benutzer-ID und ein Kennwort erforderlich, die über den globalen Benutzerexit XWBAUTH bereitgestellt werden. Alternativ kann die Bereitstellung auch in Form von Werten in den API-Befehlen wie z. B. **WEB SEND** oder **WEB CONVERSE** erfolgen. Der globale Benutzerexit XWBAUTH wird nicht aufgerufen, wenn im API-Befehl die Optionen USERNAME und PASSWORD angegeben werden. Wenn Sie einen Authentifizierungswert im API-Befehl angeben, dann wird anstelle des AUTHENTICATE-Werts, der in der URIMAP-Ressource angegeben wurde, dieser Wert verwendet.

#### **CERTIFICATE**(*bezeichnung*)

Dieses Attribut gilt für USAGE(CLIENT).

In CERTIFICATE wird die Bezeichnung des X.509-Zertifikats angegeben, das während des SSL-Handshakes als SSL-Clientzertifikat verwendet wird. Zertifikatsbezeichnungen können bis zu 32 Zeichen lang sein. Dieses Attribut wird nur dann verwendet, wenn der in der URIMAP-Definition angegebene URI für eine HTTPS-Anforderung verwendet werden soll, die von einem CICS-Clientensystem abgesetzt wurde. Hierbei entscheidet der Server, ob ein SSL-Clientzerti-

fikat angefordert werden soll. Falls dies der Fall ist, dann stellt CICS die Zertifikatsbezeichnung bereit, die in der URIMAP-Definition angegeben wurde. Wenn dieses Attribut nicht angegeben wird, dann wird das Standardzertifikat verwendet, das im Schlüsselring für die Benutzer-ID der CICS-Region definiert wurde. Das Zertifikat muss in einem Schlüsselring in der Datenbank des externen Sicherheitsmanagers gespeichert werden. Das angegebene Zertifikat oder das Standardzertifikat muss über einen privaten Schlüssel verfügen, da andernfalls die URIMAP-Definition nicht installiert werden kann. Wenn Sie kein Zertifikat verwenden wollen, dann lassen Sie dieses Feld leer und stellen Sie sicher, dass der Schlüsselring, der von der CICS-Region verwendet wird, kein Standardzertifikat umfasst. Weitere Informationen hierzu finden Sie in *Building a key ring manually*.

#### **CHARACTERSET**(*zeichensatz*)

Dieses Attribut gilt für USAGE(SERVER), wenn eine statische Antwort bereitgestellt werden muss.

In CHARACTERSET wird der aus einem bis 40 Zeichen bestehende Name des Zeichensatzes angegeben, in den CICS den Entitätshauptteil der Antwort konvertieren soll, die an den Web-Client gesendet wird. CICS unterstützt nicht alle Zeichensätze, die von IANA (Internet Assigned Numbers Authority) aufgeführt werden. In HTML coded character sets werden die IANA-Zeichensätze aufgeführt, die von CICS unterstützt werden. Der Wert dieses Attributs ist im Header "Content-Type" der Antwort enthalten.

Sie müssen CHARACTERSET angeben, wenn eine statische Antwort bereitgestellt wird und wenn im Attribut MEDIATYPE ein Texttyp angegeben ist.

#### **CIPHERS**(*wert*)

Dieses Attribut gilt für USAGE(CLIENT).

Das Attribut CIPHERS kann auf eine der beiden folgenden Arten angegeben werden:

- Eine aus maximal 56 Hexadezimalziffern bestehende Zeichenfolge, die als Liste mit bis zu 28 zweistelligen Cipher-Suite-Codes interpretiert wird.
- Der Name der Spezifikationsdatei für die SSL-Cipher-Suite, bei der es sich um eine z/OS UNIX-Datei im Unterverzeichnis security/ciphers des Verzeichnisses handelt, das im Systeminitialisierungsparameter **USSCONFIG** angegeben ist. Wenn **USSCONFIG** z. B. auf den Wert /var/cicsts/dfhconfig gesetzt wird und **CIPHERS** auf den Wert strongciphers.xml, dann lautet der vollständig qualifizierte Dateiname /var/cicsts/dfhconfig/security/ciphers/strongciphers.xml. Weitere Informationen hierzu finden Sie in *Creating an SSL cipher suite specification file*.

Wenn Sie die CEDA-Transaktion zum Definieren der Ressource verwenden, dann wird das Attribut von CICS automatisch mit einer Standardliste zulässiger Codes initialisiert. Damit CICS das Attribut initialisieren kann, muss der Systeminitialisierungsparameter KEYRING in der CICS-Region angegeben werden, in der CEDA ausgeführt wird. Wird KEYRING nicht angegeben, dann kann CICS das Attribut nicht initialisieren. Die Standardliste der Codes trägt die Bezeichnung 35363738392F303132330A1613100D15120F0C, es sei denn, der Systeminitialisierungsparameter **NISTSP800131A=CHECK** wird festgelegt. In diesem Fall lautet die Bezeichnung 35363738392F303132330A1613100D.

Sie können die Cipher-Codes anders sortieren oder aus der ursprünglichen Liste entfernen. Allerdings ist es nicht möglich, Cipher-Codes hinzuzufügen, die sich nicht in der Standardliste für die angegebene Verschlüsselungsstufe befinden. Zum Zurücksetzen des Werts auf die standardmäßige Codeliste müssen Sie alle Cipher-Suite-Codes löschen. Das Feld wird dann automatisch erneut



mit der Standardliste gefüllt.

Weitere Informationen hierzu finden Sie in Cipher suites and cipher suite specification files.

#### **CONVERTER**(*name*)

Dieses Attribut gilt für USAGE(SERVER), wenn eine anwendungsgenerierte Antwort bereitgestellt werden muss.

In CONVERTER ist der aus einem bis acht Zeichen bestehende Name eines Konvertierungsprogramms angegeben, das die Anforderung sowie die Antwort konvertiert. Normalerweise setzt ein Konvertierungsprogramm die HTTP-Anforderung in einen Kommunikationsbereich (COMMAREA) um, der von einem Anwendungsprogramm verwendet werden kann, und setzt die Ausgabe in eine HTTP-Antwort um. Es kann jedes beliebige Konvertierungsprogramm verwendet werden, das in der lokalen CICS-Region verfügbar ist.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Anders als die Beziehung zwischen dem Analyseprogramm und der TCPIP-SERVICE-Definition besteht zwischen dem Konvertierungsprogramm und der TCIPSERVICE-Definition keine Beziehung.

Wenn für das Attribut ANALYZER die Option YES angegeben ist, dann wird das Attribut CONVERTER als Eingabe für das Analyseprogramm verwendet, kann jedoch vom Analyseprogramm überschrieben werden. Wenn ein Konvertierungsprogramm verwendet wird, dann können Sie das Attribut PROGRAM der URIMAP-Definition weiterhin angeben, die von Ihnen für dieses Attribut angegebenen Werte können jedoch vom Konvertierungsprogramm überschrieben werden. Alternativ hierzu können Sie für dieses Attribut auch keinen Wert angeben und ihn durch das Konvertierungsprogramm festlegen lassen.

#### **DESCRIPTION**(*text*)

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

#### **GROUP**(*gruppenname*)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

### **HFSFILE(*name*)**

Dieses Attribut gilt für USAGE(SERVER), wenn eine statische Antwort bereitgestellt werden muss.

Das Attribut HFSFILE gibt den vollständig qualifizierten (absoluten) oder relativen Namen einer z/OS UNIX System Services-zFS-Datei an, die den Hauptteil der statischen Antwort bildet, die vom Web-Client für die HTTP-Anforderung gesendet wird. Es können bis zu 255 Zeichen verwendet werden.

Der Name kann als absoluter Pfad einschließlich aller Verzeichnisse und beginnend mit einem Schrägstrich (/) angegeben werden. Beispiel: /u/facts/images/bluefish.jpg. Alternativ hierzu können Sie auch einen relativen Pfad angeben. Beispiel: facts/images/bluefish.jpg.

- Für eine Datei für statische Antworten für eine URIMAP-Ressource, die mithilfe einer Onlinere Ressourcendefinition definiert wurde, wird entweder ein vollständig qualifizierter Dateipfad (mit einem Schrägstrich (/) als Präfix) oder ein Dateipfad relativ zum Ausgangsverzeichnis der Benutzer-ID für die CICS-Region angegeben.
- Für eine Datei für statische Antworten für eine URIMAP-Ressource, die in einem CICS-Bundle definiert wurde, ist der Dateipfad relativ zum Stammverzeichnis des CICS-Bundles angegeben. Die zFS-Datei muss im CICS-Bundle mit der URIMAP-Ressource gepackt werden. Die Attribute MEDIA-TYPE und HOSTCODEPAGE müssen angegeben werden, um CICS den Dateityp und die Codepage mitzuteilen, mit denen die Datei codiert wurde. Wenn die Datei in einem CICS-Bundle definiert ist und mit CICS Explorer in zFS exportiert wird, dann wird die Datei mit einer Codepage codiert, die identisch mit der Codepage ist, die von CICS Explorer verwendet wird. Weitere Informationen hierzu finden Sie in Referencing zFS artifacts in a bundle.

Der angegebene Wert muss ein gültiger Name einer UNIX-Datei sein:

- Er darf keine eingebetteten Leerzeichen enthalten.
- Er darf keine aufeinanderfolgenden Vorkommen des Zeichens / enthalten.
- Die Groß-/Kleinschreibung muss beachtet werden.

#### **Zulässige Zeichen:**

A-Z a-z 0-9 . / \_ # @ -

Wenn Sie TEMPLATENAME oder HFSFILE angeben, dann muss für ANALYZER die Einstellung NO festgelegt werden. Andere Attribute, die sich nur auf anwendungsgenerierte Antworten beziehen (TRANSACTION, CONVERTER und PROGRAM), dürfen nicht angegeben werden.

Wenn Sie einen Pfadabgleich durchführen wollen, dann geben Sie einen Stern (\*) als Platzhalterzeichen am Ende des Pfads der zFS-Datei und außerdem auch am Ende des Pfads an, der im Attribut PATH angegeben ist. CICS verwendet den Pfadteil einer HTTP-Anforderung, der vom Platzhalterzeichen abgedeckt wird, und setzt diesen Teil als letzten Teil des Dateipfads ein.

Sie können beispielsweise eine URIMAP-Definition mit der folgenden Angabe für das Attribut PATH erstellen:

findout/pictures/\*

Des Weiteren können Sie das Attribut HFSFILE wie folgt angeben:

/u/facts/images/\*

Die URIMAP-Definition wird zur Verarbeitung eingehender HTTP-Anforderungen verwendet:

`http://www.example.com/findout/pictures/bluefish.jpg`

Anstelle des Sterns (\*) fügt CICS `bluefish.jpg` zu dem zFS-Dateipfad hinzu, der in der URIMAP-Definition angegeben wurde, sodass die zFS-Datei `/u/facts/images/bluefish.jpg`

als statische Antwort benutzt wird.

Die Angabe für HFSFILE darf nicht nur aus einem Stern bestehen. Sie müssen mindestens eine Ebene der Verzeichnisstruktur angeben.

Wenn Sie IRIs (Internationalized Resource Identifiers) verwenden, die Unicode-Zeichen enthalten, dann müssen die Unicode-Zeichen im zFS-Dateinamen und im entsprechenden Pfad in der prozentcodierten Darstellung als Escapezeichenfolge angegeben werden. Im Web werden entsprechende Tools angeboten, die Sie bei diesem Arbeitsschritt unterstützen. Durchsuchen Sie das Web nach „Unicode percent escaped“.

Eine Abfragezeichenfolge darf nicht in die Inhalte der zFS-Datei eingesetzt werden, obwohl Sie die zFS-Datei als CICS-Dokumentschablone definieren und sie mithilfe des Attributs `TEMPLATENAME` anstelle des Attributs `HFSFILE` angeben können.

#### **HOST(hostname|\*)**

Dieses Attribut gilt für alle USAGE-Optionen.

In `HOST` wird der Hostname des URI angegeben, für den die URIMAP-Definition gilt, oder die entsprechenden IPv4- oder IPv6-Adresse. Der Name kann bis zu 116 Zeichen lang sein. In The components of a URL wird jede der Komponenten sowie die Vorgehensweise zu deren Begrenzung erläutert.

Das Attribut `HOST` muss vorhanden sein. Das Attribut `HOST` darf nur alphanumerische Zeichen, Bindestriche (-), Doppelpunkte (:) oder Punkte (.) enthalten, obwohl die Verwendung von Doppelpunkten bei Angabe eines zeichenbasierten Hostnamens anstelle einer IP-Adresse nicht zulässig ist. CICS überprüft den Hostnamen während der Definition. Ein Hostname kann in beliebiger Groß-/Kleinschreibung eingegeben werden. Wenn jedoch anstelle einer IP-Adresse ein zeichenbasierter Hostname angegeben wird, dann konvertiert das System diesen Hostnamen in der URIMAP-Definition in Kleinbuchstaben.

Wenn Sie `USAGE(SERVER)`, `USAGE(PIPELINE)`, `USAGE(ATOM)` oder `USAGE(JVMSEVER)` angeben, dann können Sie im Attribut `HOST` einen einzelnen Stern verwenden, sodass die URIMAP-Definition mit einem beliebigen Hostnamen übereinstimmt. Die Verwendung eines Sterns als Platzhalterzeichen mit einem anderen Zeichen ist im Attribut `HOST` nicht möglich. Geben Sie in diesem Attribut für diese Verwendungstypen keine Portnummer an.

URIMAP-Ressourcen unterstützen IRIs (Internationalized Resource Identifiers), die Unicode-Zeichen enthalten können. Wenn Sie einen Hostnamen angeben, der Unicode-Zeichen enthält, dann müssen Sie ihn in das Punycode-Format konvertieren, das in RFC 3492 beschrieben wird. CICS stellt kein Tool zur Durchführung dieser Konvertierung bereit, im Internet werden jedoch kostenlose Anwendungen angeboten, mit denen die Konvertierung vom Unicode-Format ins Punycode-Format durchgeführt werden kann. Wenn Sie als Hostnamen einen Stern angeben, dann ist die Verwendung des Punycode-Formats nicht erforderlich. Weitere Informationen zu IRIs finden Sie in Internationalized Resource Identifiers (IRIs).

Sie können IPv4- und IPv6-Adressen in verschiedenen zulässigen Formaten angeben. Weitere Informationen zu Adressformaten finden Sie in IP-Adressen .

Wenn CICS als HTTP-Client eingesetzt wird und wenn Sie USAGE(CLIENT) angeben, dann können Sie eine Portnummer in der Anforderung an den Server angeben:

- Verwenden Sie das Attribut PORT, um die Portnummer anzugeben. Das Attribut PORT macht die Verwendung des Attributs HOST zur Angabe einer Portnummer überflüssig.
- Aus Gründen der Kompatibilität in existierenden Programmen, die mit nativen IPv4-Adressen und Hostnamen arbeiten, können Sie das Attribut HOST jedoch weiterhin zur Angabe der Portnummer verwenden. Native IPv4-Adressen und Hostnamen sind die einzigen Formate, in denen Sie die Portnummer zusammen mit einem vorangestellten Doppelpunkt (:) angeben können. Beispiel: 1.2.3.4:80 oder hostname.com:443.
- Wenn Sie eine IPv6-Adresse (oder einen Hostnamen, der in eine IPv6-Adresse aufgelöst wird) angeben, dann müssen Sie sicherstellen, dass Sie in einer Dualmodusumgebung (IPv4 und IPv6) arbeiten und dass der Client oder der Server, mit dem Sie kommunizieren, ebenfalls in einer Dualmodusumgebung (IPv4 und IPv6) arbeitet. Weitere Informationen zu IPv6 finden Sie in Understanding IPv6 and CICS.
- Für native IPv6-Adressen müssen Sie das Attribut PORT verwenden, um die Portnummer anzugeben. IPv6-Adressen erfordern die Angabe eckiger Klammern, um die Adresse von der Portnummer zu trennen. Da eckige Klammern nicht in allen EBCDIC-Zeichensätzen feste Werte darstellen, werden sie im Attribut HOST nicht unterstützt.
- Geben Sie die Portnummer nur an, wenn sie sich vom entsprechenden Standardwert für das Schema unterscheidet (80 für HTTP ohne SSL oder 443 für HTTPS und HTTP mit SSL).
- Wenn Sie eine Portnummer im Attribut HOST und eine andere Portnummer im Attribut PORT angeben, dann wird ein Fehler zurückgegeben. Wenn Sie weder im Attribut HOST noch im Attribut PORT eine Portnummer angeben, dann wird die Standardportnummer des jeweiligen Schemas verwendet.

#### **HOSTCODEPAGE**(*codepage*)

Dieses Attribut gilt für USAGE(SERVER), wenn eine statische Antwort bereitgestellt werden muss.

Das Attribut HOSTCODEPAGE dient zur Angabe des aus einem bis zehn Zeichen bestehenden Namens der IBM Codepage (EBCDIC), in der das Textdokument angegeben ist, das die statische Antwort bildet. CICS benötigt diese Informationen, um die Codepages für den Entitätshauptteil der statischen Antwort zu konvertieren.

Das CICS-Standardformat des Namens der Host-Codepage besteht aus der Codepagenummer (oder allgemein CCSID), die aus drei bis fünf Dezimalziffern bestehen kann, die bei Bedarf mit nachfolgenden Leerzeichen aufgefüllt werden. Für die Codepage 37, die weniger als drei Ziffern vorgibt, lautet das Standardformat 037. CICS akzeptiert alle (mit nachfolgenden Leerzeichen aufgefüllten) Dezimalzahlen im Bereich zwischen 1 und 65535 als Codepagename. Dies gilt auch dann, wenn nicht das Standardformat verwendet wird.

Sie müssen HOSTCODEPAGE angeben, wenn eine statische Antwort bereitgestellt wird und wenn im Attribut MEDIATYPE ein Texttyp angegeben ist.

#### **LOCATION**(*url*)

Dieses Attribut gilt für USAGE(SERVER), USAGE(PIPELINE) und USAGE(ATOM).

Das Attribut LOCATION dient zur Angabe einer URL mit bis zu 255 Zeichen, an die die Clientanforderung gesendet wird. Die URL muss vollständig sein, also Schema, Host, Pfadkomponenten und entsprechende Begrenzungszeichen umfassen. CICS überprüft nicht, ob die URL gültig ist, sodass Sie sicherstellen müssen, dass das Ziel vorhanden ist und die URL korrekt angegeben wurde.

In der Beschreibung des Attributs PATH werden die Zeichen aufgelistet, die in einer URL nicht verwendet werden dürfen. Diese Zeichen dürfen im Attribut LOCATION nicht verwendet werden. Eine Ausnahme bildet in diesem Zusammenhang das Nummernzeichen (#), das im Attribut LOCATION als Trennzeichen vor einer Fragment-ID angegeben werden kann, die auf die URL folgt.

Das Attribut REDIRECTTYPE wird verwendet, um den Typ der Umleitung anzugeben. Wenn eine vorübergehende oder dauerhafte Umleitung angegeben wird, dann wird die URL im Attribut LOCATION für die Umleitung verwendet. Wird keine Umleitung angegeben, wird die URL im Attribut LOCATION ignoriert. Mit dem Befehl SET URIMAP können Sie die Werte im Attribut REDIRECTTYPE und im Attribut LOCATION ändern.

#### **MEDIATYPE**(*typ*)

Dieses Attribut gilt für USAGE(SERVER), wenn eine statische Antwort bereitgestellt werden muss.

Das Attribut MEDIATYPE gibt den Medientyp (Dateninhalt) einer statischen Antwort an, die von CICS für die HTTP-Anforderung bereitgestellt wird. Beispiel: image/jpg, text/html oder text/xml. Es können bis zu 56 Zeichen verwendet werden. Der Medientyp muss genau einen Schrägstrich (/) enthalten. Der Medientyp kann in Groß- oder Kleinbuchstaben eingegeben werden, er wird jedoch in der URIMAP-Definition in Kleinbuchstaben konvertiert.

Der Name für jeden formal erkannten Dateninhaltenstyp wird von IANA (Internet Assigned Numbers Authority) definiert. Eine Liste steht unter <http://www.iana.org/assignments/media-types/media-types.xhtml> zur Verfügung. CICS erstellt einen Header "Content-Type" für die Antwort und verwendet dazu den Wert aus diesem Attribut.

Für dieses Attribut ist kein Standardwert vorhanden. Es muss angegeben werden. Wenn im Attribut MEDIATYPE ein Texttyp angegeben wird, beispielsweise ein Typ, der mit der Zeichenfolge text/ beginnt oder die Zeichenfolge +xml enthält, dann müssen Sie auch das Attribut CHARACTERSET und das Attribut HOSTCODEPAGE angeben, damit die Codepagekonvertierung ausgeführt werden kann. Textmedientypen werden in RFC 3023 identifiziert. Weitere Informationen hierzu finden Sie unter der Adresse <http://www.ietf.org/rfc/rfc3023.txt>.

Für die Erstellung dynamischer (anwendungsgenerierter) Antworten wird dieses Attribut nicht verwendet. Der Medientyp für die Antwort wird im Befehl WEB SEND angegeben.

#### **PATH**(*pfad*)

Dieses Attribut gilt für alle USAGE-Optionen.

Das Attribut PATH gibt die Pfadkomponente des URI an, für den die URIMAP-Definition gilt. Die Länge kann bis zu 255 Zeichen (einschließlich des Schrägstrichs (/) am Anfang der Pfadkomponente betragen. Wird der Schrägstrich nicht angegeben, dann dürfen nur 254 Zeichen zur Angabe des Pfads verwendet werden. Die mindestens erforderliche Angabe für den Pfad besteht aus einem Schrägstrich, der für das Stammverzeichnis der URL-Struktur für den angegebenen Hostnamen steht. Sie können den Schrägstrich am Anfang der Pfadkomponente angeben oder weglassen. Wird er nicht angegeben, dann fügt ihn CICS allerdings während der Laufzeit hinzu. Ein Beispiel für einen

Pfad kann z. B. wie folgt lauten: `software/http/cics/index.html`. In The components of a URL finden Sie Erläuterungen zu den einzelnen Komponenten und wie diese mit Begrenzern versehen werden können.

Das Attribut PATH wird in gemischter Groß-/Kleinschreibung angegeben. Die Groß-/Kleinschreibung wird in der URIMAP-Definition beibehalten. Das Attribut PATH darf nur die Zeichen enthalten, die in URIs zulässig sind. Insbesondere die Zeichen `< > # % " { } | \ ^ [ ] `` sowie eingebettete Leerzeichen müssen ausgeschlossen werden. Eine Ausnahme bildet hier lediglich das Prozentzeichen (`%`), das zulässig ist, wenn es zur Einleitung einer gültigen hexadezimalen Escapezeichenfolge dient (d. h., wenn auf das Zeichen zwei gültige hexadezimale Ziffern in Groß- oder Kleinbuchstaben folgen). Das Tildezeichen (`~`) darf in CICS nicht angegeben werden und muss durch die entsprechende hexadezimale Escapezeichenfolge (`%7E`) ersetzt werden. CICS überprüft die Verwendung der Zeichen während der Definition.

URIMAP-Ressourcen unterstützen IRIs (Internationalized Resource Identifiers), die Unicode-Zeichen enthalten können. Wenn Sie einen Pfad angeben, der Unicode-Zeichen enthält, dann müssen diese Unicode-Zeichen in der prozentcodierten Darstellung als Escapezeichenfolge angegeben werden. Wenn Sie nicht über eine Anwendung verfügen, mit der Unicode-Zeichen in die prozentcodierte Darstellung konvertiert werden können, dann können Sie die im Internet angebotenen kostenlosen Anwendungen benutzen, um diese Aufgabe auszuführen. Der Pfad darf weiterhin maximal 255 Zeichen umfassen und ein Zeichen bedeutet in diesem Kontext ein einzelnes ASCII-Zeichen und nicht das ursprüngliche Unicode-Zeichen. Das kyrillische Zeichen mit der prozentcodierten Darstellung `%D0%B4` umfasst beispielsweise sechs Zeichen, die auf die Begrenzung von 255 Zeichen anzurechnen sind.

Für URIMAP-Definitionen, die sich auf CICS als HTTP-Server und Web-Services beziehen, müssen Sie, wenn die URIMAP-Definition mit mehreren Pfaden übereinstimmen soll, am Ende des Pfads einen Stern als Platzhalterzeichen verwenden. Beispiel: Der Pfad `/software/http/cics/*` führt dazu, dass die URIMAP-Definition mit allen Anforderungen übereinstimmt, deren Pfadangabe mit der Zeichenfolge links vom Stern beginnt. Die Angabe des Pfads `/*` führt dazu, dass die URIMAP-Definition mit allen Anforderungen übereinstimmt, die an den Host weitergeleitet werden, der im Attribut HOST aufgeführt ist. Wenn eine HTTP-Anforderung mit mehr als einer URIMAP-Definition übereinstimmt, wird die Definition mit dem höchsten Übereinstimmungsgrad verwendet.

Wenn eine Abfragekomponente vorhanden ist und wenn Sie die URIMAP-Definition ausschließlich auf diese spezielle Abfrage anwenden wollen, dann können Sie sie in die Pfadkomponente aufnehmen. Schließen Sie das Fragezeichen (`?`) am Anfang der Zeichenfolge ein. Die Abfragezeichenfolge darf nur die Zeichen enthalten, die in URIs zulässig sind. Eine Abfragezeichenfolge darf selbst keinen Stern (`*`) als Platzhalterzeichen enthalten, kann jedoch einem Pfad folgen, der einen Stern als Platzhalterzeichen umfasst. Wenn Sie in die URIMAP-Definition keine Abfragezeichenfolge einschließen, dann werden alle in der HTTP-Anforderung enthaltenen Abfragezeichenfolgen beim Abgleich automatisch ignoriert.

Für URIMAP-Definitionen für Atom-Feeds müssen Sie einen Stern als Platzhalterzeichen am Ende des Pfads verwenden. Der Teil des Pfads, den Sie in der URIMAP-Definition angeben, ist der Teil, der für die Atom-Feed- und die Atom-Eintrags-URLs einheitlich ist. CICS gleicht den Rest der URL mit den URLs ab, die in allen `<atom:verbindung>`-Elementen in der Atom-Konfigurationsdatei für den Feed angegeben werden.

Für URIMAP-Definitionen für die Verwendung von CICS als HTTP-Client können Sie keinen Stern als Platzhalterzeichen verwenden. Sie müssen in diesem Fall für die Anforderung den vollständigen Pfad angeben. Wenn in einem Befehl **WEB OPEN** auf die URIMAP-Definition verwiesen wird, dann wird dieser Pfad als Standardpfad für **WEB SEND**-Befehle festgelegt, die sich auf diese Verbindung beziehen. Wenn in einem Befehl **WEB SEND** auf die URIMAP-Definition verwiesen wird, dann wird der Pfad für diesen Befehl **WEB SEND** verwendet. Allerdings muss das Hostattribut für diese URIMAP-Definition mit dem im Befehl **WEB OPEN** für die Verbindung angegebenen Host übereinstimmen.

#### **PIPELINE**(*name*)

Dieses Attribut gilt für **USAGE(PIPELINE)**.

Wenn ein Client eine Anforderung für einen eingehenden Web-Service an CICS absetzt und wenn dabei der URI in dieser URIMAP-Definition angegeben wird, dann gibt das Attribut **PIPELINE** den aus einem bis acht Zeichen bestehenden Namen der **PIPELINE**-Ressourcendefinition des Web-Service an. Die **PIPELINE**-Ressourcendefinition stellt Informationen zu den Nachrichtenhandlern bereit, die die Serviceanforderung des Clients verarbeiten. Diese Ressourcendefinitionen werden in **PIPELINE resources** beschrieben.

##### **Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl **CREATE** verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

#### **PORT**(**{NO|***port***}**)

Dieses Attribut gilt nur für die Optionen **USAGE(CLIENT)** und **USAGE(JVM-SERVER)**.

Für **USAGE(CLIENT)** gibt das Attribut **PORT** die Dezimalzahl des Ports an, der von einer CICS-Anwendung für die Kommunikation mit einem Server benutzt wird. Der Wert muss eine Zahl im Bereich zwischen 1 und 65535 sein.

Die Portnummer wird mit dem **HOST**-Wert kombiniert, um das Ziel für abgehende Anforderungen für diese URIMAP-Definition zu ermitteln. Geben Sie die Portnummer nur an, wenn sie sich vom entsprechenden Standardwert für das Schema unterscheidet (80 für HTTP ohne SSL oder 443 für HTTPS und HTTP mit SSL).

Wenn Sie eine Portnummer im Attribut **HOST** und eine andere Portnummer im Attribut **PORT** angeben, dann wird ein Fehler zurückgegeben. Wenn Sie weder im Attribut **HOST** noch im Attribut **PORT** eine Portnummer angeben, dann wird die Standardportnummer des jeweiligen Schemas verwendet.

Für die Option **USAGE(JVMSEVER)** gibt das Attribut **PORT** die Portnummer an, die zum Empfangen von Anforderungen für den Zugriff auf eine Anwendung benutzt wird, die auf einem Liberty-Profilserver ausgeführt wird.

Wenn Sie für das Attribut **PORT** keinen Wert angeben, dann wird für **PORT** die Einstellung **NO** ausgewählt, um anzugeben, dass das Attribut nicht verwendet wird.

#### **PROGRAM**(*name*)

Dieses Attribut gilt für **USAGE(SERVER)**, wenn eine anwendungsgenerierte Antwort bereitgestellt werden muss.

In **PROGRAM** ist der aus einem bis acht Zeichen bestehende Name des Benutzeranwendungsprogramms angegeben, das die HTTP-Antwort erstellt. Bei Ver-

wendung von CICS als HTTP-Server ist dieses Attribut erforderlich, es sei denn, ein Analyse- oder Konvertierungsprogramm, ein Schablonenname, eine zFS-Datei oder aber eine Umleitung wird angegeben.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Wenn für das Attribut ANALYZER die Option YES angegeben ist oder wenn im Attribut CONVERTER ein Konvertierungsprogramm angegeben ist, dann wird das Attribut PROGRAM als Eingabe für das Analyse- oder Konvertierungsprogramm verwendet, kann jedoch von diesen Programmen überschrieben werden. Alternativ hierzu können Sie für dieses Attribut auch keinen Wert angeben und ihn durch das Analyse- oder Konvertierungsprogramm festlegen lassen.

**REDIRECTTYPE({NONE|TEMPORARY|PERMANENT})**

Dieses Attribut gilt für USAGE(SERVER), USAGE(PIPELINE) und USAGE(ATOM).

Im Attribut REDIRECTTYPE wird der Typ der Umleitung für Anforderungen angegeben, die mit dieser URIMAP-Definition übereinstimmen. Die im Attribut LOCATION angegebene URL wird bei Bedarf für die Umleitung verwendet.

- NONE bedeutet, dass Anforderungen nicht umgeleitet werden. Eine im Attribut LOCATION angegebene URL wird ignoriert.
- TEMPORARY bedeutet, dass Anforderungen vorübergehend umgeleitet werden. Die im Attribut LOCATION angegebene URL wird für die Umleitung verwendet und der für die Antwort benutzte Statuscode ist 302 (Found).
- PERMANENT bedeutet, dass Anforderungen dauerhaft umgeleitet werden. Die im Attribut LOCATION angegebene URL wird für die Umleitung verwendet und der für die Antwort benutzte Statuscode ist 301 (Moved Permanently).

Mit dem Befehl **SET URIMAP** können Sie die Werte im Attribut REDIRECTTYPE und im Attribut LOCATION ändern. Wenn bei der Erstellung einer URIMAP-Definition REDIRECTTYPE(TEMPORARY) oder REDIRECTTYPE(PERMANENT) angegeben wird, dann sind die folgenden Attribute optional: ANALYZER, CONVERTER, HFSFILE, PIPELINE, PROGRAM, TEMPLATENAME, TRANSACTION, USERID und WEBSERVICE. Wenn Sie einen Befehl CEMT oder EXEC CICS verwenden, um das Attribut REDIRECTTYPE auf die Einstellung NONE zu setzen, nachdem die URIMAP-Definition installiert wurde, dann werden alle aufgelisteten Attribute, die in der URIMAP-Definition angegeben sind, aktiviert.

**SCHEME({HTTP|HTTPS})**

Dieses Attribut gilt für alle USAGE-Optionen.

Das Attribut SCHEME gibt die Schemakomponente des URI an, für den die URIMAP-Definition gilt. Hierbei kann es sich entweder um HTTP (ohne SSL) oder HTTPS (mit SSL) handeln. Schließen Sie die Begrenzungszeichen :// (Doppelpunkt und zwei Schrägstriche), die im URI auf die Schemakomponente folgen, nicht mit ein.

Eine URIMAP-Definition, in der das HTTP-Schema angegeben ist, akzeptiert Web-Client-Anforderungen, die mit dem HTTP- oder dem HTTPS-Schema ausgeführt wurden. Eine URIMAP-Definition, in der das HTTPS-Schema festgelegt



ist, akzeptiert nur Web-Client-Anforderungen, die mithilfe des HTTPS-Schemas ausgeführt werden. Wenn für die Übertragung allerdings WebSphere MQ verwendet wird, dann akzeptiert eine URIMAP-Definition, in der entweder das HTTP- oder das HTTPS-Schema angegeben ist, Web-Client-Anforderungen, die mithilfe des HTTP- oder des HTTPS-Schemas ausgeführt werden.

Eingehende HTTP-Anforderungen, die in TCPIPService-Elementen mit SSL(ATTLSAWARE) empfangen werden, müssen sichere Verbindungen verwenden, damit sie stets als HTTPS behandelt werden.

#### **SOCKETCLOSE({0|*hhmmss*})**

Dieses Attribut gilt für USAGE(CLIENT).

Im Attribut SOCKETCLOSE wird angegeben, ob und für wie lange CICS eine HTTP-Clientverbindung geöffnet hält, nachdem die CICS-Anwendung ihre Verwendung eingestellt hat. Nach Abschluss der Verwendung überprüft CICS den Status der Verbindung und platziert sie im Ruhezustand in einem entsprechenden Pool. Eine ruhende Verbindung kann von derselben oder einer anderen Anwendung, die eine Verbindung zu demselben Host und Port herstellt, wiederverwendet werden.

**0** CICS schließt jede HTTP-Clientverbindung, wenn die CICS-Anwendung sie nicht mehr benutzt. CICS platziert die Verbindung nicht in einem Pool, damit sie wiederverwendet werden kann.

#### ***hhmmss***

Nachdem eine CICS-Anwendung die Nutzung der zugehörigen HTTP-Clientanwendung beendet hat, überprüft CICS den Status der Verbindung und platziert sie zur Wiederverwendung in einem Pool. Eine ruhende Verbindung, die nicht wiederverwendet wird, wird nach Ablauf der von Ihnen hier angegebenen Zeitdauer gelöscht.

Das Verbindungspooling bietet Leistungsvorteile für den HTTP-Ereignisverarbeitungsadapter bei der CICS-Ereignisverarbeitung oder wenn mehrere Aufrufe für CICS-Webunterstützungsanwendungen Verbindungsanforderungen für denselben Host und Port ausführen oder wenn Web-Service-Anwendungen mehrere Anforderungen und Antworten ausführen. Zum Aktivieren des Verbindungspoolings muss in Ihren Anwendungsprogrammen im Befehl **INVOKE SERVICE** oder **WEB OPEN** die URIMAP-Ressource angegeben werden. Weitere Informationen zum Verbindungspooling finden Sie in Connection pooling for HTTP client performance.

#### **STATUS({ENABLED|DISABLED})**

Dieses Attribut gilt für alle USAGE-Optionen.

Das Attribut STATUS gibt an, ob die URIMAP-Definition im aktivierten oder inaktivierten Status installiert werden soll. Der Standardwert ist der aktivierte Status.

Dieses Attribut wird für URIMAP-Ressourcen ignoriert, die dynamisch von einem CICS-Bundle generiert werden. Der Anfangsstatus einer URIMAP-Ressource wird aus dem Anfangsstatus des Bundles abgeleitet, das die Ressource definiert.

#### **TCPIPService(*name*)**

Dieses Attribut gilt für USAGE(SERVER), USAGE(PIPELINE) und USAGE(ATOM).

Im Attribut TCPIPService wird der aus einem bis acht Zeichen bestehende Name einer TCPIPService-Ressourcendefinition mit PROTOCOL(HTTP) angegeben, in der ein eingehender Port definiert wird, dem diese URIMAP-Defi-

nition zugeordnet ist. Wird dieses Attribut nicht angegeben, dann gilt die URIMAP-Definition für eine Anforderung an allen Ports für eingehende Daten.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

Wenn eine URIMAP-Definition mit dem HTTPS-Schema mit einer Anforderung übereinstimmt, die von einem Web-Client ausgeführt wird, dann überprüft CICS, ob der Port für eingehende Daten, der von der Anforderung verwendet wird, mit SSL arbeitet. Wenn für den Port nicht SSL angegeben ist, wird die Anforderung mit dem Statuscode 403 (Forbidden) zurückgewiesen. Wenn die URIMAP-Definition für alle Ports für eingehende Daten gilt, dann wird mit dieser Überprüfung sichergestellt, dass ein Web-Client keinen nicht sicheren Port für den Zugriff auf eine gesicherte Ressource verwenden kann. Für URIMAP-Definitionen, die mit dem HTTP-Schema arbeiten, wird keine Überprüfung durchgeführt. Die Web-Clients können somit entweder nicht gesicherte oder gesicherte Ports (SSL) für den Zugriff auf diese Ressourcen verwenden.

Sie geben die Sicherheitsmaßnahmen an, die für jeden Port in der TCP/IPSERVER-Ressourcendefinition angewendet werden sollen. Sie können auswählen, ob SSL verwendet werden soll. Wird SSL verwendet, dann müssen Sie die exakten Sicherheitsmaßnahmen auswählen, die angewendet werden sollen, z. B. die Authentifizierungsmethode, das Senden von Zertifikaten von Client und Server sowie die Nachrichtenverschlüsselung. Weitere Informationen zu den Sicherheitsfunktionen, die Sie zum Schutz Ihrer CICS-Webunterstützungsfunktion verwenden können, finden Sie in Security for CICS web support.

**TEMPLATENAME** (*name*)

Dieses Attribut gilt für USAGE(SERVER), wenn eine statische Antwort bereitgestellt werden muss.

Das Attribut TEMPLATENAME gibt den aus 1 bis 48 Zeichen bestehenden Namen einer CICS-Dokumentschablone an, die den Hauptteil der statischen Antwort bildet, die vom Web-Client für die HTTP-Anforderung gesendet wird. Er muss mithilfe einer DOCTEMPLATE-Ressourcendefinition definiert werden und das in dieser Definition festgelegte Attribut TEMPLATENAME gibt den Namen an, der in der URIMAP-Definition verwendet wird. Weitere Informationen hierzu finden Sie in CICS documents and document templates.

**Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Wenn Sie das Attribut TEMPLATENAME oder HFSFILE angeben, dann müssen Sie für das Attribut ANALYZER die Einstellung NO festlegen. Die anderen Attribute, die sich nur auf anwendungsgenerierte Antworten beziehen (TRANSACTION, CONVERTER und PROGRAM), dürfen nicht angegeben werden.

Wenn Sie einen Pfadabgleich durchführen wollen, dann geben Sie einen Stern (\*) als Platzhalterzeichen am Ende des Namens der CICS-Dokumentschablone und außerdem auch am Ende des Pfads an, der im Attribut PATH festgelegt wurde. CICS verwendet den Pfadteil einer HTTP-Anforderung, der vom Platzhalterzeichen abgedeckt wird, und setzt diesen Teil als letzten Teil des Schablonennamens ein.

Sie können beispielsweise eine URIMAP-Definition mit der folgenden Angabe für das Attribut PATH erstellen:

```
findout/about/*
```

Des Weiteren können Sie das Attribut TEMPLATENAME wie folgt angeben:

```
templates.facts.*
```

Die URIMAP-Definition wird zur Verarbeitung eingehender HTTP-Anforderungen verwendet:

```
http://www.example.com/findout/about/fish.html
```

Anstelle des Sterns (\*) fügt CICS fish.html zu dem Schablonennamen hinzu, der in der URIMAP-Definition angegeben wurde, sodass die Schablone

```
templates.facts.fish.html
```

zur Erstellung der statischen Antwort benutzt wird.

Wenn Sie für das Attribut TEMPLATENAME nur einen Stern (\*) angeben, dann bedeutet dies, dass die ausgewählte Schablone denselben Namen wie der Teil der URL hat, der dem Platzhalterzeichen im Attribut PATH entspricht.

Wenn Sie das Attribut TEMPLATENAME angeben und der URI eine Abfragezeichenfolge enthält, die jedoch im Attribut PATH nicht verwendet wird, dann übergibt CICS den Inhalt der Abfragezeichenfolge automatisch als Symbolliste an die angegebene CICS-Dokumentschablone. Wenn Sie den Inhalt der Abfragezeichenfolge in der Dokumentschablone verwenden wollen, dann nehmen Sie geeignete Variablen in die Dokumentschablone auf, die durch den Inhalt der Abfragezeichenfolge ersetzt werden können.

#### **TRANSACTION**(*name*)

Dieses Attribut gilt für USAGE(SERVER), USAGE(PIPELINE), USAGE(ATOM) und USAGE(JVMSEVER).

Im Attribut TRANSACTION wird der aus einem bis vier Zeichen bestehende Name einer Aliastransaktion angegeben, die zur Ausführung der Anwendung oder zum Starten der Pipeline benutzt werden soll.

##### **Zulässige Zeichen:**

```
A-Z a-z 0-9 $ @ # . / - _ % & ? ! : | " = ~ , ; < >
```

Die Standardaliasstransaktion ist für jeden URIMAP-Typ unterschiedlich:

- USAGE(SERVER) verwendet die CWBA-Transaktion.
- USAGE(PIPELINE) verwendet die CPIH-Transaktion.
- USAGE(ATOM) verwendet die CW2A-Transaktion.
- USAGE(JVMSEVER) verwendet die CJSa-Transaktion.

Zu Sicherheits-, Überwachungs- und Abrechnungszwecken sowie zur Begrenzung von Transaktionsklassen können Sie einen anderen Transaktionsnamen auswählen. Der für die Aliastransaktion ausgewählte Name muss allerdings immer zur Ausführung desselben Programms dienen, das im Attribut USAGE festgelegt wird und das jedes andere Programm überschreibt, das in der Transaktionsdefinition angegeben ist:

- Für USAGE(SERVER) lautet das Programm DFHWBA. Es stellt eine Verbindung zu dem Anwendungsprogramm her, das im Attribut PROGRAM der URIMAP-Definition oder im Analyseprogramm angegeben wird.

- Für USAGE(PIPELINE) lautet das Programm DFHPIDSH. Es dient zum Starten der Pipeline, die im Attribut PIPELINE angegeben ist, und zum Starten des Web-Service, der im Attribut WEBSERVICE (sofern angegeben) definiert ist.
- Für USAGE(ATOM) lautet das Programm DFHW2A. Hierbei handelt es sich um das W2-Domänenaliasprogramm.
- Für USAGE(JVMSEVER) lautet das Programm DFHSJTHP. Es dient zur Ausführung von Tasks für eine Anwendungsanforderung auf dem JVM-Server.

Nur für USAGE(SERVER): Wenn für das Attribut ANALYZER die Einstellung YES angegeben wird, dann wird der Wert im Attribut TRANSACTION als Eingabe für das Analyseprogramm verwendet. Dieser Wert kann vom Analyseprogramm jedoch überschrieben werden. Alternativ hierzu können Sie für dieses Attribut auch keinen Wert angeben und ihn durch das Analyseprogramm festlegen lassen. Das Analyseprogramm wird für andere URIMAP-Typen nicht verwendet.

#### **URIMAP(*name*)**

Gibt den Namen dieser URIMAP-Definition an. Der Name kann bis zu acht Zeichen lang sein. Das Attribut wird in gemischter Groß-/Kleinschreibung angegeben. Die Groß-/Kleinschreibung wird in der URIMAP-Definition beibehalten.

##### **Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

#### **USAGE({SERVER|CLIENT|PIPELINE|ATOM|JVMSEVER})**

Gibt an, ob diese URIMAP-Definition für CICS als HTTP-Server (SERVER), CICS als HTTP-Client (CLIENT), einen Web-Service (PIPELINE), einen Atom-Feed (ATOM) oder einen JVM-Server (JVMSEVER) vorgesehen ist. Das Attribut USAGE regelt, welche anderen Attribute in der URIMAP-Definition verwendet werden können.

- Wenn Sie SERVER angeben, dann erstellen Sie eine URIMAP-Definition für CICS als HTTP-Server. Dieser Typ einer URIMAP-Definition wird verwendet, um den URI einer eingehenden HTTP-Anforderung von einem Web-Client den entsprechenden CICS-Ressourcen zuzuordnen. Vom System kann eine anwendungsgenerierte oder eine statische Antwort bereitgestellt werden. Anforderungen, für die eine anwendungsgenerierte Antwort erforderlich ist, kommen möglicherweise für die Verarbeitung durch direkt angehängte Benutzertransaktionen in Betracht, wobei die Webanhängetask umgangen wird. Weitere Informationen hierzu finden Sie in HTTP requests are processed by directly attached user transactions.
- Wenn Sie CLIENT angeben, dann erstellen Sie eine URIMAP-Definition für CICS als HTTP-Client. Dieser Typ der URIMAP-Definition wird verwendet, wenn CICS eine Anforderung für eine HTTP-Ressource auf einem Server ausführt, damit Sie die Angabe des URI in Ihrem Anwendungsprogramm umgehen können.
- Wenn Sie PIPELINE angeben, dann erstellen Sie eine URIMAP-Definition für einen Web-Service. Dieser Typ von URIMAP-Definition wird für eine Anforderung für einen eingehenden Web-Service verwendet. Hierbei handelt es sich um eine Anforderung, mit der ein Client einen Web-Service in CICS aufruft. Der URI der eingehenden Anforderung wird den WEBSERVICE- und PIPELINE-Ressourcen zugeordnet, in denen die Art der Verarbeitung angegeben wird, die für die Nachricht ausgeführt werden soll. Anforderungen

kommen möglicherweise für die Verarbeitung durch direkt angehängte Benutzertransaktionen in Betracht, wobei die Webanhängetask umgangen wird. Weitere Informationen hierzu finden Sie in HTTP requests are processed by directly attached user transactions.

- Wenn Sie ATOM angeben, dann erstellen Sie eine URIMAP-Definition für einen Atom-Feed. Dieser Typ einer URIMAP-Definition wird für eine eingehende Anforderung von Daten verwendet, die von CICS als Atom-Feed bereitgestellt werden. Die URIMAP-Definition ordnet den Anforderungs-URI einer ATOMSERVICE-Ressourcendefinition zu, die ein Atom-Dokument definiert. Anforderungen kommen möglicherweise für die Verarbeitung durch direkt angehängte Benutzertransaktionen in Betracht, wobei die Webanhängetask umgangen wird. Weitere Informationen hierzu finden Sie in HTTP requests are processed by directly attached user transactions.
- Wenn Sie JVMSERVER angeben, dann erstellen Sie eine URIMAP-Definition für eine Webanwendung, die auf einem JVM-Server ausgeführt wird. Dieser URIMAP-Typ wird für die Zuordnung eingehender Anwendungsanforderungen zu einer CICS-Transaktion verwendet. Wenn keine der URIMAPs mit dem URI der eingehenden Anforderung übereinstimmt, dann verwendet CICS die CJS-Transaktion.

#### **USERID(*id*)**

Dieses Attribut gilt für die Einstellung USAGE(SERVER), bei der eine anwendungsgenerierte Antwort bereitgestellt werden muss, sowie für USAGE(PIPELINE), USAGE(JVMSERVER) und USAGE(ATOM).

Das Attribut USERID gibt die aus einem bis acht Zeichen bestehende Standard-Benutzer-ID an, die von jedem Client verwendet werden kann. Für eine anwendungsgenerierte Antwort oder einen Web-Service wird die Aliastransaktion unter dieser Benutzer-ID zugeordnet.

Für USAGE(JVMSERVER) wird die Benutzer-ID zur Zuordnung einer Transaktion für die Ausführung von Arbeiten auf dem Liberty-Profilserver verwendet. Die Transaktion kann in der URI-Maske angegeben werden. Alternativ hierzu kann auch die standardmäßige CJS-Transaktion verwendet werden. Die Benutzer-ID wird nicht verwendet, wenn die Basisauthentifizierung aktiviert ist. CICS erwartet eine Benutzer-ID und ein Kennwort im HTTP-Header für die Anwendungsanforderung und weist die Anforderung zurück, wenn die Authentifizierung fehlschlägt. Wenn die Sicherheitsfunktion nicht aktiviert wurde und keine Benutzer-ID im HTTP-Header oder in der URI-Maske vorhanden ist, dann wird die Standard-Benutzer-ID für die CICS-Region verwendet.

Für andere Verwendungstypen, bei denen zur Herstellung der Verbindung eine Authentifizierung erforderlich ist, sodass CICS eine authentifizierte Benutzer-ID direkt vom Client anfordert, wird die Standard-Benutzer-ID, die Sie in der URIMAP-Definition angeben, nicht verwendet. Stattdessen wird die authentifizierte Benutzer-ID des Clients verwendet. Schlägt die Authentifizierung fehl, dann wird die Anforderung zurückgewiesen. Die Authentifizierungsprozeduren werden im Attribut AUTHENTICATE der TCPIPSERVICE-Definition für die Verbindung angegeben.

Wenn für eine anwendungsgenerierte Antwort ANALYZER(YES) angegeben wird, dann wird der Wert im Attribut USERID als Eingabe für das Analyseprogramm verwendet. Dieser Wert kann vom Analyseprogramm jedoch überschrieben werden. Alternativ hierzu können Sie für dieses Attribut auch keinen Wert angeben und ihn durch das Analyseprogramm festlegen lassen. Das Analyseprogramm wird nur verwendet, wenn USAGE(SERVER) angegeben wurde. Eine von einem Client angegebene Benutzer-ID kann vom Analyseprogramm ebenfalls geändert werden. Wenn für keine der hier dargestellten Methoden

eine Benutzer-ID angegeben wird, dann wird als Standard-Benutzer-ID für eine anwendungsgenerierte Antwort die Standard-Benutzer-ID von CICS verwendet.

Für statische Antworten gilt das Attribut USERID nicht. Die Ressourcensicherheitsprüfung für statische Antworten kann mithilfe der authentifizierten Benutzer-ID eines Clients ausgeführt werden.

Wenn in der CICS-Region die Ersatzbenutzerprüfung aktiviert wurde und als Systeminitialisierungsparameter XUSER=YES angegeben wurde, dann überprüft CICS, ob die zum Installieren der URIMAP-Definition verwendete Benutzer-ID als Ersatz für die Benutzer-ID berechtigt ist, die für das Attribut USERID angegeben wurde. Weitere Informationen zur Ersatzbenutzerprüfung finden Sie in Where surrogate user checking applies.

#### **WEBSERVICE**(*name*)

Dieses Attribut gilt für USAGE(PIPELINE).

Wenn ein Client eine Anforderung für einen eingehenden Web-Service an CICS absetzt und wenn dabei der URI in dieser URIMAP-Definition angegeben wird, dann gibt das Attribut WEBSERVICE den Namen des Web-Service an. Als Name kann der zwischen einem und acht Zeichen umfassende Name einer WEBSERVICE-Ressourcendefinition angegeben werden. Alternativ kann auch ein Name mit maximal 32 Zeichen in gemischter Groß-/Kleinschreibung angegeben werden, der für einen Web-Service steht, der vom CICS-Web-Service-Assistenten generiert wird.

##### **Zulässige Zeichen:**

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Eine WEBSERVICE-Ressource definiert die Aspekte der Laufzeitumgebung für ein CICS-Anwendungsprogramm, das in einer Web-Service-Einstellung bereitgestellt wird, bei der eine Zuordnung zwischen der Anwendungsdatenstruktur und den SOAP-Nachrichten mit den CICS-Tools generiert wird.

Dieses Attribut ist optional. Wenn Sie für USAGE(PIPELINE) keine WEBSERVICE-Ressource angeben, dann muss der WEBSERVICE-Name allerdings von einem Handlerprogramm in der Pipeline aufgelöst werden oder Sie müssen anstelle einer WEBSERVICE-Ressource ein alternatives Anwendungshandlerprogramm oder Terminal-Handlerprogramm verwenden.

---

## **WEBSERVICE-Ressourcen**

Eine WEBSERVICE-Ressource definiert die Aspekte der Laufzeitumgebung für ein CICS-Anwendungsprogramm, das in einer Web-Service-Einstellung bereitgestellt wird, bei der die Zuordnung zwischen der Anwendungsdatenstruktur und den SOAP-Nachrichten mit dem CICS-Web-Service-Assistenten generiert wird. Obwohl CICS die üblichen Ressourcendefinitionsmechanismen für WEBSERVICE-Ressourcen zur Verfügung stellt, werden sie normalerweise dynamisch installiert, wobei die vom Assistenten generierte Ausgabe verwendet wird.

Von der WEBSERVICE-Ressource werden die folgenden Aspekte der Laufzeitumgebung definiert:

#### **Pipeline**

Definiert die Gruppe der Nachrichtenhandler, die für Web-Service-Anforderungen und -Antworten verwendet werden. Die WEBSERVICE-Ressource gibt eine separate PIPELINE-Ressource an, die ihrerseits die Pipelinekonfigurationsdatei angibt.

### Datei für Web-Service-Bindungen

Diese Datei enthält Informationen, die während der Laufzeit zur Ausführung der Zuordnung zwischen Anwendungsdatenstrukturen und SOAP-Nachrichten verwendet wird. Die Datei für Web-Service-Bindungen wird mit den von CICS bereitgestellten Tools generiert.

### Web-Service-Beschreibung

Die Web-Service-Beschreibung wird nur dann verwendet, wenn die Laufzeitvalidierung von SOAP-Nachrichten erforderlich ist. Die Validierung der einzelnen Nachrichten wird auf Basis des zugehörigen Schemas durchgeführt, das in die Web-Service-Beschreibung eingebettet ist.

Eine eingehende Web-Service-Anforderung (Anforderung, mit der der Client einen Web-Service in CICS aufruft) wird einer WEBSERVICE-Ressource mithilfe der URIMAP-Ressource zugeordnet. Die URIMAP-Ressource identifiziert die WEBSERVICE-Ressource, die für den URI in der eingehenden Nachricht gilt. Die WEBSERVICE-Ressource gibt die Verarbeitung an, die für die Nachricht ausgeführt werden soll.

Obwohl von CICS die üblichen Ressourcendefinitionsmechanismen für die Erstellung von WEBSERVICE-Ressourcen und für die Installation dieser Ressourcen in Ihrer CICS-Region bereitgestellt werden, können Sie stattdessen den Suchmechanismus verwenden, um die WEBSERVICE-Ressourcen dynamisch auf dem aktiven CICS-System zu installieren. Die Vorteile, die dieser Ansatz bietet, bestehen in der Reduzierung des Aufwands für die Ressourcendefinition und darin, dass CICS die bei der Entwicklung bereitgestellten Informationen direkt nutzen kann.

Zum Aufrufen des Suchmechanismus verwenden Sie den Befehl **PERFORM PIPELINE**.

Der Name einer dynamisch installierten WEBSERVICE-Ressource wird vom Namen der Datei für Web-Service-Bindungen abgeleitet, auf deren Basis die WEBSERVICE-Definition generiert wird. Er hat eine maximale Länge von 32 Zeichen. Die Namen der über die CSD oder mit dem Befehl **EXEC CICS CREATE WEBSERVICE** installierten WEBSERVICE-Definitionen sind auf acht Zeichen begrenzt. Beispiel: Eine Datei für Web-Service-Bindungen, deren zFS-Name `/samples/webservices/WSDir/InquireSingle.wsbind` lautet, generiert eine WEBSERVICE-Definition mit dem Namen `InquireSingle`.

Informationen zur Installation von Web-Service-Ressourcendefinitionen mit CEMT und CEDA finden Sie in *Installing WEBSERVICE resource definitions*. Informationen zum Arbeiten mit BAS (Business Application Services) finden Sie in *Working with BAS web service resource definitions*.

## WEBSERVICE-Ressourcen in CICS-Bundles

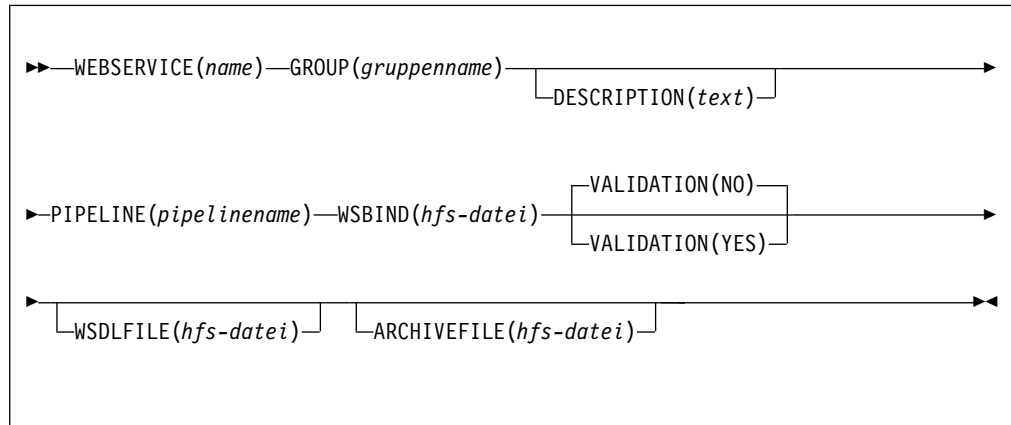
Sie können ein CICS-Bundle verwenden, um eine WEBSERVICE-Ressourcendefinition zu erstellen, zu bearbeiten und zu installieren. Wenn Sie eine WEBSERVICE-Ressource auf diese Weise erstellen, dann müssen Sie zum Verwalten des Lebenszyklus dieser Ressource das CICS-Bundle verwenden. Die unabhängige Verwaltung der Ressource ist nicht möglich.

Sie können eine WEBSERVICE-Ressource abfragen, die von einem CICS-Bundle dynamisch generiert wird. Verwenden Sie hierzu den Befehl **EXEC CICS INQUIRE WEBSERVICE** oder **CEMT INQUIRE WEBSERVICE**. Allerdings kann ein Befehl **DISCARD** nicht für eine dynamisch generierte WEBSERVICE-Ressource verwendet werden. Sie müssen den Befehl für die BUNDLE-Ressource absetzen und CICS wendet ihn

dann auf die WEBSERVICE-Ressource an. Weitere Informationen zum Definieren von Ressourcen in CICS-Bundles finden Sie in Defining CICS bundles.

## WEBSERVICE-Attribute

Beschreibt die Syntax und die Attribute der WEBSERVICE-Ressource.



### WEBSERVICE(name)

Gibt den aus einem bis acht Zeichen bestehenden Namen der WEBSERVICE-Ressource an.

#### Zulässige Zeichen:

A-Z a-z 0-9 \$ @ # . / - \_ % & ? ! : | " = ~ , ; < >

Sie sollten keine Namen verwenden, die mit der Zeichenfolge DFH beginnen, weil diese Zeichen zur Verwendung durch CICS reserviert sind. CICS kann sie löschen oder verwerfen und durch CICS-Ressourcen ersetzen, wenn Sie ein Upgrade durchführen oder neue Funktionen oder Fixes bereitstellen.

### ARCHIVEFILE(hfs-datei)

Gibt den aus 1 - 255 Zeichen bestehenden, vollständig qualifizierten oder relativen Namen eines Archivs an, das mindestens eine WSDL-Datei enthält. Das unterstützte Format für das Archiv ist .zip.

- Für ein Archiv einer WEBSERVICE-Ressource, die mithilfe der Onlineresourcendefinition definiert wurde, wird entweder ein vollständig qualifizierter Dateipfad (mit einem Schrägstrich (/) als Präfix) oder ein Dateipfad relativ zum Ausgangsverzeichnis der Benutzer-ID für die CICS-Region angegeben.
- Für ein Archiv für eine WEBSERVICE-Ressource, die in einem CICS-Bundle definiert ist, wird der Dateipfad relativ zum Stammverzeichnis des CICS-Bundles angegeben. Die zFS-Datei muss im CICS-Bundle mit der WEBSERVICE-Ressource gepackt werden. Weitere Informationen hierzu finden Sie in Referencing zFS artifacts in a bundle.

Der angegebene Wert muss ein gültiger Name einer UNIX-Datei sein:

- Er darf keine eingebetteten Leerzeichen enthalten.
- Er darf keine aufeinanderfolgenden Vorkommen des Zeichens / enthalten.
- Die Groß-/Kleinschreibung muss beachtet werden.

#### Zulässige Zeichen:

A-Z a-z 0-9 . / \_ # @ -



**GROUP**(*gruppenname*)

Jede Ressourcendefinition muss über einen GROUP-Namen verfügen. Die Ressourcendefinition wird Mitglied der Gruppe und auf dem CICS-System installiert, wenn die Gruppe installiert wird.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Alle Kleinbuchstaben, die Sie eingeben, werden in Großbuchstaben konvertiert.

Der GROUP-Name kann bis zu acht Zeichen lang sein. Kleinbuchstaben werden als Großbuchstaben behandelt.

**DESCRIPTION**(*text*)

Sie können eine Beschreibung für die Ressource angeben, die Sie in diesem Feld definieren. Der Beschreibungstext kann bis zu 58 Zeichen lang sein. Für die zu verwendenden Zeichen bestehen keine Beschränkungen. Wenn Sie jedoch Klammern verwenden, dann müssen Sie sicherstellen, dass für die jeweils linke Klammer auch immer eine zugehörige rechte Klammer vorhanden ist. Wenn Sie den Befehl CREATE verwenden, dann müssen Sie für jedes einzelne Hochkomma im Text zwei Hochkommas angeben.

**PIPELINE**(*pipelinename*)

Gibt den aus einem bis acht Zeichen bestehenden Namen der PIPELINE-Ressource an, der diese WEBSERVICE-Ressource zugeordnet ist.

**Zulässige Zeichen:**

A-Z 0-9 \$ @ #

Sofern Sie nicht den Befehl CREATE verwenden, werden alle Kleinbuchstaben, die Sie eingeben, in Großbuchstaben konvertiert.

**VALIDATION**(NO|YES)

Gibt an, ob die vollständige Validierung von SOAP-Nachrichten auf Basis des entsprechenden Schemas in der Web-Service-Beschreibung während der Laufzeit ausgeführt werden soll. Die Validierung einer SOAP-Nachricht auf Basis des zugehörigen Schemas verursacht einen erheblichen Verarbeitungsaufwand. Deswegen sollten Sie unter normalen Umständen VALIDATION(NO) angeben.

Durch die vollständige Validierung wird sichergestellt, dass alle SOAP-Nachrichten, die gesendet und empfangen werden, gültige XML-Komponenten darstellen, bei denen das XML-Schema beachtet wird. Wird VALIDATION(NO) angegeben, dann wird die Prüfung ausgeführt, um sicherzustellen, dass die Nachricht korrekt formatierte XML-Elemente enthält, eine Garantie, dass es sich auch um gültige XML-Elemente handelt, besteht dabei aber nicht.

**WSBIND**(*hfs-datei*)

Gibt einen aus 1 - 255 Zeichen bestehenden, vollständig qualifizierten oder relativen Namen der Datei für Web-Service-Bindungen unter z/OS UNIX an.

- Für eine Datei für Web-Service-Bindungen einer WEBSERVICE-Ressource, die mithilfe der Onlineresourcendefinition definiert wurde, wird entweder ein vollständig qualifizierter Dateipfad (mit einem Schrägstrich (/) als Präfix) oder ein Dateipfad relativ zum Ausgangsverzeichnis der Benutzer-ID für die CICS-Region angegeben.
- Für eine Datei für Web-Service-Bindungen für eine WEBSERVICE-Ressource, die in einem CICS-Bundle definiert ist, wird der Dateipfad relativ zum Stammverzeichnis des CICS-Bundles angegeben. Die zFS-Datei muss im

CICS-Bundle mit der WEBSERVICE-Ressource gepackt werden. Weitere Informationen hierzu finden Sie in Referencing zFS artifacts in a bundle.

Der angegebene Wert muss ein gültiger Name einer UNIX-Datei sein:

- Er darf keine eingebetteten Leerzeichen enthalten.
- Er darf keine aufeinanderfolgenden Vorkommen des Zeichens / enthalten.
- Die Groß-/Kleinschreibung muss beachtet werden.

**Zulässige Zeichen:**

A-Z a-z 0-9 . / \_ # @ -

#### **WSDLFILE**(*hfs-datei*)

Gibt den aus 1 - 255 Zeichen bestehenden, vollständig qualifizierten oder relativen Namen der Web-Service-Beschreibungsdatei (WSDL) unter z/OS UNIX an. Diese Datei wird verwendet, wenn die vollständige Laufzeitvalidierung aktiv ist.

- Für eine Web-Service-Beschreibungsdatei einer WEBSERVICE-Ressource, die mithilfe der Onlinere sourcedefinition definiert wurde, wird entweder ein vollständig qualifizierter Dateipfad (mit einem Schrägstrich (/) als Präfix) oder ein Dateipfad relativ zum Ausgangsverzeichnis der Benutzer-ID für die CICS-Region angegeben.
- Für eine Web-Service-Beschreibungsdatei für eine WEBSERVICE-Ressource, die in einem CICS-Bundle definiert ist, wird der Dateipfad relativ zum Stammverzeichnis des CICS-Bundles angegeben. Die zFS-Datei muss im CICS-Bundle mit der WEBSERVICE-Ressource gepackt werden. Weitere Informationen hierzu finden Sie in Referencing zFS artifacts in a bundle.

Der angegebene Wert muss ein gültiger Name einer UNIX-Datei sein:

- Er darf keine eingebetteten Leerzeichen enthalten.
- Er darf keine aufeinanderfolgenden Vorkommen des Zeichens / enthalten.
- Die Groß-/Kleinschreibung muss beachtet werden.

**Zulässige Zeichen:**

A-Z a-z 0-9 . / \_ # @ -

---

## Kapitel 2. CEDA-Befehle für Ressourcenmanagementtransaktionen

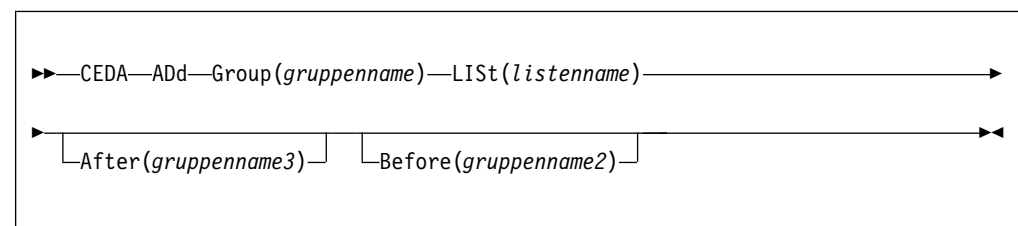
Die CEDA-Transaktion verfügt über eine Reihe von Befehlen für die Arbeit mit Ressourcendefinitionen in der CSD (CICS-Systemdefinition).

---

### Befehl CEDA ADD

Mit dem Befehl **CEDA ADD** können Sie eine Gruppe zu einer Liste in der CSD (CICS-Systemdefinition) hinzufügen.

#### Syntax



#### Beschreibung

Sie können den Befehl ADD in einer DISPLAY-Anzeige benutzen.

#### Optionen

##### **After**(gruppenname3)

Mit dieser Option können Sie die Platzierung des neuen Gruppennamens steuern. Wird BEFORE oder AFTER nicht angegeben, dann wird der Gruppenname am Ende der Liste hinzugefügt.

##### **Before**(gruppenname2)

Mit dieser Option können Sie die Platzierung des neuen Gruppennamens steuern. Wird BEFORE oder AFTER nicht angegeben, dann wird der Gruppenname am Ende der Liste hinzugefügt.

##### **Group**(gruppenname1)

Gibt den Namen der Gruppe an, die hinzugefügt werden soll. Der Name darf in der Liste noch nicht vorhanden sein. Ein generischer Gruppenname wird nicht akzeptiert. Wird keine Gruppe angegeben, dann wird der aktuelle Gruppenname hinzugefügt.

##### **LISt**(listenname)

Gibt den Namen der Liste an, zu der die Gruppe hinzugefügt werden soll. Ist die Liste noch nicht vorhanden, wird eine neue Liste erstellt. Wenn LIST nicht angegeben wird, dann wird der Gruppenname zur aktuellen Liste (sofern vorhanden) hinzugefügt. Ein generischer Listenname wird nicht akzeptiert.

#### Beispiele

1. Geben Sie Folgendes an, um eine Liste mit dem Namen LA01 zu erstellen, indem Sie eine Gruppe zu dieser Liste hinzufügen:  
ADD GROUP(GA001) LIST(LA01)

2. Geben Sie Folgendes an, um eine weitere Gruppe zur Liste LA01 hinzuzufügen, ohne dabei anzugeben, wo die Gruppe hinzugefügt werden soll:

```
ADD GROUP(GA002) LIST(LA01)
```

Die Liste LA01 hat nun folgendes Format:

GA001

GA002

3. Geben Sie Folgendes an, um eine weitere Gruppe am Anfang der Liste hinzuzufügen:

```
ADD GROUP(GA003) LIST(LA01) BEFORE(GA001)
```

Fügen Sie eine weitere Gruppe zwischen GA001 und GA002 hinzu:

```
ADD GROUP(GA004) LIST(LA01) AFTER(GA001)
```

Die Liste LA01 hat nun folgendes Format:

GA003

GA001

GA004

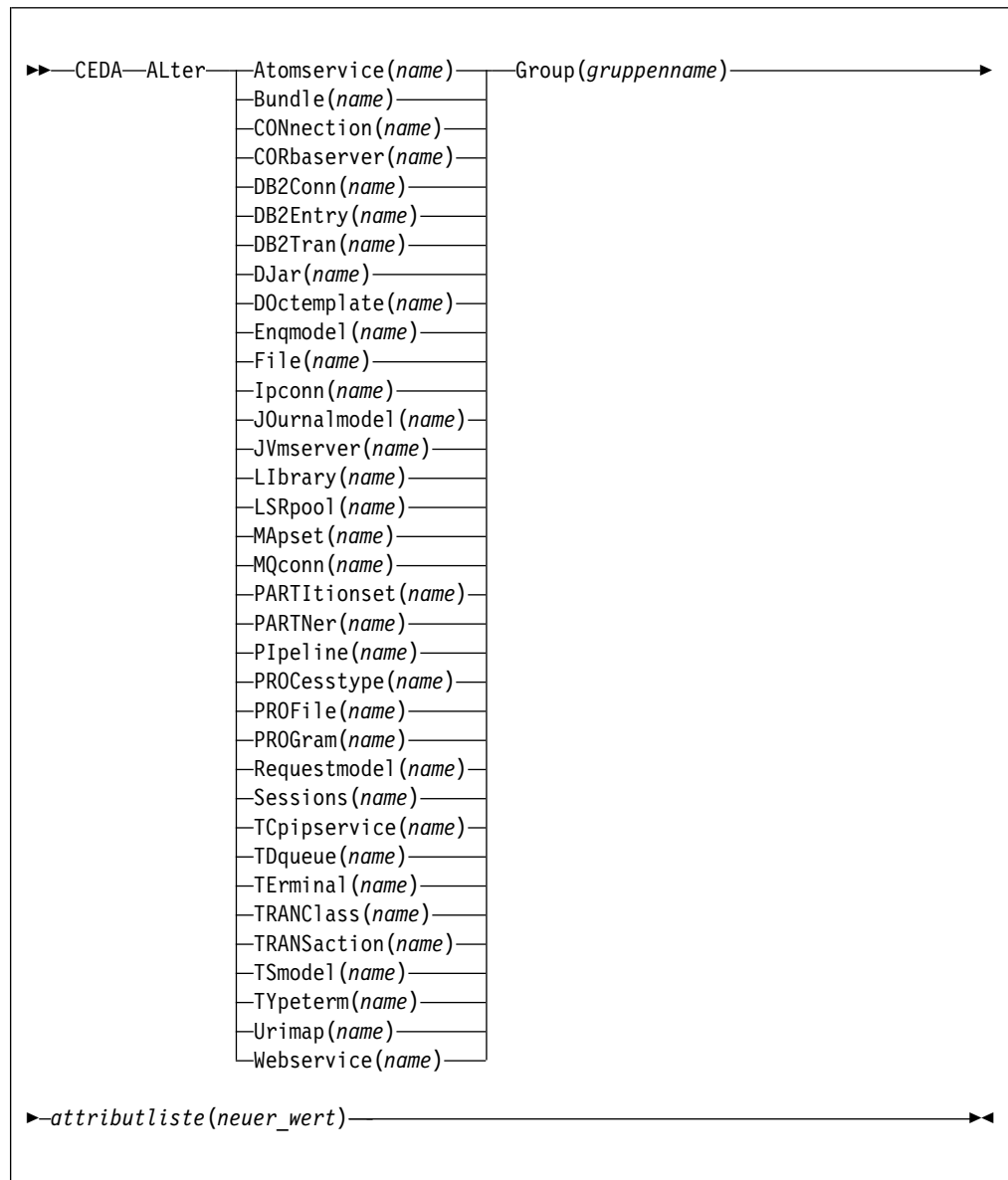
GA002

---

## Befehl CEDA ALTER

Verwenden Sie den Befehl **CEDA ALTER**, um einzelne oder alle Attribute einer vorhandenen Ressourcendefinition ändern.

## Syntax



## Beschreibung

**Important:** Verwenden Sie ALTER nicht, um den Wert der Attribute einer TYPE-TERM-Definition zu ändern, von der andere Attribute abhängig sind. Wenn Ihnen bei der Angabe für DEVICE, SESSIONTYPE oder TERMMODEL ein Fehler unterläuft, dann löschen Sie die Definition und erstellen Sie eine neue Definition mit den korrekten Werten.

Sie können Nullattributwerte angeben. Beispiel:

```
ALTER FILE(TEST) GROUP(ACT1) DESCRIPTION()
```

Wenn ein Attribut, für das Sie einen Nullwert angegeben haben, über einen Standardwert verfügt, dann hängt der verwendete Wert vom jeweiligen Feldtyp ab.

Beispiel:

- Der Befehl

```
ALTER FILE(TEST) GROUP(ACT1) RLSACCESS()
```

verhält sich so, als ob RLSACCESS nicht angegeben worden wäre. Das Attribut RLSACCESS weist einen CVDA-Standardwert auf, der ignoriert wird.

- Der Befehl

```
ALTER FILE(TEST) GROUP(ACT1) DESCRIPTION()
```

wirkt sich so aus, dass die Beschreibung (da kein Standardwert für das Feld DESCRIPTION vorhanden ist) mit Leerzeichen überschrieben wird.

- Der Befehl

```
ALTER FILE(TEST) GROUP(ACT1) PROFILE()
```

trägt den Standardwert DFHCICSA im Feld PROFILE ein. In diesem Fall ist der Standardwert eine Zeichenfolge und kein CVDA-Wert.

Änderungen an Ressourcendefinitionen in der CSD-Datei werden in einem momentan aktiven CICS-System erst nach der Installation der Definition oder der Gruppe wirksam, in der sich die Ressourcendefinition befindet.

Sie können den Befehl CEDAL ALTER in einer DISPLAY-Anzeige benutzen. Wenn Sie die Taste PF12 verwenden, nachdem Sie die gewünschten Änderungen vorgenommen haben, dann ruft CEDAL erneut die DISPLAY-Anzeige auf. In den Feldern für Datum und Uhrzeit wird die Nachricht 'ALTER SUCCESSFUL' angezeigt. Wenn Sie diesen Schritt ausführen, jedoch keine Änderungen vornehmen, dann wird der Befehl ALTER durch einen Stern (\*) ersetzt.

Mit einem generischen Namen können Sie einen Befehl ALTER verwenden, um in mehreren Ressourcendefinitionen für die gleichen Attribute eine identische Änderung vorzunehmen.

## Optionen

### **attribute list**

Gibt die Attribute an, die geändert werden sollen.

### **Group(*gruppenname*)**

Gibt den Namen der Gruppe an, die die Ressource enthält, die geändert werden soll.

### **ressource(*name*)**

Gibt den Typ und den Namen der Ressource an, deren Attribute geändert werden sollen.

## Beispiele

- Geben Sie Folgendes ein, um ein Programm als resident zu definieren:

```
ALTER PROGRAM(ERR01) GROUP(GENMODS) RESIDENT(YES)
    DATALOCATION()
```

Wenn Sie keine Attributliste angeben und

```
ALTER PROGRAM(ERR01) GROUP(GENMODS)
```

eingeben, dann gibt CEDAL die Nachricht ALTER SUCCESSFUL aus und ruft dann die Anzeige zum „Ändern durch Überschreibung“ (Overtypeto-Modify) auf.

- Geben Sie Folgendes ein, um den Status einer ganzen Programmgruppe zu ändern:

## Befehl CEDA APPEND

Mit dem Befehl **CEDA APPEND** können Sie die Gruppen in einer Liste der CSD (CICS-Systemdefinition) am Ende einer anderen Liste anfügen.

### Syntax

```
►►—CEDA—Append—LISt(listenname1)—To(listenname2)—►◄
```

### Optionen

#### LISt(*listenname1*)

Gibt die Quellenliste an, die angefügt werden soll. Ein generischer Listenname wird nicht akzeptiert.

#### To(*listenname2*)

Gibt die Zielliste an, an die angefügt werden soll. Ein generischer Listenname wird nicht akzeptiert. Wenn **listenname2** bereits vorhanden ist, dann wird die Quellenliste an diese Liste angefügt. Wenn **listenname2** nicht vorhanden ist, dann wird die Liste erstellt.

### Beispiel

Eine Liste mit dem Namen LISTA enthält die folgenden Gruppen:

```
GB001
GB002
GB003
```

Eine Liste mit dem Namen LISTB enthält die folgenden Gruppen:

```
G001
G002
G003
```

Fügen Sie LISTB wie folgt zu LISTA hinzu:

```
APPEND LIST(LISTB) TO(LISTA)
```

Daraufhin enthält LISTA die folgenden Gruppen in der hier angegebenen Reihenfolge:

```
GB001
GB002
GB003
G001
G002
G003
```

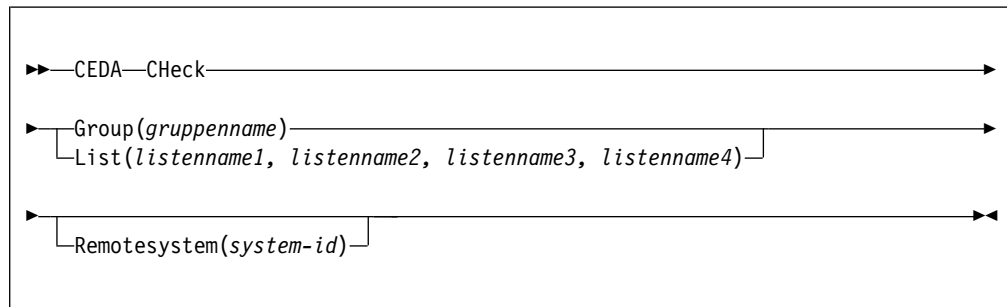
LISTB enthält weiterhin Folgendes:

```
G001
G002
G003
```

## Befehl CEDA CHECK

Mit dem Befehl **CEDA CHECK** können Sie die Konsistenz einer Gruppe von Ressourcendefinitionen in der CSD (CICS-Systemdefinition) prüfen.

### Syntax



### Beschreibung

Mit dem Befehl CHECK kann eine Gegenprüfung einer Gruppe, Liste oder Ressourcendefinitionsliste ausgeführt werden. Er sollte verwendet werden, bevor die Ressourcendefinitionen installiert werden.

Mit dem Befehl wird geprüft, ob die Ressourcendefinitionen in der Gruppe oder den Listen konsistent sind.

Beispiel: Für jede Transaktion (TRANSACTION) in der Liste, die geprüft wird, wird festgestellt, ob die angegebene PROGRAM-Definition in einer der Gruppen vorhanden ist. Der Erfolg der Prüfung bedeutet nicht zwingend, dass das in PROGRAM angegebene Programm für das aktive System zur Verfügung steht.

Listen sollten vor ihrer Verwendung für die Initialisierung von CICS während eines Erst- oder Kaltstarts geprüft werden (wenn die Listen im Systeminitialisierungsparameter GRPLIST angegeben werden).

Eine Gruppe sollte überprüft werden, bevor sie mit dem Befehl INSTALL auf dem aktiven CICS-System installiert wird.

Eine Gruppe kann überprüft werden, bevor Sie den Befehl ADD verwenden, um die Gruppe zu einer Liste hinzuzufügen. (Die Gruppe ist möglicherweise nicht abgeschlossen. In diesem Fall ist es nicht zielführend, sie isoliert zu überprüfen. Nehmen Sie sie in eine Liste der Gruppen auf, die zugehörige Ressourcendefinitionen enthalten.)

Sie können den Befehl CHECK in einer DISPLAY-Anzeige benutzen.

### Vorgehensweise von CEDA bei der Überprüfung von Definitionen

Mit dem Befehl CHECK wird die Konsistenz von Definitionen innerhalb einer Gruppe oder innerhalb aller Gruppen in einer Liste überprüft. Allerdings kann mit ihm keine Gegenprüfung aller Attribute einer Ressource durchgeführt werden. Möglicherweise erhalten Sie auch Fehlernachrichten, wenn Sie eine Gruppe installieren, obwohl mit dem Befehl CHECK keine Probleme festgestellt werden konnten.



Wenn Sie den Befehl CHECK GROUP benutzen, dann führt CEDA eine Gegenprüfung aller Ressourcen in einer angegebenen Gruppe durch, um sicherzustellen, dass die Gruppe für die Verwendung bereit ist. Sie können beispielsweise mit CHECK darüber gewarnt werden, dass eine Transaktionsdefinition in der Gruppe kein Programm in derselben Gruppe benennt. (Hierbei ist zu beachten, dass es sich hierbei möglicherweise nicht um einen Fehler handelt. Die Gruppe wurde eventuell absichtlich einer Gruppe zugeordnet, die das Programm enthält, oder Sie möchten, dass das Programm automatisch installiert wird. In diesem Fall verfügt es nicht über eine Definition.)

Wenn Sie den Befehl CHECK LIST verwenden, führt CEDA für jede Gruppe, die in der Liste oder den Listen aufgeführt ist, eine Gegenprüfung durch. Dabei wird nicht jede Gruppe nacheinander geprüft, sondern die Definitionen in allen aufgelisteten Gruppen werden zusammengeführt und dann gemeinsam überprüft. Auf diese Weise erhalten Sie eine Warnung, wenn doppelte Ressourcendefinitionen oder Referenzen auf Definitionen erkannt werden, die nicht vorhanden sind.

## Optionen

### **Group**(*gruppenname*)

Gibt die Gruppe an, die geprüft werden soll. Ein generischer Gruppenname wird nicht akzeptiert.

### **List**(*listenname1, listenname2 etc.*)

Gibt die Listen an, die geprüft werden sollen. Ein generischer Listenname wird nicht akzeptiert.

### **Remotesystem**(*system-id*)

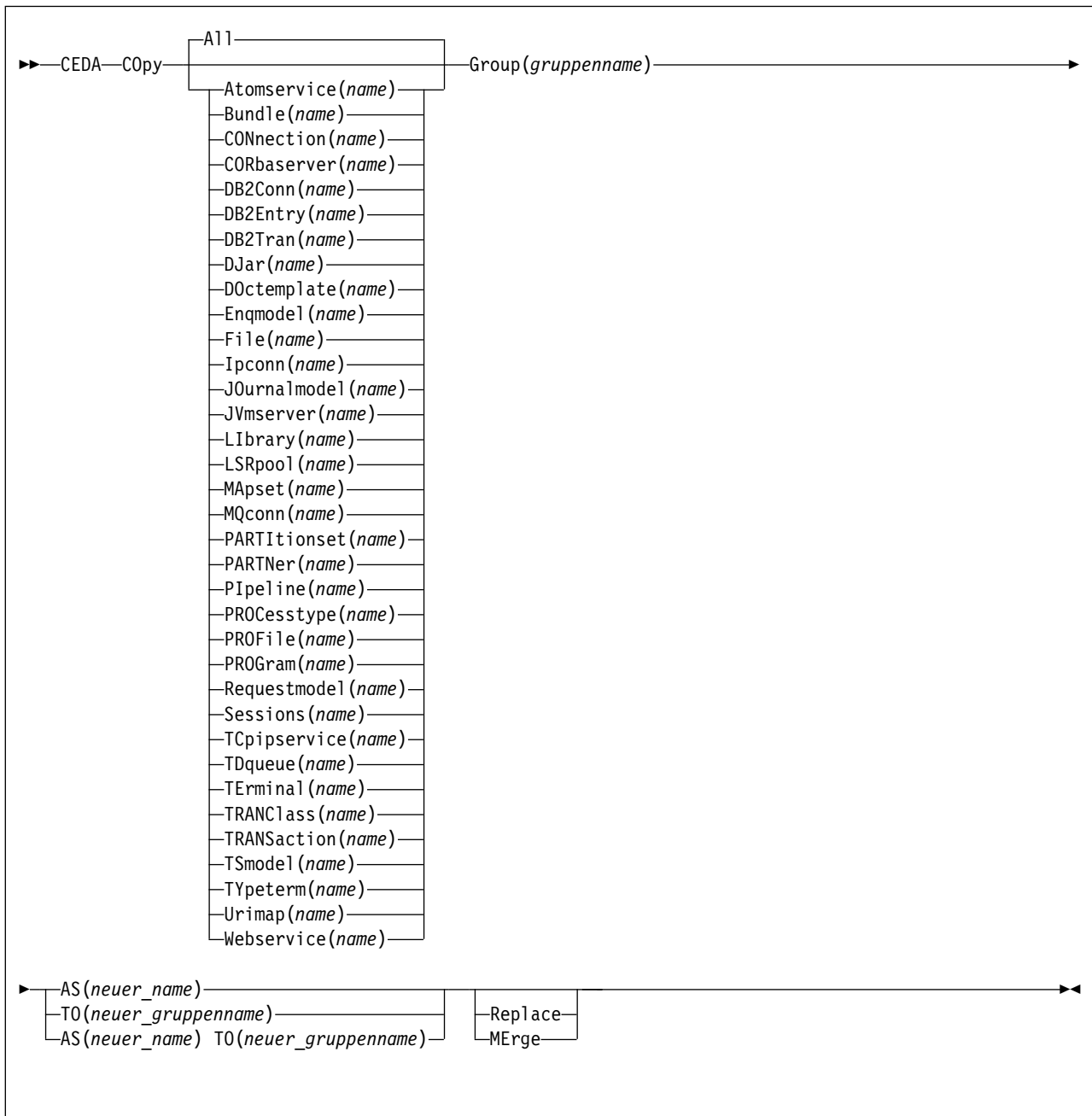
Gibt an, dass eine Überprüfung für die Gruppe oder Liste in einer CICS-Region mit einer System-ID ausgeführt werden soll, die von der System-ID der Region abweicht, in der der Befehl CHECK abgesetzt wird. Wenn diese Option nicht verwendet wird, benutzt der Befehl CHECK die System-ID der Region, von der aus der Befehl CHECK abgesetzt wird.

---

## Befehl CEDA COPY

Mit dem Befehl **CEDA COPY** können Sie eine Ressourcendefinition entweder innerhalb derselben Gruppe oder in eine andere Gruppe der CSD (CICS-Systemdefinition) kopieren.

## Syntax



## Beschreibung

Wenn Sie weder MERGE noch REPLACE angeben, werden Sie in einer Nachricht gewarnt, dass Sie versuchen, doppelte Ressourcen zu erstellen. Die COPY-Operation schlägt fehl.

## Optionen

### AS(*neuer\_name*)

Wenn Sie eine Definition innerhalb einer bestimmten Gruppe kopieren möchten, dann müssen Sie die Option AS verwenden, um sie umzubenennen. Sie

können AS auch verwenden, um eine Definition in eine andere Gruppe zu kopieren und sie gleichzeitig umzubenennen. Bei Verwendung von AS sind generische Namen nicht zulässig.

**Group(*gruppenname*)**

Gibt die Gruppe an, die die zu kopierenden Definitionen enthält.

**MErge**

Diese Option gilt, wenn doppelte Definitionsnamen in den Gruppen enthalten sind, die im Befehl COPY angegeben wurden. Wenn Sie MERGE angeben, dann werden doppelt vorhandene Definitionen in der TO-Gruppe nicht ersetzt.

**Replace**

Diese Option gilt, wenn doppelte Definitionsnamen in den Gruppen enthalten sind, die im Befehl COPY angegeben wurden. Wenn Sie REPLACE angeben, dann ersetzen die zu kopierenden Definitionen die Definitionen, die in der im Operanden TO angegebenen Gruppe enthalten sind.

**ressource(*name*)**

Gibt den Typ und den Namen der Ressource an, deren Attribute kopiert werden sollen.

**TO(*neuer\_gruppenname*)**

Sie können Definitionen in eine andere Gruppe kopieren und dabei im Operanden TO die neue Gruppe angeben.

## Beispiele

Sie können eine einzelne Ressourcendefinition in eine neue Gruppe kopieren und dabei in der Option TO die neue Gruppe angeben. Beispiel:

```
COPY SESSIONS(L122) GROUP(CICSC1) TO(CICSC2)
```

Sie können eine Ressourcendefinition innerhalb derselben Gruppe kopieren. In diesem Fall müssen Sie sie mit der Option AS umbenennen. Beispiel:

```
COPY TERMINAL(TD12) AS(TD34) GROUP(TERMVDU1)
```

Beim Kopieren zwischen verschiedenen Gruppen können Sie der Kopie einen neuen Namen zuordnen. Verwenden Sie hierzu die Option AS, um den neuen Namen anzugeben.

```
COPY PROGRAM(ABC01) GROUP(XYZ) AS(ABC02) TO(NEWXYZ)
```

(Wenn Sie für die Kopie denselben Namen wie für die Originaldefinition beibehalten wollen, dann müssen Sie unbedingt darauf achten, bei der Installation die richtige Definition zu verwenden.)

Mit der Option ALL (ohne Angabe eines Namens) können Sie alle Ressourcendefinitionen in der Gruppe in eine neue Gruppe kopieren. Beispiel:

```
COPY ALL GROUP(N21TEST) TO(N21PROD)
```

Sie können auch mehrere Ressourcendefinitionen in eine neue Gruppe kopieren. Verwenden Sie dazu die Option TO, um die neue Gruppe anzugeben.

Mit einem generischen Ressourcendefinitionsnamen können Sie alle oder bestimmte Definitionen des gleichen Ressourcentyps in die neue Gruppe kopieren. Beispiel:

```
COPY CONNECTION(*) GROUP(CICSG1) TO(CICSG2)  
COPY PROGRAM(N21++) GROUP(NTEST) TO(NPROD)
```

Wenn Sie die Option ALL mit einem generischen Namen verwenden, dann können Sie alle oder bestimmte Ressourcendefinitionen in der Gruppe in die neue Gruppe kopieren. Beispiel:

```
COPY ALL(N21*) GROUP(N21OLD) TO(N21NEW)
```

Wenn Sie die Option ALL mit einem spezifischen Namen verwenden, dann können Sie alle in der Gruppe enthaltenen Ressourcendefinitionen mit diesem Namen (die zwingend unterschiedliche Ressourcentypen aufweisen müssen) in die neue Gruppe kopieren. Beispiel:

```
COPY ALL(XMPL) GROUP(EXAMPLE) TO(EX2)
```

Wenn Sie eine Reihe von Definitionen in eine andere Gruppe kopieren wollen und wenn die Gruppen doppelte Ressourcennamen enthalten, dann müssen Sie entweder MERGE oder REPLACE angeben. Beispiel:

```
COPY ALL GROUP(TAX1) TO(TAX2) MERGE  
COPY ALL GROUP(TAX1NEW) TO(TAX1) REPLACE
```

Im folgenden Beispiel wird eine Gruppe mit dem Namen GA001 in eine Gruppe mit dem Namen GA002 kopiert, die bereits vorhanden ist. Dabei werden alle doppelten Ressourcendefinitionen durch die in der Gruppe GA001 vorhandenen Ressourcendefinitionen ersetzt.

```
COPY GROUP(GA001) TO(GA002) REPLACE
```

Im folgenden Beispiel wird die Gruppe GA003 in die Gruppe GA004 kopiert. Bei eventuell vorhandenen Duplikaten werden jedoch die Ressourcendefinitionen in Gruppe GA004 beibehalten.

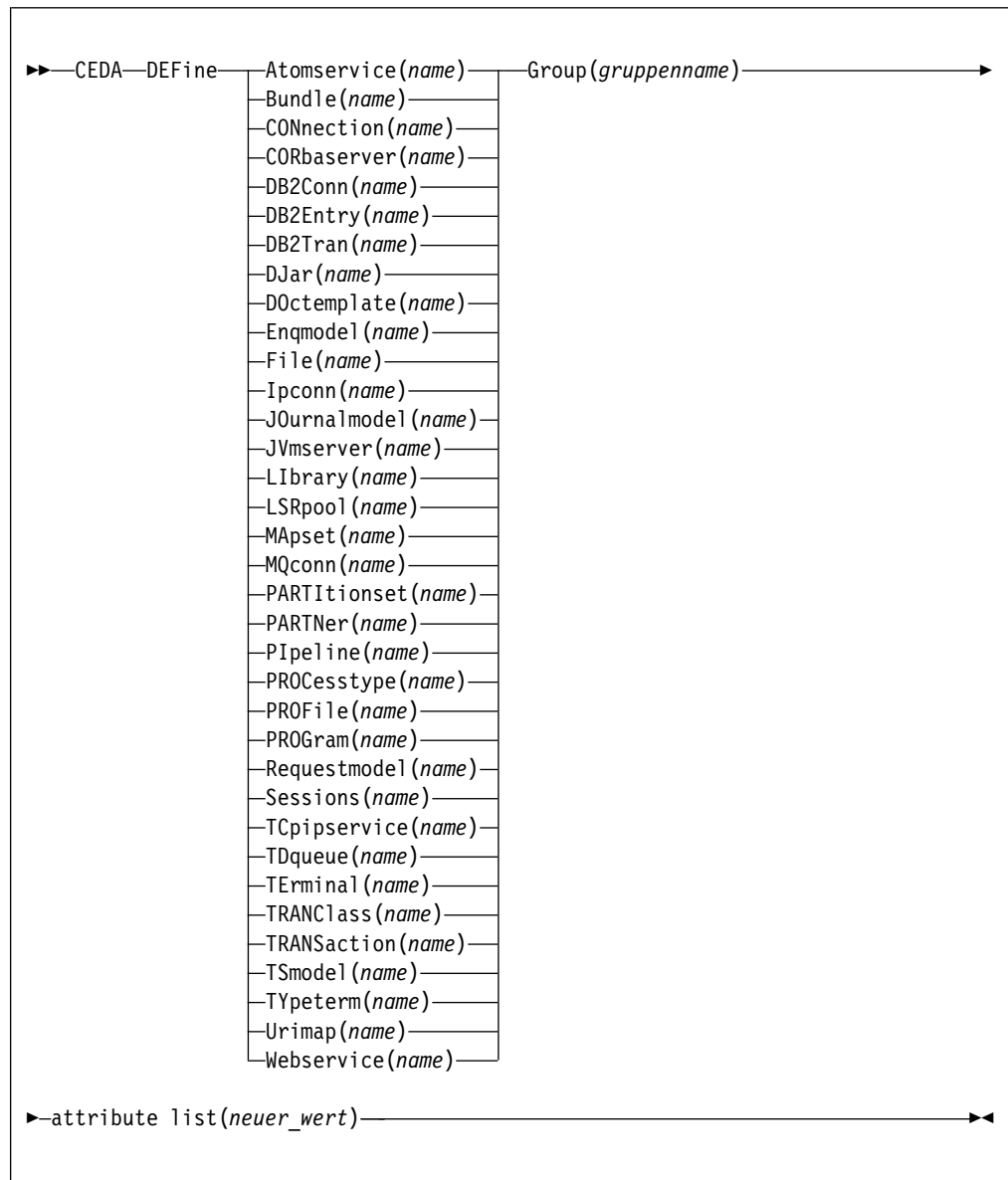
```
COPY GROUP(GA003) TO(GA004) MERGE
```

---

## Befehl CEDA DEFINE

Mit dem Befehl **CEDA DEFINE** können Sie neue Ressourcendefinitionen in der CSD (CICS-Systemdefinition) erstellen.

## Syntax



## Beschreibung

Sie können den gleichen Namen für mehrere Ressourcendefinitionen in einer Gruppe verwenden, wenn die Definitionen für unterschiedliche Ressourcentypen gelten. Beispiel:

```
DEFINE PROGRAM(N28A) GROUP(N28APPL)
DEFINE TRANSACTION(N28A) GROUP(N28APPL)
```

```
DEFINE TERMINAL(USER) GROUP(USERDEF)
DEFINE PROGRAM(USER) GROUP(USERDEF)
```

Wenn Sie über Ressourcendefinitionen mit dem gleichen Namen verfügen, dann müssen Sie sicherstellen, dass Sie die richtige Ressourcendefinition installieren. Weitere Informationen hierzu finden Sie in [Duplicate resource definition names](#).

Sie müssen den Ressourcentyp und den Namen in der Befehlszeile angeben. Sie können auch die Gruppe und andere Attribute in der Befehlszeile angeben.

Die gesamte Definition wird in einer Anzeige zum Ändern durch Überschreibung (Overtyp-to-Modify) dargestellt. Sie können die Attribute in dieser Anzeige ändern, sofern Sie nicht einen vollständigen Befehl DEFINE in der Befehlszeile angegeben haben. In diesem Fall können die Attribute NAME und GROUPNAME nicht geändert werden.

Die Definition wurde erstellt, als Sie den Befehl DEFINE eingegeben haben. Wenn Sie keine weiteren Änderungen mehr daran vornehmen wollen, dann können Sie einen weiteren Befehl über die Befehlszeile eingeben.

Wenn eine Ressourcendefinition mit dem gleichen Namen und Typ bereits in der Gruppe vorhanden ist, dann werden alle in der Befehlszeile angegebenen Attribute ignoriert und die bereits vorhandene Ressourcendefinition wird angezeigt. Anschließend können Sie die vorhandenen Werte bei Bedarf überschreiben und ändern. Wenn Sie keine Änderungen an der Definition vornehmen wollen, dann können Sie einen weiteren Befehl über die Befehlszeile eingeben.

Achten Sie unbedingt darauf, keine Werte zu überschreiben, von denen andere Attributwerte abhängig sind.

## Optionen

### **attribute list**

Die Attributliste ist vom definierten Ressourcentyp abhängig. Bestimmte Ressourcen verfügen über Attribute, die in die Definition eingebunden werden müssen. Für Attribute, für die von Ihnen kein Wert angegeben wird, werden die entsprechenden Standardwerte verwendet.

### **Group**(*gruppenname*)

Gibt den Namen der Gruppe an, in der die zu definierende Ressourcendefinition enthalten ist. Verwenden Sie keinen generischen Gruppennamen. Wenn Sie den Namen einer Gruppe angeben, die noch nicht vorhanden ist, dann wird die Gruppe erstellt.

### *ressource*(*name*)

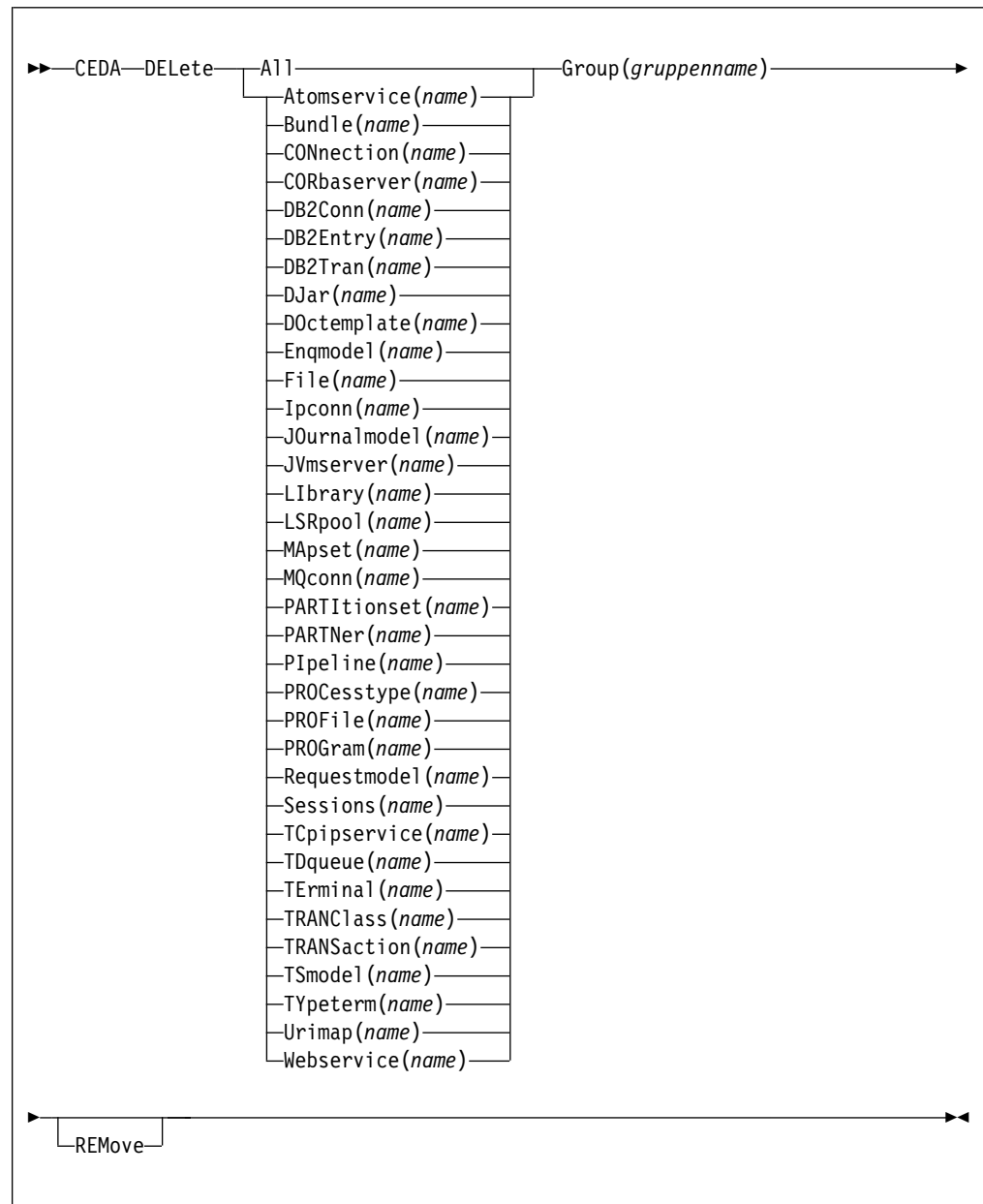
Gibt den Typ und den Namen der Ressource an, die definiert werden soll. Verwenden Sie keinen generischen Ressourcennamen.

---

## Befehl CEDA DELETE

Mit dem Befehl **CEDA DELETE** können Sie einzelne oder mehrere Ressourcendefinitionen aus der CSD-Datei löschen.

## Syntax



## Beschreibung

Das Löschen einer Ressourcendefinition unterscheidet sich vom Entfernen einer Gruppe aus einer Liste (siehe „Befehl CEDA REMOVE“ auf Seite 362). Eine gelöschte Ressourcendefinition wird tatsächlich aus der CSD-Datei entfernt.

Wenn Sie die letzte Ressource aus einer Gruppe löschen, wird auch die Gruppe automatisch gelöscht. Es dürfen keine leeren Gruppen existieren.

Dieser Befehl hat **keine** Auswirkungen auf die Definitionen, die auf dem aktiven CICS-System installiert sind. Um installierte Definitionen von dem aktiven CICS-System zu entfernen, können Sie entweder die Transaktion CEMT DISCARD oder den Befehl EXEC CICS DISCARD verwenden.

## Optionen

### **All**

Gibt an, dass alle Ressourcen aus der Gruppe gelöscht werden sollen.

### **Group**(*gruppenname*)

Gibt die Gruppe an, die die Ressource enthält. Verwenden Sie keinen generischen Gruppennamen.

### **REMove**

Gibt an, dass eine Gruppe, die gelöscht wird, aus allen Listen entfernt werden muss, in denen sie enthalten war.

### *resource*(*name*)

Gibt den Typ und den Namen der Ressource an, die gelöscht werden soll. Sie können einen generischen Ressourcennamen verwenden.

## Beispiele

- Geben Sie Folgendes ein, um alle Ressourcen in einer Gruppe zu löschen:  
`DELETE ALL GROUP(TOPS)`
- Geben Sie Folgendes ein, um alle Programme in einer Gruppe zu löschen:  
`DELETE PROGRAM(*) GROUP(NSOS)`

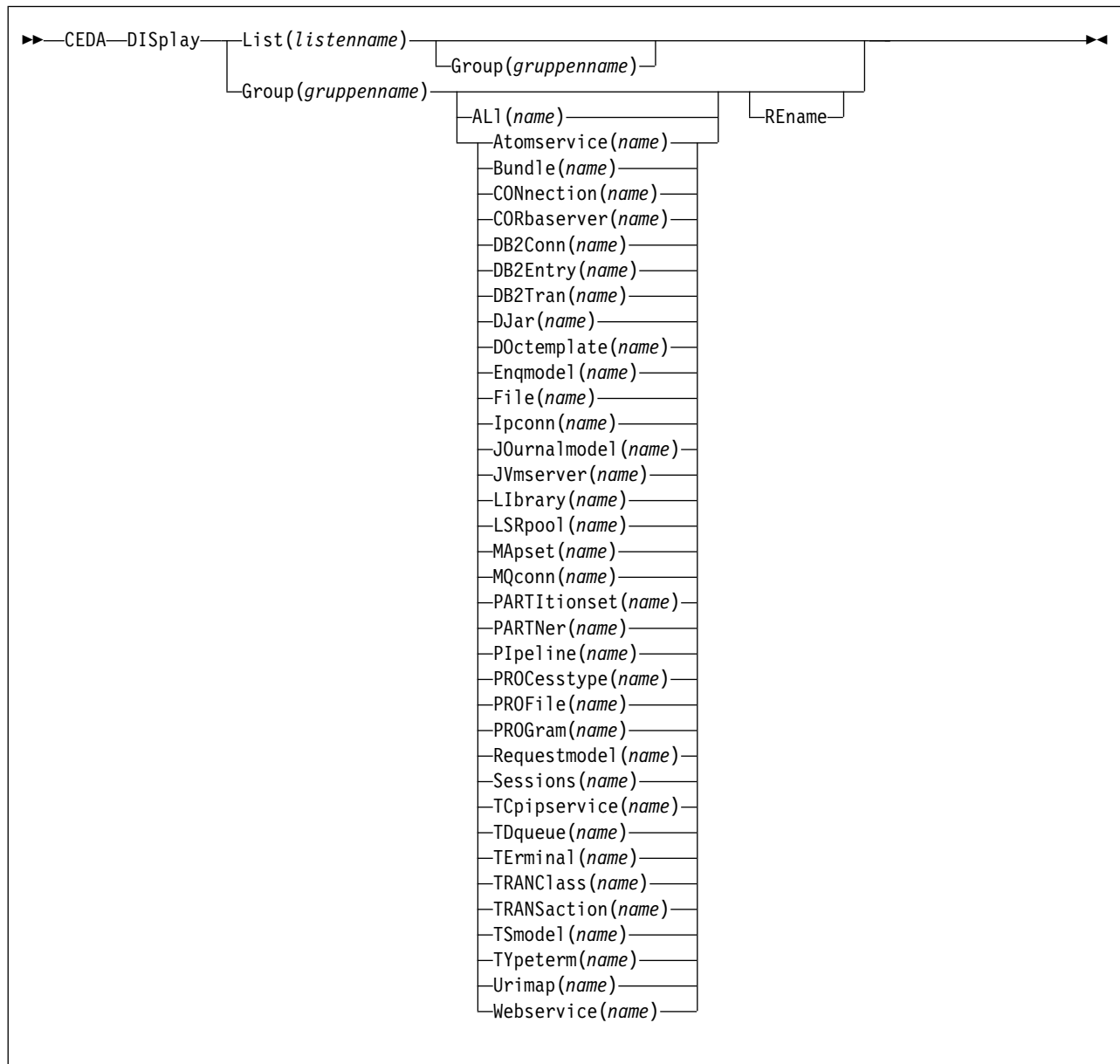
---

## Befehl CEDA DISPLAY

Mit dem Befehl **CEDA DISPLAY** können Sie einzelne oder mehrere Gruppennamen, Listennamen oder Ressourcen anzeigen.



## Syntax



## Optionen

### **Group(gruppenname)**

Gibt den Namen der Gruppe an, die angezeigt werden soll. Sie können einen generischen Gruppennamen verwenden.

### **List(listenname)**

Gibt den Namen der Liste an, die angezeigt werden soll. Sie können einen generischen Listennamen verwenden.

### **REname**

Diese Option gilt nur für GROUP. Wenn Sie RENAME angeben, können Sie Ressourcendefinitionen innerhalb der Gruppe umbenennen.

### **ressource(name)**

Gibt den Typ und den Namen der Ressource an, die angezeigt werden soll.

## Beispiele

- Geben Sie Folgendes ein, um alle Gruppen in der CSD-Datei anzuzeigen:  
`DISPLAY GROUP(*)`
- Geben Sie Folgendes ein, um alle Gruppen anzuzeigen, deren Name sieben Zeichen umfasst und mit der Zeichenfolge PROD beginnt:  
`DISPLAY GROUP(PROD+++)`
- Geben Sie Folgendes ein, um alle Listen in der CSD-Datei anzuzeigen:  
`DISPLAY LIST(*)`
- Geben Sie Folgendes ein, um alle Listen anzuzeigen, deren Name fünf Zeichen umfasst und mit der Zeichenfolge PROD beginnt:  
`DISPLAY LIST(PROD+)`

## Befehl CEDA DISPLAY GROUP

Mit dem Befehl **CEDA DISPLAY GROUP** können Sie einzelne oder mehrere Gruppennamen in der Gesamtanzeige darstellen.

Neben den Namen in der Anzeige können Sie die folgenden Befehle eingeben:

CHECK  
DISPLAY ALL  
EXPAND  
INSTALL  
LOCK  
UNLOCK

**Restriction:** In einer DISPLAY-Anzeige können Sie keine CONNECTION- und SESSIONS-Ressourcen installieren. Sie müssen in Gruppen installiert werden.

In dieser Anzeige können sämtliche Befehle bis auf ihren Anfangsbuchstaben abgekürzt werden. Der Gruppenname muss nicht eingegeben werden. Die Syntax der einzelnen Befehle finden Sie in den Informationen zum jeweiligen Befehl in diesem Abschnitt.

Um die Gruppennamen mit DISPLAY anzeigen zu können, müssen Sie einen generischen Namen verwenden.

## Befehl CEDA DISPLAY LIST

Mit dem Befehl **CEDA DISPLAY LIST** können Sie einzelne oder mehrere Listennamen in der Gesamtanzeige darstellen.

Neben den Namen in der Anzeige können Sie die folgenden Befehle eingeben:

CHECK  
DISPLAY GROUP  
EXPAND  
LOCK  
UNLOCK

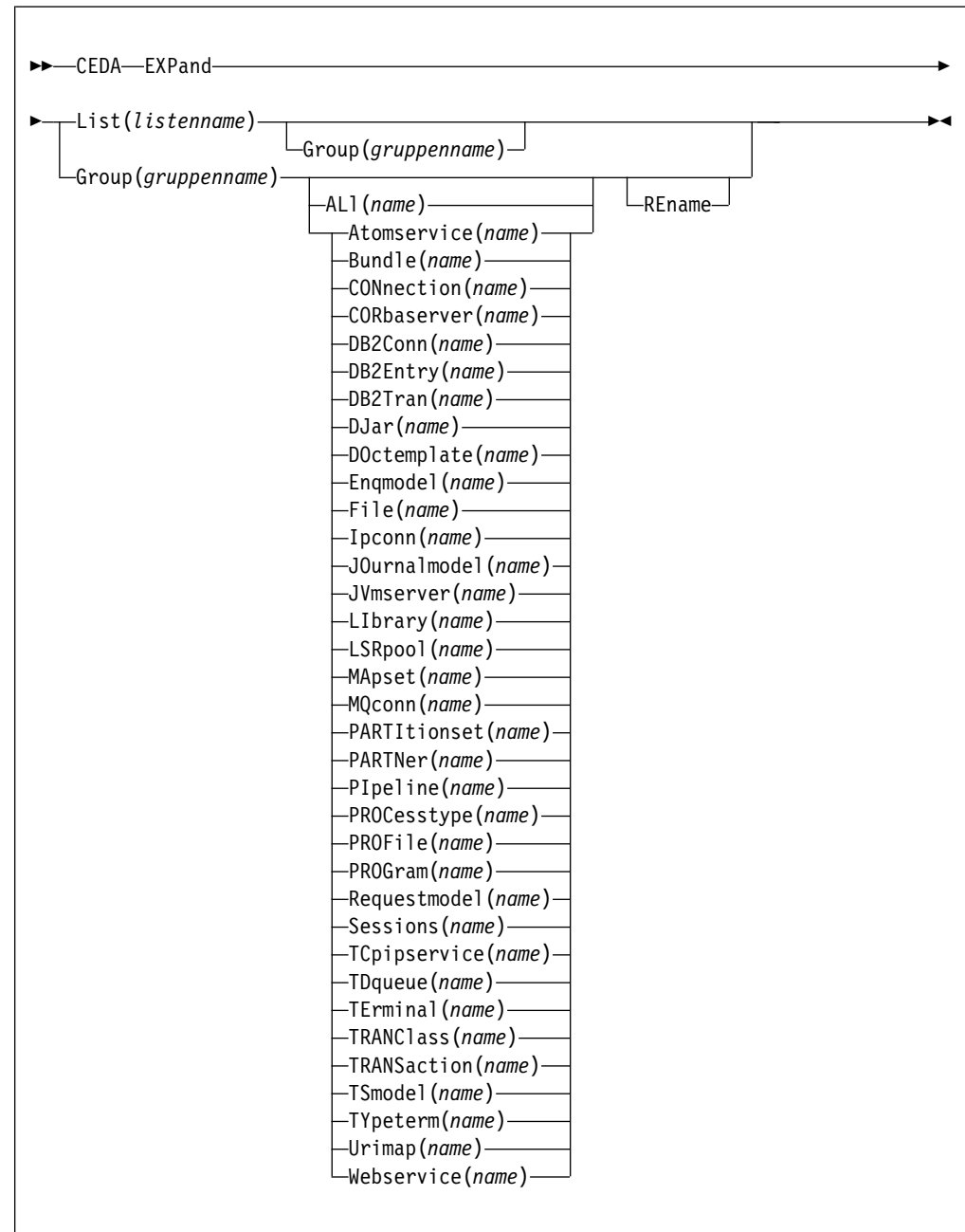
In dieser Anzeige können sämtliche Befehle bis auf ihren Anfangsbuchstaben abgekürzt werden. Der Listenname muss nicht eingegeben werden. Die Syntax der einzelnen Befehle finden Sie in den Informationen zum jeweiligen Befehl in diesem Abschnitt.

Um die Listennamen mit DISPLAY anzeigen zu können, müssen Sie einen generischen Listennamen verwenden.

## Befehl CEDA EXPAND

Mit dem Befehl **CEDA EXPAND** können Sie die Ressourcendefinitionen anzeigen, die in einzelnen oder mehreren Gruppen oder Listen enthalten sind. Sie können für die angezeigten Ressourcennamen weitere CEDA-Befehle eingeben.

### Syntax



### Optionen

#### **Group(gruppenname)**

Gibt die Gruppe an, die erweitert werden soll.

**List(listenname)**

Gibt die Liste an, die erweitert werden soll.

**REname**

Diese Option gilt nur für GROUP. Wenn Sie RENAME angeben, können Sie Ressourcendefinitionen innerhalb der Gruppe umbenennen.

**ressource(name)**

Gibt den Typ und den Namen der Ressource an, die in der erweiterten Gruppe angezeigt werden soll.

## Befehl CEDA EXPAND GROUP

Mit dem Befehl **EXPAND GROUP** können Sie die Ressourcendefinitionen anzeigen, die in einzelnen oder mehreren Gruppen enthalten sind. Sie können für die angezeigten Ressourcennamen weitere CEDA-Befehle eingeben.

Neben den Namen in der Anzeige können Sie die folgenden Befehle eingeben:

ALTER  
COPY  
DELETE  
INSTALL  
MOVE  
RENAME  
VIEW

Alle Befehle können bis auf ihren Anfangsbuchstaben abgekürzt werden. Der Gruppen- oder Listenname muss nicht eingegeben werden. Die Syntax der einzelnen Befehle finden Sie in den Informationen zum jeweiligen Befehl in diesem Abschnitt.

Sie können mit EXPAND einen generischen Gruppennamen erweitern. Beispiel:

- Geben Sie Folgendes ein, um alle Ressourcendefinitionen in allen Gruppen in der CSD-Datei anzuzeigen:

EXPAND GROUP(\*)

- Geben Sie Folgendes ein, um alle Ressourcendefinitionen in Gruppen anzuzeigen, deren Name mit der Zeichenfolge PROD beginnt und sieben Zeichen lang ist:

EXPAND GROUP(PROD+++)

Mit EXPAND können Sie eine Gruppe so erweitern, dass nur ein Ressourcentyp angezeigt wird. Der von Ihnen als Ressourcename angegebene Name kann ein generischer Name sein. Beispiel:

- Geben Sie Folgendes ein, um alle PROFILE-Definitionen in allen Gruppen in der CSD-Datei anzuzeigen:

EXPAND GROUP(\*) PROFILE(\*)

- Geben Sie Folgendes ein, um alle TERMINAL-Definitionen in einer Gruppe anzuzeigen, deren Name mit den Zeichen SZ beginnt:

EXPAND GROUP(ZEMGROUP) TERMINAL(SZ++)

Sie können Gruppen mit EXPAND erweitern und die Ressourcendefinitionen auf die mit einem gemeinsamen generischen Namen einschränken. Beispiel:

- Geben Sie Folgendes ein, um alle Ressourcendefinitionen aller Typen anzuzeigen, deren Name auf A1 endet:

EXPAND GROUP(REINDEER) ALL(\*A1)

Bei Verwendung der Option RENAME wird eine spezielle Anzeige aufgerufen, in der Sie die Namen der Ressourcendefinitionen überschreiben können. Auf diese Weise können Sie eine große Anzahl von Ressourcendefinition schnell umbenennen.

## Befehl CEDA EXPAND LIST

Mit dem Befehl **EXPAND LIST** können Sie die Gruppen anzeigen, die in einzelnen oder mehreren Listen enthalten sind. Sie können für die angezeigten Gruppennamen weitere CEDA-Befehle eingeben.

Wenn Sie eine Liste erweitern, können Sie die folgenden Befehle neben den Namen in der Anzeige eingeben:

ADD  
REMOVE

Mit EXPAND können Sie Teile einer Liste erweitern, indem Sie einen generischen Gruppennamen angeben. Beispiel:

```
EXPAND LIST(INITLIST) GROUP(DFH*)
```

Mit EXPAND können Sie mehrere Listen erweitern, indem Sie einen generischen Listennamen angeben. Geben Sie Folgendes ein, um z. B. die Gruppen in allen Listen in der CSD-Datei anzuzeigen:

```
EXPAND LIST(*)
```

Geben Sie Folgendes ein, um die Gruppen in Listen anzuzeigen, deren Name mit der Zeichenfolge PROD beginnt und fünf Zeichen lang ist:

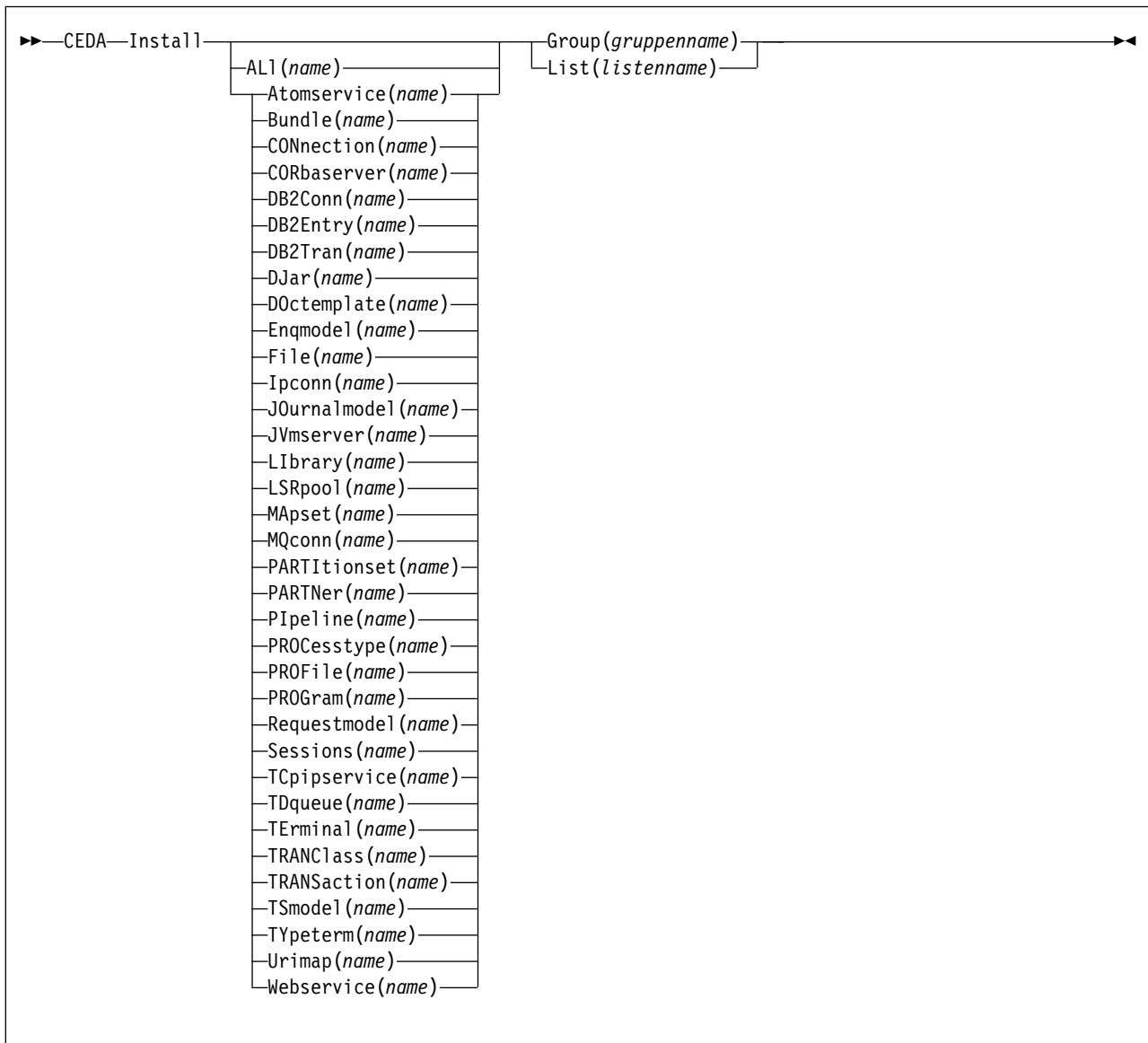
```
EXPAND LIST(PROD+)
```

---

## Befehl CEDA INSTALL

Mit dem Befehl **CEDA INSTALL** können die Ressourcendefinitionen in der angegebenen Gruppe oder Gruppenliste für das aktive CICS-System verfügbar gemacht werden.

## Syntax



## Beschreibung

Mithilfe einer Einzelressourceninstallation können Sie ausschließlich die von Ihnen benötigten Ressourcen installieren.

Bei Verwendung für Telekommunikationsressourcen sowie Ressourcen für die übergreifende Kommunikation gelten beim Befehl `INSTALL` für Einzelressourcen gewisse Einschränkungen. Die Einschränkungen gelten für Ressourcendefinitionen, die verbunden sind, z. B. für `CONNECTION`- und `SESSIONS`-Definitionen.

Die folgenden Ressourcentypen können *nur* als Teil einer Gruppe installiert werden:

- `CONNECTION`-Definitionen mit Ausnahme von `CONNECTION` mit `ACCESS-METHOD(INDIRECT)`
- `SESSIONS`-Definitionen

- TERMINAL-Definitionen für Pipeline-Terminals, die denselben Pool gemeinsam nutzen

Wenn Sie für TERMINAL-Definitionen und zugehörige TYPETERM-Definitionen den Befehl INSTALL für Einzelressourcen verwenden wollen, dann müssen Sie die TYPETERM-Definition, auf die in einer TERMINAL-Definition verwiesen wird, *vor* der entsprechenden TERMINAL-Definition installieren.

Diese Einschränkung gilt auch bei der Installation von Gruppen, die TERMINAL- und TYPETERM-Definitionen enthalten. Installieren Sie die Gruppe, die die TYPETERM-Definition enthält, vor der Gruppe mit der TERMINAL-Definition. Ähnliches gilt, wenn Sie eine Liste installieren möchten, die Gruppen mit TERMINAL- und TYPETERM-Definitionen enthält. Die Gruppe mit der TYPETERM-Definition muss auch in diesem Fall vor der Gruppe installiert werden, in der die TERMINAL-Definition enthalten ist. In einer Liste mit Gruppen von Db2-Ressourcendefinitionen (DB2CONN-, DB2ENTRY- und DB2TRAN-Definitionen) muss ebenfalls die DB2CONN-Definition in der ersten Gruppe der Liste definiert werden.

Wenn Sie die Einzelressourceninstallation für ENQMODEL verwenden wollen, dann ist zu beachten, dass die ENQMODEL-Definitionen sich normalerweise im Standardstatus ENABLED befinden. Demzufolge sind bei der Installation von ENQMODEL-Definitionen die Regeln zum Aktivieren von ENQMODEL-Definitionen zu beachten. Dies bedeutet, dass die ENQMODEL-Definitionen, aus denen verschachtelte generische ENQnames gebildet werden, nacheinander aktiviert werden müssen. Dabei wird zuerst das speziellste Element und zuletzt das am wenigsten spezielle Element verarbeitet (z. B. ABCD\* dann ABC\* dann AB\*).

Ist die angegebene Ressource auf dem aktiven CICS-System bereits vorhanden, dann wird die schon existierende Definition durch die neue Definition ersetzt.

Die vorhandene Definition wird nur dann ersetzt, wenn sie momentan nicht verwendet wird.

Wenn die Installation einzelner oder mehrerer Ressourcen in den zu installierenden Gruppen fehlschlägt, weil die betreffende Ressource momentan benutzt wird oder weil ein anderer Fehlergrund vorliegt, dann werden die folgenden Aktionen ausgeführt:

1. Der Installationsprozess wird bei der nächsten Ressource in der Gruppe fortgesetzt.
2. Nach Abschluss der Installation einer Gruppe werden alle Ressourcen im installierbaren Set zurückgesetzt, wenn die Ressource, deren Verarbeitung fehlgeschlagen ist, Mitglied des *installierbaren Sets* war. Ressourcen, die auf Einzelressourcenebene festgeschrieben wurden, werden nicht zurückgesetzt. Eine Liste der Ressourcentypen, die als Teil eines installierbaren Sets festgeschrieben werden, finden Sie in *What happens when you use the INSTALL command*.
3. Das System gibt eine Nachricht aus, in der Sie darüber informiert werden, wenn die Gruppe nur teilweise installiert wurde. Außerdem erhalten Sie für jede der Ressourcen, deren Installation fehlgeschlagen ist, ebenfalls eine Nachricht in der Nachrichtenanzeige, in der Sie über die Fehlerursache informiert werden. Für diejenigen Ressourcen, die zurückgesetzt wurden, erhalten Sie keine Nachrichten.

Darüber hinaus wird eine Nachricht in das CEMT-Protokoll geschrieben, in der Sie informiert werden, wenn die Installation mit Fehlern abgeschlossen wurde.

Sie können den Befehl CEDA INSTALL über eine Konsole ausführen. Verwenden Sie dazu den MVS-Befehl MODIFY.

## Optionen

### **Group**(*gruppenname*)

Gibt den Namen der Gruppe an, die installiert werden soll. Alternativ wird der Name der Gruppe angegeben, die die zu installierende Ressource enthält. Ein generischer Gruppenname wird nicht akzeptiert.

### **List**(*listenname*)

Gibt eine Liste der Gruppen an, die installiert werden sollen. Alternativ wird eine Liste der Gruppen angegeben, die die zu installierende Ressource enthalten. Ein generischer Listenname wird nicht akzeptiert.

### *ressource*(*name*)

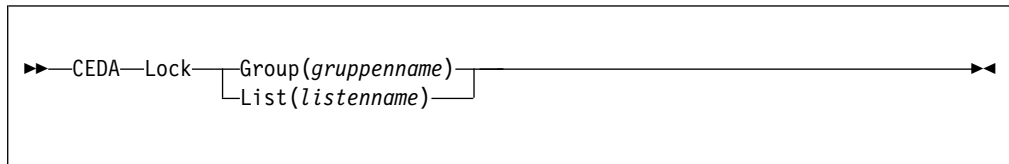
Gibt den Typ und den Namen der Ressource an, die installiert werden soll.

---

## Befehl CEDA LOCK

Mit dem Befehl **CEDA LOCK** können Sie die Zugriffsberechtigung für Aktualisierungs- und Löschoperationen auf einzelne Bediener-IDs beschränken.

## Syntax



## Beschreibung

Die Gruppe oder Liste kann von anderen RDO-Benutzern verwendet, angezeigt und kopiert werden, sie kann jedoch von diesen Benutzern nicht geändert oder gelöscht werden.

Mit LOCK können Sie eine nicht vorhandene Gruppe oder Liste sperren und die angegebene Gruppe oder Liste somit zur eigenen Verwendung in der Zukunft reservieren.

Der einzige RDO-Befehl, mit dem eine Sperre wieder freigegeben werden kann, ist der Befehl UNLOCK. Mit keinem der anderen RDO-Befehle können Gruppen oder Listen freigegeben werden. Wenn Sie z. B. alle Ressourcen in einer Gruppe oder alle Gruppen in einer Liste mit DELETE löschen, dann bleibt die Sperre dennoch erhalten.

Sie müssen entweder GROUP oder LIST angeben, auch wenn Sie die aktuelle Gruppe oder Liste sperren wollen.

Ein generischer Gruppen- oder Listenname wird nicht akzeptiert.

## Ressourcen sperren und wieder freigeben

Mit den Befehlen LOCK und UNLOCK können Sie den Aktualisierungszugriff auf eine Gruppe oder Liste steuern. Auf diese Weise können nur Bediener mit derselben Bediener-ID Änderungen durchführen.



Die Sperre gilt für die CSD-Datei und bleibt auch nach einem Neustart des CICS-Systems erhalten. Eigner der Sperre ist der Benutzer, der durch eine Kombination der generischen CICS-Anwendungs-ID (angegeben im Systeminitialisierungsparameter APPLID) und der Bediener-ID (OPIDENT) des Benutzers identifiziert wird.

OPIDENT wird dem Benutzer zugeordnet, wenn sich dieser bei dem für RDO verwendeten Terminal anmeldet. Weitere Informationen zu OPIDENT finden Sie in The CICS segment. Benutzer, die nicht angemeldet sind oder über eine andere OPIDENT verfügen, dürfen keine Operationen ausführen, die Änderungen an der gesperrten Gruppe zur Folge hätten. Allerdings ist jeder Benutzer dazu berechtigt, für eine gesperrte Gruppe folgende Befehle auszuführen:

- CHECK
- COPY
- DISPLAY
- INSTALL
- VIEW

Die Sperre kann mithilfe des Befehls UNLOCK aufgehoben werden, jedoch nur von einem Benutzer auf demselben System, der über dieselbe Bediener-ID verfügt.

## Optionen

### **Group**(*gruppenname*)

Gibt die Gruppe an, die gesperrt werden soll.

### **List**(*listenname*)

Gibt die Liste an, die gesperrt werden soll.

## Beispiele

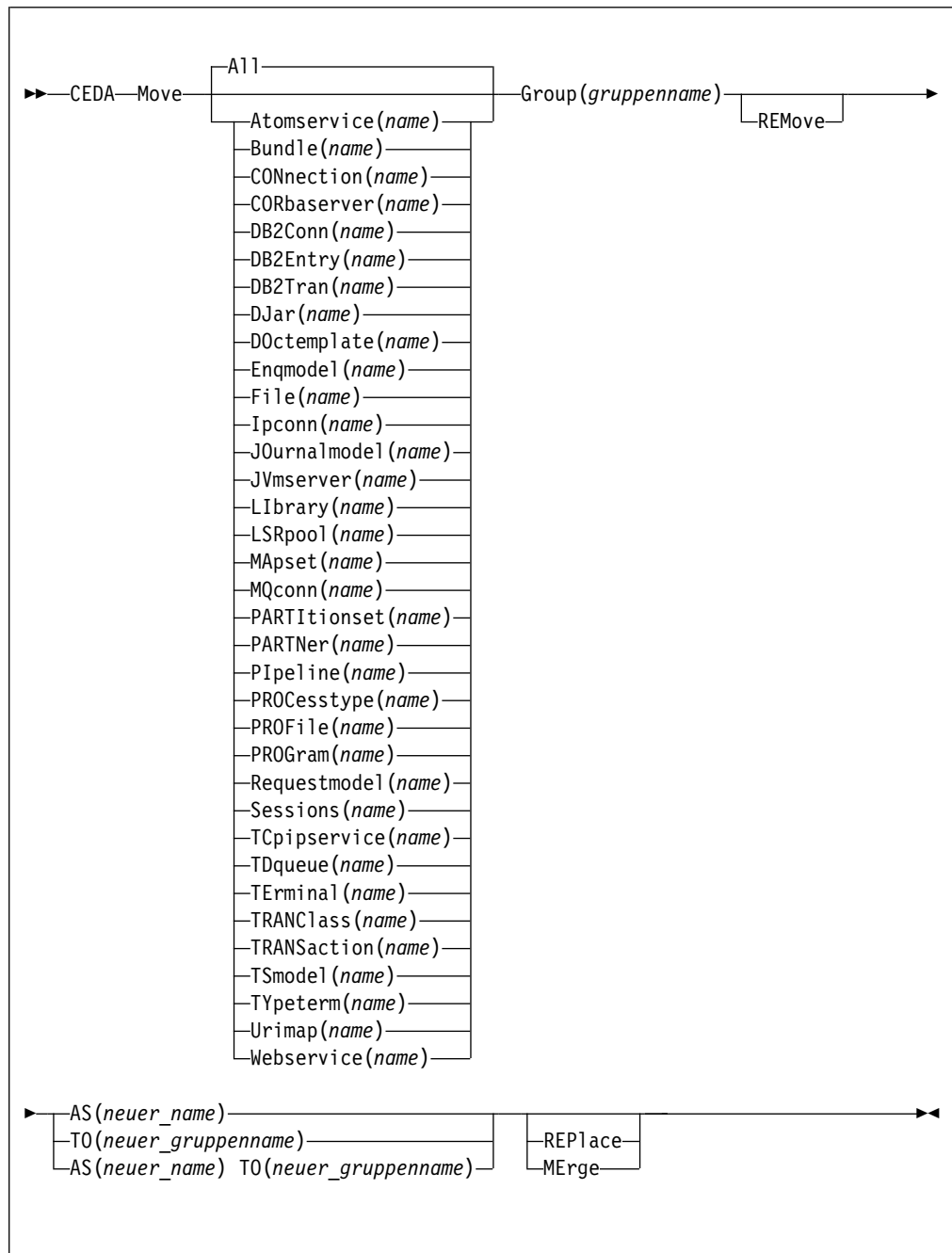
- Geben Sie Folgendes ein, um die Liste L1 zu sperren:  
LOCK LIST(L1)
- Geben Sie Folgendes ein, um die Gruppe G1 zu sperren:  
LOCK GROUP(G1)

---

## Befehl CEDA MOVE

Mit dem Befehl **CEDA MOVE** können Sie einzelne oder mehrere Ressourcendefinitionen von einer Gruppe in eine andere verschieben.

## Syntax



## Beschreibung

Mit diesem Befehl können Sie die Ressourcendefinitionen von einer in der Option GROUP angegebenen Gruppe in die Gruppe verschieben, die in der Option TO angegeben wird.

Wenn Sie die letzte Ressource in einer Gruppe mit dem Befehl MOVE in die andere, in TO angegebene Gruppe verschieben, dann wird die Gruppe automatisch gelöscht. Es dürfen keine leeren Gruppen existieren.

Wenn Sie weder MERGE noch REPLACE angeben, werden Sie in einer Nachricht gewarnt, dass Sie versuchen, doppelte Ressourcendefinitionen zu erstellen. Die Definitionen werden nicht verschoben.

## Optionen

### **AS**(*neuer\_name*)

Wenn Sie eine Definition innerhalb einer bestimmten Gruppe verschieben möchten, dann müssen Sie die Option AS verwenden, um sie umzubenennen. Sie können AS auch verwenden, um eine Definition in eine andere Gruppe zu verschieben und sie gleichzeitig umzubenennen. Bei Verwendung von AS sind generische Namen nicht zulässig.

### **Group**(*gruppenname*)

Gibt die Gruppe an, die die zu verschiebenden Definitionen enthält.

### **Merge**

Diese Option gilt, wenn doppelte Definitionsnamen in den Gruppen enthalten sind, die im Befehl MOVE angegeben wurden. Wenn Sie MERGE angeben, dann werden doppelt vorhandene Definitionen in der TO-Gruppe nicht ersetzt.

### **REMove**

Gibt an, dass eine Gruppe, die gelöscht wird, weil die letzte Ressource in der Gruppe an eine andere Position verschoben wurde, aus allen Listen entfernt werden muss, in denen sie enthalten war.

### **REPlace**

Diese Option gilt, wenn doppelte Definitionsnamen in den Gruppen enthalten sind, die im Befehl MOVE angegeben wurden. Wenn Sie REPLACE angeben, dann ersetzen die zu verschiebenden Definitionen die Definitionen, die in der im Operanden TO angegebenen Gruppe enthalten sind.

### *ressource*(*name*)

Gibt den Typ und den Namen der Ressource an, die verschoben werden soll. Der Standardwert lautet ALL und er bewirkt, dass alle Ressourcendefinitionen in einer Gruppe in eine andere Gruppe verschoben werden.

### **TO**(*neuer\_gruppenname*)

Sie können Definitionen in eine andere Gruppe verschieben und dabei im Operanden TO die neue Gruppe angeben.

## Beispiele

- Wenn Sie eine einzelne Ressourcendefinition verschieben, können Sie sie gleichzeitig umbenennen. Verwenden Sie dazu die Option AS, um den neuen Namen anzugeben. Beispiel:

```
MOVE PARTITIONSET(PSETQ1) GROUP(PSET1) AS(PSETQ4)
TO(PSET2)
```

- Sie können einen generischen Ressourcendefinitionsnamen angeben, um alle oder einzelne Definitionen desselben Ressourcentyps zu verschieben. Beispiel:

```
MOVE TRANSACTION(*) GROUP(DENTRY) TO(TEST1)
MOVE MAPSET(ACCT+++) GROUP(ACCOUNTS1) TO(ACCOUNTS2)
```

- Um alle Ressourcendefinitionen in einer Gruppe in eine neue Gruppe zu verschieben, können Sie die Option ALL verwenden. Beispiel:

```
MOVE ALL GROUP(N21TEST) TO(N21PROD)
```

- Die Option ALL kann mit einem generischen Namen verwendet werden, um alle den Kriterien des generischen Namens entsprechende Ressourcendefinitionen in der Gruppe in die neue Gruppe zu verschieben. Beispiel:

```
MOVE ALL(N21*) GROUP(N21OLD) TO(N21NEW)
```

- Die Option ALL kann mit einem spezifischen Namen verwendet werden, um alle in der Gruppe enthaltenen Ressourcendefinitionen mit diesem Namen (die zwingend unterschiedliche Ressourcentypen aufweisen müssen) in die neue Gruppe zu verschieben. Beispiel:

```
MOVE ALL(XMPL) GROUP(EXAMPLE) TO(EXAMPLE2)
```

- Geben Sie Folgendes ein, um Definitionen aus Gruppe X mit den Definitionen aus Gruppe Y zusammenzuführen und im Falle von Duplikaten die Versionen aus Gruppe Y beizubehalten.

```
MOVE GROUP(X) TO(Y) MERGE
```

- Geben Sie Folgendes ein, um Definitionen aus Gruppe X mit den Definitionen aus Gruppe Y zu kombinieren und im Falle von Duplikaten die Versionen aus Gruppe X beizubehalten:

```
MOVE GROUP(X) TO(Y) REPLACE
```

---

## Befehl CEDA REMOVE

Mit dem Befehl **CEDA REMOVE** können Sie den Namen einer Gruppe aus einer Liste entfernen.

### Syntax

►—CEDA—Remove—Group(*gruppenname*)—LIST(*listenname*)—►

### Beschreibung

Die Gruppe und alle darin enthaltenen Ressourcendefinitionen bleiben in der CSD-Datei weiterhin erhalten. Wenn die letzte Gruppe aus einer Liste entfernt wird, dann ist die Liste in der CSD-Datei nicht mehr vorhanden.

Ein generischer Listenname wird nicht akzeptiert.

Ein generischer Gruppenname kann angegeben werden, um eine große Anzahl von Gruppen oder alle Gruppen mithilfe nur eines Befehls aus einer Liste zu entfernen.

Beim Löschen einer Gruppe kann der Benutzer anfordern, dass die Gruppe aus allen Listen entfernt werden soll, in denen die Gruppe enthalten war. Wenn die letzte Gruppe aus einer Liste entfernt wird, wird die Liste gelöscht.

### Optionen

**Group**(*gruppenname*)

Gibt die Gruppe an, die entfernt werden soll.

**List**(*listenname*)

Gibt die Liste an, aus der die Gruppe entfernt werden soll.

### Beispiele

- Eine Liste mit dem Namen LL02 enthält die folgenden Gruppen:  
 G001  
 X001  
 XG001  
 G002

G003  
X002  
G004

Geben Sie Folgendes ein, um alle Gruppen zu entfernen, deren Name mit G beginnt:

```
REMOVE GROUP(G*) LIST(LL02)
```

Damit verbleiben die folgenden Gruppen in der Liste:

X001  
XG001  
X002

- Geben Sie Folgendes ein, um die Liste vollständig zu entfernen:

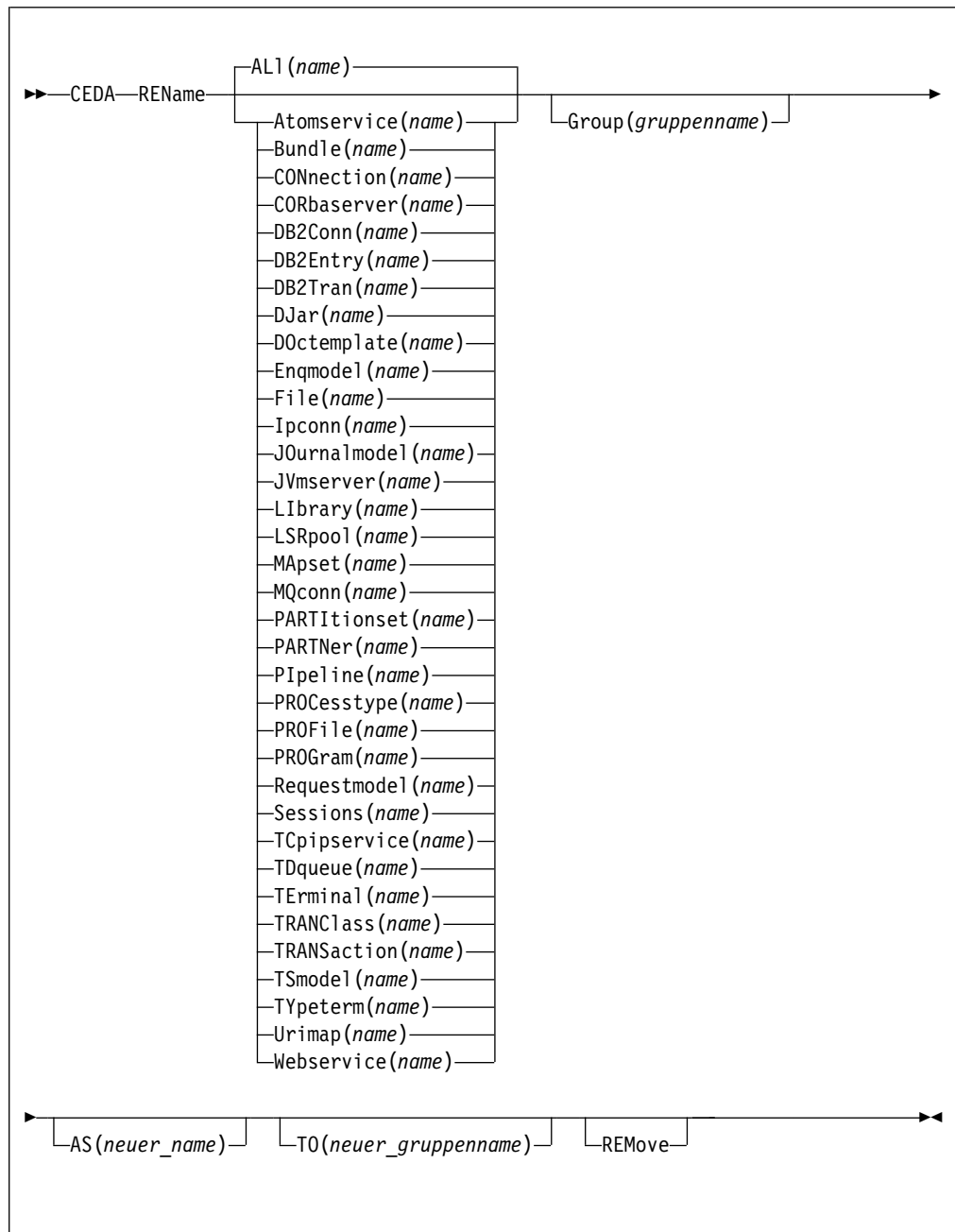
```
REMOVE GROUP(*) LIST(LL02)
```

---

## Befehl CEDA RENAME

Mit dem Befehl **CEDA RENAME** können Sie den Namen einer Ressource in der CSD (CICS-Systemdefinition) ändern oder die Ressource in eine andere Gruppe verschieben.

## Syntax



## Beschreibung

Eine Ressourcendefinition kann auch mithilfe des Befehls `DISPLAY` oder des Befehls `EXPAND` unter Angabe der Option `RENAME` umbenannt werden. Informationen zu diesen Befehlen finden Sie in „Befehl `CEDA DISPLAY`“ auf Seite 350 und „Befehl `CEDA EXPAND`“ auf Seite 353.

Einer Ressourcendefinition kann bei einer Umbenennung kein Name zugeordnet werden, der in der Zielgruppe bereits vorhanden ist.

## Optionen

### **AS**(*neuer\_name*)

Wenn Sie eine Definition innerhalb einer bestimmten Gruppe kopieren möchten, dann müssen Sie die Option AS verwenden, um sie umzubenennen. Sie können AS auch verwenden, um eine Definition in eine andere Gruppe zu verschieben und sie gleichzeitig umzubenennen. Bei Verwendung von AS sind generische Namen nicht zulässig.

### **Group**(*gruppenname*)

Gibt die Gruppe an, die die umzubenennenden oder zu verschiebenden Definitionen enthält. Ein generischer Gruppenname wird nicht akzeptiert.

### **REMove**

Gibt an, dass eine Gruppe, die gelöscht wird, aus allen Listen entfernt werden muss, in denen sie enthalten war.

### **ressource**(*name*)

Gibt den Typ und den Namen der Ressource an, die verschoben oder umbenannt werden soll. Der Standardwert lautet ALL und er bewirkt, dass alle Ressourcendefinitionen in einer Gruppe in eine andere Gruppe kopiert werden. Ein generischer Ressourcendefinitionsname wird nicht akzeptiert.

### **TO**(*neuer\_gruppenname*)

Sie können Definitionen in eine andere Gruppe verschieben und dabei im Operanden TO die neue Gruppe angeben.

## Beispiele

- Geben Sie Folgendes ein, um eine Ressource umzubenennen und sie in der aktuellen Gruppe zu belassen:

```
RENAME PROFILE(PROF1) AS(NEWPROF) GROUP(PROFS)
```

- Sie können alle Ressourcendefinitionen mit dem gleichen Namen unter Angabe eines neuen Namens umbenennen. Verwenden Sie dazu anstelle eines Ressourcentyps die Option ALL. Beispiel:

```
RENAME ALL(TVA) AS(XTVA) GROUP(XTVA1)  
RENAME ALL(USER) AS(OLDU) GROUP(USERDEF)
```

- Sie können eine Ressourcendefinition in eine neue Gruppe verschieben, die Sie in der Option TO angeben, und sie gleichzeitig umbenennen. (Dieser Arbeitsschritt kann auch mit dem Befehl MOVE ausgeführt werden.) Beispiel:

```
RENAME PROGRAM(N20ZA) AS($SOSERR) GROUP(N20) TO($MODULES)
```

- Sie können alle Ressourcendefinitionen mit dem gleichen Namen mit der Option TO aus einer Gruppe in eine andere Gruppe verschieben und sie gleichzeitig umbenennen. Beispiel:

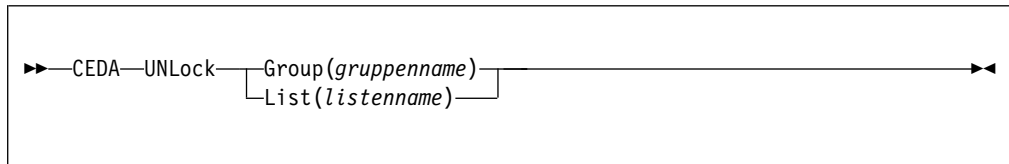
```
RENAME ALL(USER) GROUP(USERDEF) AS(TEMP) TO(TEMPGRP)
```

---

## Befehl CEDA UNLOCK

Mit dem Befehl **CEDA UNLOCK** können Sie die Sperre einer Gruppe oder Liste mit Definitionen aufheben.

## Syntax



## Beschreibung

Beim Befehl UNLOCK handelt es sich um den einzigen RDO-Befehl, mit dem eine Sperre für eine Liste oder Gruppe aufgehoben werden kann, die mit dem RDO-Befehl LOCK definiert wurde.

Mit dem Befehl UNLOCK können Sie eine nicht existierende Gruppe oder Liste entsperren.

Sie müssen entweder die Option GROUP oder LIST angeben, auch wenn Sie die aktuelle Gruppe oder Liste entsperren, da der Befehl UNLOCK mit beiden verwendet werden kann.

## Ressourcen sperren und wieder freigeben

Mit den Befehlen LOCK und UNLOCK können Sie den Aktualisierungszugriff auf eine Gruppe oder Liste steuern. Auf diese Weise können nur Bediener mit derselben Bediener-ID Änderungen durchführen.

Die Sperre gilt für die CSD-Datei und bleibt auch nach einem Neustart des CICS-Systems erhalten. Eigner der Sperre ist der Benutzer, der durch eine Kombination der generischen CICS-Anwendungs-ID (angegeben im Systeminitialisierungsparameter APPLID) und der Bediener-ID (OPIDENT) des Benutzers identifiziert wird.

OPIDENT wird dem Benutzer zugeordnet, wenn sich dieser bei dem für RDO verwendeten Terminal anmeldet. Weitere Informationen zu OPIDENT finden Sie in The CICS segment. Benutzer, die nicht angemeldet sind oder über eine andere OPIDENT verfügen, dürfen keine Operationen ausführen, die Änderungen an der gesperrten Gruppe zur Folge hätten. Allerdings ist jeder Benutzer dazu berechtigt, für eine gesperrte Gruppe folgende Befehle auszuführen:

- CHECK
- COPY
- DISPLAY
- INSTALL
- VIEW

Die Sperre kann mithilfe des Befehls UNLOCK aufgehoben werden, jedoch nur von einem Benutzer auf demselben System, der über dieselbe Bediener-ID verfügt.

## Optionen

### **Group(gruppenname)**

Gibt die Gruppe an, die entsperrt werden soll. Ein generischer Gruppenname wird nicht akzeptiert.

### **List(listenname)**

Gibt die Liste an, die entsperrt werden soll. Ein generischer Listenname wird nicht akzeptiert.



## Beispiele

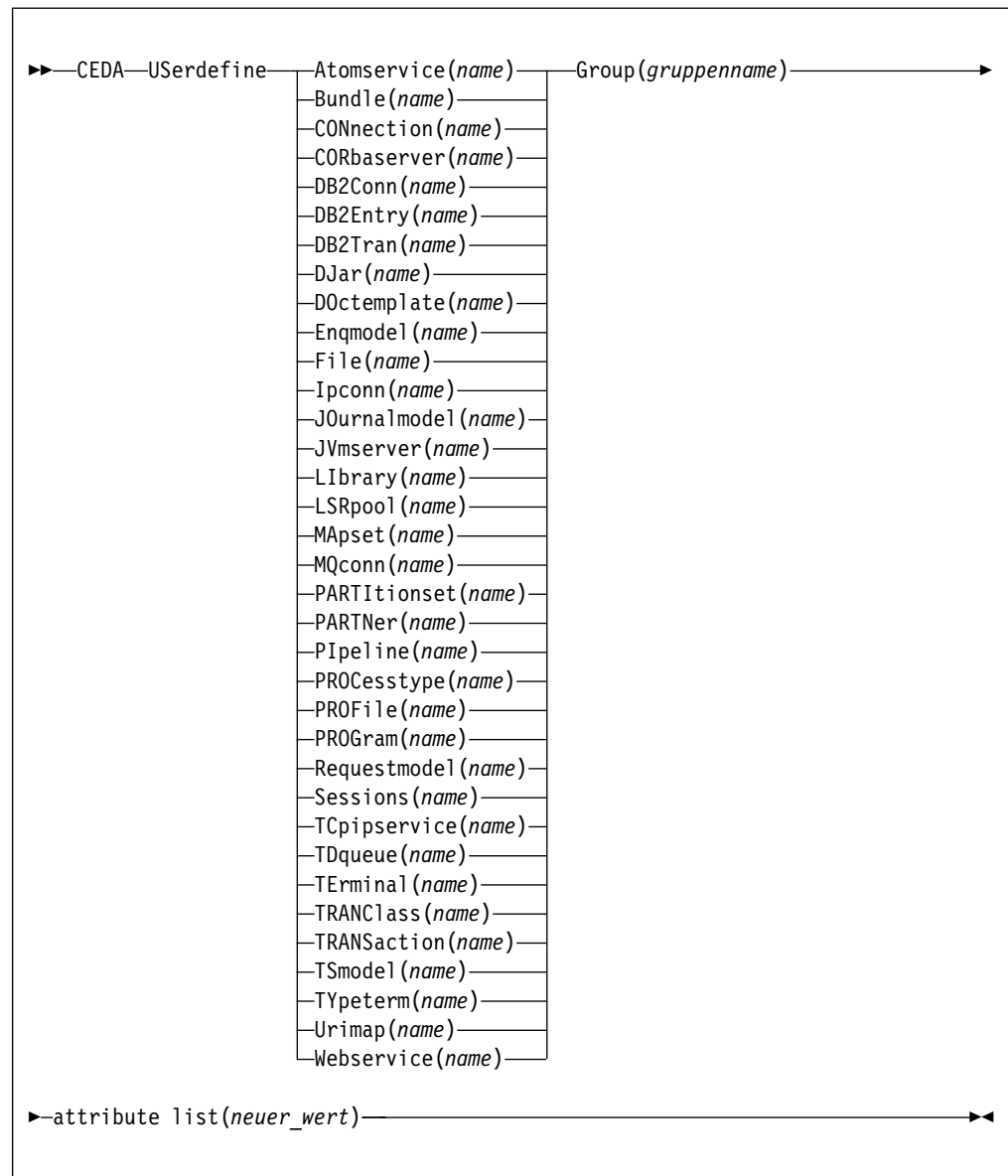
- Geben Sie Folgendes ein, um die Gruppe G1 zu entsperren:  
UNLOCK GROUP(G1)
- Geben Sie Folgendes ein, um die Liste L1 zu entsperren:  
UNLOCK LIST(L1)

---

## Befehl CEDA USERDEFINE

Mit dem Befehl **CEDA USERDEFINE** können Sie eine neue Ressourcendefinition in der CSD (CICS-Systemdefinition) mit den benutzerdefinierten Standardattributen erstellen.

### Syntax



## Beschreibung

Beim Befehl `USERDEFINE` handelt es sich um eine Alternative zum Befehl `DEFINE`. Anstatt der von CICS bereitgestellten Standardwerte benutzt der Befehl `USERDEFINE` Ihre eigenen Standardeinstellungen. In jeder anderen Hinsicht arbeitet der Befehl genau wie der Befehl `DEFINE`.

Um eigene Standardwerte einzurichten, müssen Sie mit `DEFINE` eine Pseudoressourcendefinition mit dem Namen `USER` in einer Gruppe mit dem Namen `USERDEF` erstellen. Jede Pseudoressourcendefinition muss vollständig sein. Dies bedeutet, dass eine Transaktionsdefinition beispielsweise eine Programmdefinition angeben muss, und zwar auch dann, wenn Sie immer einen Programmnamen angeben, wenn Sie mit `USERDEFINE` eine Transaktion definieren. Die Pseudoressourcendefinitionen müssen nicht vor Verwendung von `USERDEFINE` installiert werden.

Führen Sie diesen Arbeitsschritt für jeden Ressourcentyp aus, für den Sie Standardwerte festlegen wollen. Obwohl alle Definitionen in der Gruppe den gleichen Namen (`USER`) aufweisen, sind alle Definitionen individuell verschieden, weil ein jeweils unterschiedlicher Ressourcentyp definiert wird.

## Optionen

### **attribute list**

Die Attributliste ist vom definierten Ressourcentyp abhängig. Bestimmte Ressourcen verfügen über Attribute, die in die Definition eingebunden werden müssen. Für Attribute, für die von Ihnen kein Wert angegeben wird, werden die entsprechenden Standardwerte verwendet.

### **Group(*gruppenname*)**

Der Name der zu definierenden Gruppe.

## Beispiele

Die Assemblerprogrammierer einer Installation haben eine Pseudoprogrammdefinition mit dem Namen `USER` erstellt und Assembler als Standardsprache festgelegt. Sie verwenden den Befehl `USERDEFINE`, um ihre Programme in CICS zu definieren.

1. Zuerst muss hierzu ein Programm mit dem Namen `USER` in der Gruppe `USERDEF` definiert werden. Hierzu können Sie den folgenden Befehl eingeben:

```
CEDA DEFINE PROGRAM(USER) GROUP(USERDEF)
```

In der folgenden Abbildung wird die Anzeige dargestellt, die nach Eingabe dieses Befehls aufgerufen wird:

```

DEFINE PROGRAM(USER)
GROUP(USERDEF)          CICS RELEASE = 0620
OVERTYPE TO MODIFY
CEDA DEFINE
  PROGram      : USER
  Group        : USERDEF
  Description   ==>
  Language     ==>          CObo1 | Assembler | Le370 | C | P1i
  RELoad      ==> No        No | Yes
  RESident    ==> No        No | Yes
  USAge       ==> Normal    Normal | Transient
  USElpacopy   ==> No        No | Yes
  Status      ==> Enabled    Enabled | Disabled
  RSI         : 00          0-24 | Public
  Cedf        ==> Yes        Yes | No
  DAtalocation ==> Any        Below | Any
I New group USERDEF created

                                SYSID=ABCD  APPLID=DBDCCICS
                                TIME: 11.24.39  DATE: 97.359
                                PF 1 HELP 2 COM 3 END          6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL

```

2. Geben Sie für die Option LANGUAGE die Einstellung ASSEMBLER an und drücken Sie dann die Eingabetaste.

```

OVERTYPE TO MODIFY
CEDA USerdefine
  PROGram      : USER
  Group        : USERDEF
  Description   ==>
  Language     ==> Assembler CObo1 | Assembler | Le370 | C | P1i
  RELoad      ==> No        No | Yes
  RESident    ==> No        No | Yes
  USAge       ==> Normal    Normal | Transient
  USElpacopy   ==> No        No | Yes
  Status      ==> Enabled    Enabled | Disabled
  RSI         : 00          0-24 | Public
  Cedf        ==> Yes        Yes | No
  DAtalocation ==> Any        Below | Any

                                SYSID=ABCD  APPLID=DBDCCICS
                                TIME: 11.24.41  DATE: 97.359
                                PF 1 HELP 2 COM 3 END          6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL

```

Wenn Sie nun ein neues Programm definieren möchten, dann können Sie den Befehl USERDEFINE verwenden, um den Standardwert ASSEMBLER automatisch abzurufen. Wenn Sie ein neues Programm P2 in der Gruppe GRP definieren möchten, dann müssen Sie den folgenden Befehl eingeben:

```
CEDA USERDEFINE PROGRAM(P2) GROUP(GRP)
```

In der folgenden Abbildung ist die Anzeige dargestellt, die nach Eingabe dieses Befehls aufgerufen wird.

```

USERDEFINE PROGRAM(P2)
GROUP(GRP)
OVERTYPE TO MODIFY
CEDA Userdefine
  PROGram      : P2
  Group        : GRP
  Description   ==>
  Language      ==> Assembler      CObo1 | Assembler | Le370 | C | Pl1
  REload        ==> No              No | Yes
  RESident      ==> No              No | Yes
  USAge         ==> Normal          Normal | Transient
  USElpacopy    ==> No              No | Yes
  Status        ==> Enabled         Enabled | Disabled
  RS1           : 00               0-24 | Public
  Cedf          ==> Yes             Yes | No
  DAtallocation ==> Any             Below | Any
                                           APPLID=DBDCCICS
USERDEFINE SUCCESSFUL                TIME: 11.25.48  DATE: 97.359
PF 1 HELP 2 COM 3 END                6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL

```

Wie in der Abbildung ersichtlich, wurde für das Attribut LANGUAGE die Option ASSEMBLER eingefügt. Sie können die Werte der Optionen in dieser Anzeige wie in der Anzeige des Befehls DEFINE überschreiben, um die Definition abzuschließen.

Nachdem Sie die eigenen Standardwerte in einer Ressourcendefinition USER festgelegt haben, dann erhält jeder Benutzer, der den Befehl USERDEFINE für diesen Ressourcentyp verwendet, diese Standardwerte.

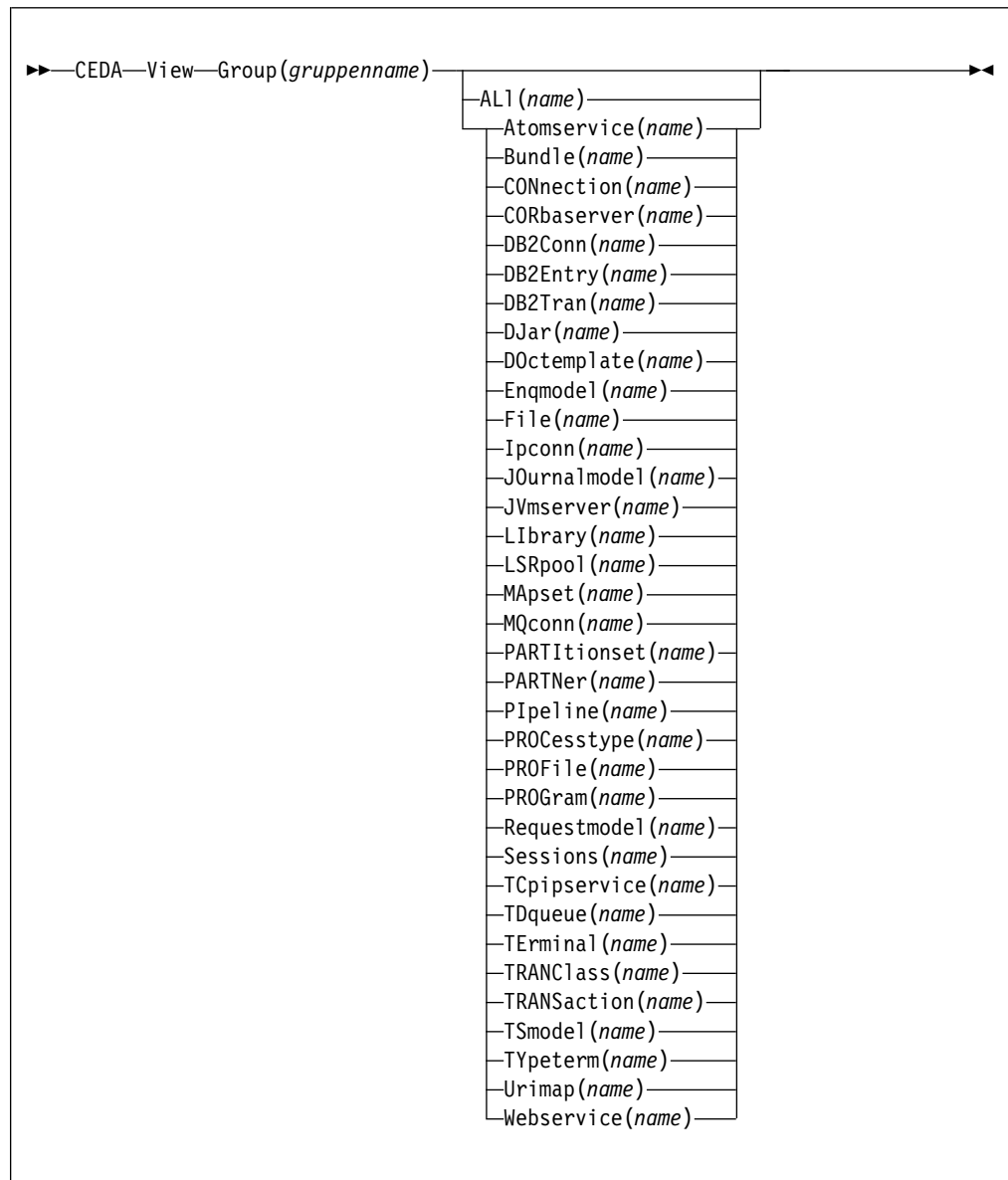
Wenn Sie USER umbenennen und eine eigene Pseudoressourcendefinition definieren, können Sie eigene Standardwerte benutzen. Normalerweise werden die Standardwerte in Ihrer Installation allerdings aus Gründen der Standardisierung vereinbart und es wird mit LOCK eine Sperre für die Gruppe USERDEF definiert.

---

## Befehl CEDA VIEW

Mit dem Befehl **CEDA VIEW** können Sie die Attribute einer Ressourcendefinition in der CSD (CICS-Systemdefinition) anzeigen.

## Syntax



## Beschreibung

Mit dem Befehl VIEW können Sie die Ressourcendefinitionsattribute in derselben Weise anzeigen wie mit dem Befehl ALTER. Allerdings ist es nicht möglich, die Definitionen zu aktualisieren.

## Optionen

### Group (gruppenname)

Gibt die Gruppe an, die angezeigt werden soll. Wenn kein Name angegeben wird, dann nimmt das System die aktuelle Gruppe an.

## Beispiele

```
VIEW TERMINAL(SZT1) GROUP(ZEMTERMS)
```

```
VIEW MAPSET(N20MAP01) GROUP(N20)
```



---

## Kapitel 3. DFHCSDUP-Befehle des Dienstprogramms zur Ressourcenverwaltung

Im vorliegenden Abschnitt werden die Befehle beschrieben, die im Dienstprogramm DFHCSDUP zur Verfügung stehen. Befehle können abgekürzt werden, aber die zulässige Mindestabkürzung weicht von den Abkürzungen bestimmter CEDA-Befehle ab.

---

### DFHCSDUP-Befehl ADD

Gruppe zu einer Liste hinzufügen.

#### Syntax von ADD

```
►►—Add—Group—(—gruppenname—)—LIst—(—listenname—)—————►
|
|   After—(—gruppenname2—)—
|   Before—(—gruppenname3—)—
|_____►
```

#### Optionen

##### **After**(*gruppenname2*)

Geben Sie AFTER an, um den neuen Gruppennamen nach dem vorhandenen Gruppennamen zu platzieren. Wenn BEFORE oder AFTER nicht angegeben wird, dann wird der Gruppename am Ende der Liste hinzugefügt.

##### **Before**(*gruppenname3*)

Geben Sie BEFORE an, um den neuen Gruppennamen vor dem vorhandenen Gruppennamen zu platzieren. Wenn BEFORE oder AFTER nicht angegeben wird, dann wird der Gruppename am Ende der Liste hinzugefügt.

##### **Group**(*gruppenname*)

Gibt den Namen der Gruppe an, die hinzugefügt werden soll. Der Name darf in der Liste noch nicht vorhanden sein. Ein generischer Gruppename wird nicht akzeptiert.

##### **LIst**(*listenname*)

Gibt den Namen der Liste an, zu der die Gruppe hinzugefügt werden soll. Ist die Liste noch nicht vorhanden, wird eine neue Liste erstellt. Ein generischer Listenname wird nicht akzeptiert.

#### Beispiele

Geben Sie Folgendes an, um eine Liste mit dem Namen LA01 zu erstellen, indem Sie eine Gruppe zu dieser Liste hinzufügen:

```
ADD GROUP(GA001) LIST(LA01)
```

Geben Sie Folgendes an, um eine weitere Gruppe zur Liste LA01 hinzuzufügen, ohne dabei anzugeben, wo die Gruppe hinzugefügt werden soll:

```
ADD GROUP(GA002) LIST(LA01)
```

Die Liste LA01 hat nun folgendes Format:

- GA001
- GA002

Geben Sie Folgendes an, um eine weitere Gruppe am Anfang der Liste hinzuzufügen:

```
ADD GROUP(GA003) LIST(LA01) BEFORE(GA001)
```

Geben Sie Folgendes an, um eine weitere Gruppe zwischen GA001 und GA002 hinzuzufügen:

```
ADD GROUP(GA004) LIST(LA01) AFTER(GA001)
```

Die Liste LA01 hat nun folgendes Format:

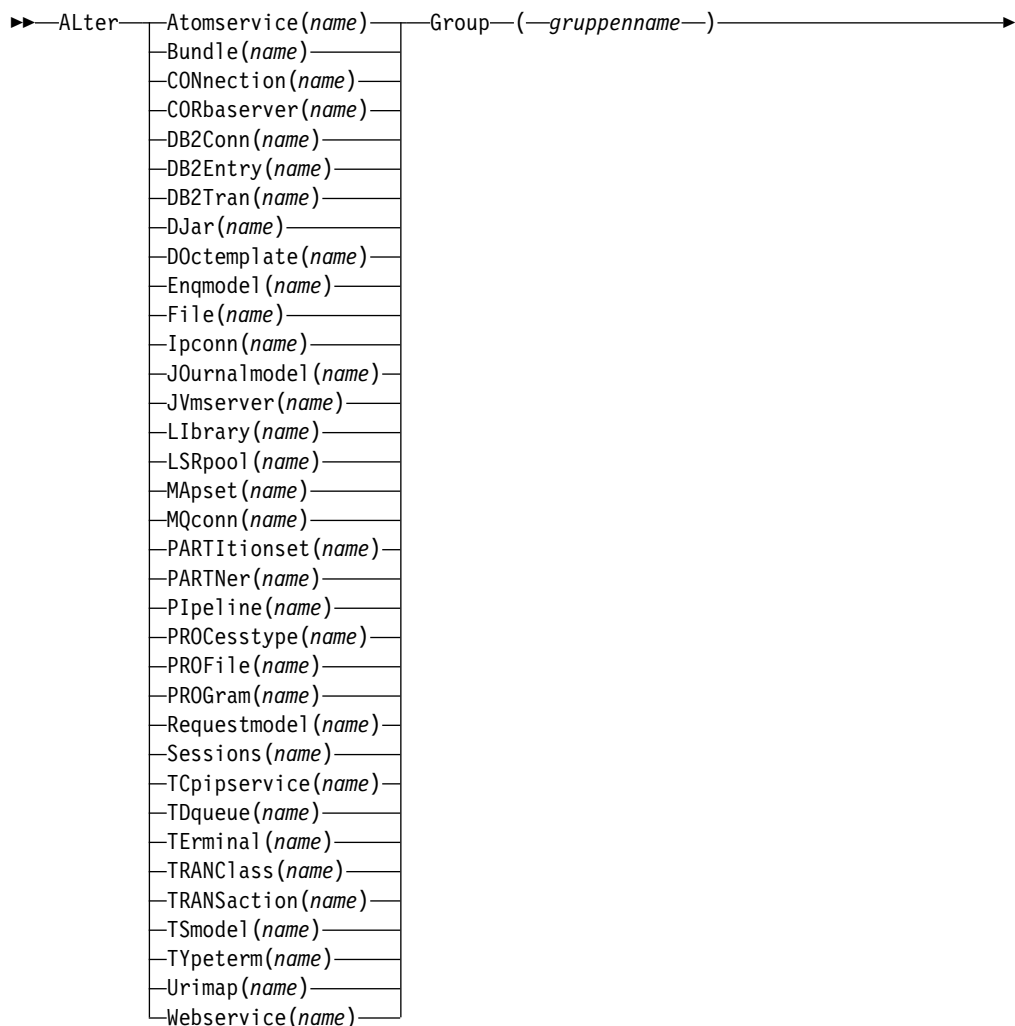
- GA003
- GA001
- GA004
- GA002

---

## DFHCSDUP-Befehl ALTER

Einzelne oder alle Attribute einer vorhandenen Ressourcendefinition ändern.

### Syntax von ALTER





## Beschreibung

Informationen zu den Attributen, die im Befehl ALTER für die unterschiedlichen Ressourcentypen angegeben werden können, sowie eine Beschreibung der Attribute und Standardwerte aller Ressourcentypen finden Sie in RDO resources.

Verwenden Sie ALTER nicht, um den Wert der Attribute einer TYPETERM-Definition zu ändern, von der andere Attribute abhängig sind. Wenn Ihnen bei der Angabe für DEVICE, SESSIONTYPE oder TERMMODEL ein Fehler unterläuft, dann löschen Sie die Definition und erstellen Sie eine neue Definition mit den korrekten Werten.

Sie können Nullattributwerte angeben. Beispiel:

```
ALTER FILE(TEST) GROUP(ACT1) DESCRIPTION()
```

Wenn ein Attribut, für das Sie einen Nullwert angegeben haben, über einen Standardwert verfügt, dann hängt der verwendete Wert vom jeweiligen Feldtyp ab:

- Im Befehl

```
ALTER FILE(TEST) GROUP(ACT1) RLSACCESS() DESCRIPTION()
```

wird für RLSACCESS der Standardwert NO verwendet und die Beschreibung wird ausgeblendet.

- Im Befehl

```
ALTER FILE(TEST) GROUP(ACT1) PROFILE()
```

wird für das Feld PROFILE der Standardwert DFHCICSA verwendet.

Änderungen an Ressourcendefinitionen in der CSD-Datei werden in einem momentan aktiven CICS-System erst nach der Installation der Gruppe wirksam, in der sich die Ressourcendefinition befindet.

## Generische Benennung im Befehl ALTER

Der Befehl ALTER akzeptiert sowohl generische Ressourcennamen als auch Gruppennamen.

Für jede Ressource in der CSD-Datei, die mit der angegebenen Kombination aus Ressourcenname und Gruppenname übereinstimmt, versucht das System, den Befehl ALTER auszuführen. Schlägt die Ausführung eines einzelnen Befehls ALTER fehl, dann wird die Verarbeitung beendet, wenn alle Versuche zur Ausführung des Befehls verarbeitet wurden.

## Optionen

### Attribute list

Gibt die Attribute an, die geändert werden sollen.

### Group(*gruppenname*)

Gibt den Namen der Gruppe an, die die Ressource enthält, die geändert werden soll.

**Resource**(*name*)

Gibt die Ressource an, deren Attribute geändert werden sollen. Sie können einen generischen Namen angeben, indem Sie die Zeichen + und \* verwenden.

**Beispiele**

Geben Sie Folgendes ein, um ein Programm als resident zu definieren:

```
ALTER PROGRAM(ERR01) GROUP(GENMODS) RESIDENT(YES)
    DATALOCATION()
```

Geben Sie Folgendes ein, um alle Programme in der Gruppe GENMOD als resident zu definieren:

```
ALTER PROGRAM(*) GROUP(GENMOD) RESIDENT(YES)
    DATALOC()
```

---

**DFHCSDUP-Befehl APPEND**

In einer Liste enthaltene Gruppen am Ende einer anderen Liste hinzufügen.

**Syntax von APPEND**

```
►►—Append—FRomcsd—(—dd-name—)—List—(—listenname1—)——————►
►—To—(—listenname2—)——————►◄
```

**Beschreibung**

In einer Liste sind keine doppelten Gruppennamen zulässig. Wenn DFHCSDUP während der APPEND-Operation doppelte Namen findet, dann werden diese Namen ignoriert und nicht angefügt. In diesem Fall enthält die DFHCSDUP-Ausgabefliste eine Warnnachricht.

**Anmerkung:** Wenn Sie eine Anfügeoperation von einer CSD-Datei in eine andere durchführen, dann sollten Sie berücksichtigen, dass mit diesem Befehl nicht die Gruppen selbst kopiert werden. Hierzu müssen Sie einen separaten Befehl COPY ausführen.

**Optionen****FRomcsd**(*dd-name*)

Gibt den DD-Namen der sekundären CSD-Datei an, aus der Sie *listenname1* anfügen.

**List**(*listenname1*)

Gibt den Namen der Liste an, die angefügt wird. Verwenden Sie keinen generischen Listennamen.

Die Liste, für die die Anfügung durchgeführt wird, kann sich in der primären CSD-Datei oder in einer anderen CSD-Datei befinden. Wenn Sie eine Anfügung aus einer anderen CSD-Datei durchführen, dann müssen Sie diese Datei mithilfe des Parameters FROMCSD angeben.

**To**(*listenname2*)

Gibt den Namen der Liste an, an die die Gruppennamen angefügt werden sollen. Wenn Sie eine Anfügung aus einer anderen CSD-Datei durchführen, dann können Sie dieser Liste den gleichen Namen zuordnen wie der Quellenliste. Verwenden Sie keinen generischen Listennamen.

Ist diese Zielliste bereits vorhanden, dann wird die Quellenliste am Ende der Zielliste angefügt. Wenn die Zielliste noch nicht vorhanden ist, dann wird sie erstellt. (Tatsächlich kopieren Sie die Quellenliste.)

## Beispiele

Eine Liste mit dem Namen LISTA enthält die folgenden Gruppen:

- GB001
- GB002
- GB003

Eine Liste mit dem Namen LISTB enthält die folgenden Gruppen:

- G001
- G002
- G003

Fügen Sie LISTB wie folgt zu LISTA hinzu:

```
APPEND LIST(LISTB) TO(LISTA)
```

Daraufhin enthält LISTA die folgenden Gruppen in der hier angegebenen Reihenfolge:

- GB001
- GB002
- GB003
- G001
- G002
- G003

LISTB enthält weiterhin Folgendes:

- G001
- G002
- G003

---

## DFHCSDUP-Befehl COPY

Ressourcendefinitionen von einer Gruppe in eine andere Gruppe kopieren.

### Syntax von COPY

►►—Copy—Group—(—*gruppename1*—)—TO(*gruppename2*)—————►



## Beschreibung

Mit dem Befehl COPY werden Ressourcendefinitionen von **gruppenname1** nach **gruppenname2** kopiert. Die Gruppe, aus der die Ressourcendefinitionen kopiert werden sollen (*gruppenname1*), können sich in der primären CSD-Datei befinden, oder aber in der CSD-Datei, die im Parameter **FROMCSD** angegeben ist.

Mit dem DFHCSDUP-Befehl COPY können Sie einzelne Ressourcendefinitionen oder alle Ressourcendefinitionen in der Gruppe kopieren. Die Ressourcendefinitionen werden in eine Gruppe kopiert, die im Parameter TO (*gruppenname2*) in der Primärdatei angegeben ist. Wenn diese Gruppe bereits vorhanden ist, dann werden die Definitionen aus der Quellengruppe (*gruppenname1*) zu den Definitionen hinzugefügt, die bereits in der Gruppe *gruppenname2* enthalten sind. Wenn die im Parameter TO angegebene Gruppe noch nicht existiert, dann wird eine neue Gruppe mit dem entsprechenden Namen erstellt.

Wenn Sie den DFHCSDUP-Befehl COPY zum Kopieren aller Ressourcendefinitionen in einer Gruppe in eine andere Gruppe verwenden und wenn die Gruppen doppelte Ressourcenamen enthalten, dann müssen Sie entweder MERGE oder REPLACE angeben, um festzulegen, wie Duplikate behandelt werden sollen. Wenn Sie MERGE angeben, dann werden doppelte Definitionen in der TO-Gruppe nicht

ersetzt. Wenn Sie REPLACE angeben, dann ersetzen die zu kopierenden Definitionen die in der TO-Gruppe enthaltenen Definitionen. Wenn Sie weder MERGE noch REPLACE angeben und Duplikate gefunden werden, dann schlägt die Kopieroperation fehl.

Wenn Sie den DFHCSDUP-Befehl COPY verwenden, um eine einzelne Ressourcendefinition in eine andere Gruppe zu kopieren, dann wird eine in der TO-Gruppe eventuell vorhandene doppelte Definition standardmäßig nicht ersetzt. Wenn Sie REPLACE angeben, dann ersetzt die zu kopierende Definition die in der TO-Gruppe enthaltene Definition.

Die DFHCSDUP-Ausgabeliste gibt Aufschluss über die kopierten Definitionen und darüber, wie mit den doppelten Definitionen verfahren wurde.

## Generische Benennung im Befehl COPY

Der DFHCSDUP-Befehl COPY akzeptiert generische Gruppennamen in der Option GROUP und der Option TO. Hierbei gelten die folgenden Regeln:

- Das einzige generische Zeichen, das im Befehl COPY zulässig ist, ist der Stern (\*), der sich am Ende des Namens befinden muss.
- Die Präfixlänge von *gruppenname1* muss größer oder gleich der Präfixlänge von *gruppenname2* sein. COPY GROUP(DFHCOMP\*) TO(USRCMP\*) ist gültig, COPY GROUP(DFHCO\*) TO(USRCMP\*) hingegen nicht.

Sie können das Sternsymbol (\*) verwenden, um Elemente von einer Gruppe mit generischem Namen in eine andere Gruppe mit generischem Namen oder von einer Gruppe mit generischem Namen in eine spezielle Gruppe zu kopieren (siehe hierzu „Beispiele“ auf Seite 380).

Der DFHCSDUP-Befehl COPY akzeptiert keine generischen Namen für individuelle Ressourcendefinitionen, die Sie mithilfe einer der Ressourcenoptionen angeben.

## Optionen

### **FRomcsd**(*dd-name*)

Gibt den DD-Namen der sekundären CSD-Datei an, aus der Sie die Gruppe *gruppenname1* kopieren.

### **Group**(*gruppenname1*)

Gibt den Namen der Gruppe an, die kopiert werden soll. Sie können einen generischen Namen für die Gruppe angeben, indem Sie einen Stern (\*) verwenden. Die Standardaktion besteht darin, die gesamte Gruppe zu kopieren. Alternativ hierzu können Sie auch eine der Ressourcentypoptionen und den Namen einer einzelnen Ressourcendefinition dieses Typs angeben, die kopiert werden soll.

### **MErge**

Gibt an, dass die ursprünglichen Definitionen in *gruppenname2* beibehalten werden, wenn *gruppenname2* bereits existiert und doppelte Definitionen auftreten.

### **Repl**ace

Gibt an, dass die Definitionen in *gruppenname1* die Definitionen in *gruppenname2* ersetzen, wenn *gruppenname2* bereits existiert und doppelte Definitionen auftreten.

### *ressource*(*name*)

Gibt den Typ sowie den Namen einer einzelnen Ressourcendefinition an, deren

Attribute Sie kopieren möchten. Im DFHCSDUP-Befehl COPY ist es nicht möglich, generische Namen für einzelne Ressourcendefinitionen zu verwenden.

**To**(*gruppename2*)

Gibt den Namen der Gruppe an, in die alle Ressourcendefinitionen oder die einzeln angegebene Ressourcendefinition kopiert werden soll(en). Wenn das Kopieren aus einer anderen CSD-Datei erfolgen soll, dann können Sie dieser Gruppe den gleichen Namen zuordnen wie der Quellengruppe. Sie können einen generischen Namen für die Gruppe angeben, indem Sie einen Stern (\*) verwenden.

## Beispiele

Im folgenden Beispiel wird eine Gruppe mit dem Namen GA001 in eine Gruppe mit dem Namen GA002 kopiert, die bereits vorhanden ist. Dabei werden alle doppelten Ressourcendefinitionen durch die in der Gruppe GA001 vorhandenen Ressourcendefinitionen ersetzt.

```
COPY GROUP(GA001) TO(GA002) REPLACE
```

Im folgenden Beispiel wird die Gruppe GA003 in die Gruppe GA004 kopiert. Bei eventuell vorhandenen Duplikaten werden jedoch die Ressourcendefinitionen in Gruppe GA004 beibehalten.

```
COPY GROUP(GA003) TO(GA004) MERGE
```

Im folgenden Beispiel werden alle von CICS bereitgestellten Gruppen in vom Benutzer benannte Gruppen mit dem Präfix USR kopiert. Dies hat zur Folge, dass DFHOPER in USROPER, DFHSTAND in USRSTAND usw. umbenannt wird.

```
COPY GROUP(DFH*) TO(USR*)
```

Im folgenden Beispiel werden alle Gruppen, deren Name mit ABCD beginnt, in die Gruppe mit dem Namen NEWGROUP kopiert:

```
COPY GROUP(ABCD*) TO(NEUGROUP)
```

Im folgenden Beispiel wird eine einzelne Ressourcendefinition von der Gruppe CICSC1 in die Gruppe CICSC2 kopiert:

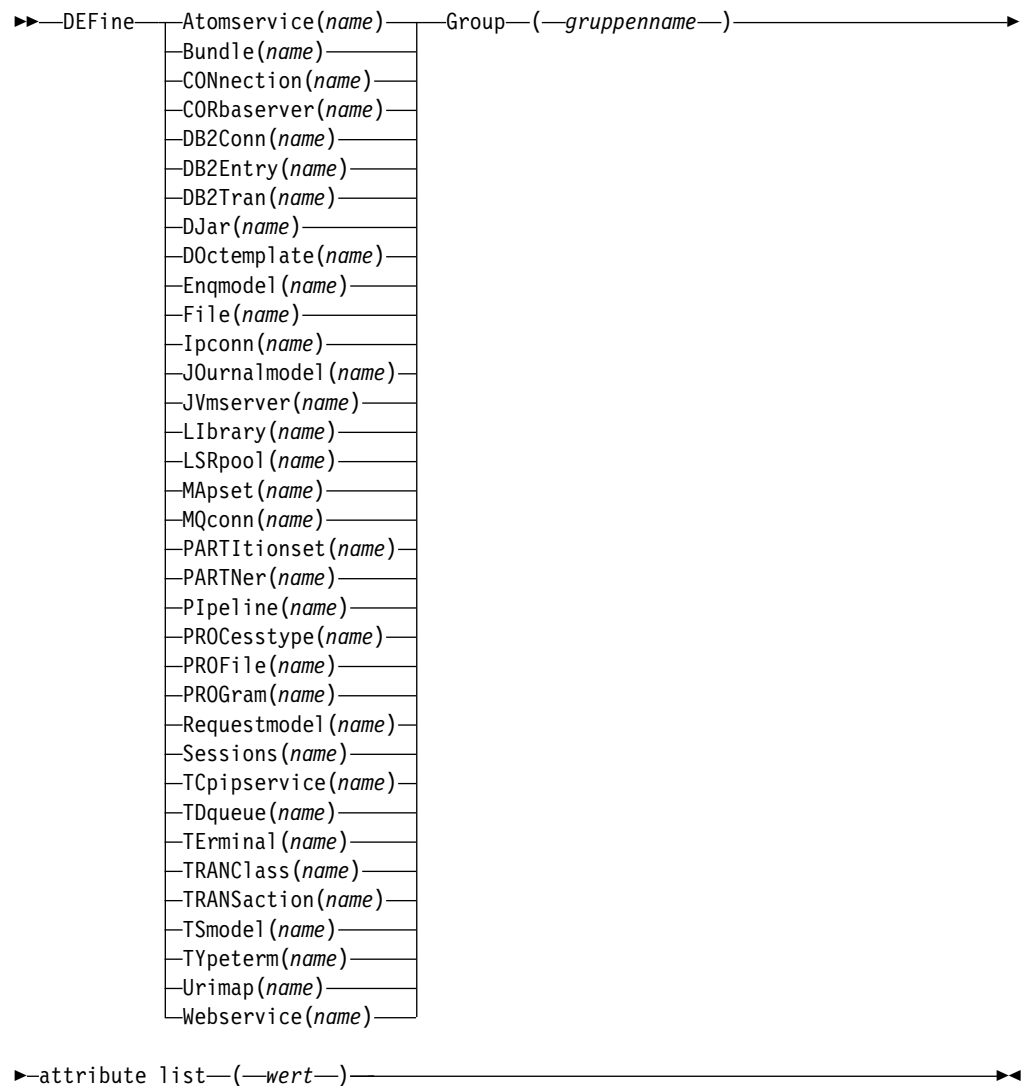
```
COPY SESSIONS(L122) GROUP(CICSC1) TO(CICSC2)
```

---

## DFHCSDUP-Befehl DEFINE

Neue Ressourcendefinitionen erstellen.

### Syntax von DEFINE



## Optionen

### attribute list

Die Attributliste ist vom definierten Ressourcentyp abhängig. Bestimmte Ressourcen verfügen über Attribute, die in die Definition eingebunden werden müssen. Für Attribute, für die von Ihnen kein Wert angegeben wird, werden die entsprechenden Standardwerte verwendet.

### Group(gruppenname)

Gibt den Namen der Gruppe an, die die zu erstellende Ressourcendefinition enthält. Verwenden Sie keinen generischen Gruppennamen. Wenn Sie den Namen einer Gruppe angeben, die noch nicht vorhanden ist, dann wird die Gruppe erstellt.

### resource(name)

Gibt den Namen der Ressource an, die definiert werden soll. Verwenden Sie keinen generischen Ressourcennamen. Die Ressourcenoption muss immer als der erste Operand des Befehls **DEFINE** angegeben werden.

## Beispiele

Sie können den gleichen Namen für mehrere Ressourcendefinitionen in einer Gruppe verwenden, wenn die Definitionen für unterschiedliche Ressourcentypen gelten. Beispiel:

```
DEFINE PROGRAM(N28A) GROUP(N28APPL)
DEFINE TRANSACTION(N28A) GROUP(N28APPL)
```

```
DEFINE TERMINAL(USER) GROUP(USERDEF)
DEFINE PROGRAM(USER) GROUP(USERDEF)
```

Im nächsten Beispiel werden zwei Konsolen für CICS definiert. Wenn eine Definition mehrere Zeilen umfasst, sind keine Fortsetzungssymbole erforderlich.

```
DEFINE TERMINAL(CON0)      GROUP(CONTERMS)
    CONSNAME(CONSJCL)  TYPETERM(DFHCONS)
    DESCRIPTION(MVS CONSOLE FOR ISSUING JCL COMMANDS)

DEFINE TERMINAL(CON1)      GROUP(CONTERMS)
    CONSNAME(CONSMAS)  TYPETERM(DFHCONS)
    DESCRIPTION(MVS MASTER CONSOLE)
```

Mit dem Befehl INITIALIZE wird eine TYPETERM-Definition, jedoch keine TERMINAL-Definition für eine Konsole generiert. Sie müssen mindestens eine Konsole definieren, um den MVS-Befehl **MODIFY** für CICS absetzen zu können.

---

## DFHCSDUP-Befehl DELETE

Einzelne Ressourcendefinition in einer Gruppe, alle Ressourcendefinitionen in einer Gruppe oder alle Gruppennamen in einer Gruppenliste löschen.

### Syntax von DELETE





## Beschreibung

Das Löschen einer Ressourcendefinition unterscheidet sich vom Entfernen einer Gruppe aus einer Liste (siehe „DFHCSDUP-Befehl REMOVE“ auf Seite 390). Eine gelöschte Ressourcendefinition wird tatsächlich aus der CSD-Datei entfernt.

### Anmerkung:

Wenn Sie die letzte Ressource aus einer Gruppe löschen, wird auch die Gruppe automatisch gelöscht. Es dürfen keine leeren Gruppen existieren.

Wird eine Gruppe gelöscht, so wird sie nicht aus allen Listen entfernt, in der sie enthalten ist.

Die Definitionen von Gruppen und Listen, die von IBM bereitgestellt werden, können nicht gelöscht werden.

Wenn Sie eine Liste löschen, dann werden die Definitionen der Ressourcen in den Gruppen, die in der Liste enthalten sind, nicht gelöscht. Dazu müssen Sie jede Gruppe einzeln löschen.

## Optionen

### **Group**(*gruppenname*)

Wenn diese Option alleine angegeben wird, dann wird in ihr der Name der Gruppe angegeben, die gelöscht werden soll. Wird auch eine Ressource angegeben, dann gibt sie die Gruppe an, zu der die Resource gehört. Verwenden Sie keinen generischen Gruppennamen.

### **List**(*listenname*)

Gibt den Namen der Liste an, die gelöscht werden soll. Verwenden Sie keinen generischen Listennamen.

### **Remove**

Wird diese Option beim Löschen der Gruppe angegeben, dann wird die Gruppe aus allen Listen entfernt, in denen sie enthalten war, es sei denn, auf dem System werden UPGRADE-Befehle ausgeführt.

### **Resource**(*name*)

Gibt den Namen der Liste an, die gelöscht werden soll. Verwenden Sie keinen generischen Ressourcennamen.

Dieser Operand kann nur zusammen mit der Option GROUP verwendet werden.

## Beispiele

Eine Liste in der primären CSD-Datei mit dem Namen LISTA enthält die folgenden Gruppen:

- GB001
- GB002

Die Gruppe GB001 enthält die folgenden Ressourcendefinitionen:

```
TERMINAL(CON0)  
TERMINAL(CON1)  
TERMINAL(TEST)
```

Mit dem folgenden Befehl wird die Ressourcendefinition für das Terminal TEST aus der Gruppe GB001 gelöscht:

```
DELETE TERMINAL(TEST) GROUP(GB001)
```

Mit dem folgenden Befehl werden alle Ressourcendefinitionen in der Gruppe GB002 gelöscht:

```
DELETE GROUP(GB002)
```

In der Gruppenliste LISTA verbleibt somit nur noch die Gruppe GB001. Mit dem folgenden Befehl werden alle Gruppennamen in der Gruppenliste LISTA gelöscht:

```
DELETE LIST(LISTA)
```

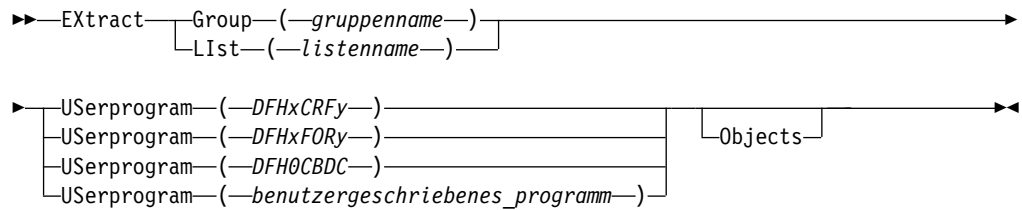
**Anmerkung:** Die Ressourcendefinitionen in den Gruppen in LISTA werden nicht gelöscht.

---

## DFHCSDUP-Befehl EXTRACT

Eine Ressourcendefinition, Gruppe oder Liste aus der CSD-Datei extrahieren.

### Syntax von EXTRACT



### Beschreibung

Mit dem Befehl **EXTRACT** können Sie Ressourcendefinitionsdaten der CSD-Datei entweder aus einer Liste oder einer Gruppe extrahieren und ein Benutzerprogramm aufrufen, um die extrahierten Daten zu verarbeiten. Das Benutzerprogramm wird im Parameter **USERPROGRAM** angegeben.

**Anmerkung:** Programmierinformationen zum Codieren von Benutzerprogrammen für den Befehl EXTRACT finden Sie in Invoking a user program from DFHCSDUP.

#### Optionen

##### **Group**(*gruppenname*)

Gibt nur die Ressourcendefinitionen innerhalb der benannten Gruppe an. Sie können einen generischen Gruppennamen angeben.

##### **List**(*listenname*)

Gibt nur die Ressourcendefinitionen innerhalb der Gruppen an, die in der benannten Liste enthalten sind. Sie können einen generischen Listennamen nur verwenden, wenn Sie die Option OBJECTS nicht verwenden.

##### **Objects**

Gibt die Details zu jeder Ressourcendefinition zurück. Sie können Ressourcendefinitionsdaten auf zwei Detailebenen extrahieren:

- Ohne die Option OBJECTS extrahiert der Befehl entweder die Namen aller Gruppen in einer angegebenen Liste oder die Namen aller Ressourcendefinitionen innerhalb einer bestimmten Gruppe.
- Mit der Option OBJECTS werden auch alle Ressourcendefinitionsattribute extrahiert.

Sie müssen OBJECTS für die bereitgestellten Beispielbenutzerprogramme DFHxCRFy und DFHxFORy angeben. Für DFH0CBDC und für benutzergeschriebene Benutzerprogramme ist die Option optional.

##### **Userprogram**(*benutzergeschriebenes\_programm*)

Gibt den Namen des vom Benutzer geschriebenen Programms an, mit dem die vom Befehl **EXTRACT** abgerufenen Daten verarbeitet werden sollen. Sie müssen einen Wert für USERPROGRAM angeben.

CICS stellt drei Typen von Beispielbenutzerprogrammen bereit: DFHxCRFy, DFHxFORy und DFH0CBDC. Der Buchstabe *x* im Programmnamen wird für Assembler oder PL/I durch \$ und für COBOL durch 0 ersetzt. Der Buchstabe *y*

im Programmnamen gibt die Programmiersprache an. Hierbei steht y=A für die Assemblerversion, y=C für die COBOL-Version und y=P für die PL/I-Version.

Für jedes benutzergeschriebene Programm muss eine bestimmte DD-Karte verwendet werden:

- DFHxCRFy benötigt die DD-Karte CRFOUT.
- DFHxFORy benötigt die DD-Karte FOROUT.
- DFH0CBDC benötigt die DD-Karte CBDOUT.

Alle anderen Benutzerprogramme stehen im Quellenformat in CICSTS55.CICS.SDFHSAMP zur Verfügung. Die Assemblerversionen sind außerdem in einem vorgenerierten Format in CICSTS55.CICS.SDFHLOAD verfügbar.

## Beispiele

Im folgenden Befehl wird das bereitgestellte Benutzerprogramm DFH0CBDC verwendet, um die Ressourcendefinitionen in der Gruppe DFHTYPE zu extrahieren und die **DEFINE**-Befehle zu erstellen, die für ihre Erstellung benötigt werden. Diese Befehle werden in der Datei gespeichert, die in der DD-Anweisung CBDOUT angegeben ist.

```
EXTRACT GROUP(DFHTYPE) USERPROGRAM(DFH0CBDC) OBJECTS
```

---

## DFHCSDUP-Befehl INITIALIZE

Neu definierte Datei zur Verwendung als CSD-Datei vorbereiten.

### Syntax von INITIALIZE

►—INITialize —►

### Beschreibung

Sie müssen Ihre CSD-Datei initialisieren, bevor Sie einen der anderen DFHCSDUP-Befehle oder die RDO-Transaktionen verwenden können. Nach der Initialisierung der CSD-Datei müssen Sie diese Funktion nicht nochmals ausführen.

Die Standardeinträge für die von CICS bereitgestellten Ressourcendefinitionen werden in der CSD-Datei erstellt. Mit dem Befehl INITIALIZE können diese Definitionen zu Gruppen zusammengefasst werden. Diese Gruppen werden anschließend in einer Gruppenliste mit dem Namen DFHLIST definiert. Diese Liste enthält nur die von CICS bereitgestellten Gruppen, die für ein CICS-System erforderlich sind.

CICS unterstützt RDO für transiente Daten. Die Gruppe DFHDCTG enthält Beispieldefinitionen aller von CICS bereitgestellten Warteschlangen. Sie können die Namen anderer Warteschlangen hinzufügen, die gleichzeitig in DFHDCTG installiert werden sollen. Platzieren Sie DFHDCTG am Anfang der Liste DFHLIST, so dass die Warteschlangen in einer möglichst frühen Phase der CICS-Initialisierung verfügbar gemacht werden können.

Wenn Sie eine andere Gruppe verwenden, um die von CICS bereitgestellten Warteschlangen zu installieren, dann sollten Sie sich vergewissern, dass diese Gruppe sich am Anfang der ersten Liste befindet, die mit GRPLIST im Rahmen eines Erst- oder Kaltstarts installiert wird.

Sie können andere Ressourcendefinitionen für transiente Daten in andere Gruppen einordnen, aus denen sie dann entweder während eines Erst- oder Kaltstarts oder zu einem geeigneten Zeitpunkt nach Abschluss der Initialisierung installiert werden können.

INITIALIZE erstellt außerdem einen Steuerdatensatz am Anfang der CSD-Datei. Dieser Datensatz enthält Felder, in denen das CICS-Release und die aktuelle Servicestufe angegeben sind, das bzw. die für die CSD angewendet wurde. Außerdem finden Sie dort Felder, die das Datum und die Uhrzeit der Erstellung der CSD-Datei sowie den Zeitpunkt (Datum und Uhrzeit) der letzten Aktualisierung enthalten. Diese beiden Felder erscheinen in der Hardcopyliste der CSD-Datei, die mit dem Befehl LIST erstellt werden kann.

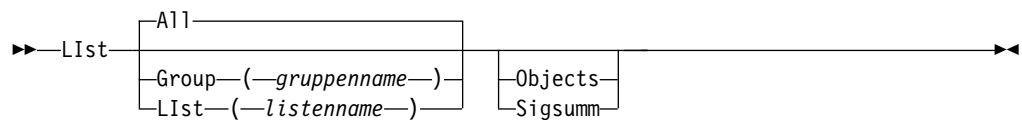
Wenn Sie eine neu definierte wiederherstellbare Datei zur Verwendung als CSD-Datei vorbereiten wollen, dann müssen Sie sie mit INITIALIZE im Nicht-RLS-Modus initialisieren, weil eine wiederherstellbare Datei nicht für die Ausgabe im Stapelbetrieb im RLS-Modus geöffnet werden kann. Die Datei muss aber für die Ausgabe geöffnet werden, um sie zu initialisieren.

---

## DFHCSDUP-Befehl LIST

Erstellt Listen mit dem aktuellen Status der CSD-Datei.

### Syntax von LIST



### Beschreibung

Die Listen werden zusammen mit den bei der Befehlsverarbeitung ausgegebenen Nachrichten an die SYSOUT-Datei gesendet. Die Inhalte aller qualifizierten Gruppen oder Listen werden ausgegeben.

Die Listen enthalten Ausgabedaten, mit denen die Releaseinformationen für die CSD (CICS-Systemdefinition) bestätigt werden können. Unter der Überschrift *CREATED BY RELEASE:* wird das Release angezeigt, in dem die CSD erstellt wurde. Sofern vorhanden, folgt die Angabe für *UPGRADED TO RELEASE:*, wenn für die CSD ein Upgrade durchgeführt wurde. In diesem Fall wird auch das Zielrelease aufgeführt.

Wenn die CSD gewartet wurde, wird eine zusätzliche Überschrift angezeigt: *LAST CSD MAINTENANCE UPGRADE USED dateiname AT PTF ptf-nummer LEVEL*. Die Wartung, die mit dem Befehl *DFHCSDUP UPGRADE USING (DFHCURDM)* angewendet wurde, gibt den PTF-Stand der verwendeten Komponente *DFHCURDM* an.

### Optionen

#### Group (gruppenname)

Gibt nur die Ressourcendefinitionen innerhalb der benannten Gruppe an. Sie können einen generischen Gruppennamen angeben.

**List(listenname)**

Gibt nur die Ressourcendefinitionen innerhalb der Gruppen an, die in der benannten Liste enthalten sind. Sie können einen generischen Listennamen nur verwenden, wenn Sie die Option OBJECTS nicht verwenden. Der einzige Befehl, für den ein generischer Listenname nicht akzeptiert wird, ist LIST LIST(listenname) OBJECTS.

**Objects**

Gibt den Detaillierungsgrad an, der für die einzelnen Ressourcendefinitionen erforderlich ist. Sie können Ressourcendefinitionsdaten in zwei Detaillierungsgraden extrahieren:

- Ohne die Option OBJECTS extrahiert der Befehl entweder die Namen aller Gruppen in einer angegebenen Liste oder die Namen aller Ressourcendefinitionen innerhalb einer bestimmten Gruppe.
- Mit der Option OBJECTS werden auch alle Ressourcendefinitionsattribute (einschließlich der Felder für die Definitionssignatur) extrahiert.

**Sigsumm**

Zeigt die Definitionssignatur für jede der angezeigten Ressourcendefinitionen an.

**Beispiele**

Die von den verschiedenen Befehlen generierten Listen lauten wie folgt:

- LIST ALL
  - Namen von definierten Listen und Gruppen
  - Zusammenfassung der Listen
  - Zusammenfassung der Gruppen

Mit dem Befehl LIST ALL können Zusammenfassungen aller Definitionen von Listen und Gruppen ausgegeben werden, die sich in der CSD-Datei befinden.

- LIST ALL OBJECTS
  - Namen von definierten Listen und Gruppen
  - Zusammenfassung der Listen
  - Zusammenfassung der Gruppen
  - Objekte in Gruppen

Mit dem Befehl LIST ALL OBJECTS können Zusammenfassungen aller Definitionen von Listen und Gruppen, die sich in der CSD-Datei befinden, zusammen mit den Eigenschaften der Ressourcen in allen Gruppen ausgegeben werden.

- LIST GROUP(gruppenname)
  - Zusammenfassung der Gruppen
  - Gruppennamen können generisch sein

Mit dem Befehl LIST GROUP können die Namen aller Ressourcen in einzelnen oder mehreren Gruppen zusammengefasst werden. Sie sind in jeder Gruppe in Ressourcentypkategorien zusammengefasst, z. B. für Maskengruppen und Programme.

- LIST GROUP(gruppenname) OBJECTS
  - Zusammenfassung der Gruppen
  - Objekte in Gruppen
  - Gruppennamen können generisch sein

Mit diesem Befehl können Sie die Eigenschaften von Ressourcen, die auf Basis des Ressourcentyps zusammengefasst sind, in tabulatorbasierten Aufstellungen

darstellen. Die Erstellungszeit für jede der Ressourcen wird mit sämtlichen Attributen wie anfänglich mit den Befehlen DEFINE und ALTER oder durch Migration aus einer CICS-Tabelle festgelegt, angegeben. Die Eigenschaften von Transaktionen und Profilen werden in den gleichen Unterkategorien angeordnet, die auch in der Anzeige für CEDA DEFINE dargestellt werden.

- LIST GROUP(*gruppenname*) SIGSUMM
  - Gruppennamen können generisch sein

Mit diesem Befehl können Sie die Definitionssignatur der Ressourcen, die auf Basis des Ressourcentyps zusammengefasst sind, in tabulatorbasierten Aufstellungen darstellen.

- LIST LIST(*listenname*)
  - Zusammenfassung der Listen
  - Listennamen können generisch sein

Die Inhalte einzelner oder mehrerer Gruppenlisten haben ein tabulatorbasiertes Format. Die Gruppen werden in der gleichen Reihenfolge angezeigt, in der ihre Positionen in der List angeordnet sind. Diese Reihenfolge wird mit den Befehlen ADD und APPEND festgelegt, die in der CEDA-Transaktion zum Erstellen der Liste benutzt wurden.

- LIST LIST(*listenname*) OBJECTS
  - Zusammenfassung der Listen
  - Objekte von Gruppen in Liste
  - Generische Listennamen sind nicht zulässig

Verwenden Sie diesen Befehl, um die Merkmale aller Ressourcen zu tabulieren, die in einem CICS-System beim Systemstart definiert werden sollen. Sie werden durch mindestens einen Listennamen im Systeminitialisierungsparameter GRPLIST=(*liste1,liste2,liste3,liste4*) identifiziert. Die Namen aller Gruppen in der Liste werden in der Zusammenfassung der Listen aufgeführt. Anschließend werden für jede in der Liste enthaltene Gruppe die Eigenschaften der individuellen Ressourcen in der Gruppe tabuliert.

Die Tabulierung für „Objekte von Gruppen in Liste“ ordnet die Gruppen in derselben Reihenfolge an, in der sie zur Gruppenliste hinzugefügt wurden. Diese Reihenfolge ist wichtig, wenn doppelte Einträge vorhanden sind, weil Definitionen derselben Ressource möglicherweise in mehr als einer Gruppe enthalten sind. Wenn eine Liste dieses Typs beim Systemstart verwendet wird, dann werden im Falle doppelter Einträge die Ressourcendefinitionen verwendet, die zu der Gruppe gehören, die innerhalb der Liste an letzter Stelle steht.

---

## DFHCSDUP-Befehl PROCESS

Die Wartung auf die CSD-Datei für einen bestimmten APAR (Authorized Program Analysis Report) anwenden.

### Syntax von PROCESS

►►—PROCESS—Apar—(—*apar-nummer*—)—————►►

### Beschreibung

Mit dem Befehl PROCESS APAR kann die Wartung für einen bestimmten APAR (Authorized Program Analysis Report) auf Ihre CSD-Datei angewendet werden. Verwenden Sie diesen Befehl nur in Übereinstimmung mit den Anweisungen im zugehörigen PTF-Begleitschreiben.

## Optionen

### **Apar**(*apar-nummer*)

Die Nummer des APAR für die Bereitstellung der Wartung. PROCESS APAR- (PQ12417) wird beispielsweise verwendet, um die Wartung auf den APAR PQ12417 anzuwenden.

---

## DFHCSDUP-Befehl REMOVE

Einen Gruppennamen aus einer Liste entfernen.

### Syntax von REMOVE

►► ~~Remove~~ ~~Group~~ (~~—gruppenname—~~) ~~LIst~~ (~~—listenname—~~) ~~—————~~ ◀◀

## Beschreibung

Die Gruppe und alle darin enthaltenen Ressourcendefinitionen bleiben in der CSD-Datei weiterhin erhalten.

## Optionen

### **Group**(*gruppenname*)

Gibt den Namen der Gruppe an, die entfernt werden soll. Verwenden Sie keinen generischen Gruppennamen.

### **LIst**(*listenname*)

Gibt den Namen der Liste an, aus der eine Gruppe entfernt werden soll. Verwenden Sie keinen generischen Listennamen. Wenn die letzte Gruppe aus einer Liste entfernt wird, dann ist die Liste in der CSD-Datei nicht mehr vorhanden.

## Beispiele

Eine Liste mit dem Namen LL02 enthält die folgenden Gruppen:

G001 G002 G003 G004

Geben Sie Folgendes ein, um die Gruppe G003 zu entfernen:

REMOVE GROUP(G003) LIST(LL02)

Damit verbleiben die folgenden Gruppen in der Liste:

G001 G002 G004

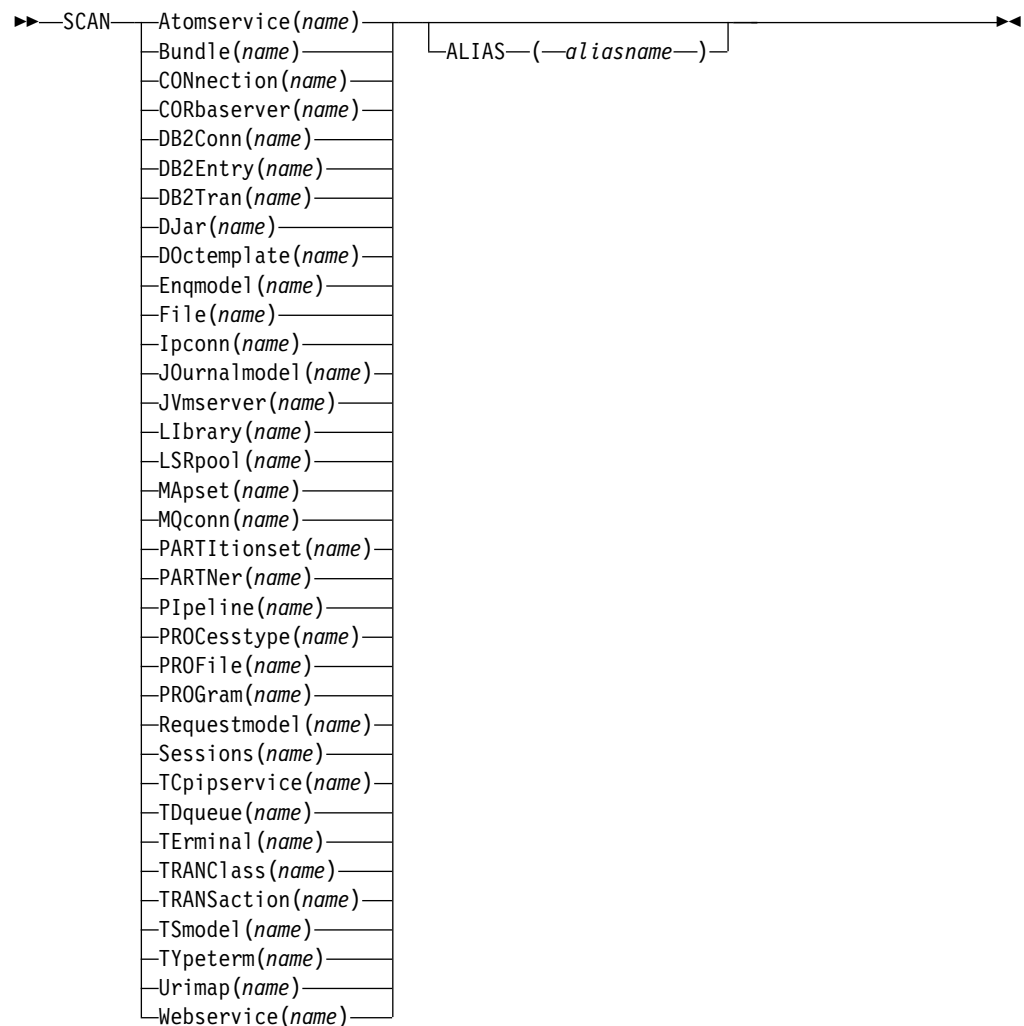
---

## DFHCSDUP-Befehl SCAN

Alle von IBM bereitgestellten Gruppen und benutzerdefinierten Gruppen für eine angegebene Ressource mit SCAN durchsuchen. Die Definition der abgeglichenen Ressource in einer von IBM bereitgestellten Gruppe wird mit den Definitionen der entsprechenden abgeglichenen Ressource in den Benutzergruppen verglichen.

### Syntax von SCAN





## Beschreibung

Informationen zu den Ressourcentypen, die im Befehl SCAN angegeben werden können, sowie eine Beschreibung der Attribute und Standardwerte aller Ressourcentypen finden Sie in RDO resources.

Mit dem Befehl SCAN können alle von IBM bereitgestellten Gruppen in der CSD (CICS-Systemdefinition) nach einer Ressourcendefinition mit einem angegebenen Namen und Typ durchsucht werden. Anschließend wird eine Nachricht ausgegeben, die die Ergebnisse der Suche enthält. Die benutzerdefinierten Gruppen werden anschließend nach derselben Ressourcendefinition durchsucht. Das Ergebnis dieses Vorgangs kann wie folgt lauten:

- Wenn eine von IBM bereitgestellte Gruppe und mindestens eine benutzerdefinierte Gruppe die Ressourcendefinition enthalten, dann wird die Definition in der von IBM bereitgestellten Gruppe mit der Definition in den Benutzergruppen verglichen. Anschließend wird eine Nachricht ausgegeben, in der angegeben ist, ob die Definition in der von IBM bereitgestellten Gruppe mit den Definitionen in den Benutzergruppen übereinstimmt.
- Wenn die Ressourcendefinition in den benutzerdefinierten Gruppen nicht gefunden wird, gibt das System eine Nachricht aus.

- Wenn die Ressourcendefinition in keiner der von IBM bereitgestellten Gruppen, jedoch in mindestens einer der benutzerdefinierten Gruppen gefunden werden kann, dann wird eine Nachricht ausgegeben, in der die entsprechenden Gruppen angegeben werden.

Wird ein Wert für *aliasname* angegeben, dann werden die Benutzergruppen mithilfe dieser Angabe für *aliasname* durchsucht.

#### Anmerkung:

1. Die DFHCOMPx-Kompatibilitätsgruppen werden nicht zusammen mit den von IBM bereitgestellten Gruppen durchsucht, sondern zusammen mit den benutzerdefinierten Gruppen.
2. Das Attribut DESCRIPTION wird im Vergleich nicht verwendet.

Sie können mit dem Befehl SCAN nach Unterschieden zwischen den von IBM bereitgestellten Definitionen, an denen Sie Änderungen vorgenommen haben, und den neuesten von IBM bereitgestellten Versionen nach einem Upgrade suchen.

## Optionen

### Alias(*aliasname*)

Gibt den Aliasnamen für den Ressourcentyp an, nach dem in den benutzerdefinierten Gruppen gesucht werden soll.

Dieser Operand ist optional.

### Resource(*name*)

Gibt den Namen des Ressourcentyps an, nach dem in den von IBM bereitgestellten Gruppen und in den benutzerdefinierten Gruppen gesucht werden soll, falls *aliasname* nicht angegeben wurde. Die Ressourcenoption muss immer als der erste Operand des Befehls SCAN angegeben werden.

## Beispiele

Gehen Sie wie folgt vor, um die CSD nach CEDA-Befehlen für eine Transaktion zu durchsuchen:

```
SCAN TRANSACTION(CEDA)
```

Das Ergebnis lautet wie folgt:

```
DFH5130 I PRIMARY CSD OPENED; DDNAME: DFHCSD
DFH5633 I TRANSACTION CEDA FOUND IN GROUP DFHSPI
DFH5631 I TRANSACTION CEDA FOUND IN GROUP A1
          MATCHES THE IBM SUPPLIED DEFINITION IN GROUP DFHSPI
DFH5631 I TRANSACTION CEDA FOUND IN GROUP A2
          MATCHES THE IBM SUPPLIED DEFINITION IN GROUP DFHSPI
DFH5632 I TRANSACTION CEDA FOUND IN GROUP DFHCOMP1
          DOES NOT MATCH THE IBM SUPPLIED DEFINITION
          IN GROUP DFHSPI
DFH5101 I SCAN COMMAND EXECUTED SUCCESSFULLY.
```

Gehen Sie wie folgt vor, um die CSD nach CEDA-Befehlen für eine Transaktion mit dem Aliasnamen AEDA zu durchsuchen:

```
SCAN TRANSACTION(CEDA) ALIAS(AEDA)
```

Das Ergebnis lautet wie folgt:

```
DFH5130 I PRIMARY CSD OPENED; DDNAME: DFHCSD
DFH5633 I TRANSACTION CEDA FOUND IN GROUP DFHSPI
DFH5631 I TRANSACTION AEDA FOUND IN GROUP A3
          MATCHES THE IBM SUPPLIED DEFINITION IN GROUP DFHSPI
DFH5101 I SCAN COMMAND EXECUTED SUCCESSFULLY.
```

---

## DFHCSDUP-Befehl SERVICE

Die Wartung für Ihre CSD-Datei anwenden.

### Syntax von SERVICE

```
►►—Service—FROmcsd—(—dd-name—)—LEvel—(—nnn—)—————►►
```

### Beschreibung

Möglicherweise müssen Sie gelegentlich (zwischen CICS-Releases) eine Serviceroutine anwenden, um präventive oder korrektive Wartungsmaßnahmen für Ihre CSD-Datei auszuführen.

Mit dem Befehl SERVICE können Sie eine neue Kopie der CSD-Datei auf Basis der bereits vorhandenen CSD-Datei erstellen. Alle Definitionen werden beibehalten, wobei die Korrekturen (falls vorhanden) angewendet werden.

### Optionen

#### FROmcsd (*dd-name*)

Gibt den DD-Namen der aktuellen CSD-Datei an, die für den Befehl als sekundäre CSD-Datei eingestuft wird.

#### LEvel (*nnn*)

Ihrer CSD-Datei ist eine aktuelle Servicestufe zugeordnet, deren Anfangswert bei der Initialisierung der Datei auf 000 gesetzt war. Wird die Serviceroutine angewendet, dann hat dies beim Wechsel von der aktuellen Stufe auf die Zielstufe eine Erhöhung der Servicestufe jeweils um den Wert 1 zur Folge.

Dieser Operand gibt die Zielservicestufe an, auf die die CSD-Datei aktualisiert werden soll. Sie muss um den Wert 1 höher sein als die aktuelle Stufe von FROMCSD. Geben Sie diesen Wert als aus drei Stellen bestehende Ganzzahl (z. B. LEVEL(001)) an.

---

## DFHCSDUP-Befehl UPGRADE

Von CICS bereitgestellte Ressourcendefinitionen in einer primären CSD-Datei ändern.

### Syntax von UPGRADE

```
►►—Upgrade—┐
              └—USing—(—dateiname—)—┐ └—Replace—┐—————►►
```

### Beschreibung

Führt ein Upgrade der von IBM bereitgestellten Definitionen in der CSD durch. Definitionen können zu DFH-Gruppen hinzugefügt, in diesen geändert oder aus DFH-Gruppen gelöscht werden. Beachten Sie hierbei, dass gelöschte Definitionen

zu Kompatibilitätsgruppen mit Namen hinzugefügt werden, die das Format DFH-COMP $n$  haben. Dadurch wird es möglich, die CSD gemeinsam mit früheren Releases von CICS zu nutzen, nachdem Sie den Upgradebefehl ausgeführt haben.

Verwenden Sie den DFHCSDUP-Befehl LIST, um den Status der CSD-Datei vor oder nach dem Upgrade zu überprüfen. Der Befehl meldet das Release, unter dem eine CSD erstellt wurde, und das Release, auf das die CSD aktualisiert wurde.

Der Upgradebefehl kann außerdem verwendet werden, um ein beliebiges Paket mit von IBM bereitgestellten Ressourcendefinitionen auf die CSD-Datei anzuwenden. Die Definitionen für die CICS-Beispielprogramme und -Transaktionen können z. B. mit der Anweisung UPGRADE in die CSD-Datei übertragen werden.

## Optionen

### Replace

Geben Sie die Option REPLACE an, wenn Sie die Ausführung des Befehls UPGRADE wiederholen müssen (z. B. aufgrund eines zuvor aufgetretenen Problems).

### USing(*dateiname*)

Um IBM Funktionen unter CICS zu installieren, müssen Sie UPGRADE USING(*dateiname*) angeben. Der Befehl UPGRADE USING(DFHRDJPN) wird beispielsweise verwendet, um die Definitionen der DBCS-Funktion (DBCS = Double-Byte Character Set; Doppelbytezeichensatz) in der CSD-Datei zu speichern.

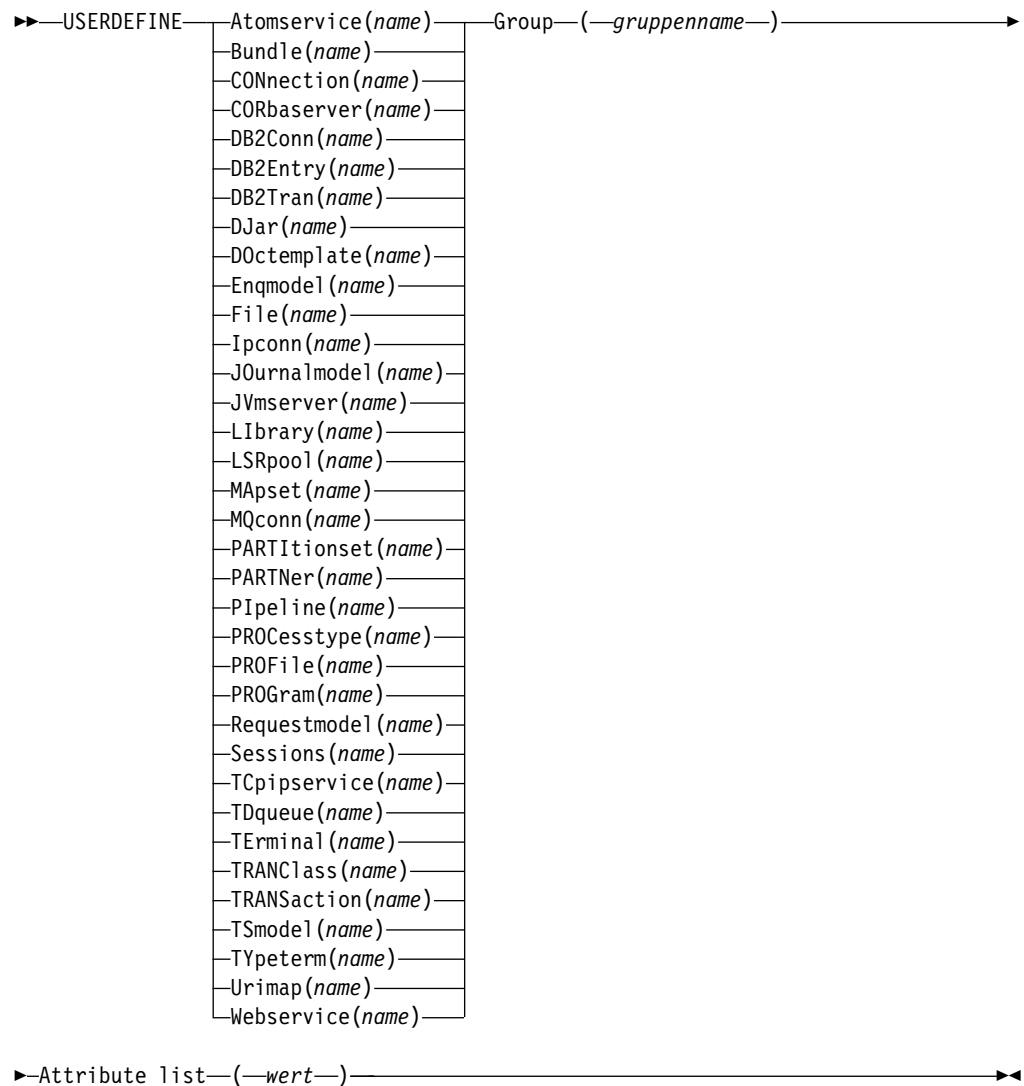
Zum Upgrade einer CSD darf der Operand **USING** nicht angegeben werden. Alle von IBM bereitgestellten Definitionen aus einem **beliebigen** Release werden gelöscht und anschließend wird die CSD-Datei initialisiert. Aus diesem Grund ist es nicht erforderlich, das Quellenrelease anzugeben.

---

## DFHCSDUP-Befehl USERDEFINE

Neue Ressourcendefinitionen mithilfe eigener Standardwerte anstelle der von CICS bereitgestellten Standardwerte erstellen.

### Syntax von USERDEFINE



## Beschreibung

Beim Befehl USERDEFINE handelt es sich um eine Alternative zum Befehl DEFINE. Anstelle der von CICS bereitgestellten Standardwerte werden im Befehl USERDEFINE Ihre eigenen Standardwerte benutzt, um eine Ressourcendefinition zu erstellen. In jeder anderen Hinsicht arbeitet der Befehl genau wie der Befehl DEFINE.

Zum Einrichten eigener Standardwerte für den Befehl USERDEFINE verwenden Sie den normalen Befehl DEFINE, um Ressourcendefinitionen mit dem Namen USER in einer Gruppe mit dem Namen USERDEF zu erstellen:

- Erstellen Sie eine Ressourcendefinition mit dem Namen USER in der Gruppe USERDEF für jede Ressource, für die Sie Standardwerte angeben wollen. Beispiel: Wenn Sie Standardwerte für die PROGRAM-, TRANSACTION- und TCPIPService-Ressourcendefinitionen bereitstellen wollen, dann erstellen Sie die Ressourcendefinitionen PROGRAM(USER), TRANSACTION(USER) und TCPIPService(USER) in der Gruppe USERDEF. Hierbei spielt es keine Rolle, dass alle Ressourcendefinitionen in der Gruppe USERDEF den Namen USER haben. Sie sind eindeutig, weil sie über verschiedene Ressourcentypen verfügen. Alle Ressourcendefinitionen in der Gruppe USERDEF, die nicht den Namen USER haben, werden vom Befehl USERDEFINE ignoriert.

- Geben Sie in jeder Ressourcendefinition in der Gruppe USERDEF die Standardwerte an, die angewendet werden sollen, wenn Sie den Befehl USERDEFINE zum Erstellen einer Ressource dieses Typs verwenden wollen. Wenn Sie z. B. in den PROGRAM-Ressourcendefinitionen, die mit dem Befehl USERDEFINE erstellt werden, als Standardsprache 'Assembler' festlegen wollen, dann geben Sie den folgenden Befehl DEFINE ein, um die Ressourcendefinition zu erstellen:

```
DEFINE PROGRAM(USER) GROUP(USERDEF) LANGUAGE(ASSEMBLER)
```

- Jede Ressourcendefinition in der Gruppe USERDEF muss eine vollständige, gültige Ressourcendefinition sein. Für eine Transaktionsdefinition muss beispielsweise eine Programmdefinition angegeben werden, und zwar auch dann, wenn Sie immer einen Programmnamen angeben, wenn Sie mit dem Befehl USERDEFINE eine Transaktion definieren.
- Sie müssen die Ressourcendefinitionen nicht in der Gruppe USERDEF installieren.

Wenn Sie Ressourcendefinitionen in der Gruppe USERDEF erstellt haben, dann können Sie den Befehl USERDEFINE zum Definieren dieser Typen von Ressourcen verwenden. Die von Ihnen festgelegten Standardwerte werden in den Ressourcendefinitionen verwendet. Beispiel: Wenn Sie eine PROGRAM-Ressourcendefinition in der Gruppe USERDEF erstellt haben, in der LANGUAGE(ASSEMBLER) angegeben ist, dann wird mit dem folgenden Befehl eine Ressourcendefinition für das Programm P2 in der Gruppe GRP erstellt und dabei 'Assembler' als Sprache angegeben:

```
USERDEFINE PROGRAM(P2) GROUP(GRP)
```

## Optionen

### **Attribute list**(wert)

Die Attributliste ist vom definierten Ressourcentyp abhängig. Bestimmte Ressourcen verfügen über Attribute, die in die Definition eingebunden werden müssen. Eine Beschreibung der Attribute und Standardwerte aller Ressourcentypen finden Sie in RDO resources. Für Attribute, für die von Ihnen kein Wert angegeben wird, werden die entsprechenden Standardwerte verwendet.

### **Group**(gruppenname)

Gibt den Namen der Gruppe an, in der die zu erstellende Ressourcendefinition enthalten sein wird. Verwenden Sie keinen generischen Gruppennamen. Wenn Sie den Namen einer Gruppe angeben, die noch nicht vorhanden ist, dann wird die Gruppe erstellt.

### **resource**(name)

Gibt den Namen der Ressource an, die definiert werden soll. Verwenden Sie keinen generischen Ressourcenamen. Die Ressourcenoption muss immer als der erste Operand des Befehls USERDEFINE angegeben werden.

---

## DFHCSDUP-Befehl VERIFY

Interne Sperren für Gruppen und Listen entfernen.

### **Syntax von VERIFY**

►►—VERIFY—◄◄

## Beschreibung

Verwenden Sie den Befehl VERIFY nur, wenn die CSD-Datei nicht verwendet wird und keine Rücksetzungsverarbeitungsoperationen für die CSD-Datei anstehen. Verwenden Sie den Befehl vorzugsweise nur dann, wenn keine CICS-Systeme, die die CSD-Datei möglicherweise benutzen, aktiv sind. Insbesondere sollten Sie den Befehl VERIFY nicht verwenden, während CICS-Systeme im RLS-Zugriffsmodus auf die CSD-Datei zugreifen können.

Der Befehl VERIFY hat Auswirkungen auf die gesamte CSD-Datei und ist für die Verwendung in Extremsituationen vorgesehen, in denen die Datensätze zu internen Sperren nicht mehr aktualisiert werden können. Diese Datensätze werden normalerweise entfernt, nachdem die Ausführung einer Funktion, mit der die CSD-Datei geändert wird, abgeschlossen ist. Allerdings wurde diese Ausführung möglicherweise nicht abgeschlossen, wenn ein Systemfehler bei der Ausführung der CEDA-Transaktion aufgetreten ist oder wenn die Ausführung eines Offlinedienstprogramms nicht abgeschlossen werden konnte. Die Sperren können verhindern, dass CEDA-Benutzer auf bestimmte Gruppen und Listen in der CSD-Datei zugreifen können.

Beachten Sie hierbei, dass der Befehl VERIFY nur interne Sperren entfernt. Auf die normalen Benutzersperren, die mit dem Befehl LOCK in der CEDA-Transaktion angewendet werden, hat der Befehl hingegen keine Auswirkungen.





---

## Kapitel 4. DFHCSDUP: Syntax- und Befehlsregeln

Verwenden Sie die folgenden Regeln für die Syntax und Vorbereitung von Befehlen für das Programm DFHCSDUP.

Geben Sie die Befehle in den Spalten 1 bis 71 der 80-stelligen Eingabedatensätze ein. Sie können Schlüsselwortwerte angeben, deren Länge die Zeilenlänge überschreitet, wenn Sie das Fortsetzungszeichen (Stern (\*)) am Ende einer Zeile (in Spalte 72) verwenden. Nachfolgende Zeilen beginnen in Spalte 1. Sie können diese Funktion z. B. benutzen, um Werte für XTPNAME mit bis zu 128 Hexadezimalzeichen anzugeben.

Die Befehlsschlüsselwörter können auch in Form von Abkürzungen und in gemischter Groß-/Kleinschreibung angegeben werden. Die entsprechenden Eingaben werden in der Befehlssyntax der jeweiligen Befehlsbeschreibung angegeben. Die Mindestabkürzung wird in der Befehlssyntax in Großbuchstaben angegeben, die optionalen Zeichen hingegen in Kleinbuchstaben. Beispiel:

Alter Connection(name) Group(gruppenname)

Führende Leerzeichen werden ignoriert und Leerzeichen zwischen Schlüsselwörtern und Operanden sind zulässig.

Kommentardatensätze sind zulässig. Sie müssen jedoch in Spalte 1 mit einem Stern (\*) gekennzeichnet werden. Kommentarelemente sind in Datensätzen, die einen Befehl enthalten, nicht zulässig.

Nullsätze zwischen Befehlen werden ignoriert.

Beim Codieren der Parameter **GROUP**, **LIST**, **TO** und **TYPESGROUP** müssen die Konventionen für Gruppen- und Listennamen eingehalten werden. Wenn Sie eine generische Spezifikation für den Parameter GROUP oder LIST im Befehl LIST verwenden, dann können Sie die Symbole Stern (\*) und Plus (+) in derselben Weise wie für CEDA benutzen.

Der Parameter **FROMCSD** muss einen gültigen DD-Namen enthalten, der den Regeln für die JCL des Betriebssystems entspricht.

Ein Beispiel für eine gültige Befehlsfolge finden Sie in Abb. 6 auf Seite 400. Weitere Beispiele für Befehle sind in den folgenden Befehlsbeschreibungen aufgeführt.

```

*                               SET UP INITIAL CSD FILE
INITialize
*
LIst LIst(DFHLIST) Objects
*                               UPGRADE FROM EARLIER RELEASE
UPgrade
*
LI Group(PPTM1)
LI G(SETM*)
*                               CREATE GROUP PCTZ4
Copy G(PCTM1) To(PCTZ4)
C G(SETMP3) T(PCTZ4) Replace
LI G(P++M+)
*                               CREATE LIST MODLIST
APpend LIst(TESTLIST) TO(MODLIST) FRomcsd(CSDF1)
AP LI(SECLIST) To(MODLIST) FR(CSDF1)
AP LI(DFHLIST) To(MODLIST)
*
LI ALL OBJECTS

```

*Abbildung 6. Beispielbefehle des Programms DFHCSDUP*

---

## Kapitel 5. Ressourcendefinitionsmakros

Obwohl Sie die meisten CICS-Ressourcen auf unterschiedliche Weise definieren können, müssen Sie für einige Ressourcen CICS-Makros verwenden.

---

### CLT: Befehlslistentabelle

Die Befehlslistentabelle (CLT = Command List Table) wird von der *erweiterten Wiederherstellungsfunktion* (XRF = Extended Recovery Facility) verwendet und enthält eine Liste mit MVS-Systembefehlen und -Nachrichten an den Bediener, die während der Übernahme abgesetzt werden. Normalerweise haben diese Befehle die Funktion, alternative Systeme zur Übernahme der Steuerung von den zugehörigen aktiven Systemen anzuweisen, die sich in derselben MRO-Konfiguration (MRO = Multiregion Operation; Mehrregionenbetrieb) befinden.

Die Befehlslistentabelle enthält auch den Namen des alternativen Systems mit dem Jobnamen des aktiven Systems, zu dessen Abbruch es berechtigt ist. (Siehe hierzu den Operanden DFHCLT TYPE=LISTSTART FORALT.) Auf diese Weise wird eine Sicherheitsprüfung bereitgestellt, mit der verhindert werden kann, dass der falsche Job abgebrochen wird, wenn das alternative System die Steuerung übernimmt.

Darüber hinaus stellt das Makro DFHCLT TYPE=INITIAL JES Routing-Informationen zur Verfügung, die zum Senden von Abbruchbefehlen an das entsprechende MVS-System benötigt werden. Wenn Sie mit XRF arbeiten, dann **müssen** Sie über eine Befehlslistentabelle verfügen. Sie wird nur vom alternativen CICS-System verwendet.

Aus Sicherheitsgründen sollten Sie für die CLT eine Programmverbindung zu einer Bibliothek herstellen, die zur Verwendung von APF berechtigt ist. Zur Entlastung des virtuellen Speichers sollten Sie die Programmverbindung mithilfe einer MODE-Steueranweisung mit der Angabe AMODE(31),RMODE(ANY) herstellen. Die Tabelle sollte bei der Herstellung der Programmverbindung als wiedereintrittsfähig definiert werden. Die CLT wird nicht in den CICS-Nukleus geladen.

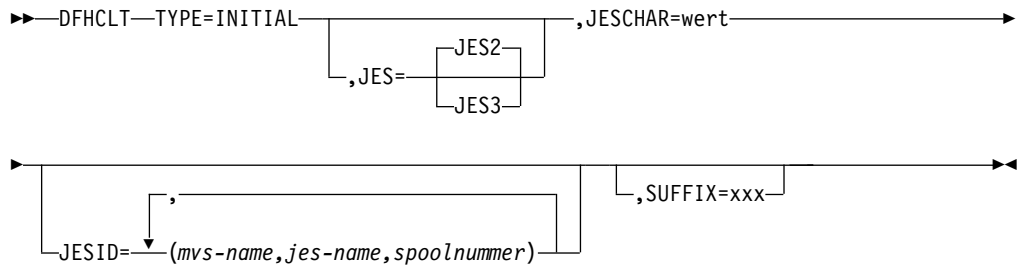
Ihre CLT kann die folgenden Anweisungen enthalten:

- DFHCLT TYPE=INITIAL
- DFHCLT TYPE=LISTSTART
- DFHCLT TYPE=COMMAND
- DFHCLT TYPE=WTO
- DFHCLT TYPE=LISTEND
- DFHCLT TYPE=FINAL (siehe hierzu TYPE=FINAL (end of table))

**Anmerkung:** Obwohl die Befehlslistentabelle gemeinsam mit einer Reihe alternativer Systeme genutzt werden kann, sollten Sie sicherstellen, dass MVS während der Übernahme nicht zu viele redundante Befehle erhält. Im Mehrregionenbetrieb und bei Verwendung einer Befehlslistentabelle mit Befehlen für mehrere Regionen sendet Region 1 z. B. gültige Befehle an andere Regionen, die ihrerseits redundante Befehle an Region 1 und an die anderen Regionen senden würden.

## Programmabschnitt: DFHCLT TYPE=INITIAL

Mit dem Makro DFHCLT TYPE=INITIAL wird der Einstiegspunkt und die Anfangsadresse der zu definierenden CLT (Command List Table; Befehlslistentabelle) eingerichtet.



Allgemeine Informationen zu Makros vom Typ TYPE=INITIAL einschließlich der Verwendung des Operanden SUFFIX finden Sie in TYPE=INITIAL (control section).

### JES={JES2 v JES3}

Gibt an, welche Version von JES verwendet wird. Wenn Sie über aktive und alternative CICS-Systeme in verschiedenen CPCs verfügen, müssen Sie in beiden CPCs die gleiche Version von JES verwenden.

Wenn Sie JES3 verwenden, benötigen Sie für die vollständige Unterstützung aller CLT-Funktionen in einer Umgebung mit zwei CPCs das Release 2.2.0. JES2 akzeptiert Befehle, die von Programmen ausgegeben werden. Für JES3 wurde diese Funktion nur für Release 2.2.0 eingeführt. Bei einem früheren JES3-Release ist es möglich, dass eine Übernahme in einer Umgebung mit zwei CPCs nur dann fortgesetzt werden kann, wenn der Bediener zuvor die Ausführung des fehlerhaften aktiven CICS-Systems manuell abgebrochen hat. Diese Situation tritt normalerweise nur auf, wenn das aktive System nicht erkennt, dass ein Fehler aufgetreten ist und die Ausführung fortsetzt.

### JESCHAR=wert

Gibt das aus einem Zeichen bestehende Präfix an, das für Befehle verwendet werden soll, die an JES übergeben werden. Wenn Sie dieses Schlüsselwort nicht angeben, gilt Folgendes:

- JESCHAR=\$ ist der Standardwert für JES=JES2
- JESCHAR=\* ist der Standardwert für JES=JES3

### JESID=((mvs-name, jes-name, spoolnummer) [, (mvs-name, jes-name, spoolnummer), ...])

Gibt den JES-Routing-Code an, der dem MVS- und dem JES-Namen eines aktiven CICS-Systems entspricht. Sie müssen diese Option verwenden, wenn aktive und alternative CICS-Systeme sich in unterschiedlichen CPCs befinden.

Sie können verschiedene Gruppen von *mvs-name*, *jes-name* und *spoolnummer* angeben, sodass die CLT verwendet werden kann, um auf weitere CPC/JES-Kombinationen zu verweisen.

#### mvs-name

Dies ist die SID, die im SYS1.PARMLIB-Member SMFPRMxx für den CPC angegeben wurde, auf dem das aktive CICS-System ausgeführt wird.

#### jes-name

Dies ist der Name des JES2- oder JES3-Subsystems für das JES-System, unter dessen Steuerung das aktive System ausgeführt wird. Er wird im MVS/ESA-Systemgenerierungsmakro SCHEDULR definiert und außerdem im SYS1.PARMLIB-Member IEFSSNxx.

### **spoolnummer**

Für JES2 wird hier die Spool-Member-Nummer für den Mehrfachzugriff von JES2 für das aktive CICS-System angegeben. Sie wird im JES2-Initialisierungsparameter MASDEF SSID(*n*) definiert. Für JES3 ist dies der Prozessname von JES3 für das aktive CICS-System. Die Definition erfolgt im JES3-Initialisierungsparameter MAINPROC NAME=*name*. Weitere Informationen hierzu finden Sie in z/OS JES2 Initialization and Tuning Guide oder z/OS JES3 Initialization and Tuning Guide.

## **Alternative Systeme angeben: DFHCLT TYPE=LISTSTART**

Dieses Makro definiert den Start der Gruppe von Befehlen und Nachrichten, die vom alternativen CICS-System abgesetzt werden, wenn es die Steuerung vom aktiven CICS-System übernimmt. (Möglicherweise gibt es keine Befehle oder Nachrichten, aber Sie benötigen trotzdem eine CLT, damit Berechtigungsprüfungen durchgeführt werden können.)

►►DFHCLT—TYPE=LISTSTART—————►  
►—,FORALT=(*anwendungs-id1*,*jobname1*)———►  
  └, (*anwendungs-id2*,*jobname2*),...┘

**FORALT=((*anwendungs-id1*,*jobname1*) [, (*anwendungs-id2*,*jobname2*), ...] )**

Gibt Paare alternativer und aktiver CICS-Systeme an.

### **anwendungs-id1**

Der Name des alternativen CICS-Systems, das die Gruppe von Befehlen und Nachrichten absetzt, wenn es die Steuerung übernimmt. Dieser Name muss eine **spezifische Anwendungs-ID (APPLID)** sein, die im Systeminitialisierungsparameter APPLID definiert ist. Sie wird für die Berechtigungsprüfung verwendet.

### **jobname1**

Der Name des aktiven CICS-Systems, von dem das alternative System die Steuerung übernimmt. Dieser Name muss der **MVS-Jobname (JOBNAME)** für das aktive CICS-System sein. Er wird zur Sicherheitsprüfung verwendet, um sicherzustellen, dass das alternative System nicht versucht, Jobs mit anderen Namen abzubereiten.

Sie können hier weiterführende Aktionen ausführen, wobei mehr Paare aus *anwendungs-id* und *jobname* verwendet werden, sodass Sie eine Befehlslistentabelle (CLT = Command List Table) für mehrere alternative CICS-Systeme verwenden können.

## **Übernahmebefehle angeben: DFHCLT TYPE=COMMAND**

Mit diesem Makro können Sie die Befehle angeben, die vom alternativen CICS-System während einer Übernahme verwendet werden sollen.

►►DFHCLT—TYPE=COMMAND—,COMMAND=befehlszeichenfolge—————►

### **COMMAND=befehlszeichenfolge**

Definiert einen Befehl, der zur Ausführung an MVS übergeben wird. Dieser Befehl wird von CICS nicht interpretiert.

Der Befehl, der am häufigsten in dieser Weise abgesetzt wird, ist CEBT PERFORM TAKEOVER.

Beim Mehrregionenbetrieb (MRO = Multiregion Operation), bei dem eine einfache Hierarchie aus **Masterregionen** und **abhängigen Regionen** besteht, kann eine Masterregion, in der ein Fehler auftritt, diesen Befehl an alle abhängigen Regionen absetzen, wenn auch diese in einen anderen CPC verschoben werden müssen.

In MRO-Umgebungen mit einem höheren Komplexitätsgrad kann eine Masterregion, in der ein Fehler auftritt, diesen Befehl an die **Koordinatorregion** absetzen und der Koordinator kann denselben Befehl an andere Masterregionen und abhängige Regionen in derselben Hierarchie von Regionen absetzen. Aus diesem Grund können mehrere MRO-Regionen gemeinsam in einen anderen CPC verschoben werden, ohne dass hierzu ein Bedienereingriff erforderlich ist.

Im Folgenden sind einige Beispiele aufgeführt:

- Eine Masterregion ohne Koordinator sendet einen Befehl an eine abhängige Region:

```
DFHCLT TYPE=COMMAND,
      COMMAND='MODIFY CICSDEP,CEBT PERFORM
      TAKEOVER'
```

- Eine Masterregion sendet einen Befehl an die zugehörige Koordinatorregion:

```
DFHCLT TYPE=COMMAND,
      COMMAND='MODIFY CICSCRD,CEBT PERFORM
      TAKEOVER'
```

- Eine Koordinatorregion sendet Befehle an Masterregionen und abhängige Regionen:

```
DFHCLT TYPE=COMMAND,
      COMMAND='MODIFY CICSMA,CEBT PERFORM
      TAKEOVER'
DFHCLT TYPE=COMMAND,
      COMMAND='MODIFY CICSDEP,CEBT PERFORM
      TAKEOVER'
```

- Sie können auch andere Befehle an andere Jobs absetzen, die unter MVS ausgeführt werden:

```
DFHCLT TYPE=COMMAND,
      COMMAND='MODIFY jobname,command
      string'
```

## Nachrichten an Bediener: DFHCLT TYPE=WTO

Diese beiden Anweisungen definieren eine Nachricht, die an den Systembediener ausgegeben wird.

```
►►DFHCLT—TYPE=WTO—,WTOL=adresse—adresse—WTO—'nachricht_an_bedienener'—►
►└┐,ROUTCDE=(zahl)└┐└┐,DESC=(zahl)└┐,MF=L—►◄
```

### WTOL=adresse

Gibt die Adresse eines WTO-Makros im Listenformat an, das die Nachricht und alle zugehörigen Routen- und Deskriptorcodes definiert.

Die Operanden MF (Makroformat), ROUTCDE (Routing-Code) und DESC (Deskriptor) des WTO-Makros werden in z/OS MVS Programming: Authorized Assembler Services Reference (Volume 1) beschrieben.

Beispiel: Senden einer Anforderung an den Bediener.

```

                DFHCLT TYPE=WTO,
                WTOL=wtoad
wtoad      WTO  'switch local terminals, please',
                MF=L

```

## Befehlsliste schließen: DFHCLT TYPE=LISTEND

In dieser Anweisung wird das Ende der Befehlsgruppe und der Nachrichten definiert, die von einem alternativen System abgesetzt werden, wenn es die Steuerung von einem aktiven System übernimmt.

```

▶▶—DFHCLT—TYPE=LISTEND—▶▶

```

---

## PDIR: DL/I-Verzeichnis

Beim PDIR (Program Specification Block Directory; Verzeichnis für Programmspezifikationsblöcke) handelt es sich um ein Verzeichnis aller fernen PSBs (Remote Program Specification Blocks; ferne Programmspezifikationsblöcke), auf die über das CICS-System zugegriffen wird.

Für die Verwendung von fernem DL/I müssen Sie das PDIR für CICS definieren. Für DBCTL müssen Sie die DL/I-Verzeichnisse mithilfe der von IMS bereitgestellten Makros für DBCTL definieren.

Um ein Verzeichnis für Programmspezifikationsblöcke (PDIR) zu erstellen, müssen Sie die Makros DFHDLPSB TYPE=INITIAL, TYPE=ENTRY und TYPE=FINAL angeben. Nach TYPE=FINAL (siehe TYPE=FINAL (end of table)) muss Folgendes angegeben werden:

```

END DFSIDIR0

```

## Programmabschnitt: DFHDLPSB TYPE=INITIAL

Mit dem Makro TYPE=INITIAL können Sie den Programmabschnitt (CSECT) für die Tabelle einrichten und die erforderlichen Steueranweisungen des Verbindungseditors erstellen.

Das Makro DFHDLPSB TYPE=INITIAL weist das folgende Format und die folgenden Operanden auf:

```

▶▶—DFHDLPSB—TYPE=INITIAL—▶▶

```

Allgemeine Informationen zu den TYPE=INITIAL-Makros sowie zur Verwendung des Operanden SUFFIX finden Sie in TYPE=INITIAL (control section).

## Programmspezifikationsblöcke: DFHDLPSB TYPE=ENTRY

Mit dem Makro TYPE=ENTRY wird ein Eintrag definiert, der im PDIR generiert werden soll.

Das Makro DFHDLPSB TYPE=ENTRY weist das folgende Format und die folgenden Operanden auf:

```

▶▶—DFHDLPSB—TYPE=ENTRY—,PSB=psb-name—,MXSSASZ=wert—
                                     |
                                     |,RMTNAME=name|
▶▶

```

►,SYSIDNT=name◄

#### **TYPE=ENTRY**

Gibt an, dass ein Eintrag im PDIR generiert werden soll. Die maximale Anzahl von Einträgen, die in das PDIR aufgenommen werden können, beträgt 32760.

#### **PSB=psb-name**

Gibt den Namen des Programmspezifikationsblocks (PSB) an, auf den über die ferne DL/I-Komponente zugegriffen wird. Im Eintrag müssen Werte für die Operanden SYSIDNT und MXSSASZ (und wahlweise auch RMTNAME) angegeben werden.

#### **MXSSASZ=wert**

Gibt die maximale Größe eines Segmentsucharguments in Byte an, das für diesen PSB verwendet werden soll.

**Anmerkung:** Ein übermäßig hoher Wert für MXSSASZ wirkt sich negativ auf die Leistung aus und kann dazu führen, dass ein Datenstrom geliefert wird, der zu groß für das verbundene CICS-System ist.

#### **RMTNAME=name**

Gibt den Namen an, unter dem der PSB auf dem fernen System oder in der fernen Region bekannt ist. Der Standardwert ist der Wert für *psb-name*, der im Operanden PSB angegeben ist.

#### **SYSIDNT=name**

Gibt den aus vier Zeichen bestehenden alphanumerischen Namen des fernen Systems oder der fernen Region an, für das bzw. die der PSB angewendet werden kann. Der angegebene Name muss der Name der CONNECTION-Definition für das ferne System sein.

Wenn die SYSIDNT des lokalen Systems angegeben ist, wird die Anforderung an DBCTL weitergeleitet und nicht per Funktionsverlagerung an eine ferne Region übertragen. Auf diese Weise kann dasselbe PDIR auf beiden Seiten der Verbindung verwendet werden, wenn dies erforderlich ist. Allerdings ist es nicht erforderlich, bei der Kommunikation mit DBCTL über ein PDIR zu verfügen.

---

## **FCT: Dateisteuertabelle**

Die Dateisteuertabelle (FCT = File Control Table) beschreibt für CICS die BDAM-Benutzerdateien (BDAM = Basic Direct Access Method), die von der Dateiverwaltung verarbeitet werden.

CICS-Benutzerdateien entsprechen physischen Dateien, die für MVS definiert und dem CICS-System zugeordnet werden müssen, bevor sie verwendet werden können.

#### **Anmerkung:**

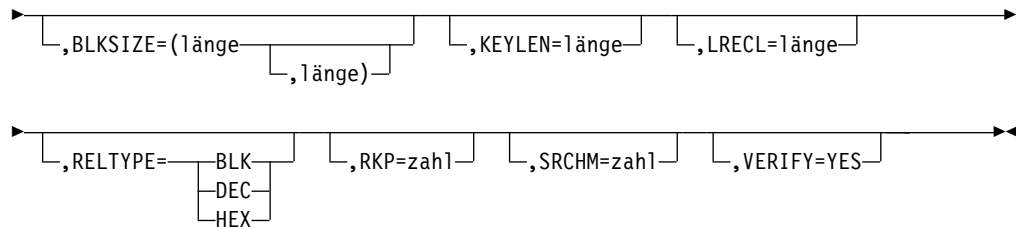
1. Verwenden Sie zum Definieren von VSAM-Dateien eine FILE-Ressource.
2. Da die CICS-Dateiverwaltung nur VSAM- und BDAM-Dateien verarbeitet, müssen Sie alle sequenziellen Dateien als partitionsübergreifende Ziele mit einer TDQUEUE-Ressource definieren.

Die folgenden Makros geben Dateimerkmale und einige der Merkmale von BDAM-Dateien an, auf die von den Dateien verwiesen wird:

- DFHFCT TYPE=INITIAL richtet den Anfang einer Dateisteuertabelle ein.







#### TYPE=FILE

Gibt an, dass dieses Makro die Merkmale einer Datei beschreibt.

#### ACCMETH=BDAM

Gibt an, dass der Datei die BDAM (Basic Direct Access Method) der Datei zugeordnet ist.

#### BLKKEYL=länge

Geben Sie diesen Operanden mit einem Dezimalwert zwischen 1 und 255 an, der die Länge in Byte des physischen Schlüssels im physischen BDAM-Datensatz darstellt. Sie müssen diesen Operanden nur für Dateien codieren, die auf Datasets mit physischen Schlüsseln verweisen (d. h. auf diejenigen, für die SERVREQ=KEY angegeben ist). Wenn eine Datei blockierte Datensätze enthält und die Entblockung mit einem logischen Schlüssel (d. h., einem in jedem logischen Datensatz eingebetteten Schlüssel) ausgeführt werden soll, muss die Länge des logischen Schlüssels mit dem Operanden KEYLEN angegeben werden.

Falls erforderlich, kann CICS einen Datensatz mit einem ENQ-Argument unter exklusive Steuerung stellen, indem der Dateiname, die Blockreferenz und der physische Schlüssel verknüpft werden. Eine ENQ-Anweisung wird unter Verwendung von maximal 255 Byte dieses Arguments abgesetzt. Überschreitet die Länge des Arguments 255 Byte, dann wird mit ENQ ein Bereich von Schlüsseln unter exklusive Steuerung gestellt.

#### BLKSIZE=(länge[,länge])

Geben Sie diesen Operanden mit der Länge des Blocks (in Byte) an. Das Verfahren zum Berechnen des Werts für BLKSIZE ist vom Wert für RECFORM abhängig. Für Blöcke mit der Angabe UNDEFINED oder VARIABLE muss als Länge die maximale Blocklänge angegeben werden. Für Blöcke mit der Angabe FIXED wird der Wert für BLKSIZE wie folgt berechnet:

$BLKSIZE = LRECL$  für entblockte Datensätze.

$BLKSIZE = (LRECL \times \text{Blockungsfaktor})$  für geblockte Datensätze.

Wenn der BLKSIZE-Wert im Datensteuerblock (DCB = Data Control Block) generiert werden soll, dann müssen Sie diesen Wert im zweiten Parameter des Operanden angeben. Beispiel: `BLKSIZE=(250,250)`. Hier bezieht sich die erste Angabe von 250 auf die Dateisteuertabelle (FCT = File Control Table) und die zweite Angabe von 250 auf den Datensteuerblock (DCB = Data Control Block). Wird der zweite Parameter nicht angegeben, dann wird der DCB ohne einen BLKSIZE-Wert generiert.

#### Anmerkung:

1. CICS setzt voraus, dass die Blockgröße der BDAM-Datei der Größe entspricht, die Sie im ersten Parameter BLKSIZE angegeben haben. Wenn der angegebene Wert kleiner als die tatsächliche Blockgröße der Datei ist, werden bei Verwendung der Datei möglicherweise Fehler im Speicher oder andere unvorhersehbare Ergebnisse angezeigt.

2. Wenn Sie den zweiten Parameter (DCB-Wert) angeben, muss der Wert, den Sie codieren, immer die tatsächliche Blockgröße sein. Es wird empfohlen, entweder den zweiten Parameter wegzulassen oder für ihn den gleichen Wert wie für den ersten Parameter anzugeben.

**DISP={OLDvSHR}**

Geben Sie diesen Operanden an, um die Disposition der Datei anzugeben, die dieser Datei zugeordnet wird. Wenn für diese Datei keine JCL-Anweisung vorhanden ist, wenn sie geöffnet wird, dann wird vor dem Öffnen eine dynamische Zuordnung der Datei mit dieser Disposition durchgeführt. Wenn eine JCL-Anweisung vorhanden ist, hat sie Vorrang vor dieser Disposition.

**OLD** Die Disposition der Datei wird auf OLD gesetzt, wenn die dynamische Zuordnung ausgeführt wird.

**SHR** Die Disposition der Datei wird auf SHR gesetzt, wenn die dynamische Zuordnung ausgeführt wird.

**Anmerkung:** Sie müssen die Disposition der Datei entweder mit dem Operanden DISP oder wie folgt angeben:

- In einer JCL-Anweisung
- Mit CEMT SET
- Mit EXEC CICS SET

Wenn Sie die Disposition in einer JCL-Anweisung angeben, dann geben Sie den Namen der Datei ebenfalls in der JCL-Anweisung an.

**DSNAME=name**

Geben Sie für den JCL-Dateinamen (DSNAME), der für diese Datei verwendet werden soll, einen Wert mit einer Länge zwischen 1 und 44 Zeichen an. Wenn für diese Datei keine JCL-Anweisung vorhanden ist, wenn sie geöffnet wird, dann wird vor dem Öffnen eine dynamische Zuordnung der Datei mit diesem Dateinamen (DSNAME) durchgeführt. Wenn eine JCL-Anweisung vorhanden ist, hat sie Vorrang gegenüber diesem Dateinamen (DSNAME).

Sie müssen den Dateinamen entweder mit dem Operanden DSNAME oder wie folgt angeben:

- In einer JCL-Anweisung
- Mit CEMT SET
- Mit EXEC CICS SET

Wenn Sie die Datei in einer JCL-Anweisung angeben, dann **müssen** Sie die Disposition ebenfalls in der JCL-Anweisung angeben.

**Anmerkung:** Sie definieren die Datei für die CICS-Systemdefinition (CSD-Datei) mit den Systeminitialisierungsparametern und nicht in der Dateisteuertabelle (FCT = File Control Table).

**FILE=name**

Geben Sie für diesen Operanden einen aus einem bis acht Zeichen bestehenden symbolischen Namen an, mit dem dieser FCT-Eintrag identifiziert werden kann. Dieser Name wird als **Dateiname** bezeichnet und wird von CICS oder den CICS-Anwendungsprogrammen verwendet, um auf die Datei zu verweisen, der dieser FCT-Eintrag zugeordnet wurde.

Zusätzlich zur Identifikation des FCT-Eintrags wird dieser Name auch als DD-NAME verwendet, wenn die zugeordnete Datei CICS zugeordnet wird. Die Zuordnung wird entweder mit JCL-Anweisungen im Jobstrom für den Start oder dynamisch mithilfe der Werte für DSNAME und DISP in der FCT durchgeführt.

Verwenden Sie für eigene Dateien keine Dateinamen, die mit der Zeichenfolge DFH beginnen, weil nur CICS dazu berechtigt ist, Dateinamen zu verwenden, die mit DFH beginnen. Darüber hinaus kann die Verwendung der Zeichenfolge FCT als Dateinamenspräfix zu Assemblierungsfehlern führen.

**FILSTAT=({ENABLED∨DISABLED∨UNENABLED},{OPENED∨CLOSED})**

Geben Sie diesen Operanden an, um den Anfangsstatus der Datei anzugeben.

Der erste Operand bestimmt den anfänglichen Aktivierungsstatus der Datei. Er wird nur während eines Erst- oder Kaltstarts verwendet. (Bei einem Warm- oder Notfallstart wird der Dateistatus anhand des Status bestimmt, der zum Zeitpunkt des vorherigen Systemabschlusses vorlag.)

Der zweite Operand gibt an, ob versucht wurde, die Datei am Ende der CICS-Initialisierung zu öffnen. Er gilt für Erst- sowie für Kalt-, Warm- und Notfallstarts.

**ENABLED**

Für diese Datei ist eine normale Verarbeitung zulässig.

**DISABLED**

Jede Anforderung für diese Datei, die von einem Anwendungsprogramm stammt, führt dazu, dass die Bedingung DISABLED an das Programm übergeben wird.

**UNENABLED**

Diese Option ist nur mit der Option CLOSED gültig. Sie kann verwendet werden, um zu verhindern, dass die Datei beim ersten Verweis geöffnet wird. Wird versucht, auf die Datei zuzugreifen, während sie sich in diesem Status befindet, führt dies zur Ausgabe der Bedingung NOTOPEN.

**OPENED**

Die Datei wird nach der CICS-Initialisierung durch eine automatisch eingeleitete CICS-Transaktion (CSFU) geöffnet. (Bei einem Warm- oder Notfallstart verbleibt eine Datei im Status UNENABLED, wenn dieser Status bereits zum Zeitpunkt des vorherigen Systemabschlusses vorlag. Bei einer CSFU-Transaktion wird die Option OPENED für eine Datei im Status UNENABLED ignoriert, sodass die Datei geschlossen bleibt.)

**CLOSED**

Die Datei bleibt so lange geschlossen, bis eine Anforderung zum Öffnen mit der Hauptterminalfunktion, dem Befehl EXEC CICS SET oder mit einer impliziten Öffnungsoperation angefordert wird.

Für die verschiedenen Kombinationen der Anfangsstatus werden Dateien wie folgt geöffnet:

**(ENABLED,CLOSED)**

Die Datei wird beim ersten Verweis geöffnet. Dies ist der Standardwert.

**(ENABLED,OPENED)**

Die Datei wird nach der CICS-Initialisierung durch die automatisch eingeleitete Transaktion (CSFU) geöffnet, es sei denn, eine Benutzeranwendung oder eine Hauptterminalfunktion hat sie zuerst geöffnet.

**(DISABLED,CLOSED)**

Die Datei wird nur durch eine explizite OPEN-Anforderung (z. B. der Hauptterminaltransaktion) geöffnet.

**(DISABLED,OPENED)**

Die Datei wird nach der CICS-Initialisierung durch die automatisch eingeleitete Transaktion (CSFU) geöffnet.

leitete Transaktion (CSFU) geöffnet, es sei denn, eine Benutzeranwendung oder eine Hauptterminalfunktion hat sie zuerst geöffnet.

**(UNENABLED,CLOSED)**

Die Datei wird nur durch eine explizite OPEN-Anforderung geöffnet. Nach dem Öffnen weist die Datei den Status (ENABLED,OPENED) auf.

**Anmerkung:** Aus Leistungsgründen wird der standardmäßige Eintrag der CSD-Datei für die CSFU-Transaktion mit DTIMOUT=10 (Sekunden) definiert. Dies kann zu einer abnormalen Beendigung aufgrund einer Transaktionszeitlimitüberschreitung führen, wenn beim CICS-Start eine Verzögerung beim Öffnen einer Datei auftritt. Eine Erläuterung zum DTIMOUT-Wert finden Sie im Abschnitt zu den TRANSACTION-Attributen.

**JID={NOvzahl}**

Geben Sie diesen Operanden an, wenn für diesen FCT-Eintrag die Aktivitäten für das automatische Journal ausgeführt werden sollen und wenn das Journal angegeben werden soll, das zum Aufzeichnen der Journaldaten verwendet werden soll. Die Operationen, die die Aufzeichnung von Datensätzen im Journal zur Folge haben, werden im Parameter JREQ angegeben.

**NO** Für diese Datei werden keine Aktivitäten für die automatische Journalführung ausgeführt.

**zahl**

Die Journalkennung, die für die automatische Journalführung verwendet werden soll. Hier kann eine beliebige Zahl im Bereich zwischen 01 und 99 angegeben werden. Die Zahl wird an die Buchstaben DFHJ angefügt, um einen Journalnamen im Format DFHJnn zu erhalten, der einem allgemeinen Protokolldatenstrom der MVS-Systemprotokollfunktion zugeordnet wird.

**Anmerkung:** Die automatische Journalführung kann angegeben werden, wenn die Dateiaktivität für die nachfolgende Verarbeitung durch Sie aufgezeichnet werden soll (z. B. zur E/A-Wiederherstellung für eine benutzerdefinierte Datei). Dieser Vorgang darf nicht mit der automatischen Protokollierung (Angabe LOG=YES) verwechselt werden, die erforderlich ist, wenn CICS für Dateien eine Zurücksetzung durchführen muss, um unvollständige Taskaktivitäten während eines Wiederanlaufs nach Systemabsturz oder beim dynamischen Zurücksetzen von Transaktionen zu entfernen.

**JREQ={ALLv(anforderung[,anforderung,...])}**

Geben Sie diesen Operanden für die Dateioperationen an, die automatisch im Journal aufgezeichnet werden sollen. Außerdem können Sie angeben, ob die Journalführungsoperation **synchron** oder **asynchron** zur Dateiaktivität ausgeführt werden soll.

Wenn eine synchrone Journaloperation für eine READ-Anforderung ausgeführt wird, dann wird die Steuerung erst an das Programm zurückgegeben, von dem die Dateisteuerungsanforderung abgesetzt wurde, nachdem die Datenleseoperation in die Journaldatei geschrieben wurde. Wenn eine synchrone Journaloperation für eine WRITE-Anforderung ausgeführt wird, dann wird die Ausgabeoperation in die Datei erst dann eingeleitet, nachdem die Daten in die Journaldatei geschrieben wurden.

Wenn eine asynchrone Journaloperation für eine READ-Anforderung ausgeführt wird, dann kann die Steuerung zurückgegeben werden, sobald die gelesenen Daten in den E/A-Puffer des Journals verschoben wurden. Wenn eine asynchrone Journaloperation für eine WRITE-Anforderung ausgeführt wird,

dann kann die Ausgabeoperation in die Datei eingeleitet werden, sobald die Daten in den E/A-Puffer des Journals verschoben wurden.

Synchronisationsstandardwerte stellen eine asynchrone Operation für READ-Anforderungen und eine synchrone Operation für WRITE-Operationen bereit.

Wenn Sie die automatische Journalführung angefordert haben, dann geben die Inhalte des Journals die tatsächlichen Änderungen an einer Datei möglicherweise nicht exakt wieder, weil die Anforderung im Journal aufgezeichnet wird, bevor die Antwort der E/A-Operation getestet wurde.

Wenn dieser Operand weggelassen wird und JID angegeben wird, dann wird für JREQ der Standardwert JREQ=(WU,WN) verwendet.

Folgende Werte für *anforderung* sind möglich:

- ALL** Alle Dateiaktivitäten mit asynchronen READ- und synchronen WRITE-Operationen werden im Journal aufgezeichnet.
- ASY** Für WRITE-Operationen wird eine asynchrone Journaloperation ausgeführt.
- RO** Für READ ONLY-Operationen wird eine Journaloperation ausgeführt.
- RU** READ UPDATE-Operationen werden im Journal aufgezeichnet.
- SYN** Für READ-Operationen wird eine synchrone Journaloperation ausgeführt.
- WN** WRITE NEW-Operationen werden im Journal aufgezeichnet.
- WU** WRITE UPDATE-Operationen werden im Journal aufgezeichnet.

#### **KEYLEN=länge**

Geben Sie diesen Operanden mit der Länge des logischen Schlüssels für die Entblockung der BDAM-Datei an, auf die diese Datei verweist.

Der logische Schlüssel für BDAM-Dateien wird mithilfe des Operanden RKP eingebettet und lokalisiert. Die Länge des physischen Schlüssels wird im Operanden BLKKEYL angegeben. Sie kann sich von dem Wert unterscheiden, der für KEYLEN angegeben wurde.

Dieser Operand muss immer angegeben werden, wenn logische Schlüssel in geblockten BDAM-Dateien verwendet werden.

#### **LOG={NO✓YES}**

Dieser Operand gibt die Wiederherstellungsattribute der Datei an. Geben Sie LOG=YES an, wenn die automatische Protokollierung verwendet werden soll. Auf diese Weise können Sie unvollständige Änderungen an der Datei zurücksetzen (Wiederherstellung), auf die durch diese Datei verwiesen wird, wenn ein Wiederanlauf nach Systemabsturz oder der Abbruch einer Transaktion aufgetreten ist. Wenn eine Änderung (Aktualisierung, Löschung oder Hinzufügung) für die Datei ausgeführt wurde, wird das Image vor der Änderung automatisch im CICS-Systemprotokoll aufgezeichnet. (Die automatische Protokollierung darf nicht mit der automatischen Journalführung verwechselt werden.)

**NO** Die automatische Protokollierung wird nicht ausgeführt.

**YES**

Die automatische Protokollierung wird ausgeführt.

Wenn eine Anforderung zum Ändern des Inhalts der Datei abgesetzt wird, auf die von der Datei verwiesen wird, dann wird der Datensatz, der aktualisiert, hinzugefügt oder gelöscht werden soll, in die Warteschlange einge-

stellt. Dabei wird die Datensatz-ID zusammen mit der Adresse des CICS-Steuerblocks verwendet, der der Basisdatei zugeordnet ist. Diese Einreihung in die Warteschlange (sog. Enqueue) wird beibehalten, bis die Task abgeschlossen ist oder die Anwendung eine Synchronisationspunktanforderung absetzt, um das Ende einer logischen Arbeitseinheit zu signalisieren. Auf diese Weise wird die Integrität der geänderten Daten sichergestellt.

Da die Enqueues somit über einen längeren Zeitraum erhalten bleiben, kann es zu einer Enqueue-Aussperrung kommen, wenn ein Anwendungsprogramm, das auf diese Datei zugreift, mehrere logische Arbeitseinheiten für die Datei ausführt, ohne jede logische Arbeitseinheit separat für CICS zu definieren, indem eine Synchronisationspunktanforderung abgesetzt wird. Auch Tasks mit langer Laufzeit können Speicherressourcen belegen.

#### **LRECL=länge**

Geben Sie diesen Operanden mit der maximal zulässigen Länge (in Byte) des logischen Datensatzes an. Der angegebene Wert gibt auch die Länge der Datensätze in einer fernen Datei mit fester Länge an. Weitere Informationen zu fernen Dateien finden Sie im Abschnitt zum Makro DFHFCT TYPE=REMOTE.

#### **RECFORM=([{UNDEFINED∨VARIABLE∨FIXED}], [{BLOCKED∨UNBLOCKED}], [DCB format])**

Geben Sie diesen Operanden an, um das Format von physischen Datensätzen in der Datei zu beschreiben.

Für BDAM-Dateien bezieht sich der Begriff **Blockung** auf die CICS-Blockung, die keine Bedeutung für BDAM hat. Sie müssen BLOCKED oder UNBLOCKED für alle Dateien angeben, für deren Format FIXED oder VARIABLE angegeben wurde.

#### **BLOCKED**

Geben Sie diese Option an, wenn jeder physische Datensatz von CICS als ein Block betrachtet werden soll, der aus mehr als einem logischen Datensatz besteht.

#### **DCB**

Geben Sie diese Option zusammen mit dem Datensatzformat an, das in den DCB eingefügt werden soll. Beispiel:  
RECFORM=(FIXED,BLOCKED,FBS).

Der Unterparameter für das DCB-Format des Operanden RECFORM stellt die einzige Möglichkeit dar, Informationen zum Datensatzformat in den DCB einzubinden, wenn die Dateisteuertabelle (FCT = File Control Table) assembliert wird. Die ersten beiden Unterparameter des Operanden RECFORM generieren keine Informationen im DCB.

#### **FIXED**

Die Datensätze weisen eine feste Länge auf.

#### **UNBLOCKED**

Geben Sie diese Option an, wenn keine CICS-Blockstruktur verwendet werden soll. Dies bedeutet, dass ein logischer CICS-Datensatz für jeden physischen BDAM-Datensatz vorhanden ist.

#### **UNDEFINED**

Die Länge der Datensätze ist nicht definiert. (Wenn Sie eine Datei als UNDEFINED festlegen, dann sollten Sie acht zusätzliche Byte für das Zählerfeld vorsehen, wenn Sie den Wert für BLKSIZE berechnen.)

#### **VARIABLE**

Die Datensätze weisen eine variable Länge auf.

**RELTYPE={BLKvDECvHEX}**

Geben Sie diesen Operanden an, wenn die relative Adressierung im Blockreferenzabschnitt des Felds für die Datensatzkennung der BDAM-Datei verwendet werden soll, auf die durch diese Datei verwiesen wird. Wenn der Operand RELTYPE nicht angegeben wird, dann wird die absolute Adressierung verwendet (MBBCCHHR).

**BLK** Das System verwendet die relative Blockadressierung.

**DEC** Das System verwendet das gezonte Dezimalformat.

**HEX** Das System verwendet das hexadezimale relative Track- und Datensatzformat.

**RKP=zahl**

Geben Sie diesen Operanden mit der Startposition des Schlüsselfelds in dem Datensatz relativ zum Anfang des Datensatzes an. Bei Datensätzen mit variabler Länge muss dieser Operand Speicherplatz für das vier Byte umfassende LLbb-Feld am Anfang jedes logischen Datensatzes umfassen. Dieser Operand muss für Dateien, bei denen jeder logische Datensatz Schlüssel enthält, oder beim Browsing immer angegeben werden.

**SERVREQ=(anforderung[,anforderung],...)**

Geben Sie diesen Operanden an, um die Typen der Serviceanforderungen zu definieren, die für die Datei verarbeitet werden können. Die folgenden Parameter können eingebunden werden:

**ADD**

Dieser Operand gibt an, dass Datensätze zur Datei hinzugefügt werden können.

**BROWSE**

Dieser Operand gibt an, dass Datensätze in sequenzieller Reihenfolge aus der Daten abgerufen werden können.

**KEY**

Diese Option gibt an, dass Datensätze aus der Datei abgerufen oder zu der Datei hinzugefügt werden können. Dieser Parameter muss angegeben werden, wenn die Datei, auf die durch die Datei verwiesen wird, eine BDAM-Datei mit Schlüssel ist. Für andere Dateien muss die Option nicht angegeben werden.

**NOEXCTL**

Diese Option gibt an, dass Datensätze nicht der exklusiven Steuerung unterstellt werden sollen, wenn ein Lesevorgang zur Aktualisierung angefordert wird.

Wenn Sie NOEXCTL nicht angeben, dann wird standardmäßig die exklusive BDAM-Steuerung bereitgestellt. Auf diese Weise wird die Integrität auf dem System sichergestellt. Für BDAM können Sie bei SERVREQ=NOEXCTL die Einstellung LOG=YES angeben. Auf diese Weise wird lediglich eine CICS-Enqueue angefordert, die exklusive BDAM-Steuerung jedoch unterdrückt. Dadurch wird die CICS-Integrität für die Aktualisierung nur bis zu einem Synchronisationspunkt bereitgestellt.

**Anmerkung:** Die CICS-Enqueue wird auf Datensezebene in der CICS-Region ausgeführt und bis zu einem Synchronisationspunkt beibehalten. Die exklusive BDAM-Steuerung arbeitet hingegen auf Basis eines physischen Blocks, wird systemweit benutzt und nur solange aufrecht erhalten, bis die Aktualisierung abgeschlossen ist.



**READ**

Diese Option gibt an, dass die Datensätze in dieser Datei gelesen werden können. Wenn Sie BROWSE oder UPDATE angeben, dann nimmt das System READ an.

**UPDATE**

Diese Option gibt an, dass die Datensätze in dieser Datei geändert werden können.

**SRCHM=zah1**

Geben Sie diesen Operanden an, wenn die Suche über mehrere Spuren nach Datensätzen mit Schlüssel bereitgestellt werden soll. Dieser Operand kann nur für BDAM-Dateien mit Schlüssel angegeben werden.

**zah1**

Die Anzahl der Spuren oder Blöcke, die durchsucht werden sollen. Der Standardwert ist 0.

**VERIFY=YES**

Geben Sie diesen Operanden an, wenn Sie die Parität von Plattendatensätzen überprüfen wollen, nachdem sie geschrieben wurden. Wenn dieser Operand nicht angegeben wird, werden die Datensätze nach einer Schreibanforderung nicht überprüft.

**Zusammenfassungstabelle**

Dieser Abschnitt soll Sie bei der Verwendung des Makros DFHFCT TYPE=FILE zum Definieren Ihrer Dateien unterstützen. Jede Anweisung TYPE=FILE beschreibt die Merkmale der Datei und auch der Datei, auf die die Datei verweist.

Tabelle 12. DFHFCT TYPE=FILE-Anweisungen für BDAM-Dateien

	Geblockt mit Schlüssel	Geblockt ohne Schlüssel	Entblockt mit Schlüssel	Entblockt ohne Schlüssel
BLKKEYL	E		E	
SRCHM	O		O	
VERIFY	O	O	O	O
RELTYPE	E <sup>1</sup>	E <sup>1</sup>	E <sup>1</sup>	E <sup>1</sup>
LRECL	E	E	E	E
BLKSIZE	E <sup>3</sup>	E	E <sup>2</sup>	E
KEYLEN	E <sup>5</sup>			
RKP	E <sup>4</sup>		E <sup>4</sup>	
RECFORM	O	O	O	O
FILSTAT	O	O	O	O
SERVREQ	E <sup>6</sup>	O	E <sup>6</sup>	O

**Hinweise:**

**E**      Erforderlich

**O**      Optional

1. Erforderlich, wenn die relative Typadressierung verwendet werden soll.
2. Bei Angabe von SERVREQ=BROWSE oder SERVREQ=ADD muss dieser Wert für entblockte Datensätze BLKSIZE + BLKKEYL sein.
3. Bei Angabe von SERVREQ=BROWSE oder SERVREQ=ADD muss dieser Wert für geblockte Datensätze (LRECL x Blockungsfaktor) + BLKKEYL sein.
4. Erforderlich, wenn ein Schlüssel in den logischen Datensätzen vorhanden ist.
5. Erforderlich, wenn die Entblockung nach Schlüssel für BDAM ausgeführt wird.
6. SERVREQ=KEY ist erforderlich.

## DFHFCT-Beispiel

Ein Beispiel für einen FCT-Eintrag für eine BDAM-Datei.

In Abb. 7 wird die Codierung dargestellt, die zum Erstellen eines FCT-Eintrags für eine BDAM-Datei erforderlich ist.

```
DFHFCT TYPE=FILE,
        FILE=DAM83,
        ACCMETH=BDAM,
        SERVREQ=(READ,BROWSE,KEY),
        BLKSIZE=172,
        RECFORM=(FIXED,BLOCKED),
        LRECL=86,
        RELTYPE=HEX,
        KEYLEN=6,
        BLKKEYL=6,
        RKP=0,
        FILSTAT=(ENABLED,OPENED)
```

```
*
*
*
*
*
*
*
*
*
*
*
```

Abbildung 7. Beispiel für Dateisteuertabelle: BDAM-Datei

## MCT: Überwachungssteuertabelle

Die Überwachungssteuertabelle (MCT = Monitoring Control Table) definiert die Benutzerdatenfelder in CICS-Leistungsklassendatensätzen für die Überwachung und beschreibt, wie sie bei Ereignisüberwachungspunkten (EMPs = Event Monitoring Points) bearbeitet werden. Darüber hinaus dient sie dazu, die Auswahl der systemdefinierten Felder für die Leistungsklassendaten, die aufgezeichnet werden sollen, zu steuern.

Detaillierte Informationen zur Überwachungssteuertabelle und zur Leistung finden Sie in The monitoring control table (MCT).

Die Überwachungssteuertabelle wird nur in folgenden Situationen benötigt:

- Beim Aufruf von EMPs in Ihren Anwendungsprogrammen.
- Beim Ausschluss bestimmter systemdefinierter Felder von der Aufzeichnung.
- Bei Verwendung von Überwachungsressourcenklassen.
- Bei der Erfassung zusätzlicher Überwachungsleistungsdaten für die Ressourcenmanager, die von Ihrer Transaktion verwendet werden.
- Beim Inaktivieren der Datenkomprimierung für Überwachungsdatensätze.

Wenn keine Überwachungssteuertabelle in der CICS-Region vorhanden ist, dann verwendet CICS die folgenden Standardwerte:

- Die Leistungs-, Transaktionsressourcen- und Ausnahmeüberwachungsklassen sind verfügbar.
- Alle systemdefinierten CICS-Datenfelder werden erfasst.
- Die Datenkomprimierung wird für Überwachungsdatensätze ausgeführt.

In Ihren Anwendungsprogrammen können Sie EMPs unter Verwendung von MONITOR aufrufen.

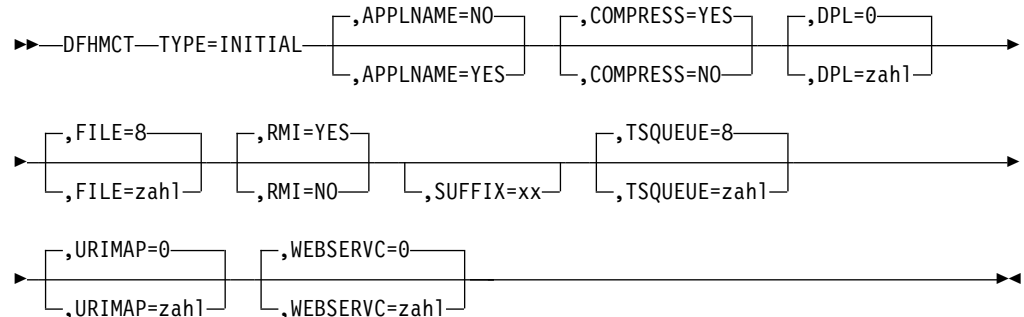
Die Überwachungssteuertabelle besteht aus den folgenden Makroinstruktionen:

- Programmabschnitt DFHMCT TYPE=INITIAL

- Ereignisüberwachungspunkte des Benutzers (DFHMCT TYPE=EMP)
- Steuerdatenaufzeichnung (DFHMCT TYPE=RECORD)
- Ende der Überwachungssteuertabelle (DFHMCT TYPE=FINAL; siehe TYPE=FINAL (end of table))

## Programmabschnitt: DFHMCT TYPE=INITIAL

Der Programmabschnittsname für die MCT (Monitoring Control Table; Überwachungssteuertabelle) wird mit dem Makro DFHMCT TYPE=INITIAL eingerichtet. Dieses Makro dient außerdem zur Erstellung der Steueranweisungen für den Verbindungseitor, die für die nachfolgende Programmverbindung erforderlich sind.



Allgemeine Informationen zu Makros vom Typ TYPE=INITIAL einschließlich der Verwendung des Operanden SUFFIX finden Sie in TYPE=INITIAL (control section).

### Anmerkung:

1. Die kombinierte Länge aller Daten für die Transaktionsressourcenüberwachung darf den Wert von 32244 nicht überschreiten.
2. Einige Optionen gelten nur, wenn die Transaktionsressourcenüberwachung aktiviert ist. Sie können die Transaktionsressourcenklassenüberwachung beim Start aktivieren, indem Sie MNRES=ON (zusammen mit MN=ON) als Systeminitialisierungsparameter angeben. Alternativ hierzu können Sie auch die Überwachungsfunktionstransaktion CEMN, den Befehl CEMT oder **EXEC CICS SET MONITOR** verwenden, um die Transaktionsressourcenüberwachung dynamisch zu aktivieren.

### APPLNAME={NO|YES}

Diese Option gibt an, dass Sie die Unterstützung für die Anwendungsbenennung verwenden möchten, die von der CICS-Überwachung bereitgestellt wird.

Die Anwendungsbenennung ist eine Aktivierungsfunktion, die es Anwendungsprogrammen ermöglicht, spezielle CICS-Ereignisüberwachungspunkte aufzurufen. Daten, die bei diesen von CICS generierten EMPs erfasst werden, können von jedem CICS-Überwachungssoftwarepaket verwendet werden.

**NO** Die Unterstützung für die Anwendungsbenennung ist in der CICS-Region nicht aktiviert und die Ereignisüberwachungspunkte für die Anwendungsbenennung DFHAPPL.1 und DFHAPPL.2 werden nicht generiert.

**YES** Die Anwendungsbenennungsunterstützung wird in der CICS-Region aktiviert. Wenn Sie die MCT (Monitoring Control Table; Überwachungssteuertabelle) assemblieren, generiert CICS die Ereignisüberwachungspunkte für die Anwendungsbenennung (DFHAPPL.1 und DFHAPPL.2). Beachten Sie hierbei, dass die Überwachungsdaten, die

bei diesen EMPs mit einem Befehl **MONITOR** verschoben werden, der diese EMPs für die Anwendungsbenennung aufruft, bis zum Abschluss der Task oder aber bis zur Änderung der Task durch einen weiteren Aufruf der EMPs mit einem nachfolgenden Befehl **EXEC CICS MONITOR** aufbewahrt werden.

Die EMPs für die Anwendungsbenennung (DFHAPPL) werden von CICS so erstellt, als ob sie mit den Parametern des Makros **TYPE=EMP** definiert worden wären:

```
DFHMCT TYPE=EMP,CLASS=PERFORM,                                X
                                ID=(DFHAPPL.1),FIELD=(1,APPLNAME) X
                                PERFORM=(MOVE(0,4))
DFHMCT TYPE=EMP,CLASS=PERFORM,                                X
                                ID=(DFHAPPL.2),                  X
                                PERFORM=(MOVE(4,8))
```

Weitere Informationen zur Vorgehensweise bei der Verwendung der Ereignisüberwachungspunkte für die Anwendungsbenennung in Ihren Anwendungen finden Sie in Application naming event monitoring points.

#### **COMPRESS={YES|NO}**

Diese Option gibt an, ob die Datenkomprimierung für die CICS SMF 110-Überwachungsdatensätze ausgeführt werden soll, die von der CICS-Überwachungsfunktion generiert werden.

**YES** Bei Verwendung dieses Standardwerts wird CICS angewiesen, die Datenkomprimierung für die CICS SMF 110-Überwachungsdatensätze, die von der CICS-Überwachungsfunktion erstellt werden, auszuführen. Informationen zur Datenkomprimierung finden Sie in Data compression for monitoring records.

**NO** Die Datenkomprimierung für die CICS SMF 110-Überwachungsdatensätze wird inaktiviert.

#### **DPL={0|zahl}**

Diese Option gibt die maximal zulässige Anzahl von DPL-Anforderungen (DPL = Distributed Program Link; Verbindung zu verteilten Programmen) an, für die CICS die Transaktionsressourcenüberwachung durchführen soll. Diese Option gilt nur, wenn die Transaktionsressourcenüberwachung aktiviert ist (siehe hierzu Hinweis 2).

Die standardmäßigen CICS-Leistungsklassendaten für die Überwachung umfassen Summen für *alle* Programme (einschließlich der DPLs), auf die über eine Transaktion zugegriffen wird. Die Transaktionsressourcenüberwachung hingegen erfasst Informationen zu einzelnen DPLs (Distributed Program Links), und zwar bis zu der angegebenen Anzahl. Die folgenden Daten werden erfasst:

- Programmname
- System-ID (sysid), an die die Anforderung weitergeleitet wird
- Anzahl der DPL-Anforderungen (DPL = Distributed Program Link)

**0** Diese Standardeinstellung inaktiviert die Transaktionsressourcenüberwachung für alle DPLs auch dann, wenn MNRES=ON aktiv ist.

**zahl** Gibt die maximal zulässige Anzahl von DPL-Anforderungen im Bereich zwischen 0 und 64 an, für die CICS die Transaktionsressourcenüberwachung ausführen soll. CICS erfasst die Überwachungsleistungsdaten auf Ressourcenebene für jede DPL-Anforderung, die von einer Transaktion abgesetzt wird, und zwar bis zu dem in *zahl* angegebenen Maximum. Wenn die Transaktion eine höhere Anzahl von DPL-Anforderungen absetzt, als in dieser Variablen angegeben ist, dann werden

die Anforderungen, die das Maximum übersteigen, ignoriert. Allerdings erhalten sie ein Flag, das anzeigt, dass die Transaktion das DPL-Limit überschritten hat.

**FILE={8|zahl}**

Diese Option gibt die maximal zulässige Anzahl von Dateien an, für die CICS die Transaktionsressourcenüberwachung durchführen soll. Diese Option gilt nur, wenn die Transaktionsressourcenüberwachung aktiviert ist (siehe hierzu Hinweis 2).

Die standardmäßigen CICS-Leistungsklassendaten für die Überwachung umfassen Summen für *alle* Dateien, auf die über eine Transaktion zugegriffen wird. Die Transaktionsressourcenüberwachung hingegen erfasst Informationen zu einzelnen Dateien, und zwar bis zu der angegebenen Anzahl. Die folgenden Daten werden erfasst:

- Dateiname
- Anzahl und Gesamtzeitdauer von GET-Anforderungen für Dateien
- Anzahl und Gesamtzeitdauer von PUT-Anforderungen für Dateien
- Anzahl und Gesamtzeitdauer von Anforderung zum Durchsuchen von Dateien
- Anzahl und Gesamtzeitdauer von Anforderung zum Hinzufügen von Dateien
- Anzahl und Gesamtzeitdauer von Anforderung zum Löschen von Dateien
- Gesamtzahl und Gesamtzeitdauer aller Anforderungen für die Datei
- Anzahl der Anforderungen für die Dateizugriffsmethode
- Wartezeit und Anzahl der Wartezyklen bei der Dateiein/-ausgabe
- Wartezeit und Anzahl der Wartezyklen bei der Dateiein/-ausgabe (RLS-Modus)
- Wartezeit und Anzahl der Wartezyklen bei der Dateiein/-ausgabe für Coupling-Facility-Datentabellen (CFDTs)
- Wartezeit und Anzahl der Wartezyklen für die exklusive Steuerung
- Wartezeit und Anzahl der Wartezyklen für VSAM-Zeichenfolgen

**8** Mit dieser Standardeinstellung wird CICS angewiesen, die Transaktionsressourcenüberwachung für maximal acht Dateien auszuführen.

**zahl** Gibt die maximale Anzahl von Dateien im Bereich zwischen 0 und 64 an, für die CICS die Transaktionsressourcenüberwachung ausführen soll. CICS erfasst die Überwachungsleistungsdaten auf Ressourcenebene für jede Datei, auf die von einer Transaktion zugegriffen wird, und zwar bis zu dem in *zahl* angegebenen Maximum. Wenn die Transaktion auf eine höhere Anzahl von Dateien zugreift, als in dieser Variablen angegeben ist, dann werden die Dateien, die das Maximum übersteigen, ignoriert. Allerdings erhalten sie ein Flag, das anzeigt, dass die Transaktion das Dateilimit überschritten hat.

Wenn Sie FILE=0 angeben, dann werden die Daten der Transaktionsressourcenüberwachung für Dateien nicht erfasst, auch wenn MNRES=ON aktiv ist.

**RMI={YES|NO}**

Diese Option gibt an, ob zusätzliche Leistungsklassendaten für die Überwachung für die Ressourcenmanager erfasst werden sollen, die von Ihren Transaktionen verwendet werden.

**YES** Diese Standardeinstellung weist CICS an, zusätzliche Überwachungsleistungsdaten für die Ressourcenmanager zu erfassen, die von Ihren Transaktionen verwendet werden.

Informationen zu den erfassten Daten finden Sie in Performance data in group DFHRMI.

**NO** Diese Einstellung weist CICS an, keine zusätzlichen Überwachungsleistungsdaten für die Ressourcenmanager zu erfassen, die von Ihren Transaktionen verwendet werden.

**TSQUEUE={8|zahl}**

Diese Option gibt die maximal zulässige Anzahl von Warteschlangen für temporären Speicher an, für die CICS die Transaktionsressourcenüberwachung durchführen soll. Diese Option gilt nur, wenn die Transaktionsressourcenüberwachung aktiviert ist (siehe hierzu Hinweis 2).

Die standardmäßigen CICS-Leistungsklassendaten für die Überwachung umfassen Summen für *alle* Warteschlangen für temporären Speicher, auf die über eine Transaktion zugegriffen wird. Die Transaktionsressourcenüberwachung hingegen erfasst Informationen zu einzelnen Warteschlangen für temporären Speicher, und zwar bis zu der angegebenen Anzahl. Die folgenden Daten werden erfasst:

- Name der Warteschlange für temporären Speicher
- Anzahl und Gesamtzeitdauer der GET-Anforderungen der Warteschlange für temporären Speicher
- Anzahl und Gesamtzeitdauer der GET-Anforderungen der Warteschlange für temporären Speicher an den gemeinsam genutzten temporären Speicher
- Anzahl und Gesamtzeitdauer der PUT-Anforderungen der Warteschlange für temporären Speicher an den temporären Hilfsp Speicher
- Anzahl und Gesamtzeitdauer der PUT-Anforderungen der Warteschlange für temporären Speicher an den temporären Hauptspeicher
- Anzahl und Gesamtzeitdauer der PUT-Anforderungen der Warteschlange für temporären Speicher an den gemeinsam genutzten temporären Speicher
- Gesamtzahl und Gesamtzeitdauer aller Anforderungen für die Warteschlange für temporären Speicher
- Gesamtlänge aller Elemente, die aus dem temporären Speicher abgerufen wurden
- Gesamtlänge aller Elemente, die aus dem gemeinsam genutzten temporären Speicher abgerufen wurden
- Gesamtlänge aller Elemente, die in den temporären Hilfsp Speicher geschrieben wurden
- Gesamtlänge aller Elemente, die in den temporären Hauptspeicher geschrieben wurden
- Gesamtlänge aller Elemente, die in den gemeinsam genutzten temporären Speicher geschrieben wurden
- Wartezeit und Anzahl der Wartezyklen bei der Dateiein-/ausgabe des temporären Speichers
- Wartezeit und Anzahl der Wartezyklen bei der Dateiein-/ausgabe des gemeinsam genutzten temporären Speichers

**8** Mit dieser Standardeinstellung wird CICS angewiesen, die Transaktionsressourcenüberwachung für maximal acht Warteschlangen für temporären Speicher auszuführen.

**zahl** Gibt die maximale Anzahl der Warteschlangen für temporären Speicher im Bereich zwischen 0 und 64 an, für die CICS die Transaktionsressourcenüberwachung ausführen soll. CICS erfasst die Überwachungsleistungsdaten auf Ressourcenebene für jede Warteschlange für temporären Speicher, auf die von einer Transaktion zugegriffen wird, und zwar bis zu dem in *zahl* angegebenen Maximum. Wenn die Transaktion auf eine höhere Anzahl von Warteschlangen für temporären Speicher

zugreift, als in dieser Variablen angegeben ist, dann werden die Warteschlangen für temporären Speicher, die das Maximum übersteigen, ignoriert. Allerdings erhalten sie ein Flag, das anzeigt, dass die Transaktion das Limit für Warteschlangen für temporären Speicher überschritten hat.

Wenn Sie TSQUEUE=0 angeben, dann werden die Daten der Transaktionsressourcenüberwachung für Warteschlangen für temporären Speicher nicht erfasst, auch wenn MNRES=ON aktiv ist.

#### **URIMAP={0|zahl}**

Diese Option gibt die maximal zulässige Anzahl von URIMAPs an, die im Befehl **WEB OPEN URIMAP** angegeben sind und für die CICS die Transaktionsressourcenüberwachung durchführen soll. Diese Option gilt nur, wenn die Transaktionsressourcenüberwachung aktiviert ist (siehe hierzu Hinweis 2).

Die standardmäßigen CICS-Leistungsklassendaten für die Überwachung umfassen Summen für alle URIMAPs, die von einer Transaktion benutzt werden. Die Transaktionsressourcenüberwachung hingegen erfasst Informationen zu einzelnen URIMAPs, und zwar bis zu der angegebenen Anzahl. Die folgenden Daten werden erfasst:

- URIMAP-Name
- Die Zeitdauer, die die Benutzertask auf den Abschluss der **WEB OPEN URIMAP**-Anforderungen gewartet hat, die von der Benutzertask für diese URIMAP abgesetzt wurden
- Die Zeitdauer, die die Benutzertask auf den Abschluss der folgenden Anforderungen gewartet hat, die von der Benutzertask für ein Sitzungstoken für diese URIMAP abgesetzt wurden:
  - **WEB RECEIVE**-Anforderungen
  - Empfangsseite der **WEB CONVERSE**-Anforderungen
- Die Zeitdauer, die die Benutzertask auf den Abschluss der folgenden Anforderungen gewartet hat, die von der Benutzertask für ein Sitzungstoken für diese URIMAP abgesetzt wurden:
  - **WEB SEND**-Anforderungen
  - Sendeseite der **WEB CONVERSE**-Anforderungen

**0** Diese Standardeinstellung inaktiviert die Transaktionsressourcenüberwachung für alle URIMAPs auch dann, wenn MNRES=ON aktiv ist.

**zahl** Gibt die maximale Anzahl von URIMAPs im Bereich zwischen 0 und 64 an, für die CICS die Transaktionsressourcenüberwachung ausführen soll. CICS erfasst die Überwachungsleistungsdaten auf Ressourcenebene für jede URIMAP, die im Befehl **WEB OPEN URIMAP** angegeben wird, der von einer Transaktion abgesetzt wird, und zwar bis zu dem in *zahl* angegebenen Maximum. Wenn die Transaktion eine höhere Anzahl von eindeutigen URIMAPs öffnet, als in dieser Variablen angegeben ist, dann werden die Anforderungen, die das Maximum übersteigen, ignoriert. Allerdings erhalten sie ein Flag, das anzeigt, dass die Transaktion das URIMAP-Limit überschritten hat.

#### **WEBSERV={0|zahl}**

Diese Option gibt die maximal zulässige Anzahl von WEBSERVICES an, die im Befehl **INVOKE SERVICE** angegeben sind und für die CICS die Transaktionsressourcenüberwachung durchführen soll. Diese Option gilt nur, wenn die Transaktionsressourcenüberwachung aktiviert ist (siehe hierzu Hinweis 2).

Die standardmäßigen CICS-Leistungsklassendaten für die Überwachung umfassen Summen für alle WEBSERVICES, die von einer Transaktion benutzt werden. Die Transaktionsressourcenüberwachung hingegen erfasst Informationen zu einzelnen WEBSERVICES, und zwar bis zu der angegebenen Anzahl. Die folgenden Daten werden erfasst:

- WEBSERVICE-Name
- Pipelinename
- Die Zeitdauer, die die Benutzertask auf den Abschluss der **INVOKE SERVICE**-Anforderungen gewartet hat, die von der Benutzertask für diesen WEBSERVICE abgesetzt wurden

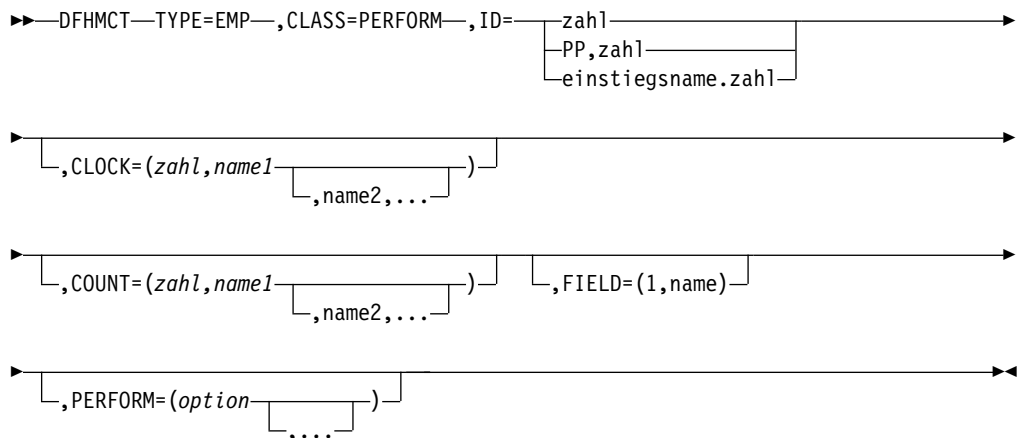
**0** Diese Standardeinstellung inaktiviert die Transaktionsressourcenüberwachung für alle WEBSERVICES auch dann, wenn MNRES=ON aktiv ist.

**zahl** Gibt die maximale Anzahl von WEBSERVICES im Bereich zwischen 0 und 64 an, für die CICS die Transaktionsressourcenüberwachung ausführen soll. CICS erfasst die Überwachungsleistungsdaten auf Ressourcenebene für jeden eindeutigen WEBSERVICE, der im Befehl **INVOKE SERVICE** angegeben wird, der von einer Transaktion abgesetzt wird, und zwar bis zu dem in **zahl** angegebenen Maximum. Wenn die Transaktion eine höhere Anzahl von eindeutigen WEBSERVICES aufruft, als in dieser Variablen angegeben ist, dann werden die Anforderungen, die das Maximum übersteigen, ignoriert. Allerdings erhalten sie ein Flag, das anzeigt, dass die Transaktion das WEBSERVIC-Limit überschritten hat.

## Ereignisüberwachungspunkte des Benutzers: DFHMCT TYPE=EMP

Mit dem Makro DFHMCT TYPE=EMP können Sie angeben, wie die Benutzerdatenfelder in den Datensätzen für die Leistungsklassendaten zu jedem Ereignisüberwachungspunkt (EMP = Event Monitoring Point) des Benutzers hinzugefügt oder für einen Ereignisüberwachungspunkt des Benutzers geändert werden. Für jeden Ereignisüberwachungspunkt des Benutzers, an dem Benutzerdaten benötigt werden, muss ein TYPE=EMP-Makro angegeben werden.

Das Makro TYPE=EMP muss zwischen dem Makro TYPE=INITIAL und der ersten Makroanweisung TYPE=RECORD angegeben werden.





**TYPE=EMP**

Gibt an, dass dieses Makro die Benutzerdaten definiert, die an einem Ereignisüberwachungspunkt des Benutzers erfasst werden sollen.

**CLASS=PERFORM**

Geben Sie diesen Operanden mit den Überwachungsklassen an, für die Benutzerdaten für diesen Ereignisüberwachungspunkt des Benutzers erfasst werden sollen. Der Wert für PERFORM muss angegeben werden. Der entsprechende Operand PERFORM muss ebenfalls angegeben werden.

**ID={*zahl* | (PP,*zahl*) | *einstiegsname.zahl*}**

Geben Sie diesen Operanden mit der Kennung des Ereignisüberwachungspunkts des Benutzers an, bei dem die in diesem Makro definierten Benutzerdaten erfasst werden sollen. Wird eines der Formate *zahl* oder (PP,*zahl*) angegeben, wird der standardmäßige Einstiegsname USER angegeben.

***zahl***

Eine dezimale Ganzzahl im Bereich zwischen 1 und 255. Für Ereignisüberwachungspunkte des Benutzers sind Kennungen zwischen 1 und 199 verfügbar. Die Zahlen zwischen 200 und 255 sind für die Ereignisüberwachungspunkte der IBM Lizenzprogramme reserviert. Geben Sie diese Zahlen ein, wenn Sie Benutzerdaten an Ereignisüberwachungspunkten erfassen wollen, die im Code von IBM Lizenzprogrammen codiert sind.

**(PP,*zahl*)**

Eine EMP-Kennung eines IBM Lizenzprogramms. Diese Angabe entspricht der Angabe des ID-Werts '199 + Zahl'. Der Wert für *zahl* ist eine dezimale Ganzzahl im Bereich zwischen 1 und 56.

***einstiegsname.zahl***

Erlaubt die mehrfache Verwendung von *zahl*, einer dezimalen Ganzzahl im Bereich zwischen 1 und 255. Demzufolge sind UNIQUE.3, DSN.3 und 3 drei unterschiedliche Ereignisüberwachungspunkte. Es können maximal 98 Einstiegsnamen für eine bestimmte Zahl angegeben werden. Außerdem stellt jeder Zähler, jede Systemzeit oder relative Position in Byte, auf den bzw. die von ihnen verwiesen wird, ein Objekt dar, das sich von den Objekten unterscheidet, auf die von den anderen verwiesen wird.

In den folgenden Beschreibungen gibt jeder Verweis auf eine Konstante eine hexadezimale Konstante mit bis zu acht hexadezimalen Ziffern an. Kürzere Zeichenfolgen werden links mit Nullen aufgefüllt. Um z. B. den Dezimalwert 14 zu addieren oder zu subtrahieren, wird die Konstante als 0000000E oder kurz E (keine Anführungszeichen erforderlich) codiert.

Verweise auf die Felder DATA1 und DATA2 beziehen sich auf die beiden binären Vollwörter, die vom Ereignisüberwachungspunkt des Benutzers bereitgestellt werden, der im Anwendungsprogramm angegeben ist. Diese Werte werden in den Operanden DATA1 und DATA2 des Befehls EXEC CICS MONITOR zum Definieren von Ereignisüberwachungspunkten des Benutzers angegeben. Abhängig von den codierten Optionen können die Felder DATA1 und DATA2 als Zahlen, Masken für die Ausführung logischer Operationen oder Zeiger auf weitere Informationen interpretiert werden.

Jeder Verweis auf eine Zahl bezieht sich auf eine dezimale Ganzzahl in dem Bereich, der in der Beschreibung der Option definiert ist.

**CLOCK=(*zahl*,*name1* [,*name2*,...])**

Mit diesem Operanden wird einzelnen oder mehreren Systemzeiten ein informeller Name zugewiesen. Der informelle Name einer Systemzeit wird im zuge-

hörigen Wörterbucheintrag angezeigt und steht für den Nachprozessor zur Verwendung (z. B. als Spaltenüberschrift) zur Verfügung.

Die Zeichenfolge *name1* ist der Systemzeit zugeordnet, die während der MCT-Generierung in *zahl* angegeben wird. Wenn dieser Operand angegeben wird, dann wird *name2* der Systemzeit '*zahl* + 1' zugewiesen. In ähnlicher Weise werden alle nachfolgenden Namen nachfolgenden Systemzeiten zugewiesen. Jede Systemzeit, die nicht in dieser Option angegeben wird, empfängt den Einstiegsnamenwert aus dem Operanden ID (der Standardwert ist USER).

Der Wert für *zahl* muss im Bereich zwischen 1 und 256 liegen. Die angegebenen Namen müssen jeweils eine Zeichenfolge mit bis zu acht Zeichen darstellen. Wenn eine Zeichenfolge einzelne oder mehrere Leerzeichen oder Kommas enthält, muss sie in Anführungszeichen eingeschlossen werden.

#### **COUNT=(*zahl*,*name1*[,*name2*,...])**

Mit diesem Operanden wird einzelnen oder mehreren Zählerfeldern ein informeller Name zugewiesen. Der informelle Name eines Zählerfelds wird im zugehörigen Wörterbucheintrag angezeigt und steht für den Nachprozessor zur Verwendung (z. B. als Spaltenüberschrift) zur Verfügung.

Die Zeichenfolge *name1* ist dem Zählerfeld zugeordnet, das während der MCT-Generierung in *zahl* angegeben wird. Wenn dieser Operand angegeben wird, dann wird *name2* dem Zählerfeld '*zahl* + 1' zugewiesen. In ähnlicher Weise werden alle nachfolgenden Namen nachfolgenden Zählerfeldern zugewiesen. Jedes Zählerfeld, das nicht in dieser Option angegeben wird, empfängt den Einstiegsnamenwert aus dem Operanden ID (der Standardwert ist USER).

Der Wert für *zahl* muss im Bereich zwischen 1 und 256 liegen. Die angegebenen Namen müssen jeweils eine Zeichenfolge mit bis zu acht Zeichen darstellen. Wenn eine Zeichenfolge einzelne oder mehrere Leerzeichen oder Kommas enthält, muss sie in Anführungszeichen eingeschlossen werden.

#### **FIELD=(1,*name*)**

Mit diesem Operanden wird einem Benutzerbytefolgefeld ein informeller Name zugeordnet. Der informelle Name wird im zugehörigen Wörterbucheintrag angezeigt und steht für den Nachprozessor zur Verwendung (z. B. als Spaltenüberschrift) zur Verfügung.

Für *name* muss eine Zeichenfolge mit bis zu acht Zeichen angegeben werden. Wenn die Zeichenfolge einzelne oder mehrere Leerzeichen oder Kommas enthält, muss sie in Anführungszeichen eingeschlossen werden.

#### **PERFORM=(*option*[,...])**

Geben Sie diesen Operanden an, wenn CLASS=PERFORM angegeben wurde. Er gibt an, dass Informationen zu den Benutzerfeldern des Leistungsklassendatensatzes dieses Ereignisüberwachungspunkts hinzugefügt oder dort geändert werden.

Die Benutzerfelder für jeden Benutzer, der durch einen separaten Einstiegsnamen im Operanden ID gekennzeichnet ist, kann Folgendes umfassen:

1. Bis zu 256 Zähler.
2. Bis zu 256 Systemzeiten, die jeweils aus einem Akkumulator mit 8 Byte und einem Zähler mit 4 Byte bestehen.
3. Eine Bytefolge mit bis zu 8192 Byte.

**Anmerkung:** Wenn die kombinierten Größen der Objekte (Systemzeiten, Zähler und Felder), die in den angegebenen Optionen impliziert werden, den Wert von 16384 Byte überschreiten, dann treten bei der Assemblierung Fehler auf. Dies kann vermieden werden, indem weniger Objekte verwendet werden.

Dazu werden entweder weniger Daten erfasst oder die Verweise auf Systemzeiten und Zähler werden zu Clustern zusammengefasst, um implizierte, jedoch nicht benutzte Objekte zu vermeiden.

**Anmerkung:** Wenn Sie Benutzerdaten definieren, die an einem Ereignisüberwachungspunkt des Benutzers erfasst werden sollen, wird dadurch die Größe aller CICS-Datensätze für die Leistungsklassenüberwachung erweitert. Jeder CICS-Überwachungsdatensatz hat dieselbe Größe wie der größte Datensatz. Dies muss bei der Angabe von Benutzerdatenfeldern berücksichtigt werden.

Aktionen werden für die Benutzerfelder entsprechend der angegebenen Optionen ausgeführt.

Der Operand PERFORM kann mit PER abgekürzt werden. Für den Operanden PERFORM sind die folgenden Optionen gültig:

**ADDCNT(*zahl*, {*konstante* | DATA1 | DATA2})**

Der Wert für das in *zahl* angegebene Benutzerzählerfeld muss um den Wert von *konstante* oder um den Wert des Felds DATA1 oder DATA2 erhöht werden. Der Wert für *zahl* ist eine dezimale Ganzzahl im Bereich zwischen 1 und 256.

**EXCNT(*zahl*, {*konstante* | DATA1 | DATA2})**

Für den Wert des Benutzerzählerfelds, der in *zahl* angegeben ist, muss anhand des Werts für *konstante* oder anhand des Werts im Feld DATA1 oder DATA2 eine Operation mit einem logischen exklusiven Oder ausgeführt werden. Der Wert für *zahl* ist eine dezimale Ganzzahl im Bereich zwischen 1 und 256.

**MLTCNT(*zahl1*, *zahl2*)**

Eine Reihe benachbarter Benutzerzählerfelder muss aktualisiert werden, indem die Werte in den benachbarten Vollwörtern in einem Bereich hinzugefügt werden, der durch das Feld DATA1 adressiert wird. Zur Verwendung dieser Option müssen die Felder DATA1 und DATA2 vom Ereignisüberwachungspunkt des Benutzers übergeben werden.

Die Benutzerzählerfelder, die aktualisiert werden müssen, beginnen mit dem Feld, das in *zahl1* angegeben ist. Die Anzahl der Benutzerzählerfelder, die aktualisiert werden, entspricht dem kleineren der beiden Werte für *zahl2* und für das Feld DATA2. Wenn im Feld DATA2 der Wert null angegeben ist, dann wird der Wert für *zahl2* verwendet. Die Reihe benachbarter Vollwörter, die zum Hinzufügen in den Benutzerzählerfeldern verwendet werden, beginnt mit der Adresse, die im Feld DATA1 angegeben ist. Aufeinanderfolgende Vollwörter werden zu aufeinanderfolgenden Benutzerzählerfeldern hinzugefügt.

Bei den Werten für *zahl1* und *zahl2* handelt es sich um dezimale Ganzzahlen im Bereich zwischen 1 und 256. Die Anzahl der generierten Benutzerzähler ist (*zahl1* + *zahl2* - 1). Dieser Wert muss ebenfalls im Bereich zwischen 1 und 256 liegen.

**Anmerkung:** In jedem Makro DFHMCT TYPE=EMP kann immer nur eine der beiden Optionen MLTCNT und MOVE verwendet werden.

**MOVE(*zahl13*, *zahl14*)**

Eine Zeichenfolge mit Daten soll in das Benutzerbytefolgefeld verschoben werden. Zur Verwendung dieser Option müssen die beiden Felder DATA1 und DATA2 vom Ereignisüberwachungspunkt des Benutzers übergeben werden.

Das Benutzerbytefolgefeld wird beginnend bei der relativen Adresse, die in *zahl3* angegeben ist, aktualisiert. Die zu verschiebenden Daten beginnen bei der Adresse, die im Feld DATA1 angegeben ist. Die maximal zulässige Länge der Daten, die verschoben werden können, wird in *zahl4* (in Byte) angegeben und die tatsächliche Länge der Daten, die verschoben werden sollen, wird im Wert des Felds DATA2 angegeben. Wenn der Wert von DATA2 gleich null ist, wird die Länge der in *zahl4* angegebenen Daten verschoben.

Der Wert von *zahl3* ist eine dezimale Ganzzahl im Bereich zwischen 0 und 8191, der Wert von *zahl4* ist eine dezimale Ganzzahl im Bereich zwischen 1 und 8192. Die maximal zulässige Länge des Benutzerzeichenfelds beträgt (*zahl3* + *zahl4*) und muss im Bereich zwischen 1 und 8192 liegen.

**Anmerkung:** In jeder Makroinstruktion DFHMCT TYPE=EMP kann immer nur eine der beiden Optionen MLTCNT und MOVE verwendet werden.

**NACNT(*zahl*, {*konstante* | DATA1 | DATA2})**

Für den Wert des Benutzerzählerfelds, der in *zahl* angegeben ist, muss anhand des Werts für *konstante* oder anhand des Werts im Feld DATA1 oder DATA2 eine Operation mit einem logischen Und ausgeführt werden. Der Wert für *zahl* ist eine dezimale Ganzzahl im Bereich zwischen 1 und 256.

**ORCNT(*zahl*, {*konstante* | DATA1 | DATA2})**

Für den Wert des Benutzerzählerfelds, der in *zahl* angegeben ist, muss anhand des Werts für *konstante* oder anhand des Werts im Feld DATA1 oder DATA2 eine Operation mit einem logischen inklusiven Oder ausgeführt werden. Der Wert für *zahl* ist eine dezimale Ganzzahl im Bereich zwischen 1 und 256.

**PCLOCK(*zahl*)**

Die durch die Zahl angegebene Systemzeit soll gestoppt werden. Der aus vier Byte bestehende Zähler im Feld für die Systemzeit des Benutzers wird markiert, um anzugeben, dass die Systemzeit nun gestoppt wurde. Der Akkumulator wird auf die Summe seiner Inhalte vor dem vorherigen SCLOCK und der verstrichenen Zeitdauer zwischen diesem SCLOCK und diesem PCLOCK gesetzt. Der Wert für *zahl* ist eine dezimale Ganzzahl im Bereich zwischen 1 und 256.

**PCPUCLK(*zahl*)**

Diese Option führt die gleiche Funktion wie PCLOCK aus, verwendet jedoch die CPU-Zeit der Haupttask von CICS anstelle der abgelaufenen Zeit.

**SCLOCK(*zahl*)**

Die in *zahl* angegebene Systemzeit soll gestartet werden. Der Wert des aus vier Byte bestehenden Zählers im Feld für die Systemzeit des Benutzers wird um den Wert 1 erhöht und markiert, um den Laufstatus anzuzeigen. Der Wert für *konstante* ist eine dezimale Ganzzahl im Bereich zwischen 1 und 256.

**SCPUCLK(*zahl*)**

Diese Option führt die gleiche Funktion wie SCLOCK aus, verwendet jedoch die CPU-Zeit der Haupttask von CICS anstelle der abgelaufenen Zeit.

**SUBCNT(*zahl*, {*konstante* | DATA1 | DATA2})**

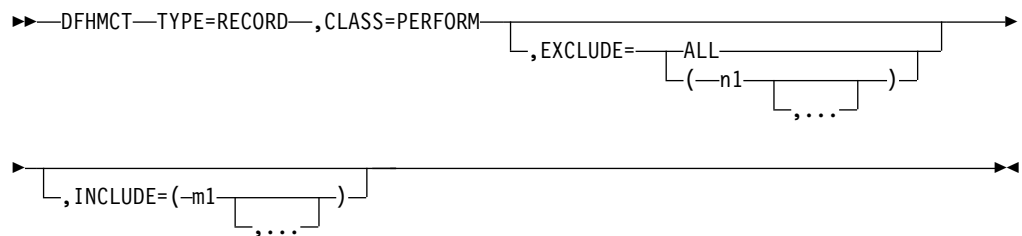
Der Wert für das in *zahl* angegebene Benutzerzählerfeld muss um den Wert von *konstante* oder um den Wert des Felds DATA1 oder DATA2 reduziert werden. Der Wert für *zahl* ist eine dezimale Ganzzahl im Bereich zwischen 1 und 256.

## DELIVER

Die für diese Task bis zum aktuellen Zeitpunkt aufgelaufenen Leistungsklassendaten werden an die Überwachungspuffer übergeben. Alle aktiven Systemzeiten werden gestoppt. Der Abschnitt für die Leistungsklasse des Überwachungsbereichs für diese Task wird auf X'00' zurückgesetzt. Eine Ausnahme bilden hierbei der Schlüsselfelder (transid, termid) und alle Daten, die aufgrund des Aufrufs der DFHAPPL-Sonder-EMPs gespeichert werden. Alle Systemzeiten, die mit dieser Option gestoppt wurden, werden für den neuen Abrechnungszeitraum von null an erneut gestartet. Die Felder für den oberen Grenzwert werden auf die aktuellen Werte zurückgesetzt.

## Steuerdatenaufzeichnung: DFHMCT TYPE=RECORD

Mit dem Makro DFHMCT TYPE=RECORD können die Felder für Leistungsklassendaten identifiziert werden, die zur Überwachung ausgewählt wurden.



### TYPE=RECORD

Gibt an, dass Überwachungsdaten für ausgewählte Felder für Leistungsklassendaten aufgezeichnet werden.

### CLASS=PERFORM

Geben Sie diesen Operanden an, um Felder für Leistungsklassendaten aufzuzeichnen. PERFORM kann mit PER abgekürzt werden.

### EXCLUDE={ALL | (n1 [, ...])}

Geben Sie diesen Operanden an, um zu verhindern, dass einzelne oder mehrere CICS-Felder von der Überwachungsfunktion gemeldet werden. Standardmäßig werden alle dokumentierten Felder für Leistungsklassen gemeldet.

Der Operand EXCLUDE wird immer vor dem Operanden INCLUDE verarbeitet, und zwar unabhängig von der Reihenfolge, in der die beiden Operanden angegeben werden. (Der Operand INCLUDE ist nur relevant, wenn der Operand EXCLUDE angegeben wird.)

### ALL

Diese Option verhindert, dass alle Felder, die für den Ausschluss auswählbar sind, gemeldet werden. Die folgenden Felder können nicht ausgeschlossen werden:

- 1
- 2
- 4
- 5
- 6
- 89

Wenn Sie die Mehrzahl der Felder ausschließen, bestimmte Felder jedoch einschließen wollen, dann geben Sie den Operanden INCLUDE gleichzeitig mit EXCLUDE=ALL an.

In Tabelle 13 werden die Felder aufgeführt, die für den Ausschluss auswählbar sind. Jedem Feld ist ein Gruppenname zugeordnet, der die Gruppe der Felder angibt, zu der es gehört. Jedes Feld verfügt außerdem über eine eigene numerische Feldkennung.

Um eine Gruppe von Feldern auszuschließen, müssen Sie den Namen der Gruppe (Zeichenfolge) als *n1* angeben. Beispiel: EXCLUDE=(DFHTASK).

Um ein einzelnes Feld auszuschließen, müssen Sie die numerische Kennung des Felds als *n1* angeben. Beispiel: EXCLUDE=(98,70).

**Anmerkung:** In numerischen Kennungen dürfen keine führenden Nullen angegeben werden. Geben Sie keine numerischen Kennungen von Feldern an, die nicht für den Ausschluss auswählbar sind.

Sie können Kombinationen von Namen und numerischen Kennungen angeben. Beispiel: EXCLUDE=(DFHFILE,DFHTERM,112,64).

*Tabelle 13. Aus- oder einschließbare Datengruppen und Felder*

Gruppenname	Feld-ID	Feldname	Beschreibung
DFHCBTS	200	PRCSNAME	CICS BTS-Prozessname
DFHCBTS	201	PRCSTYPE	CICS BTS-Prozesstyp
DFHCBTS	202	PRCSID	CICS BTS-Prozess-ID
DFHCBTS	203	ACTVTYID	CICS BTS-Aktivitäts-ID
DFHCBTS	204	ACTVTYNM	CICS BTS-Aktivitätsname
DFHCBTS	205	BARSYNCT	CICS BTS-Zähler für synchrone Ausführung von Prozessen/Aktivitäten
DFHCBTS	206	BARASYCT	CICS BTS-Zähler für asynchrone Ausführung von Prozessen/Aktivitäten
DFHCBTS	207	BALKPACT	CICS BTS-Zähler für Verbindung von Prozessen/Aktivitäten
DFHCBTS	208	BADPROCT	CICS BTS-Zähler für Definition von Prozessen
DFHCBTS	209	BADACTCT	CICS BTS-Zähler für Definition von Aktivitäten
DFHCBTS	210	BARSPACT	CICS BTS-Zähler für Zurücksetzen von Prozessen/Aktivitäten
DFHCBTS	211	BASUPACT	CICS BTS-Zähler für Aussetzen von Prozessen/Aktivitäten
DFHCBTS	212	BARMPACT	CICS BTS-Zähler für Fortsetzen von Prozessen/Aktivitäten
DFHCBTS	213	BADCPACT	CICS BTS-Zähler für Anforderungen zum Löschen von Aktivitäten oder Abbrechen von Prozessen/Aktivitäten
DFHCBTS	214	BAACQPCT	CICS BTS-Zähler zum Anfordern von Prozessen/Aktivitäten
DFHCBTS	215	BATOTPCT	CICS BTS-Zähler für Summe der Prozess-/Aktivitätsanforderungen
DFHCBTS	216	BAPRDCCT	CICS BTS-Zähler zum Löschen/Abrufen/Einreihen von Prozesscontainern
DFHCBTS	217	BAACDCCT	CICS BTS-Zähler zum Löschen/Abrufen/Einreihen von Aktivitätscontainern
DFHCBTS	218	BATOTCCT	CICS BTS-Zähler für Summe der Anforderungen für Prozess-/Aktivitätscontainer

Tabelle 13. Aus- oder einschließbare Datengruppen und Felder (Forts.)

Gruppenname	Feld-ID	Feldname	Beschreibung
DFHCBTS	219	BARATECT	CICS BTS-Zähler für Anforderungen zum Abrufen und Neuuzuordnen
DFHCBTS	220	BADFIECT	CICS BTS-Zähler für Anforderungen zum Definieren von Eingabeereignissen
DFHCBTS	221	BATIAECT	CICS BTS-Zähler für Anforderungen für zeitgeberbezogene Ereignisse
DFHCBTS	222	BATOTECT	CICS BTS-Zähler für Summe der ereignisbezogenen Anforderungen
DFHCHNL	321	PGTOTCCT	Anzahl der CICS-Anforderungen für Kanalcontainer
DFHCHNL	322	PGBRWCCT	Anzahl der Anforderungen zum Durchsuchen von Kanalcontainern
DFHCHNL	323	PGGETCCT	Anzahl der GET CONTAINER- und GET64 CONTAINER-Anforderungen für Kanalcontainer
DFHCHNL	324	PGPUTCCT	Anzahl der PUT CONTAINER- und PUT64 CONTAINER-Anforderungen für Kanalcontainer
DFHCHNL	325	PGMOVCCT	Anzahl der MOVE CONTAINER-Anforderungen für Kanalcontainer
DFHCHNL	326	PGGETCDL	Länge der Daten in den Containern für alle GET CONTAINER CHANNEL- und GET64 CONTAINER CHANNEL-Befehle
DFHCHNL	327	PGPUTCDL	Länge der Daten in den Containern für alle PUT CONTAINER CHANNEL- und PUT64 CONTAINER CHANNEL-Befehle
DFHCHNL	328	PGCRECCT	Anzahl der Container, die mit MOVE-, PUT CONTAINER- und PUT64 CONTAINER-Anforderungen für Kanalcontainer erstellt wurden
DFHCHNL	329	PGCSTHWM	Maximalwert (oberer Grenzwert) für zugeordneten Containerspeicher
DFHCICS	25	CFCAPICT	Zähler für CICS OO-Basisklassenanforderungen
DFHCICS	103	EXWTTIME	Wartezeit für Transaktionsausnahmebedingungen
DFHCICS	112	RTYPE	Leistungsdatensatztyp
DFHCICS	130	RSYSID	System-ID für Transaktionsrouting
DFHCICS	131	PERRECNT	Zähler für Leistungsdatensätze
DFHCICS	167	SRVCLASS	MVS Workload Manager-Serviceklassenname
DFHCICS	168	RPTCLASS	MVS Workload Manager-Berichtsklassenname
DFHCICS	351	OADID	Vom Adapter zu den Ursprungsdaten hinzugefügte Adapterkennung
DFHCICS	352	OADATA1	Vom Adapter zu den Ursprungsdaten hinzugefügte Daten
DFHCICS	353	OADATA2	Vom Adapter zu den Ursprungsdaten hinzugefügte Daten
DFHCICS	354	OSDATA3	Vom Adapter zu den Ursprungsdaten hinzugefügte Daten
DFHCICS	359	ONETWKID	Netz-ID, aus der diese Arbeitsanforderungstransaktion stammt
DFHCICS	360	OAPPLID	Anwendungs-ID (APPLID) der CICS-Region, aus der diese Arbeitsanforderung stammt
DFHCICS	361	OSTART	Zeitpunkt, zu dem die ursprüngliche Task gestartet wurde
DFHCICS	362	OTRANNUM	Nummer der ursprünglichen Task

Tabelle 13. Aus- oder einschließbare Datengruppen und Felder (Forts.)

Gruppenname	Feld-ID	Feldname	Beschreibung
DFHCICS	363	OTRAN	Transaktions-ID (TRANSID) der ursprünglichen Task
DFHCICS	364	OUSERID	Ursprüngliche Userid-2 oder Userid-1, abhängig von der ursprünglichen Task
DFHCICS	365	OUSERCOR	Korrelator für Ersteller
DFHCICS	366	OTCPSVCE	Name für ursprünglichen TCPIP SERVICE
DFHCICS	367	OPORTNUM	Vom ursprünglichen TCPIP SERVICE verwendete Portnummer
DFHCICS	369	OCLIPORT	TCP/IP-Portnummer des ursprünglichen Clients
DFHCICS	370	OTRANFLG	Flags für ursprüngliche Transaktion
DFHCICS	371	OFCTYNME	Funktionsname der ursprünglichen Transaktion
DFHCICS	372	OCLIPADR	IP-Adresse des ursprünglichen Clients oder Telnet-Clients
DFHCICS	373	PHNTWKID	Netz-ID des CICS-Systems einer unmittelbar vorangegangenen Task auf einem anderen CICS-System, dem diese Task zugeordnet ist
DFHCICS	374	PHAPPLID	Anwendungs-ID (APPLID) der Daten zum vorherigen Hop
DFHCICS	375	PHSTART	Startzeitpunkt der direkt vorangegangenen Task auf einem anderen CICS-System, dem diese Task zugeordnet ist
DFHCICS	376	PHTRANNO	Tasknummer der direkt vorangegangenen Task auf einem anderen CICS-System, dem diese Task zugeordnet ist
DFHCICS	377	PHTRAN	Transaktions-ID (TRANSID) der direkt vorangegangenen Task auf einem anderen CICS-System, dem diese Task zugeordnet ist
DFHCICS	378	PHCOUNT	Anzahl der Anforderungen von einem CICS-System an ein anderes CICS-System, mit denen eine Task eingeleitet werden soll, der diese Task zugeordnet ist
DFHCICS	402	EICTOTCT	Zähler der EXEC CICS API- und SPI-Anforderungen
DFHCICS	405	TIASKTCT	Zähler für ASKTIME-Anforderungen
DFHCICS	406	TITOTCT	Zähler für Summe der TIME-Anforderungen
DFHCICS	408	BFDGSTCT	Zähler für BIF DIGEST-Anforderungen
DFHCICS	409	BFTOTCT	Zähler für Summe der BIF-Anforderungen
DFHCICS	415	ECSIGECT	Zähler für SIGNAL EVENT-Anforderungen
DFHCICS	416	ECEFOPCT	Zähler für Ereignisfilteroperationen
DFHCICS	417	ECEVNTCT	Zähler der erfassten Ereignisse
DFHCICS	418	ECSEVCCT	Anzahl der synchronen Ausgabeereignisse, die von der Benutzertask erfasst wurden
DFHCICS	449	MPPRTXCD	Anzahl der Schwellenwerte für Richtlinientaskregeln, die von dieser Task überschritten wurden
DFHCICS	464	NCGETCT	Anzahl der GET COUNTER- und DCOUNTER-Anforderungen
DFHCICS	466	MPSRECT	Anzahl der Auswertungen der Richtliniensystemregeln für die Task
DFHCICS	467	MPSRACT	Anzahl der Auswertungen der Richtliniensystemregeln mit 'true' und Auslösung einer Nachricht oder eines Ereignisses



Tabelle 13. Aus- oder einschließbare Datengruppen und Felder (Forts.)

Gruppenname	Feld-ID	Feldname	Beschreibung
DFHCICS	480	PTSTART	Der Startzeitpunkt der direkt vorangegangenen oder übergeordneten Task auf demselben CICS-System, dem die Task zugeordnet ist
DFHCICS	481	PTTRANNO	Tasknummer der direkt vorangegangenen oder übergeordneten Task auf demselben CICS-System, dem die Task zugeordnet ist
DFHCICS	482	PTTRAN	Transaktions-ID (TRANSID) der direkt vorangegangenen oder übergeordneten Task auf demselben CICS-System, dem die Task zugeordnet ist
DFHCICS	483	PTCOUNT	Anzahl der Anforderung von einer Task zum Einleiten einer anderen Task auf demselben CICS-System, dem diese Task zugeordnet ist, z. B. mit einem Befehl <b>RUN TRANSID ASYNCHRONOUS</b> oder <b>START</b>
DFHDATA	179	IMSREQCT	IMS-Anforderungszähler (DBCTL)
DFHDATA	180	DB2REQCT	Db2-Anforderungszähler
DFHDATA	186	IMSWAIT	IMS-Wartezeit (DBCTL)
DFHDATA	187	DB2RDYQW	Db2 Readyq-Wartezeit
DFHDATA	188	DB2CONWT	Db2-Verbindungswartezeit
DFHDATA	395	WMQREQCT	MQ-Anforderungszähler
DFHDATA	396	WMQGETWT	MQ GETWAIT-Wartezeit
DFHDATA	397	WMQASRBT	IBM MQ API SRB-Zeit
DFHDEST	41	TDGETCT	TD-Get-Zähler
DFHDEST	42	TDPUTCT	TD-Put-Zähler
DFHDEST	43	TDPURCT	TD-Purge-Zähler
DFHDEST	91	TDTOTCT	TD-Gesamtzähler
DFHDEST	101	TDIOWTT	TD-E/A-Wartezeit
DFHDEST	403	TDILWTT	TD-Wartezeit für partitionsinterne Sperren
DFHDEST	404	TDELWTT	TD-Wartezeit für partitionsübergreifende Sperren
DFHDOCH	223	DHDELCT	Löschzähler für Dokumenthandler
DFHDOCH	226	DHCRECT	Erstellungszähler für Dokumenthandler
DFHDOCH	227	DHINSCT	Einfügungszähler für Dokumenthandler
DFHDOCH	228	DHSETCT	Set-Zähler für Dokumenthandler
DFHDOCH	229	DHRETCT	Abrufzähler für Dokumenthandler
DFHDOCH	230	DHTOTCT	Gesamtzähler für Dokumenthandler
DFHDOCH	240	DHTOTDCL	Insgesamt erstellte Dokumentlänge für Dokumenthandler
DFHFEPI	150	SZALLOCT	FEPI-Zuordnungszähler
DFHFEPI	151	SZRCVCT	FEPI-Empfangszähler
DFHFEPI	152	SZSENDCT	FEPI-Sendungszähler
DFHFEPI	153	SZSTRCT	FEPI-Startzähler
DFHFEPI	154	SZCHROUT	Gesendete FEPI CHARS
DFHFEPI	155	SZCHRIN	Empfangene FEPI CHARS
DFHFEPI	156	SZWAIT	FEPI-Aussetzzeit
DFHFEPI	157	SZALLCTO	FEPI-Zähler für Zuordnungszeitlimit
DFHFEPI	158	SZRCVTO	FEPI-Zähler für Empfangszeitlimit
DFHFEPI	159	SZTOTCT	FEPI-Gesamtzähler
DFHFILE	36	FCGETCT	FC-Get-Zähler
DFHFILE	37	FCPUTCT	FC-Put-Zähler
DFHFILE	38	FCBRWCT	FC-Suchzähler
DFHFILE	39	FCADDCT	FC-Hinzufügungszähler

Tabelle 13. Aus- oder einschließbare Datengruppen und Felder (Forts.)

Gruppenname	Feld-ID	Feldname	Beschreibung
DFHFILE	40	FCDELCT	FC-Löschzähler
DFHFILE	63	FCIOWTT	FC-E/A-Wartezeit
DFHFILE	70	FCAMCT	FC-Zähler für Zugriffsmethode
DFHFILE	93	FCTOTCT	FC-Gesamtzähler
DFHFILE	174	RLSWAIT	RLS FC-E/A-Wartezeit
DFHFILE	175	RLSCPUT	CPU-Zeit (SRB) für RLS-Dateianforderung
DFHFILE	176	CFDTPWAIT	CFDT-E/A-Wartezeit
DFHFILE	426	FCXCWTT	Wartezeit der Dateisteuerung für exklusive Steuerung eines VSAM-Steuerintervalls
DFHFILE	427	FCVSWTT	Wartezeit der Dateisteuerung für eine VSAM-Zeichenfolge
DFHJOUR	10	JCIOWTT	Journal-E/A-Wartezeit
DFHJOUR	58	JNLWRTCT	Zähler für Journalschreiboperationen
DFHJOUR	172	LOGWRTCT	Schreiboperationszähler für Protokolldatenstrom
DFHMAPP	50	BMSMAPCT	BMS MAP-Zähler
DFHMAPP	51	BMSINCT	BMS IN-Zähler
DFHMAPP	52	BMSOUTCT	BMS OUT-Zähler
DFHMAPP	90	BMSTOTCT	BMS-Gesamtzähler
DFHPROG	55	PCLINKCT	LINK-Zähler für Programm
DFHPROG	56	PCXCTLCT	XCTL-Zähler für Programm
DFHPROG	57	PCLOADCT	LOAD-Zähler für Programm
DFHPROG	71	PGMNAME	Programmname
DFHPROG	72	PCLURMCT	LINK_URM-Zähler für Programm
DFHPROG	73	PCDPLCT	DPL-Zähler für Programm
DFHPROG	113	ABCODEO	Ursprünglicher Abbruchcode
DFHPROG	114	ABCODEC	Aktueller Abbruchcode
DFHPROG	115	PCLOADTM	Programmladezeit
DFHPROG	286	PCDLCSDL	Länge der Daten in den Containern aller DPL-Anforderungen (DPL = Distributed Program Link; Verbindung zu verteilten Programmen) mit der Option CHANNEL
DFHPROG	287	PCDLCRD	Länge der Daten in den Containern aller DPL RETURN CHANNEL-Befehle
DFHPROG	306	PCLNKCCT	Anzahl lokaler LINK-Anforderungen mit der Option CHANNEL
DFHPROG	307	PCXCLCCT	Anzahl der XCTL-Anforderungen mit der Option CHANNEL
DFHPROG	308	PCDPLCCT	Anzahl der DPL-Anforderungen mit der Option CHANNEL
DFHPROG	309	PCRTNCCT	Anzahl der fernen RETURN-Anforderungen mit der Option CHANNEL
DFHPROG	310	PCRTNCLD	Länge der Daten in den Containern aller fernen RETURN CHANNEL-Befehle
DFH SOCK	241	SOIOWTT	E/A-Wartezeit für Eingangssocket
DFH SOCK	242	SOBYENCT	Verschlüsselte Byte für sicheres Socket
DFH SOCK	243	SOBYDECT	Entschlüsselte Byte für sicheres Socket
DFH SOCK	245	TCPSRVCE	TCP/IP-Servicename
DFH SOCK	246	PORTNUM	TCP/IP-Service-Port-Nummer
DFH SOCK	288	ISALLOCT	Anzahl der Zuordnungsanforderungen für IPCONNS
DFH SOCK	289	SOEXTRCT	Zähler für Socketextraktionsanforderungen

Tabelle 13. Aus- oder einschließbare Datengruppen und Felder (Forts.)

Gruppenname	Feld-ID	Feldname	Beschreibung
DFH SOCK	290	SOCNPSCT	Anforderungszähler für Erstellung nicht persistenter Sockets
DFH SOCK	291	SOCPSCT	Anforderungszähler für Erstellung persistenter Sockets
DFH SOCK	292	SONPSHWM	Oberer Grenzwert für nicht persistente Sockets
DFH SOCK	293	SOPSHWM	Oberer Grenzwert für persistente Sockets
DFH SOCK	294	SORCVCT	Zähler für Socketempfangsanforderungen
DFH SOCK	295	SOCHRIN	Empfangene Socketzeichen
DFH SOCK	296	SOSENDCT	Zähler für Socketsendeanforderungen
DFH SOCK	297	SOCHROUT	Gesendete Socketzeichen
DFH SOCK	298	SOTOTCT	Gesamtzähler für Socketanforderungen
DFH SOCK	299	SOOIOWTT	E/A-Wartezeit für Ausgangsocket
DFH SOCK	300	ISIOWTT	Verstrichene Zeit, die eine Benutzertask an diesem Ende einer IPCONN auf die Steuerung gewartet hat
DFH SOCK	301	SOMSGIN1	Empfangsanforderungszähler für Eingangsocket
DFH SOCK	302	SOCHRIN1	Empfangene Zeichen für Eingangsocket
DFH SOCK	303	SOMSGOU1	Sendeanforderungszähler für Eingangsocket
DFH SOCK	304	SOCHROU1	Gesendete Zeichen für Eingangsocket
DFH SOCK	305	ISIPICNM	Name der IPCONN, deren TCPIP SERVICE zum Anschließen der Benutzertask verwendet wurde
DFH SOCK	318	CLIPADDR	IP-Adresse des ursprünglichen Clients oder Telnet-Clients
DFH SOCK	319	ISALWTT	Wartezeit für IPIC-Sitzungszuordnung
DFH SOCK	320	SOCIPHER	Cipher-Suite-Code für SSL-Handshake zur Verwendung für die eingehende Verbindung
DFH SOCK	330	CLIPPORT	Portnummer des Clients oder Telnet-Clients
DFH STOR	33	SCUSRHWM	Oberer Grenzwert für Benutzerspeicher (UDSA)
DFH STOR	54	SCUGETCT	Get-Zähler für Benutzerspeicher (UDSA)
DFH STOR	87	PCSTGHWM	Oberer Grenzwert für Programmspeicher
DFH STOR	95	SCUSRSTG	Belegung des Benutzerspeichers (bytes-ms) (UDSA)
DFH STOR	105	SCUGETCT	Get-Zähler für Benutzerspeicher über 16-MB-Grenze (EUDSA)
DFH STOR	106	SCUSRHWM	Oberer Grenzwert für Benutzerspeicher über 16-MB-Grenze (EUDSA)
DFH STOR	107	SCUSRSTG	Belegung des Benutzerspeichers (bytes-ms) über 16-MB-Grenze (EUDSA)
DFH STOR	108	PC24BHWM	Oberer Grenzwert für Programmspeicher unter 16-MB-Grenze
DFH STOR	116	SC24CHWM	Oberer Grenzwert für Benutzerspeicher unter 16-MB-Grenze (CDSA)
DFH STOR	117	SCCGETCT	Get-Zähler für Benutzerspeicher unter 16-MB-Grenze (CDSA)
DFH STOR	118	SC24COCC	Belegung des Benutzerspeichers (bytes-ms) unter 16-MB-Grenze (CDSA)
DFH STOR	119	SC31CHWM	Oberer Grenzwert für Benutzerspeicher über 16-MB-Grenze (ECDSA)
DFH STOR	120	SCCGETCT	Get-Zähler für Benutzerspeicher über 16-MB-Grenze (ECDSA)
DFH STOR	121	SC31COCC	Belegung des Benutzerspeichers (bytes-ms) über 16-MB-Grenze (ECDSA)

Tabelle 13. Aus- oder einschließbare Datengruppen und Felder (Forts.)

Gruppenname	Feld-ID	Feldname	Beschreibung
DFHSTOR	122	PC31RHWM	Oberer Grenzwert für Programmspeicher (ERDSA)
DFHSTOR	139	PC31AHWM	Oberer Grenzwert für Programmspeicher über 16-MB-Grenze
DFHSTOR	142	PC31CHWM	Oberer Grenzwert für Programmspeicher (ECDSA)
DFHSTOR	143	PC24CHWM	Oberer Grenzwert für Programmspeicher (CDSA)
DFHSTOR	144	SC24SGCT	Zähler für gemeinsam genutzten Speicher der Getmain-Anforderungen unter 16-MB-Grenze (CDSA und SDSA)
DFHSTOR	145	SC24GSHR	Mithilfe einer getmain-Anforderung zugeordnete Byte des gemeinsam genutzten Speichers unter 16-MB-Grenze (CDSA und SDSA)
DFHSTOR	146	SC24FSHR	Mithilfe einer freemain-Anforderung freigegebener Speicherplatz des gemeinsam genutzten Speichers unter 16-MB-Grenze (CDSA und SDSA)
DFHSTOR	147	SC31SGCT	Zähler für getmain-Anforderungen des gemeinsam genutzten Speichers über 16-MB-Grenze (ECDSA und ESDSA)
DFHSTOR	148	SC31GSHR	Mithilfe einer getmain-Anforderung zugeordnete Byte des gemeinsam genutzten Speichers über 16-MB-Grenze (ECDSA und ESDSA)
DFHSTOR	149	SC31FSHR	Mithilfe einer freemain-Anforderung freigegebene Byte des gemeinsam genutzten Speichers über 16-MB-Grenze (ECDSA und ESDSA)
DFHSTOR	160	PC24SHWM	Oberer Grenzwert für Programmspeicher (SDSA)
DFHSTOR	161	PC31SHWM	Oberer Grenzwert für Programmspeicher (ESDSA)
DFHSTOR	162	PC24RHWM	Oberer Grenzwert für Programmspeicher (RDSA)
DFHSTOR	441	SC64CGCT	Get-Zähler für Benutzerspeicher oberhalb der Speicherlinie (GCDSA)
DFHSTOR	442	SC64CHWM	Oberer Grenzwert für Benutzerspeicher oberhalb der Speicherlinie (GCDSA)
DFHSTOR	443	SC64UGCT	Get-Zähler für Benutzerspeicher oberhalb der Speicherlinie (GUDSA)
DFHSTOR	444	SC64UHWM	Oberer Grenzwert für Benutzerspeicher oberhalb der Speicherlinie (GUDSA)
DFHSTOR	445	SC64SGCT	Get-Zähler für gemeinsamen Speicher oberhalb der Speicherlinie (GCDSA oder GUDSA)
DFHSTOR	446	SC64GSHR	Mithilfe einer GETMAIN-Anforderung zugeordnete Byte des gemeinsam genutzten Speichers oberhalb der Speicherlinie (GCDSA oder GUDSA)
DFHSTOR	447	SC64FSHR	Mithilfe einer FREEMAIN-Anforderung freigegebene Byte des gemeinsam genutzten Speichers oberhalb der Speicherlinie (GCDSA oder GUDSA)
DFHSYNC	60	SPSYNCCT	Zähler für Synchronisationspunkte
DFHSYNC	173	SYNCTIME	Abgelaufene Zeit für Synchronisationspunkte
DFHSYNC	177	SRVSYWTT	Synchronisationspunktwarezeit für CFDT-Server

Tabelle 13. Aus- oder einschließbare Datengruppen und Felder (Forts.)

Gruppenname	Feld-ID	Feldname	Beschreibung
DFHSYNC	196	SYNCDLY	Verzögerungszeit für Synchronisationspunkte
DFHSYNC	199	OTSINDWT	Wartezeit für unbestätigte OTS
DFHTASK	7	USRDISPT	Zuteilungszeit für Benutzertasks
DFHTASK	8	USRCPUT	CPU-Zeit für Benutzertasks
DFHTASK	14	SUSPTIME	Aussetzzeit für Benutzertasks
DFHTASK	31	TRANNUM	Tasknummer
DFHTASK	59	ICPUINCT	IC-Zähler für put/initiate
DFHTASK	64	TASKFLAG	Feld für Fehlerkennzeichen
DFHTASK	65	ICSTACCT	Anzahl lokaler START-Anforderungen mit der Option CHANNEL
DFHTASK	66	ICTOTCT	IC-Gesamtzähler
DFHTASK	82	TRNGRPID	Transaktionsgruppen-ID
DFHTASK	97	NETUOWPX	Netzname des ursprünglichen Terminals oder Systems
DFHTASK	98	NETUOWSX	ID der Arbeitseinheit auf dem Ursprungssystem
DFHTASK	102	DISPWTT	Zuteilungswartezeit für Benutzertasks
DFHTASK	109	TRANPRI	Transaktionspriorität
DFHTASK	123	GNQDELAY	Globale ENQ-Verzögerungszeit für Task
DFHTASK	124	BRDGTRAN	Transaktions-ID für 3270-Brücke
DFHTASK	125	DSPDELAY	Verzögerungszeit für erste Zuteilung
DFHTASK	126	TCLDELAY	Verzögerungszeit für erste Zuteilung aufgrund von TRANCLASS
DFHTASK	127	MXTDELAY	Verzögerung für erste Zuteilung aufgrund von MXT
DFHTASK	128	LMDELAY	Verzögerungszeit des Sperrenmanagers
DFHTASK	129	ENQDELAY	ENQ-Verzögerungszeit für Task
DFHTASK	132	RMUOWID	Arbeitseinheiten-ID für Wiederherstellungsmanager
DFHTASK	163	FCTYNAME	Name der Transaktionsfunktion
DFHTASK	164	TRANFLAG	Transaktionsflags
DFHTASK	166	TCLSNNAME	Transaktionsklassenname
DFHTASK	170	RMITIME	Abgelaufene Zeit für Ressourcenmanagerschnittstelle (RMI = Resource Manager Interface)
DFHTASK	171	RMISUSP	Aussetzzeit für Ressourcenmanagerschnittstelle (RMI = Resource Manager Interface)
DFHTASK	181	WTEXWAIT	Wartezeit für EXEC CICS WAIT EXTERNAL
DFHTASK	182	WTCEWAIT	Wartezeit für EXEC CICS WAITCICS und WAIT EVENT
DFHTASK	183	ICDELAY	Verzögerungszeit für Intervallsteuerung
DFHTASK	184	GVUPWAIT	Wartezeit für Zuteilung
DFHTASK	190	RRMSURID	ID für RRMS/MVS-Arbeitseinheit mit Wiederherstellung (URID)
DFHTASK	191	RRMSWAIT	RRMS/MVS-Wartezeit
DFHTASK	192	RQRWAIT	Wartezeit für Anforderungsempfänger
DFHTASK	193	RQPWAIT	Wartezeit für Anforderungsprozessor
DFHTASK	194	OTSTID	OTS-Transaktions-ID (TID)
DFHTASK	195	RUNTRWTT	CICS BTS-Wartezeit für synchrone Ausführung von Prozessen/Aktivitäten
DFHTASK	247	DSCHMDLY	Verzögerungszeit für TCB-Änderungsmodus
DFHTASK	249	QRMODDLY	Zuteilungswartezeit für QR-Tasksteuerblock der Benutzertask

Tabelle 13. Aus- oder einschließbare Datengruppen und Felder (Forts.)

Gruppenname	Feld-ID	Feldname	Beschreibung
DFHTASK	250	MXTOTDLY	CICS-Verzögerungszeit für offenen Tasksteuerblock im L8- und L9-Modus
DFHTASK	251	TCBATTCT	CICS-Zähler für Tasksteuerblockverbindung
DFHTASK	252	DSTCBHWM	Zähler für Benutzertaskspitzenwert für geöffnete Tasksteuerblöcke
DFHTASK	253	JVMTIME	Abgelaufene Zeit für CICS-JVM
DFHTASK	254	JVMSUSP	Aussetzzeit für CICS-JVM
DFHTASK	255	QRDISPT	Zuteilungszeit für QR-Tasksteuerblock der Benutzertask
DFHTASK	256	QRCPUT	CPU-Zeit für QR-Tasksteuerblock der Benutzertask
DFHTASK	257	MSDISPT	Zuteilungszeit für MS-Tasksteuerblock der Benutzertask
DFHTASK	258	MSCPUT	CPU-Zeit für MS-Tasksteuerblock der Benutzertask
DFHTASK	259	L8CPUT	CPU-Zeit für L8-Tasksteuerblock der Benutzertask
DFHTASK	261	S8CPUT	CPU-Zeit für S8-Tasksteuerblock der Benutzertask
DFHTASK	262	KY8DISPT	Zuteilungszeit für Tasksteuerblock mit Schlüssel 8 der Benutzertask
DFHTASK	263	KY8CPUT	CPU-Zeit für Tasksteuerblock mit Schlüssel 8 der Benutzertask
DFHTASK	264	KY9DISPT	Zuteilungszeit für Tasksteuerblock mit Schlüssel 9 der Benutzertask
DFHTASK	265	KY9CPUT	CPU-Zeit für Tasksteuerblock mit Schlüssel 9 der Benutzertask
DFHTASK	266	L9CPUT	CPU-Zeit für L9-Tasksteuerblock der Benutzertask
DFHTASK	268	DSTCBMWT	Wartezeit bei Abweichung für Tasksteuerblock der Benutzertask
DFHTASK	269	RODISPT	Zuteilungszeit für RO-Tasksteuerblock der Benutzertask
DFHTASK	270	ROCPUT	CPU-Zeit für RO-Tasksteuerblock der Benutzertask
DFHTASK	271	X8CPUT	CPU-Zeit für X8-Tasksteuerblock der Benutzertask
DFHTASK	272	X9CPUT	CPU-Zeit für X9-Tasksteuerblock der Benutzertask
DFHTASK	273	JVMITIME	Abgelaufene Zeit für Initialisierung der CICS-JVM
DFHTASK	275	JVMRTIME	Abgelaufene Zeit für Zurücksetzung der CICS-JVM
DFHTASK	279	DSMMSCWT	Verzögerungszeit für MVS-Speicherengpass
DFHTASK	281	MAXSTDLY	Verzögerungszeit für CICS-SSL-Tasksteuerblock
DFHTASK	282	MAXXTDLY	Verzögerungszeit für CICS-XP-Tasksteuerblock
DFHTASK	283	MAXTTDLY	Verzögerungszeit für Tasksteuerblock des CICS-JVM-Server-Threads
DFHTASK	285	PTPWAIT	Wartezeit für Partner der 3270-Brücke
DFHTASK	345	ICSTACDL	Länge der Daten in den Containern aller lokal ausgeführten START CHANNEL-Anforderungen

Tabelle 13. Aus- oder einschließbare Datengruppen und Felder (Forts.)

Gruppenname	Feld-ID	Feldname	Beschreibung
DFHTASK	346	ICSTRCCT	Anzahl der START CHANNEL-Anforderungen zur Intervallsteuerung, die auf fernen Systemen ausgeführt werden sollen
DFHTASK	347	ICSTRCDL	Länge der Daten in den Containern aller fern ausgeführten START CHANNEL-Anforderungen
DFHTASK	348	ROMODDLY	Zuteilungswartezeit für RO-Tasksteuerblock der Benutzertask
DFHTASK	349	SOMODDLY	Zuteilungswartezeit für SO-Tasksteuerblock der Benutzertask
DFHTASK	400	T8CPUT	CPU-Zeit für T8-Tasksteuerblock der Benutzertask
DFHTASK	401	JVMTHDWT	Wartezeit für JVMSERVER-Thread. Dies gilt nicht für Liberty-JVM-Server.
DFHTASK	429	DSAPTHWT	Wartezeit für pthread-Zuordnung des Dispatchers
DFHTASK	430	CECMCHTP	CEC-Maschinentyp
DFHTASK	431	CECMDLID	CEC-Modellnummer
DFHTASK	432	LPARNAME	Der (in EBCDIC codierte) Name der logischen Partition (LPAR) auf dem Prozessor, auf dem die CICS-Region ausgeführt wird
DFHTASK	433	MAXTASKS	MXT oder MAXTASKS für die CICS-Region beim Zuordnen der Benutzertask
DFHTASK	434	CURTASKS	Aktuelle Tasks in CICS-Region beim Zuordnen der Benutzertask
DFHTASK	436	CPUTONCP	Für die Task erforderliche Prozessorzeit auf einem Standardprozessor
DFHTASK	437	OFFLCPUT	Für die Task erforderliche Prozessorzeit, die für die Auslagerung auf einen Sonderprozessor verfügbar ist
DFHTASK	451	ACAPPLNM	Der aus 64 Zeichen bestehende Name der Anwendung in den Anwendungskontextdaten
DFHTASK	452	ACPLATNM	Der aus 64 Zeichen bestehende Name der Plattform in den Anwendungskontextdaten
DFHTASK	453	ACMAJVER	Die Hauptversion der Anwendung in den Anwendungskontextdaten in Form eines aus vier Byte bestehenden Binärwerts
DFHTASK	454	ACMINVER	Die Nebenversion der Anwendung in den Anwendungskontextdaten in Form eines aus vier Byte bestehenden Binärwerts
DFHTASK	455	ACMICVER	Die Mikroversion der Anwendung in den Anwendungskontextdaten in Form eines aus vier Byte bestehenden Binärwerts
DFHTASK	456	ACOPERNM	Der aus 64 Zeichen bestehende Name der Operation in den Anwendungskontextdaten
DFHTASK	470	ASTOTCT	Die Gesamtzahl asynchroner EXEC CICS-API-Befehle, die von der Benutzertask abgesetzt wurden. Hierzu gehören die Befehle <b>RUN TRANSID</b> , <b>FETCH CHILD</b> und <b>FETCH ANY</b> .
DFHTASK	471	ASRUNCT	Die Anzahl der <b>EXEC CICS RUN TRANSID</b> -Befehle, die von der Benutzertask abgesetzt wurden
DFHTASK	472	ASFTCHCT	Die Anzahl der <b>EXEC CICS FETCH CHILD</b> - und <b>EXEC CICS FETCH ANY</b> -Befehle, die von der Benutzertask abgesetzt wurden

Tabelle 13. Aus- oder einschließbare Datengruppen und Felder (Forts.)

Gruppenname	Feld-ID	Feldname	Beschreibung
DFHTASK	473	ASFREET	Die Anzahl der <b>EXEC CICS FREE CHILD</b> -Befehle, die von der Benutzertask abgesetzt wurden
DFHTASK	475	ASFTCHWT	Die abgelaufene Zeit, die die Benutzertask nach Absetzen eines Befehls <b>EXEC CICS FETCH CHILD</b> oder <b>EXEC CICS FETCH ANY</b> auf eine untergeordnete Task gewartet hat, die nicht abgeschlossen wurde
DFHTASK	476	ASRNATWT	Die abgelaufene Zeit für die Verzögerung der Benutzertask aufgrund der Einschränkungen einer asynchronen untergeordneten Task, die über die asynchrone Servicedomäne verwaltet wird
DFHTEMP	11	TSIOWTT	E/A-Wartezeit für TS
DFHTEMP	44	TSGETCT	Zähler für TS-Get-Anforderungen
DFHTEMP	46	TSPUTACT	Zusätzlicher Zähler für TS-Put-Anforderungen
DFHTEMP	47	TSPUTMCT	Hauptzähler für TS-Put-Anforderungen
DFHTEMP	92	TSTOTCT	TS-Gesamtzähler
DFHTEMP	178	TSSHWAIT	E/A-Wartezeit für gemeinsam genutzten TS
DFHTEMP	460	TSGETSCT	Zähler für gemeinsam genutzte TS-Get-Anforderungen
DFHTEMP	461	TSPUTSCT	Zähler für gemeinsam genutzte TS-Put-Anforderungen
DFHTERM	9	TCIOWTT	E/A-Wartezeit für TC
DFHTERM	34	TCMSGIN1	Eingabenachrichten für TC-Hauptfunktion
DFHTERM	35	TCMSGOU1	Ausgabenachrichten für TC-Hauptfunktion
DFHTERM	67	TCMSGIN2	Eingabenachrichten für alternative TC-Funktion
DFHTERM	68	TCMSGOU2	Ausgabenachrichten für alternative TC-Funktion
DFHTERM	69	TCALLOCT	Zähler für TC-Zuordnungen
DFHTERM	83	TCCHRIN1	CHARS-Eingabe für TC-Hauptfunktion
DFHTERM	84	TCCHROU1	CHARS-Ausgabe für TC-Hauptfunktion
DFHTERM	85	TCCHRIN2	CHARS-Eingabe für alternative TC-Funktion
DFHTERM	86	TCCHROU2	CHARS-Ausgabe für alternative TC-Funktion
DFHTERM	100	IRIOWTT	E/A-Wartezeit für IR
DFHTERM	111	LUNAME	LU-Name des z/OS Communications Server-Terminals
DFHTERM	133	LU61WTT	TC-E/A-Wartezeit - LU6.1
DFHTERM	134	LU62WTT	TC-E/A-Wartezeit - LU6.2
DFHTERM	135	TCM62IN2	Eingabenachrichten für alternative TC-Funktion - LU6.2
DFHTERM	136	TCM62OU2	Ausgabenachrichten für alternative TC-Funktion - LU6.2
DFHTERM	137	TCC62IN2	CHARS-Eingabe für alternative TC-Funktion - LU6.2
DFHTERM	138	TCC62OU2	CHARS-Ausgabe für alternative TC-Funktion - LU6.2
DFHTERM	165	TERMINFO	Terminalinformationen
DFHTERM	169	TERMCNM	Verbindungsname für Terminalsitzung
DFHTERM	197	NETID	Netz-ID für netzinternen Namen
DFHTERM	198	RLUNAME	Netzname für netzinternen Namen
DFHTERM	343	TCALWTT	Wartezeit für Zuordnung von MRO-, LU6.1- und LU6.2-Sitzungen
DFHWEBB	224	WBREADCT	Zähler für Webleseanforderungen
DFHWEBB	225	WBWRITCT	Zähler für Webschreibanforderungen



Tabelle 13. Aus- oder einschließbare Datengruppen und Felder (Forts.)

Gruppenname	Feld-ID	Feldname	Beschreibung
DFHWEBB	231	WBRCVCT	Zähler für Webempfangsanforderungen
DFHWEBB	232	WBCHRIN	Empfangene Webzeichen
DFHWEBB	233	WSENDCT	Zähler für Websendeanforderungen
DFHWEBB	234	WBCHROUT	Gesendete Webzeichen
DFHWEBB	235	WBTOTCT	Zähler für Gesamtzahl der Webanforderungen
DFHWEBB	236	WBREPRCT	Zähler für Leseanforderungen für Web-Header oder Formularfelder
DFHWEBB	237	WBREPWCT	Zähler für Schreibenanforderungen für Web-Header oder Formularfelder
DFHWEBB	238	WBEXTRCT	Zähler für Webextraktionsanforderungen
DFHWEBB	239	WBBRWCT	Zähler für Websuchanforderungen
DFHWEBB	331	WBREDOCT	Zähler für Leseanforderungen für Web-Header, Client
DFHWEBB	332	WBWRTOCT	Zähler für Schreibenanforderungen für Web-Header, Client
DFHWEBB	333	WBRCVIN1	Zähler für Empfangs- oder Konvertierungsanforderungen des Web-Clients
DFHWEBB	334	WBCHRIN1	Empfangene Webzeichen, Client
DFHWEBB	335	WBSNDOU1	Zähler für Sende- oder Konvertierungsanforderungen des Web-Clients
DFHWEBB	336	WBCHROU1	Gesendete Webzeichen, Client
DFHWEBB	337	WBPARSCT	Zähler für URL-Auswertungsanforderungen des Web-Clients
DFHWEBB	338	WBRWOCT	Zähler für Suchanforderungen des Web-Clients
DFHWEBB	340	WBIWBSCT	Zähler für EXEC CICS INVOKE SERVICE- oder WEBSERVICE-Anforderungen
DFHWEBB	341	WBREPRDL	Zähler für Lesebyte für Web-Header oder Formularfelder
DFHWEBB	342	WBREPWDL	Zähler für Schreibbyte für Web-Header oder Formularfelder
DFHWEBB	380	WBURIMNM	URIMAP-Ressourcenname
DFHWEBB	381	WBIPLNM	PIPELINE-Ressourcenname
DFHWEBB	382	WBATMSNM	ATOMSERVICE-Ressourcenname
DFHWEBB	383	WBSVCENM	WEBSERVICE-Ressourcenname
DFHWEBB	384	WBSVOPNM	WEBSERVICE-Operationsname
DFHWEBB	385	WBPROGNM	PROGRAM-Ressourcenname
DFHWEBB	386	WBSFCRCT	Zähler für SOAPFAULT-Erstellungsanforderungen
DFHWEBB	387	WBSFTOTCT	Gesamtzähler für SOAPFAULT-Anforderungen
DFHWEBB	388	WBISSFCT	EXEC CICS INVOKE SERVICE SOAP-Fehler
DFHWEBB	390	WBSREQBL	Länge des SOAP-Anforderungshauptteils
DFHWEBB	392	WBSRSPBL	Länge des SOAP-Antworthauptteils
DFHWEBB	412	MLXSSTD	Summe der eingehenden konvertierten Dokumentlänge
DFHWEBB	413	MLXMLTCT	Zähler für EXEC CICS TRANSFORM-Anforderungen
DFHWEBB	420	WSACBLCT	Zähler für Buildanforderungen für WS-Addressing-Kontext
DFHWEBB	421	WSACGTCT	Zähler für Get-Anforderungen für WS-Addressing-Kontext
DFHWEBB	422	WSAEPCCCT	Zähler für Erstellungsanforderungen für WS-Addressing-Endpunktreferenzen (EPR)
DFHWEBB	423	WSATOTCT	Gesamtzähler für WS-Addressing-Anforderungen

Tabelle 13. Aus- oder einschließbare Datengruppen und Felder (Forts.)

Gruppenname	Feld-ID	Feldname	Beschreibung
DFHWEBB	424	WBJSNRQL	JSON-Nachrichtenanforderungslänge für JSON-Web-Service-Anwendungen
DFHWEBB	425	WBJSNRPL	JSON-Nachrichtenantwortlänge für JSON-Web-Service-Anwendungen

#### **INCLUDE=(m1[,...])**

Geben Sie diesen Operanden an, damit einzelne oder mehrere CICS-Felder von der Überwachungsfunktion gemeldet werden können. Standardmäßig werden alle dokumentierten Felder für Leistungsklassen aufgelistet. Daher ist dieser Operand nur von Bedeutung, wenn Sie den Operanden EXCLUDE in demselben Makro angeben.

Die Felder, die für den Einschluss in diesen Operanden angegeben werden können, sind identisch mit den Feldern, die für den Ausschluss ausgewählt werden können. (Siehe hierzu die Beschreibung des Operanden EXCLUDE.) Jedem Feld ist eine numerische Feldkennung zugeordnet. Um ein Feld einzuschließen, müssen Sie als numerische Kennung des Felds *m1* angeben. Sie können mehrere numerische Kennungen angeben.

**Anmerkung:** Geben Sie in numerischen Kennungen keine führenden Nullen an. Geben Sie keine numerischen Kennungen von Feldern an, die nicht für den Ausschluss auswählbar sind und deshalb standardmäßig eingeschlossen werden.

Der Operand EXCLUDE wird immer zuerst berücksichtigt. Der Operand INCLUDE überschreibt (sofern er angegeben wird), einige seiner Auswirkungen. Mit dem folgenden Code wird z. B. die Erfassung und Auflistung von PUT-Anweisungen für die Dateisteuerung und die Gesamtzahl der Dateisteuerungsanweisungen sichergestellt, der Suchzähler der Dateisteuerung und andere Dateisteuerungsfelder werden jedoch ausgeschlossen:

```
EXCLUDE=DFHFILE,
INCLUDE=(37,93)
```

Wenn Sie die Mehrzahl der Felder ausschließen, einige jedoch einschließen wollen, dann können Sie die Operanden in der im folgenden Beispiel dargestellten Weise verwenden:

```
EXCLUDE=ALL,
INCLUDE=(DFHTERM,97,98)
```

Diese Vorgehensweise eignet sich besser als die einzelne Angabe aller Felder, die ausgeschlossen werden sollen.

CICSTS55.SDFHSAMP stellt die folgenden Beispiele für Überwachungssteuertabellen zur Verfügung:

- Für Terminalverwaltungsregionen (TOR = Terminal-Owning Regions): DFHMCTT\$
- Für Anwendungsverwaltungsregionen (AOR = Application-Owning Regions): DFHMCTA\$
- Für Anwendungsverwaltungsregionen (AOR) mit DBCTL: DFHMCTD\$
- Für Dateiverwaltungsregionen (FOR = File-Owning Regions): DFHMCTF\$

Diese Beispiele zeigen, wie die Operanden EXCLUDE und INCLUDE zur Reduzierung der Größe des Leistungsklassendatensatzes verwendet werden können, um so das Datenvolumen zu verkleinern, das von CICS in SMF geschrieben werden muss.

## DFHMCT-Beispiele

Diese Beispiele zeigen die Verwendung der Makros MCT TYPE=INITIAL, DFHMCT TYPE=RECORD und DFHMCT TYPE=EMP zur Erstellung des Steuerabschnitts für die MCT (Monitoring Control Table; Überwachungssteuertabelle), zur Angabe der Ereignisüberwachungspunkte (EMPs = Event Monitoring Points) des Benutzers und zum Ausschluss der systemdefinierten Felder.

Abb. 8 veranschaulicht die Codierung für die Steueranweisungen im Makro DFHMCT TYPE=INITIAL. Im Beispiel ist das Suffix C2 angegeben, sodass der Name dieser Überwachungssteuertabelle DFHMCTC2 lautet. Die anderen Parameter werden auf ihre Standardwerte gesetzt.

DFHMCT	TYPE=INITIAL,SUFFIX=C2,COMPRESS=YES,	*
	APPLNAME=NO,RMI=NO,FILE=8,DPL=0,TSQUEUE=8	
DFHMCT	TYPE=FINAL	
	END	

Abbildung 8. Beispiel: DFHMCT TYPE=INITIAL

Abb. 9 veranschaulicht die Codierung zum Erstellen einer Überwachungssteuertabelle (MCT = Monitoring Control Table) für zwei Ereignisüberwachungspunkte des Benutzers.

DFHMCT	TYPE=INITIAL,SUFFIX=MF,COMPRESS=YES	
DFHMCT	TYPE=EMP,	*
	ID=180,	*
	CLASS=PERFORM,	*
	PERFORM=(SCLOCK(1),ADDCNT(2,1))	
DFHMCT	TYPE=EMP,	*
	ID=181,	*
	CLASS=PERFORM,	*
	PERFORM=PCLOCK(1)	
DFHMCT	TYPE=FINAL	
	END	

Abbildung 9. Beispiel: DFHMCT TYPE=EMP, Ereignisüberwachungspunkte des Benutzers

Abb. 10 auf Seite 442 veranschaulicht die Codierung zum Erstellen einer Überwachungssteuertabelle (MCT = Monitoring Control Table), die spezielle Datenfelder aus dem Leistungsdatensatz ausschließt. Wenn Sie über keine Anwendungen verfügen, die bestimmte CICS-Funktionen verwenden, dann können Sie die Leistungsdatengruppen und die Felder, die zu diesen Funktionen gehören, ohne Risiko ausschließen. Das Beispiel zeigt die DFHMCT TYPE=RECORD-Makros zum Ausschließen der Felder für diese Funktionen:

### Front-End-Programmierschnittstelle (FEPI = Front End Programming Interface)

DFHMCT	TYPE=RECORD,CLASS=PERFORM,	*
	EXCLUDE=(DFHFEPI)	

### Business Transaction Services (BTS)

DFHMCT	TYPE=RECORD,CLASS=PERFORM,	*
	EXCLUDE=(DFHCBTS,195,196)	

## Unterstützung für Java-Anwendungen

DFHMCT TYPE=RECORD,CLASS=PERFORM,  
EXCLUDE=(253,254,273,275)

\*

DFHMCT	TYPE=INITIAL,SUFFIX=CB,COMPRESS=YES, APPLNAME=NO,RMI=NO,FILE=16,TSQUEUE=12	*
DFHMCT	TYPE=RECORD,CLASS=PERFORM, EXCLUDE=(DFHFEPI)	*
DFHMCT	TYPE=RECORD,CLASS=PERFORM, EXCLUDE=(DFHCBTS,195,196)	*
DFHMCT	TYPE=RECORD,CLASS=PERFORM, EXCLUDE=(253,254,273,275)	*
DFHMCT	TYPE=FINAL END	

Abbildung 10. Beispiel: DFHMCT TYPE=RECORD, Felder ausschließen

## PLT: Programmlistentabelle

Eine Programmlistentabelle (PLT = Program List Table) gibt Programme an, die während des CICS-Systemstarts und -Systemabschlusses ausgeführt werden sollen. Außerdem werden in der PLT Gruppen von Programmen angegeben, die Sie zusammen aktivieren und inaktivieren wollen.

Möglicherweise möchten Sie aus den folgenden Gründen mehrere PLTs angeben:

- Zum Angeben einer Liste mit Programmen, die in der zweiten und dritten Initialisierungsphase des CICS-Systemstarts ausgeführt werden sollen. Weitere Einzelheiten zu den Initialisierungsphasen finden Sie in Using post-initialization (PLTPI) programs. Programmierinformationen zu den Einschränkungen, die für die Verwendung von Programmen in den Initialisierungsphasen gelten, finden Sie in Writing initialization programs. Die ausgewählte Liste muss während der Initialisierung mit dem Systeminitialisierungsparameter PLTPI=*name* angegeben werden. Hierbei steht *name* für ein aus einem oder zwei Zeichen bestehendes Suffix der PLT oder für den vollständigen Namen der PLT, die die erforderliche Programmliste enthält.

Aus Gründen der Benutzerfreundlichkeit wird die Liste der Programme, die zur Ausführung während der Initialisierung ausgewählt wird, als die Liste 'PLTPI' bezeichnet.

- Zum Angeben einer Liste mit Programmen, die Sie während der ersten und zweiten Wartephase beim kontrollierten Systemabschluss ausführen möchten. Die ausgewählte Liste muss während der Initialisierung mit dem Systeminitialisierungsparameter PLTSD=*name* angegeben werden. Hierbei steht *name* für ein aus einem oder zwei Zeichen bestehendes Suffix der PLT oder für den vollständigen Namen der PLT, die die erforderliche Programmliste enthält.

Die im Systeminitialisierungsparameter PLTSD angegebene PLT kann beim Systemabschluss mit der Option PLT im Befehl CEMT PERFORM SHUTDOWN überschrieben werden.

Aus Gründen der Benutzerfreundlichkeit wird die Liste der Programme, die zur Ausführung während des Systemabschlusses ausgewählt wird, als die Liste 'PLTSD' bezeichnet.

- Zum Angeben einer Liste von Programmen, die mithilfe eines Hauptterminalbefehls ENABLE oder DISABLE als Gruppe aktiviert oder inaktiviert werden sollen. Diese Verwendung von PLTs bedeutet, dass ein Hauptterminalbediener eine Gruppe von Programmen mit nur einem Befehl aktivieren oder inaktivieren kann, anstatt für jedes Programm einen separaten Befehl eingeben zu müssen.

1

1

1

1

1



rend der Wiederherstellung benötigt werden. Definieren Sie die Benutzerexitprogramme in der CSD-Datei. Andernfalls ist CICS möglicherweise nicht in der Lage, nach Abschluss der CICS-Initialisierung auf diese Programme zuzugreifen (z. B. in dem Befehl **EXEC CICS DISABLE**). Allerdings sollten Sie hierbei beachten, dass die von RDO definierten Eigenschaften während der zweiten Phase der Initialisierung keine Auswirkung haben.

- Programme, die nach dem Eintrag PROGRAM=DFHDELIM in einer PLTPI aufgeführt sind, werden während der dritten Phase der Initialisierung ausgeführt. Wenn diese Programme zum Aktivieren von Benutzerexits verwendet werden, müssen die Benutzerexitprogramme auch in der CSD-Datei definiert sein oder müssen automatisch installiert werden können.
- Programme, die vor dem Eintrag PROGRAM=DFHDELIM in einer PLTSD aufgeführt sind, werden während der ersten Wartephase des Systemabschlusses ausgeführt.
- Programme, die nach dem Eintrag PROGRAM=DFHDELIM in einer PLTSD aufgeführt sind, werden während der zweiten Wartephase des Systemabschlusses ausgeführt.

Für die zweite Phase der Initialisierung und die zweite Wartephase benötigen PLT-Programme keine Programmressourcendefinitionen. Wenn sie nicht definiert sind, werden sie vom System (unabhängig von den Systeminitialisierungsparametern für die automatische Programminstallation) automatisch installiert. Dies bedeutet, dass der Exit für die automatische Installation nicht aufgerufen wird, damit die Definition geändert werden kann. Die Programme werden mit den folgenden Attributen definiert:

```
LANGUAGE(ASSEMBLER) STATUS(ENABLED) CEDF(NO)
DATALOCATION(BELOW) EXECKEY(CICS)
EXECUTIONSET(FULLAPI)
```

Demzufolge verfügen die vom System automatisch installierten Programme für CONCURRENCY standardmäßig über die Einstellung QUASIRENT und die standardmäßige API-Einstellung CICSAPI.

- Für diese threadsicheren PLT-Programme, die mit dem OPENAPI-Wert für das Attribut API definiert werden, oder bei denen es sich um C- oder C++-Programme handelt, die mit der Compileroption XPLINK kompiliert wurden, müssen Sie eine geeignete Ressourcendefinition bereitstellen. Alternativ hierzu können Sie für Language Environment-konforme Programme auch die Laufzeitoption CICS-VAR verwenden, um die richtigen Werte für CONCURRENCY und API zu setzen. Weitere Informationen hierzu finden Sie in Defining runtime options for Language Environment.

PLT-Programme für die dritte Phase der Initialisierung und die erste Wartephase können mit der automatischen Programminstallation definiert werden, und zwar abhängig von den Systeminitialisierungsparametern für die automatische Programminstallation. Wird die automatische Programminstallation nicht verwendet, dann müssen diese Programme über Programmressourcendefinitionen in der CSD-Datei verfügen.

## DFHPLT-Beispiel

Die zum Generieren einer PLT (Program List Table; Programmlistentabelle) erforderliche Codierung kann anhand eines Beispiels verdeutlicht werden.

In Abb. 11 auf Seite 446 und Abb. 12 auf Seite 446 ist die Codierung dargestellt, die zum Generieren einer Programmlistentabelle (PLT = Program List Table) erforder-

lich ist.

```
*
* LIST OF PROGRAMS TO BE EXECUTED SEQUENTIALLY DURING SYSTEM
* INITIALIZATION.
* REQUIRED SYSTEM INITIALIZATION PARAMETER: PLTPI=I1
*
  DFHPLT TYPE=INITIAL,SUFFIX=I1
*
* The following programs are run in the first pass of PLTPI
*
  DFHPLT TYPE=ENTRY,PROGRAM=TRAQA EXECUTED DURING 2ND INIT. PHASE
  DFHPLT TYPE=ENTRY,PROGRAM=TRAQB (PROGRAMS SHOULD ALSO BE DEFINED
  DFHPLT TYPE=ENTRY,PROGRAM=TRAQC BY RDO)
*
  DFHPLT TYPE=ENTRY,PROGRAM=DFHDELIM
*
*
* The following programs are run in the second pass of PLTPI
*
  DFHPLT TYPE=ENTRY,PROGRAM=TRASA EXECUTED DURING 3RD INIT. PHASE
  DFHPLT TYPE=ENTRY,PROGRAM=TRASB (PROGRAMS MUST ALSO BE DEFINED
  DFHPLT TYPE=ENTRY,PROGRAM=TRASC BY RDO)
  DFHPLT TYPE=FINAL
*
  END
```

Abbildung 11. Programmlistentabelle PLTPI: Beispiel

```
*
*
* LIST OF PROGRAMS TO BE EXECUTED SEQUENTIALLY DURING SYSTEM
* TERMINATION
* REQUIRED SYSTEM INITIALIZATION PARAMETER: PLTSD=T1
*
  DFHPLT TYPE=INITIAL,SUFFIX=T1
*
* The following programs are run in the 1st pass of PLTSD
*
*
  DFHPLT TYPE=ENTRY,PROGRAM=TRARA EXECUTED DURING 1st QUIESCE PHASE
  DFHPLT TYPE=ENTRY,PROGRAM=TRARB (PROGRAMS MUST ALSO BE DEFINED
  DFHPLT TYPE=ENTRY,PROGRAM=TRARC BY RDO)
*
  DFHPLT TYPE=ENTRY,PROGRAM=DFHDELIM
*
*
* The following programs are run in the 2nd pass of PLTSD
*
  DFHPLT TYPE=ENTRY,PROGRAM=TRAFA EXECUTED DURING 2nd QUIESCE PHASE
  DFHPLT TYPE=ENTRY,PROGRAM=TRAFB (PROGRAMS MUST ALSO BE DEFINED
  BY RDO)
*
  DFHPLT TYPE=FINAL
*
  END
```

Abbildung 12. Programmlistentabelle PLTSD: Beispiel

## RST: Wiederherstellbare Servicetabelle

**Hinweis zur Terminologie:** DBCTL bezieht sich auf die CICS-IMS/ESA-DBCTL-Schnittstelle (DBCTL = Database Control).



Die wiederherstellbare Servicetabelle (RST = Recoverable Service Table) wird für die CICS DBCTL XRF-Unterstützung (XRF = Extended Recovery Facility) verwendet. Sie enthält eine Beschreibung der DBCTL-Konfiguration. Wenn ein DBCTL-Fehler festgestellt wird, dann verwendet das **aktive** CICS-System die RST zusammen mit dem MVS-Subsystem VERIFY, um zu ermitteln, ob ein passendes alternatives DBCTL-Subsystem vorhanden ist. Das **alternative** CICS-System verwendet die RST, um zu überprüfen, ob ein DBCTL-Subsystem vorhanden ist. Aus Sicherheitsgründen sollte für die RST eine Programmverbindung zu einer Bibliothek mit APF-Berechtigung hergestellt werden. Die RST wird nicht zusammen mit dem CICS-Nukleus geladen.

Sie können in einer wiederherstellbaren Servicetabelle die folgenden Makros angeben:

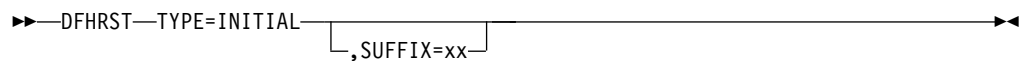
- DFHRST TYPE=INITIAL richtet den Programmabschnitt ein.
- DFHRST TYPE=RSE gibt den Start eines wiederherstellbaren Serviceelements (RSE = Recoverable Service Element) an. Ein RSE besteht aus einer (nicht leeren) Gruppe von Kennungen funktional entsprechender DBCTL-Subsysteme, den CICS-Anwendungs-IDs, die diesen DBCTL-Subsystemen zugeordnet sind, sowie den Anwendungskennungen der CICS-Systeme, die die DBCTL-Subsysteme verwenden.
- DFHRST TYPE=SBSYS gibt eines der DBCTL-Subsysteme in einem RSE an.
- DFHRST TYPE=FINAL schließt die RST ab (siehe TYPE=FINAL (end of table)).

Die vom System zu verwendende RST muss in der Systeminitialisierungstabelle mithilfe des folgenden Systeminitialisierungsparameters definiert werden:



## Programmabschnitt: DFHRST TYPE=INITIAL

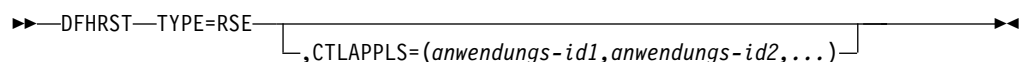
Das Makro DFHRST TYPE=INITIAL dient zum Einrichten eines Programmabschnitts für eine wiederherstellbare Servicetabelle (RST = Recoverable Service Table).



Allgemeine Informationen zu den TYPE=INITIAL-Makros sowie zur Verwendung des Operanden SUFFIX finden Sie in TYPE=INITIAL (control section).

## Wiederherstellbare Serviceelemente: DFHRST TYPE=RSE

Das Makro DFHRST TYPE=RSE enthält Informationen zu einem wiederherstellbaren Serviceelement (RSE = Recoverable Service Element) und zu den CICS-Systemen, die erwartungsgemäß eine Verbindung zu den DBCTL-Subsystemen im RSE herstellen.



**TYPE=RSE**

Gibt den Anfang einer Gruppe mit Kennungen funktional entsprechender DBCTL-Systeme an. Funktional entsprechende DBCTL-Subsysteme haben den gleichen Datenbanknamen (DB).

**CTLAPPLS=(anwendungs-id1,anwendungs-id2,...)**

Gibt die Anwendungskennungen der CICS-Systeme an, die zum Neustart der DBCTL-Subsysteme im RSE berechtigt sind.

**DBCTL-Subsysteme: DFHRST TYPE=SUBSYS**

Das Makro DFHRST TYPE=SUBSYS enthält Informationen zu den speziellen DBCTL-Subsystemen in einem RSE (Recoverable Service Element; wiederherstellbares Serviceelement).

Für jedes Subsystem in einem RSE muss ein Makro DFHRST TYPE=SUBSYS vorhanden sein.

```

▶▶DFHRST—TYPE=SUBSYS—,SYBSYSID=subsystem-id—————▶
|
|,JOBNAME=(jobname1,jobname2,...)
|
▶

```

**TYPE=SUBSYS**

Definiert eines der DBCTL-Subsysteme in einem RSE.

**SUBSYSID=subsystem-id**

Geben Sie diesen Operanden mit dem aus vier Zeichen bestehenden Namen der DBCTL-Subsystem-ID an. Diese Kennung muss innerhalb der RST (Recoverable Service Table; wiederherstellbare Servicetabelle) eindeutig sein.

**JOBNAME=(jobname1,jobname2,...)**

Geben Sie diesen Operanden mit den MVS **JOBNAME(S)** an, die dem DBCTL-Subsystem, das in der Parameteranweisung SUBSYSID angegeben ist, zugeordnet sind. Hierbei handelt es sich um eine Form der Sicherheitsprüfung. Wenn CICS die Ausführung des DBCTL-Subsystems abbrechen muss, dann ist dies nur zulässig, wenn der entsprechende MVS-Jobname, der dem DBCTL-Subsystem zugeordnet ist, im Parameter **JOBNAME** angegeben wurde.

**DFHRST-Beispiel**

Die zum Generieren einer RST (Recoverable Service Table; wiederherstellbare Servicetabelle) erforderliche Codierung kann anhand eines Beispiels verdeutlicht werden.

In Abb. 13 auf Seite 449 ist die Codierung zum Generieren einer RST dargestellt.

```

DFHRST  TYPE=INITIAL
        ,SUFFIX=K1
DFHRST  TYPE=RSE
        ,CTLAPPLS=(applid1,applid2,applid3)
DFHRST  TYPE=SUBSYS
        ,SUBSYSID=CTL1
        ,JOBNAME=(job1,job2,job3,job4)
DFHRST  TYPE=SUBSYS
        ,SUBSYSID=CTL2
        ,JOBNAME=(job5,job6,job7,job8)
DFHRST  TYPE=FINAL
END

```

Abbildung 13. Wiederherstellbare Servicetabelle: Beispiel

## SRT: Systemwiederherstellungstabelle

Die Systemwiederherstellungstabelle (SRT = System Recovery Table) enthält eine Liste mit Codes für abnormale Beendigungen, die von CICS abgefangen werden. Nachdem CICS eine abnormale Beendigung abgefangen hat, wird versucht, den Betrieb aufrecht zu erhalten, indem die Ausführung der fehlerhaften Task abgebrochen wird.

Sie können die standardmäßige Wiederherstellungsaktion ändern, indem Sie ein eigenes Wiederherstellungsprogramm schreiben. Hierzu können Sie den globalen Benutzerexitpunkt XSRAB im Systemwiederherstellungsprogramm (SRP = System Recovery Program) benutzen. (Programmierungsinformationen zum Exit XSRAB finden Sie in System recovery program exit XSRAB.)

Beachten Sie hierbei, dass eine Wiederherstellung versucht wird, wenn eine Benutzertask (jedoch keine Systemtask) zu dem Zeitpunkt, zu dem die abnormale Beendigung auftritt, die Steuerung hat.

In einer Systemwiederherstellungstabelle können die folgenden Makros codiert werden:

- DFHSRT TYPE=INITIAL richtet den Programmabschnitt ein.
- DFHSRT TYPE=SYSTEM|USER gibt die Abbruchcodes an, die verarbeitet werden müssen.
- DFHSRT TYPE=FINAL schließt die SRT ab (siehe TYPE=FINAL (end of table)).

### Programmabschnitt: DFHSRT TYPE=INITIAL

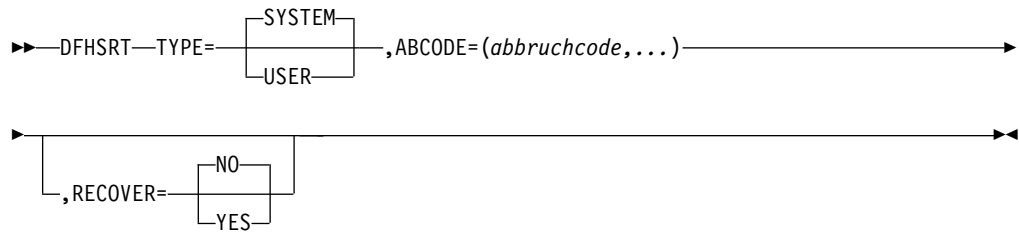
Mit dem Makro DFHSRT TYPE=INITIAL kann der Programmabschnitt der Systemwiederherstellungstabelle (SRT = System Recovery Table) generiert werden.

►► DFHSRT TYPE=INITIAL [ ,SUFFIX=xx ] ◀◀

Allgemeine Informationen zu den TYPE=INITIAL-Makros sowie zur Verwendung des Operanden SUFFIX finden Sie in TYPE=INITIAL (control section).

### Abbruchcodes: DFHSRT TYPE=SYSTEM|USER

Die Makros DFHSRT TYPE=SYSTEM und TYPE=USER geben System- und Benutzerabbruchcodes an, die von CICS abgefangen werden sollen.



#### **TYPE={SYSTEM|USER}**

Gibt den Typ des Abbruchcodes an, der abgefangen werden soll.

##### **SYSTEM**

Der Abbruchcode ist ein Abbruchcode des Betriebssystems, der einem MVS-Abbruchcode vom Typ *Sxxx* entspricht.

##### **USER**

Beim Abbruchcode handelt es sich um einen Benutzerabbruchcode (einschließlich CICS), der einem MVS-Abbruchcode vom Typ *Unnnn* entspricht.

#### **ABCODE=(abbruchcode,...)**

Geben Sie diesen Operanden mit dem Abbruchcode an, der abgefangen werden soll. Wenn Sie einen einzelnen Abbruchcode angeben, dann sind keine Klammern erforderlich.

Wenn Sie TYPE=SYSTEM angeben, dann muss dieser Abbruchcode aus drei hexadezimalen Ziffern (*xxx*) bestehen, die für den Abbruchcode vom Typ *Sxxx* des MVS-Systems stehen.

Wenn Sie TYPE=USER angeben, dann muss dieser Abbruchcode aus einer Dezimalzahl (*nnnn*) bestehen, die für den Benutzerteil des MVS-Abbruchcodes vom Typ *Unnnn* steht. Hierbei handelt es sich normalerweise um dieselbe Zahl, die auch bei der CICS-Nachricht verwendet wird, die ausgegeben wird, bevor CICS versucht, eine abnormale Beendigung durchzuführen.

#### **RECOVER={YES|NO}**

Dieser Operand gibt an, ob Codes zu der Systemwiederherstellungstabelle (SRT = System Recovery Table) hinzugefügt oder aus ihr entfernt werden sollen.

##### **YES**

Geben Sie diesen Operanden an, um die angegebenen Codes zur Systemwiederherstellungstabelle hinzuzufügen.

##### **NO**

Geben Sie diesen Operanden an, um die angegebenen Codes aus der Systemwiederherstellungstabelle zu entfernen.

#### **Anmerkung:**

1. CICS fängt die folgenden Abbruchcodes automatisch ab und führt eine Wiederherstellung durch:

```

001,002,013,020,025,026,030,032,033,034,035,
036,037,03A,03B,03D,052,053,067,0D3,0D4,0D5,
0D6,0D7,0D8,0E0,0F3,100,113,137,202,213,214,
237,283,285,313,314,337,400,413,437,513,514,
613,614,637,713,714,737,813,837,913,A13,A14,
B13,B14,B37,D23,D37,E37

```

Der Abbruchcode 0F3 bezieht sich auf verschiedene Maschinenfehlerbedingungen. Außerdem bezieht er sich auf die APR-Bedingung (APR = Alternate Pro-

cessor Retry)), die nur dann auftreten kann, wenn die Verarbeitung auf einem Multiprozessor durchgeführt wird. Mit dem von CICS bereitgestellten Wiederherstellungscode wird eine Wiederherstellung nach Maschinenstörungen aufgrund von Anweisungsfehlern versucht, wobei davon ausgegangen wird, dass es sich nicht um permanente Störungen handelt. Außerdem wird versucht, die Wiederherstellung nach APR-Bedingungen durchzuführen.

CICS versucht eine Wiederherstellung nach Ausgabe dieser standardmäßigen Abbruchcodes, wenn Sie die Systemwiederherstellungstabelle (SRT) wie folgt angeben:

```
DFHSRT TYPE=INITIAL
DFHSRT TYPE=FINAL
END
```

Die Standardcodes müssen nicht einzeln angegeben werden.

2. Wenn CICS andere Fehler behandeln soll, können Sie die Systemwiederherstellungstabelle wie folgt codieren:

```
DFHSRT TYPE=INITIAL
DFHSRT TYPE=SYSTEM,or USER,
        ABCODE=(user or system codes),
        RECOVER=YES
DFHSRT TYPE=FINAL
END
```

3. Wenn CICS nicht versuchen soll, nach Auftreten eines der standardmäßigen Abbruchcodes eine Wiederherstellung durchzuführen, dann müssen Sie die Codes mit **RECOVER=NO** oder ohne den Parameter **RECOVER** angeben.
4. CICS versucht, die Wiederherstellung durchzuführen, wenn eine Benutzertask (jedoch keine Systemtask) zum Zeitpunkt des Auftretens der abnormalen Beendigung die Steuerung hat.

## DFHSRT-Beispiel

Die zum Generieren einer SRT (System Recovery Table; Systemwiederherstellungstabelle) erforderliche Codierung kann anhand eines Beispiels verdeutlicht werden.

In Abb. 14 ist die Codierung zum Generieren einer SRT dargestellt.

DFHSRT TYPE=INITIAL,	*
SUFFIX=K1	
DFHSRT TYPE=SYSTEM,	*
ABCODE=777,	*
RECOVER=YES	
DFHSRT TYPE=USER,	
ABCODE=(888,999),	*
RECOVER=YES	
DFHSRT TYPE=USER,	*
ABCODE=020	
DFHSRT TYPE=FINAL	
END	

Abbildung 14. Systemwiederherstellungstabelle: Beispiel

## TCT: Terminalsteuertabelle

Verwenden Sie das Makro DFHTCT zum Definieren logischer SNA-Einheiten (LUs), die die Verwendung von Codes für logische Einheiten (LDCs = Logical Device Codes) sowie von sequenziellen, über BSAM angeschlossenen Einheiten unterstützen.

Ein CICS-System kann mit Terminals, sequenziellen Einheiten, logischen Einheiten und anderen Systemen kommunizieren. Die Terminalsteuertabelle (TCT = Terminal Control Table) definiert jede der Einheiten in der Konfiguration. Jeder TCT-Eintrag definiert die optionalen und variablen Funktionen der Einheit für CICS und gibt die optionalen und variablen Funktionen von CICS an, die verwendet werden sollen.

CICS verwendet zur Kommunikation mit Terminals eine Zugriffsmethode für die Telekommunikation (Telecommunications Access Method). Hierbei kann es sich um z/OS Communications Server oder (für sequenzielle Einheiten) BSAM handeln. Sie können eine dieser Komponenten oder beide auf Ihrem System verwenden.

Ein Terminal kann eine Telekommunikationseinheit (z. B. IBM 3279 Color Display Station) oder ein Subsystem (z. B. IBM 4700 Finance Communication System) sein. Terminals können sich an einem lokalen Standort (über Kanal angeschlossen) oder einem fernen Standort (über Verbindung angeschlossen) befinden.

Sie können eine sequenzielle Einheit verwenden, um ein CICS-Terminal zu simulieren. Sie können einen Kartenleser oder Stanzer, Zeilendrucker, eine DASD-Einheit (Plattenlaufwerk) oder ein Magnetbandlaufwerk als sequenzielle Einheit einsetzen.

Eine logische Einheit (LU = Logical Unit) ist ein Port, über den ein Benutzer eines SNA-Netzes Zugriff auf die Netzfunktionen erhält.

Ein System kann z. B. ein weiteres CICS-System, IBM 8815 Scanmaster, IBM Displaywriter oder eine APPC/PC-Einheit sein. Die übergreifende Kommunikation mit CICS-Systemen kann zwischen den folgenden Einheiten stattfinden:

- Zwischen unterschiedlichen Prozessoren (**Intersystem Communication** oder **ISC**) mit dem Protokoll LUTYPE 6.1 oder LUTYPE 6.2 oder mithilfe eines Transitsystems als **indirekter Verbindung**. (Die übergreifende Kommunikation mit Nicht-CICS-Systemen verwendet ebenfalls ISC.)
- Innerhalb desselben Prozessors (**Mehrregionenbetrieb** oder **MRO**) mithilfe der regionsübergreifenden Kommunikation (IRC = Interregion Communication). (Sie können LUTYPE 6.2 ISC auch innerhalb desselben Prozessors verwenden.)

## DFHTCT-Makrotypen

Die von Ihnen angegebenen DFHTCT-Makros hängen von der Einheit, die Sie definieren, und von der verwendeten Zugriffsmethode ab.

Sie beginnen immer mit einem der folgenden Makros:

DFHTCT TYPE=INITIAL,... (Siehe hierzu „Programmabschnitt: DFHTCT TYPE=INITIAL“ auf Seite 453.)

Zum Assemblieren der Terminalsteuertabelle (TCT = Terminal Control Table) für die Migration von für RDO auswählbaren Definitionen in die CSD-Datei steht ein spezielles Makro zur Verfügung:

DFHTCT TYPE=GROUP,... (Siehe hierzu „TCT-Definitionen migrieren: DFHTCT TYPE=GROUP“ auf Seite 456.)

Sie können Ihre Einheiten in jeder gewünschten Reihenfolge definieren. Jedes Gerät benötigt mindestens ein Makro, wobei bei mehreren Makros in bestimmten Fällen eine bestimmte Reihenfolge vorgegeben ist. Sie erhalten entsprechende Anweisun-

gen, wenn dies der Fall ist. In der folgenden Tabelle werden die Makros aufgeführt, die Sie für jeden Einheiten- oder Systemtyp benötigen.

*Tabelle 14. DFHTCT-Makrotypen*

Einheit oder System	DFHTCT-Makros
Codes für logische Einheiten.	DFHTCT TYPE=LDC,... DFHTCT TYPE=LDCLIST,...
Sequenzielle Einheiten.	DFHTCT TYPE=SDSCI,... DFHTCT TYPE=LINE,... DFHTCT TYPE=TERMINAL,...
Ferne Nicht-SNA-LUs, für Transaktionsrouting.	DFHTCT TYPE=REMOTE,... oder: DFHTCT TYPE=REGION,... DFHTCT TYPE=TERMINAL,...

Am Ende Ihrer Makros müssen Sie Folgendes angeben:

```
DFHTCT TYPE=FINAL
END
```

Dieses Makro wird in TYPE=FINAL (end of table) beschrieben.

#### Hinweise:

#### SYSIDNT- und TRMIDNT-Operanden

CICS akzeptiert sowohl Groß- als auch Kleinbuchstaben für SYSIDNT und TRMIDNT, die Zeichen in Kleinschreibung werden jedoch nicht auf Duplikate hin überprüft. Die Assemblierung einer Terminalsteuertabelle, die SYSIDNT- oder TRMIDNT-Angaben in Kleinschreibung enthält, führt zur Ausgabe eines Makro-Hinweises (MNOTE). Wenn Sie das System auf Duplikate überprüfen wollen, dann verwenden Sie nur Großbuchstaben für SYSIDNT und TRMIDNT.

#### TCT assemblieren

Die Assemblierung und die Herstellung einer Programmverbindung für eine Terminalsteuertabelle führt zur Erstellung von zwei separaten Lademodulen. Die Assemblierung einer Terminalsteuertabelle mit Suffix (Quellenname DFHTCTxx) führt zur Generierung einer einzelnen Textdatei. Allerdings werden zwei Member erstellt, wenn die Herstellung der Programmverbindung zu einer Ladebibliothek durchgeführt wird:

1. DFHTCTxx: Dieses Member enthält die nicht für RDO auswählbaren Definitionen im Steuerblockformat.
2. DFHRDxx: Dieses Member enthält die für RDO auswählbaren Definitionen (SNA-LU und System) im RDO-Befehlsformat.

Diese Operation wird ausgeführt, und zwar **unabhängig davon, ob Sie RDO verwenden wollen**. Sie müssen die Existenz dieser beiden Tabellen berücksichtigen, wenn Sie assemblierte Terminalsteuertabellen zwischen Ladebibliotheken kopieren oder verschieben wollen.

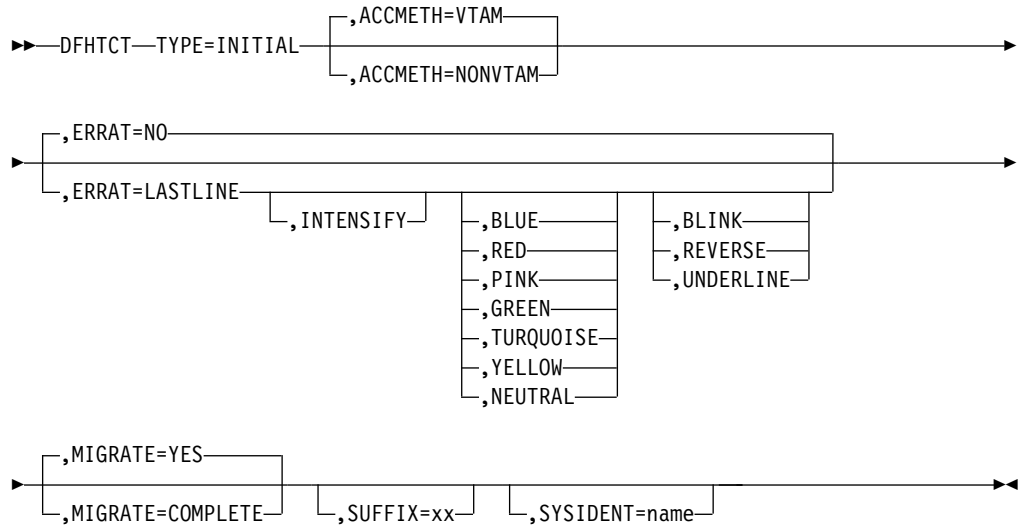
Wenn Sie die Terminalsteuertabelle erneut assemblieren, nachdem Sie CICS gestartet haben, dann werden alle Änderungen bei einem Warm- oder Notfallstart übernommen.

#### Programmabschnitt: DFHTCT TYPE=INITIAL

Sie können ein Makro DFHTCT TYPE=INITIAL vor allen anderen Makros angeben, mit denen Ihre Ressourcen definiert werden.

Das Makro TYPE=INITIAL dient zur Ausführung der folgenden beiden Aktionen:

1. Einrichten des Speicherbereichs, in dem die Terminalsteuertabelle (TCT = Terminal Control Table) assembliert wird.
2. Angeben der Informationen, die für die gesamte Terminalsteuertabelle oder für die einzelnen Nicht-z/OS Communications Server-Einträge (sowie z/OS Communications Server-LDC-Definitionen) gelten.



**Anmerkung:** SYSIDENT dient ausschließlich für die übergreifende Kommunikation.

**Anmerkung:** Allgemeine Informationen zu den TYPE=INITIAL-Makros finden Sie in TYPE=INITIAL (control section).

#### ACCMETH=( [VTAM,] NONVTAM )

Mit diesem Operanden werden die Zugriffsmethoden angegeben, die auf dem aktiven CICS-System benötigt werden. VTAM wird jetzt als z/OS Communications Server bezeichnet.

##### VTAM

Geben Sie diese Option an, wenn Sie z/OS Communications Server als Zugriffsmethode verwenden wollen.

##### NONVTAM

Geben Sie diese Option an, wenn Sie andere Telekommunikationszugriffsmethoden als z/OS Communications Server verwenden wollen. Zu diesen Zugriffsmethoden gehört BSAM (für sequenzielle Einheiten).

**Anmerkung:** Der Standardwert ist die Annahme, dass VTAM und NONVTAM beide verwendet werden.

#### ERRATT={NO v ([LASTLINE] [, INTENSIFY] [, {BLUE v RED v PINK v GREEN v TURQUOISE v YELLOW v NEUTRAL}] [, {BLINK v REVERSE v UNDERLINE}]) }

Gibt an, wie Fehlernachrichten auf allen 3270-Bildschirmeinheiten angezeigt werden. Sie können entweder den Standardwert NO beibehalten oder eine beliebige Kombination von LASTLINE, INTENSIFY, einer der Farben und einer der Hervorhebungen verwenden. Bei Angabe von INTENSIFY, einer der Farben oder einer der Hervorhebungen wird die Verwendung von LASTLINE erzwungen.



Attribute, die für eine bestimmte Einheit nicht gültig sind, werden ignoriert.

**NO** Alle Fehlnachrichten werden an der aktuellen Cursorposition und ohne zusätzliche Attribute angezeigt.

**LASTLINE**

Alle Fehlnachrichten werden beginnend am Anfang der untersten Zeile des Bildschirms so angezeigt, dass die gesamte Nachricht in die Anzeige passt.

**Achtung:** Wenn Nachrichten in schneller Folge empfangen werden, dann können sie sich überlagern. Die zuerst ausgegebenen Nachrichten werden dann möglicherweise aus dem Anzeigebereich verschoben, bevor der Bediener sie lesen konnte.

**INTENSIFY**

Die Fehlnachrichten werden in intensiver Hervorhebung und in der letzten Zeile der Anzeige dargestellt.

**BLUE∨RED∨PINK∨GREEN∨TURQUOISE∨YELLOW∨ NEUTRAL**

Die Fehlnachrichten werden in der angegebenen Farbe und in der letzten Zeile der Anzeige dargestellt.

**BLINK∨REVERSE∨UNDERLINE**

Die Fehlnachrichten werden hervorgehoben und in der letzten Zeile der Anzeige dargestellt.

**MIGRATE={YES∨COMPLETE}**

Dieser Operand steuert die Erstellung von TCT-Einträgen für z/OS Communications Server-Einheiten, die für die Onlineresourcendefinition (RDO = Resource Definition Online) **auswählbar** sind. Die einzige Möglichkeit, mit der für RDO auswählbare Ressourcen von der Makroquelle in die CSD-Datei verschoben werden können, besteht in der Verwendung von DFHCSDUP (siehe hierzu die Beschreibung unter YES).

**YES**

YES gibt an, dass Sie die erforderlichen Daten für die Migration Ihrer für RDO auswählbaren Ressourcen generieren wollen. Die Datensätze, die anhand der Makroquelle generiert werden, sind für die Verwendung als Eingabe für das Dienstprogramm DFHCSDUP vorgesehen. Für jede für RDO auswählbare Ressource wird eine MNOTE-Warnung abgesetzt.

**COMPLETE**

Die Verwendung von COMPLETE hat zur Folge, dass TCT-Einträge für die für RDO auswählbaren Einheiten nicht anhand der Makroquelle generiert werden. Für jede Einheit wird bei der Assemblierung ein Makro-Hinweis (MNOTE) generiert. Dies bedeutet, dass der TCT-Makroquellencode nach der Migration Ihrer Definitionen beibehalten werden kann.

Wenn Sie die Assemblierung einer Terminalsteuertabelle für Ressourcen fortsetzen wollen, die nicht für RDO auswählbar sind, dann fahren Sie mit der Verwendung von MIGRATE=COMPLETE fort.

**SYSIDNT={CICS∨name}**

Dieser aus einem bis vier Zeichen bestehende Name stellt einen privaten Namen dar, den das CICS-System verwendet, um sich selbst zu identifizieren. Wenn Sie DFHTCT TYPE=REGION-Makros verwenden, um ferne Terminals zu definieren, dann müssen Sie diesen Operanden angeben. Er wird verwendet, um festzustellen, ob ein ferner oder lokaler TCT-Terminal eintrag von jedem TYPE=TERMINAL-Makro generiert wird, das auf das Makro TYPE=REGION folgt. Wenn der Wert für SYSIDNT im Makro TYPE=REGION mit dem Wert

für SYSIDNT im Makro TYPE=INITIAL identisch ist, dann wird eine lokale Definition generiert. Andernfalls wird eine ferne Definition generiert.

Der Wert, den Sie für diesen Operanden angeben, wird ausschließlich verwendet, um den Namen der lokalen Region während der Assemblierung der Terminalsteuertabelle zu definieren. Sie müssen den Namen der lokalen Region zu Ausführungszwecken außerdem mit dem Systeminitialisierungsparameter SYSIDNT definieren.

### TCT-Definitionen migrieren: DFHTCT TYPE=GROUP

Mit diesem Makro können Sie Gruppen benennen, in die TCT-Definitionen eingefügt werden sollen, wenn Sie auf die Onlinere sourcedefinition (RDO = Resource Definition Online) umstellen.

Dieses Makro kann so oft wie erforderlich und an einem beliebigen Punkt in der Makroquelle angegeben werden. Bei jedem Vorkommen des Makros definiert es eine CSD-Dateigruppe, in die nachfolgende Definitionen eingefügt werden, bis das nächste Makro DFHTCT TYPE=GROUP auftritt.

►►DFHTCT—TYPE=GROUP—┐  
└, GROUP=name┘

#### GROUP=name

Geben Sie diesen Operanden mit dem Namen der Gruppe an, auf die nachfolgende Definitionen migriert werden sollen. Der Name kann maximal acht alphanumerische Zeichen umfassen, darf jedoch nicht mit der Zeichenfolge DFH beginnen. Der Standardname lautet TCTxx. Dabei steht xx für den Wert, der im Makro DFHTCT TYPE=INITIAL für SUFFIX angegeben wird. Wird ein Fehler festgestellt, dann wird der vorhandene Gruppenname weiterhin verwendet.

Wenn eine Gruppe mit dem von Ihnen angegebenen Namen nicht bereits vorhanden ist, dann wird sie erstellt. Andernfalls werden nachfolgende Definitionen zu dieser Gruppe hinzugefügt.

### DFHTCT-Codes für logische Einheiten: z/OS Communications Server (nicht 3270)

Bestimmte Typen logischer Einheiten können verwendet werden, um Zugriff auf mehrere Ressourcen innerhalb eines Subsystems zu erhalten. Eine Kartenstanzeinheit kann beispielsweise an eine logische 3770-Einheit angeschlossen sein. Das CICS-Anwendungsprogramm kann die Stanzer Ausgabe über BMS (Basic Mapping Support) und die 3770-Einheit an die Kartenstanzeinheit weiterleiten. Die von CICS bereitgestellte Komponente zur Erteilung der Berechtigung für die Kommunikation mit Einheiten in logischen Einheiten dieses Typs wird als *Code für logische Einheiten* (LDC = Logical Device Code) bezeichnet.

Bestimmte Typen logischer Einheiten können verwendet werden, um Zugriff auf mehrere Ressourcen innerhalb eines Subsystems zu erhalten. Eine Kartenstanzeinheit kann beispielsweise an eine logische 3770-Einheit angeschlossen sein. Das CICS-Anwendungsprogramm kann die Stanzer Ausgabe über BMS (Basic Mapping Support) und die 3770-Einheit an die Kartenstanzeinheit weiterleiten. Die von CICS bereitgestellte Komponente zur Erteilung der Berechtigung für die Kommunikation mit Einheiten in logischen Einheiten dieses Typs wird als **Code für logische Einheiten** (LDC = Logical Device Code) bezeichnet.

Obwohl es sich um z/OS Communications Server-Einheiten handelt, werden hier anders als bei anderen z/OS Communications Server-Einheiten Makrodefinitionen benötigt.

Folgende logischen Einheiten unterstützen LDCs:

- Logische 3601-Einheit
- Logische 3770-Stapelverarbeitungseinheit
- Logische 3770-Stapeldatenaustauscheinheit
- Logische 3790-Stapeldatenaustauscheinheit
- Logische LUTYPE4-Einheit

Um auf eine solche Einheit in einem CICS-Anwendungsprogramm oder in der CMSG-Transaktion für die Nachrichtenverteilung zu verweisen, müssen Sie ein mnemonisches LDC-Zeichen angeben, das von CICS in einen numerischen LDC-Wert umgesetzt wird. Wenn CICS einen Ausgabedatenstrom an eine logische Einheit sendet, dann umfasst dieser auch den LDC-Wert im Funktionsverwaltungsheader (FMH = Function Management Header). Wenn die logische Einheit den Datenstrom empfängt, dann verwendet sie den LDC-Wert, um zu ermitteln, welche Komponente die Ausgabe empfangen soll, oder um bestimmte Standardaktionen auszuführen.

Jedes mnemonische LDC-Zeichen, auf das verwiesen werden soll, muss in der Terminalsteuertabelle (TCT = Terminal Control Table) definiert sein. Hierbei können optional der zugehörige LDC-Wert und bestimmte Einheitenmerkmale zur Verwendung durch die BMS-Funktionen definiert werden. Diese LDC-Informationen sind entweder in der **System-LDC-Tabelle** oder in einer **erweiterten lokalen LDC-Liste** enthalten. Sie geben die folgenden DFHTCT-Makros an, um die System-LDC-Tabelle oder eine erweiterte lokale LDC-Liste anzugeben:

- Geben Sie die DFHTCT TYPE=LDC-Makros an, um Einträge in der System-LDC-Tabelle zu generieren. Sie können bestimmte LDC-Standardinträge generieren, die von CICS bereitgestellt werden. Beispiel:

```
DFHTCT TYPE=LDC,LDC=SYSTEM
```

generiert die folgenden Einträge in der System-LDC-Tabelle.

*Tabelle 15. Einträge der System-LDC-Tabelle*

Mnemonisches LDC-Zeichen	LDC-Wert	Einheit	Seitengröße (zeile,spalte)
DS	1	3604-Tastaturanzeige	6,40
JP	2	3610-Dokumentdrucker	1,80
PB	3	Sparbuch- und Dokumentdrucker	1,40
LP	4	3618-Zeilendrucker für Verwaltungszwecke	50,80
MS	5	3604-Magnetstreifencodierer	1,40
CO	0	Konsolenmedium oder standardmäßige Druckdateigruppe	
R1	32	Anweisungseingabemedium	1,80
H1	32	Anweisungsausgabemedium	1,80
P1	48	Druckmedium oder Druckdateigruppe	50,80
W1	128	Textverarbeitungsmedium 1.	50,80
W2	144	Textverarbeitungsmedium 2.	50,80
W3	160	Textverarbeitungsmedium 3.	50,80
W4	192	Textverarbeitungsmedium 4.	50,80

Sie können LDCs auch speziell definieren, um LDC-Einträge zur System-LDC-Tabelle hinzuzufügen. Beispiel:

```
DFHTCT TYPE=LDC,
      LDC=XX,
      DVC=BLUPRT,
      PGESIZE=(12,80),
      PGESTAT=PAGE
DFHTCT TYPE=LDC,
      LDC=YY,
      DVC=BLUPCH,
      PGESIZE=(1,80),
      PGESTAT=AUTOPAGE
```

Diese Angabe dient zum Hinzufügen der folgenden Einträge zur System-LDC-Tabelle.

Tabelle 16. Mit LDCs definierte Einträge der System-LDC-Tabelle

Mnemonisches LDC-Zeichen	LDC-Wert	Einheit	Seitengröße (zeile,spalte)	PGESTAT
XX	48	LU-Stapelverarbeitungsdrucker	12,80	PAGE
YY	32	Kartenausgabe einer logischen Stapelverarbeitungseinheit	1,80	AUTOPAGE

- Anstelle der System-LDC-Tabelle können Sie die folgende Reihe von DFHTCT TYPE=LDC-Makros angeben, um eine erweiterte lokale LDC-Liste zu erstellen. Außerdem können Sie Standardeinträge generieren. Beispiel:

```
LDC1 DFHTCT TYPE=LDC,LOCAL=INITIAL
* the next line generates default CO,R1,H1,P1 LDCs
DFHTCT TYPE=LDC,LDC=BCHLU
DFHTCT TYPE=LDC,DVC=BLUPRT,LDC=AA,
      PGESIZE=(6,30)
DFHTCT TYPE=LDC,DVC=BLUPCH,LDC=BB,
      PGESIZE=(1,80)
DFHTCT TYPE=LDC,DVC=BLUON,LDC=CC,
      PGESIZE=(1,132)
DFHTCT TYPE=LDC,LOCAL=FINAL
```

Diese Angabe dient zum Generieren einer erweiterten lokalen LDC-Liste mit dem Namen LDC1, die die folgenden Einträge enthält:

Tabelle 17. Einträge der erweiterten lokalen LDC-Liste

Mnemonisches LDC-Zeichen	LDC-Wert	Einheit	Seitengröße (zeile,spalte)
CO	0	Konsolenmedium oder standardmäßige Druckerdateigruppe	
R1	32	Anweisungseingabemedium	1,80
H1	32	Anweisungsausgabemedium	1,80
P1	48	Druckmedium oder Druckdateigruppe	50,80
AA	48	BLUPRT - LU-Stapelverarbeitungsdrucker	6,30

Tabelle 17. Einträge der erweiterten lokalen LDC-Liste (Forts.)

Mnemonisches LDC-Zeichen	LDC-Wert	Einheit	Seitengröße (zeile,spalte)
BB	32	BLUPCH - Kartenausgabe einer logischen Stapelverarbeitungseinheit	1,80
CC	0	BLUCON - Konsoldrucker für logische Stapelverarbeitungseinheit	1,132

Wenn Sie eine logische Einheit in der Terminalsteuertabelle (TCT = Logical Device Codes) definieren, dann können Sie die zugehörigen LDCs auf die beiden folgenden Arten angeben:

1. Geben Sie ein Makro DFHTCT TYPE=LDCLIST an, um eine lokale Liste mit mnemonischen LDC-Zeichen (und optional der zugehörigen LDC-Werte) zu definieren. Beispiel:

```
LDC2 DFHTCT TYPE=LDCLIST,
      LDC=(DS,JP,PB=5,LP,MS)
```

In dem Makro DFHTCT TYPE=TERMINAL, das zum Definieren der logischen Einheit verwendet wird, geben Sie im Operanden LDC den Namen der lokalen Liste an, der im Makro DFHTCT TYPE=LDCLIST definiert wurde. Beispiel:

```
DFHTCT TYPE=TERMINAL,
      TRMTYPE=3600,
      LDC=LDC2, ...
```

Diese Angabe dient zur Zuordnung der LDCs DS, JP, PB, LP und MS zur logischen 3601-Einheit, die von Ihnen definiert wird. Die LDC-Werte werden entweder in der lokalen Liste angegeben oder aus der System-LDC-Tabelle abgerufen. Wenn BMS diese mnemonischen LDC-Zeichen verwendet, dann müssen auch die Seitengröße und der Seitenstatus aus der System-LDC-Tabelle verfügbar sein.

**Anmerkung:** Eine lokale Liste, die mit dem Makro DFHTCT TYPE=LDCLIST definiert wird, kann von einer Reihe von logischen 3601-Einheiten, logischen LUTYPE 4-Einheiten und logischen Stapelverarbeitungseinheiten gemeinsam genutzt werden.

2. In dem Makro DFHTCT TYPE=TERMINAL, das zum Definieren der logischen Einheit verwendet wird, geben Sie im Operanden LDC den Namen einer erweiterten lokalen LDC-Liste an. Beispiel:

```
LDC1 DFHTCT TYPE=LDC,LOCAL=INITIAL
      DFHTCT TYPE=LDC,LDC=BCHLU
      DFHTCT TYPE=LDC,DVC=BLUPRT,LDC=AA,
      PGSIZE=(6,30)
      DFHTCT TYPE=LDC,DVC=BLUPCH,LDC=BB,
      PGSIZE=(1,80)
      DFHTCT TYPE=LDC,DVC=BLUCON,LDC=CC,
      PGSIZE=(1,132)
      DFHTCT TYPE=LDC,LOCAL=FINAL
      DFHTCT TYPE=TERMINAL,TRMTYPE=BCHLU,
      LDC=LDC1, ...
```

Diese Angabe dient zur Zuordnung der LDCs CO, R1, H1, P1, AA, BB und CC zur logischen Stapelverarbeitungseinheit, die von Ihnen definiert wird. Die zugehörigen LDC-Werte und Einheitenmerkmale für BMS-Funktionen werden in der erweiterten lokalen LDC-Liste mit dem Namen LDC1 beschrieben.

Wenn CICS eine Ausgabe- oder Nachrichtenverteilungsoperation mit einem bestimmten mnemonischen LDC-Zeichen für eine logische Einheit anfordert, dann

wird versucht, das mnemonische Zeichen anhand der Liste (in beliebiger Form) aufzulösen, die im Operanden LDC des Makros DFHTCT TYPE=TERMINAL angegeben ist. Wenn der LDC sich nicht in der lokalen Liste oder in der erweiterten lokalen Liste befindet, dann ist der angegebene LDC für diesen Terminaleintrag nicht gültig. In diesem Fall wird X'00' in den Abschnitt mit dem Code für logische Einheiten des FMH (Function Management Header; Funktionsverwaltungsheader) eingefügt und es wird kein Bestimmungsname eingefügt.

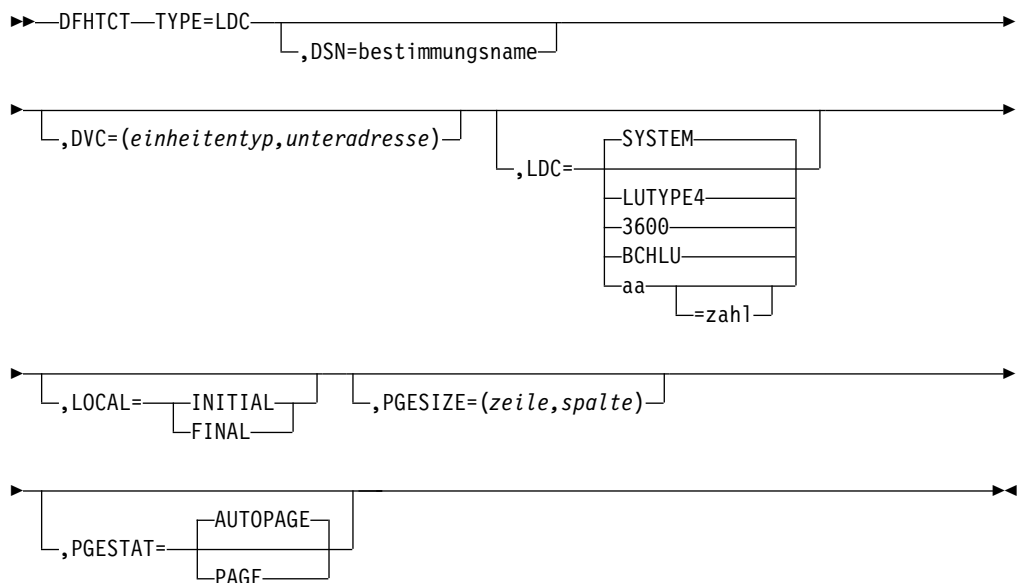
Wenn eine BMS-Funktion für einen LDC angefordert wird und wenn das mnemonische LDC-Zeichen erfolgreich aufgelöst werden konnte, dann wird auf die Einheitenmerkmale (z. B. Einheitenname und Bestimmungsname) für die BMS-Funktion zugegriffen. Wenn sich der LDC in einer erweiterten lokalen LDC-Liste befindet, dann sind diese Merkmale im Eintrag der ermittelten erweiterten lokalen Liste gespeichert. Andernfalls wird die System-LDC-Tabelle nach dem LDC und den zugehörigen Einheitenmerkmalen durchsucht.

### Codes für logischen Einheiten - Makro DFHTCT TYPE=LDC

Das Makro DFHTCT TYPE=LDC kann nur mit logischen 3600-, LUTYPE4- oder 3770-Stapelverarbeitungseinheiten sowie mit logischen 3770- und 3790-Stapeldatenaustauscheinheiten verwendet werden.

Sie sind für die Einrichtung der LDC-Struktur verantwortlich, die für das Terminal verwendet werden soll.

Die Erweiterung dieses Makros ist unabhängig von der Position, an der es innerhalb der TCT-Definition angegeben wird, gleich.



#### name

Geben Sie diesen Operanden mit dem Namen der erweiterten lokalen LDC-Liste an. Er muss mit dem Namen übereinstimmen, der im Operanden LDC des Makros DFHTCT TYPE=TERMINAL angegeben wurde und ist nur erforderlich, wenn LOCAL=INITIAL angegeben wird.

#### TYPE=LDC

Geben Sie diesen Operanden an, wenn ein LDC für die System-LDC-Tabelle oder die erweiterte lokale LDC-Liste definiert wird.

**DSN=bestimmungsname**

Geben Sie diesen Operanden mit dem Namen an, der von BMS für die Zielauswahl bei einer logischen Stapeldatenaustauscheinheit verwendet werden soll. Weiterführende Informationen zur Zielauswahl finden Sie in den jeweiligen CICS-Subsystemhandbüchern.

**DVC=(einheitentyp,unteradresse)**

Geben Sie diesen Operanden mit dem Einheitentyp an, der dem LDC zugeordnet ist, der für eine BMS-Anforderung verwendet werden soll. Dieser Operand kann nur zusammen mit dem Operanden LDC=aa[=nnn] angegeben werden.

**einheitentyp**

Dieser Wert kann wie folgt angegeben werden:

*Tabelle 18. Einträge für DVC=einheitentyp*

Einheitentyp	Erläuterung
3604	Tastaturanzeige
3610	Dokumentdrucker für Einzelformulare oder Journaldrucker (einschließlich Dokument-/Journaldrucker für eine 3612-Einheit)
3612	Sparbuchkomponente für 3612-Einheit
3618	Momentan ausgewählte Transportvorrichtung
3618P	Primäre Transportvorrichtung
3618S	Sekundäre Transportvorrichtung
3618B	Beide Transportvorrichtungen
BLUCON	Konsoldrucker für logische Stapelverarbeitungseinheit
BLUPRT	Druckerkomponente für logische Stapelverarbeitungseinheit
BLURDR	Anweisungseingabekomponente für logische Stapelverarbeitungseinheit
BLUPCH	Anweisungsausgabekomponente für logische Stapelverarbeitungseinheit
WPMED1	Textverarbeitungsmedium 1
WPMED2	Textverarbeitungsmedium 2
WPMED3	Textverarbeitungsmedium 3
WPMED4	Textverarbeitungsmedium 4

Die Einheitentypen BLUPRT, BLURDR, BLUPCH und BLUCON werden an eine logische Stapelverarbeitungseinheit, logische Stapeldatenaustauscheinheit oder eine logische LUTYPE4-Einheit angeschlossen.

Die Optionen WPMED1, WPMED2, WPMED3 und WPMED4 gelten nur für logische LUTYPE4-Einheiten. Die Komponente, für die diese Optionen gelten, wird mit einer speziellen Implementierung für logischen Einheiten des Typs 4 definiert.

**unteradresse**

Geben Sie diesen Operanden mit der Unteradresse der Medien an. Der Bereich liegt zwischen 0 und 15, der Standardwert lautet 0. Der Wert 15 gibt eine beliebige Unteradresse an. Die Unteradresse unterscheidet zwischen zwei Einheiten desselben Einheitentyps (z. B. (BLUPRT,0) und (BLUPRT,1)), bei denen es sich um zwei Druckkomponenten handeln kann, die an eine logische Einheit angeschlossen sind.

**LDC={SYSTEMvLUTYPE4v3600vBCHLUv(aa[=nnn])}**

Geben Sie diesen Operanden mit dem mnemonischen und numerischen Wert für LDC an. Nur die Option LDC=aa[=nnn] kann zusammen mit den Operanden DVC, PGESIZE und PGESTAT verwendet werden.

## SYSTEM

Die folgenden Systemstandardwerte für LDCs für 3600-Einheiten, Stapelverarbeitungseinheiten und logische LUTYPE4-Einheiten werden eingerichtet:

Tabelle 19. Standardmäßige LDCs des Systems

Mnemonisches LDC-Zeichen	LDC-Wert	Einheit	Seitengröße (Zeile, Spalte)
DS	1	3604-Tastaturanzeige	6,40
JP	2	3610-Dokumentdrucker	1,80
PB	3	Sparbuch- und Dokumentdrucker	1,40
LP	4	3618-Zeilendrucker für Verwaltungszwecke	50,80
MS	5	3604-Magnetstreifencodierer	1,40
CO	0	Konsolenmedium oder standardmäßige Druckdateigruppe	
R1	32	Anweisungseingabemedium	1,80
H1	32	Anweisungsausgabemedium	1,80
P1	48	Druckmedium oder Druckdateigruppe	50,80
W1	128	Textverarbeitungsmedium 1.	50,80
W2	144	Textverarbeitungsmedium 2.	50,80
W3	160	Textverarbeitungsmedium 3.	50,80
W4	192	Textverarbeitungsmedium 4.	50,80

## LUTYPE4

Für eine logische LUTYPE4-Einheit (Textverarbeitung) müssen mnemonische LDC-Systemstandardeinstellungen eingerichtet werden. Diese bestehen aus den mnemonischen Zeichen CO, R1, P1, H1, W1, W2, W3 und W4, den entsprechenden LDC-Werten und den entsprechenden Seitengrößen.

## 3600

Mnemonische LDC-Systemstandardeinstellungen für die 3600-Einheit müssen eingerichtet werden. Diese bestehen aus den mnemonischen Zeichen DS, JP, PB, LP und MS, den entsprechenden LDC-Werten und den entsprechenden Angaben für die Seitengröße und den Seitenstatus.

## BCHLU

Es müssen Systemstandardwerte für mnemonische LDC-Zeichen für eine logische Stapelverarbeitungseinheit eingerichtet werden. Diese bestehen aus den mnemonischen Zeichen CO, R1, P1 und H1, den entsprechenden LDC-Werten und den entsprechenden Angaben für die Seitengröße und den Seitenstatus.

**aa** Das zweistellige mnemonische Zeichen, das für diesen LDC verwendet werden soll.

## nnn

Der numerische Wert, der dem LDC in der LDC-Liste des Systems oder in der erweiterten lokalen LDC-Liste zugeordnet werden soll. Der Wert in der Systemliste wird als Standardwert für diesen LDC verwendet, wenn in einer lokalen LDC-Liste (d. h. der erweiterten Liste), die einem TCTTE zugeordnet ist, kein Wert gefunden wird. Für eine 3600-Einheit muss ein Wert angegeben werden. Für eine logische Stapelverarbeitungseinheit, eine logische Stapeldatenaustauscheinheit oder eine



logische LUTYPE4-Einheit muss kein Wert angegeben werden. Wird dennoch ein solcher Wert angegeben, dann muss er dem LDC-Wert für den Einheitentyp entsprechen.

#### **LOCAL={INITIALvFINAL}**

Es muss eine erweiterte lokale LDC-Liste generiert werden.

##### **INITIAL**

Diese Option gibt den Anfang einer erweiterten lokalen LDC-Liste an.

##### **FINAL**

Diese Option gibt das Ende einer erweiterten lokalen LDC-Liste an.

**Anmerkung:** LOCAL=INITIAL oder FINAL sind möglicherweise nicht in demselben Makro DFHTCT TYPE=LDC wie die anderen Operanden angegeben. Alle DFHTCT TYPE=LDC-Einträge, die nach LOCAL=INITIAL und vor LOCAL=FINAL angegeben sind, bilden Bestandteile einer erweiterten lokalen LDC-Liste. Die Einträge, die außerhalb der Struktur dieser Gruppe angegeben werden, werden zur System-LDC-Tabelle hinzugefügt.

Im Folgenden ist ein Beispiel für eine erweiterte lokale LDC-Liste aufgeführt:

```
DFHTCT TYPE=TERMINAL,TRMIDNT=BTCH,      *
TRMTYPE=BCHLU,ACCMETH=VTAM,LDC=LDCA      *
LDCA DFHTCT TYPE=LDC,LOCAL=INITIAL
DFHTCT TYPE=LDC,DVC=BLUPRT,LDC=AA,        *
PGESIZE=(6,30)
DFHTCT TYPE=LDC,DVC=BLUPCH,LDC=BB,        *
PGESIZE=(1,80)
DFHTCT TYPE=LDC,DVC=BLUCON,LDC=CC,        *
PGESIZE=(1,132),PGESTAT=AUTOPAGE
DFHTCT TYPE=LDC,LOCAL=FINAL
```

**Anmerkung:** VTAM wird jetzt als z/OS Communications Server bezeichnet.

#### **PGESIZE=(zeile,spalte)**

Geben Sie diesen Operanden mit der logischen Seitengröße an, die mit diesem LDC verwendet werden soll, wenn BMS-Anforderungen verarbeitet werden.

Das Produkt aus *zeile* und *spalte* darf 32767 nicht überschreiten.

#### **PGESTAT={AUTOPAGEvPAGE}**

Gibt an, ob die Einheit das Autopaging verwenden soll oder nicht. Autopaging bedeutet, dass BMS-Nachrichten mit mehreren Seiten fortlaufend gedruckt werden, ohne dass hierzu ein Bedienereingriff erforderlich ist. Dieses Verhalten ist normalerweise für einen Drucker erforderlich. (Gegensatz hierzu ist die Anforderung für Nachrichten mit mehreren Seiten, die auf einer 3270-Bildschirm-einheit angezeigt werden, wenn der Bediener das Lesen einer Seite abschließen möchte, bevor die nächste Seite angefordert wird.)

Nur BMS SEND-Befehle mit der Option PAGING verwenden das Autopaging. BMS SEND-Befehle mit TERMINAL oder SET verwenden das Autopaging hingegen nicht.

##### **AUTOPAGE**

Geben Sie diesen Wert für Drucker an.

##### **PAGE**

Geben Sie diesen Wert für Bildschirm-einheiten an.

Wenn die Standardwerte für PGESIZE oder PGESTAT, die im Operanden LDC angegeben sind, überschrieben werden sollen, dann geben Sie einen speziellen

LDC an, dessen mnemonisches Zeichen überschrieben werden soll. Geben Sie den überschreibenden LDC in der LDC-Tabelle vor dem Operanden LDC an.

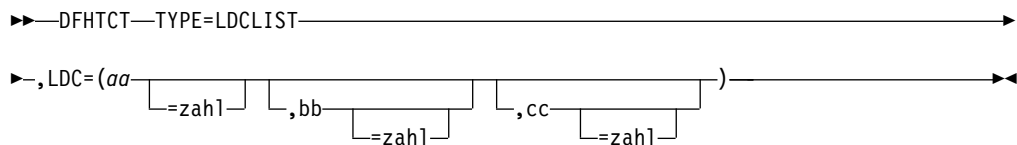
PGESTAT=AUTOPAGE kann verwendet werden, um die Angabe für PGESTAT in DFHTCT TYPE=TERMINAL zu überschreiben.

## Codes für logischen Einheiten - DFHTCT TYPE=LDCLIST

Das Makro DFHTCT TYPE=LDCLIST, das mit logischen 3600-Einheiten, logischen LUTYPE4-Einheiten und logischen Stapelverarbeitungseinheiten verwendet werden kann, erlaubt Ihnen die Erstellung einer allgemeinen Liste mit Codes für logische Einheiten (LDCs = Logical Device Codes), die von mehreren TCT-Termineinträgen gemeinsam genutzt werden können.

Sie sind für die Einrichtung der LDC-Struktur verantwortlich, die für das Terminal verwendet werden soll.

Die Erweiterung dieses Makros ist unabhängig von der Position, an der es innerhalb der TCT-Definition angegeben wird, gleich.



### *listenname*

Dies ist der erforderliche Name für die LDC-Liste. Auf diesen Namen wird von TCTTEs mithilfe des Operanden LDC in DFHTCT TYPE=TERMINAL verwiesen.

### **TYPE=LDCLIST**

Es wird eine LDC-Liste definiert.

### **LDC=(aa [=nnn] [,bb [=nnn]] [,cc [=nnn]] [,...])**

Geben Sie diesen Operanden mit den LDCs (mnemonische Zeichen und optional dem numerischen LDC-Wert) in dieser Liste an.

### **(aa [=nnn] [,bb [=nnn]] [,cc [=nnn]] [,...])**

Generiert die LDCs in der Liste.

### **aa,bb,cc...**

Die zweistelligen mnemonischen Zeichen für die LDCs in dieser Liste.

### **nnn**

Ein Dezimalwert im Bereich zwischen 1 und 255, der einem LCD zugeordnet werden soll. Wird kein Wert angegeben, dann wird für diesen LDC der Systemstandardwert aus der mit dem Makro DFHTCT TYPE=LDC definierten Tabelle verwendet. Dieser Wert muss für eine logische Stapelverarbeitungseinheit oder eine logische LUTYPE4-Einheit nicht angegeben werden. Wird er dennoch angegeben, dann muss er mit dem LDC-Wert der Einheit übereinstimmen. LDCs für Einheiten, die an eine logische Stapelverarbeitungseinheit oder logische LUTYPE4-Einheit angeschlossen sind, werden unter dem Parameter LDC des Makros DFHTCT TYPE=LDC aufgelistet.

## **DFHTCT-Beispiele: LDC**

Ein Beispiel, in dem dargestellt wird, wie das Makro DFHTCT TYPE=LDC für eine logische 3770-Stapelverarbeitungseinheit angegeben wird.

DFHTCT	TYPE=LDC,	*
	LDC=XX,	*
	DVC=BLUPRT,	*
	PGESIZE=(12,80),	*
	PGESTAT=PAGE	
DFHTCT	TYPE=LDC,	*
	LDC=YY,	*
	DVC=BLUPCH,	*
	PGESIZE=(1,80),	*
	PGESTAT=AUTOPAGE	
DFHTCT	TYPE=LDC,	*
	LDC=SYSTEM	

Abbildung 15. LDCs für logische 3770-Stapelverarbeitungseinheiten - TCT-Beispiel

## Sequenzielle Einheiten

CICS verwendet BSAM, um sequenzielle Terminals wie z. B. Kartenleser, Zeilendrucker, Magnetbandeinheiten und DASDs zu steuern. Sequenzielle Terminals ermöglichen die Simulation tatsächlicher Terminals und können verwendet werden, bevor die tatsächlichen Terminals verfügbar werden. Sequenzielle Terminals können außerdem beim Testen neuer Anwendungen benutzt werden.

Um eine sequenzielle Einheit zu definieren, müssen Sie die folgenden Makroanweisungen angeben:

```
DFHTCT TYPE=INITIAL,
        ACCMETH=(NONVTAM)    Definieren der Zugriffs-
                               methode
```

(Definieren Sie die folgenden Makroinstruktionen fortlaufend.)

```
DFHTCT TYPE=SDSCI,
        DSCNAME=isadscn,      Definieren der Eingabe-
        DDNAME=indd, ...      datei
DFHTCT TYPE=SDSCI,
        DSCNAME=osadscn,      Definieren der Ausgabe-
        DDNAME=outdd, ...     datei
DFHTCT TYPE=LINE,
        ISADSCN=isadscn,
        OSADSCN=osadscn, ...
DFHTCT TYPE=TERMINAL,
        TRMIDNT=name, ...
```

Die beiden in den DFHTCT TYPE=SDSCI-Makros definierten Dateien simulieren ein CICS-Terminal, das unter dem im Operanden TRMIDNT des Makros DFHTCT TYPE=TERMINAL angegebenen Namen bekannt ist. Die DSCNAMEs der Ein- und Ausgabedateien müssen in den Operanden ISADSCN und OSADSCN des Makros DFHTCT TYPE=LINE angegeben werden.

Der Datenendeanzeiger (EODI = End of Data Indicator) für sequenzielle Einheiten kann mithilfe des Systeminitialisierungsparameters EODI geändert werden.

## JCL für sequenzielle Einheiten

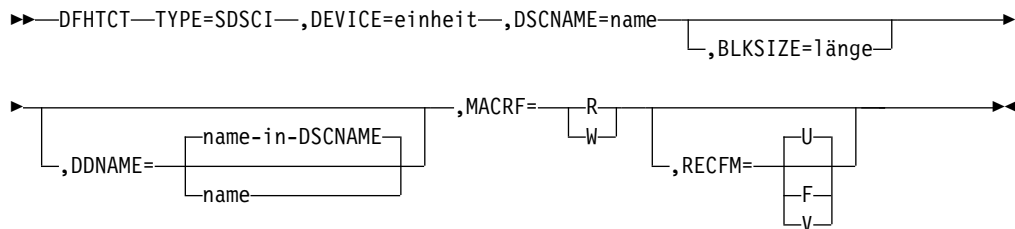
Die DDNAME-Operanden der DFHTCT TYPE=SDSCI-Makros dienen zur Angabe des DD-Namens der Datendefinitionsanweisungen, die sie im CICS-Startjobstrom angeben müssen.

```
//indd DD ...      Eingabedatei
//outdd DD ...     Ausgabedatei
```

Hierbei steht *indd* für die Datei, die die Eingabedaten eines simulierten Terminals enthält, und *outdd* für die Datei, an die die Ausgabe des simulierten Terminals gesendet wird.

## Sequenzielle Einheiten: DFHTCT TYPE=SDSCI

Das Makro DFHTCT TYPE=SDSCI gibt die Merkmale der Eingabe- und Ausgabedateien an, die ein CICS-Terminal simulieren.



### BLKSIZE=länge

Geben Sie diesen Operanden mit der maximalen Länge eines Blocks in Byte an.

Der Standardwert ist BLKSIZE=0. Wird dieser Operand nicht angegeben, kann die Blockgröße in der Datendefinitionsanweisung angegeben werden, die der Datei zugeordnet ist. Eine ausführlichere Erläuterung zu diesem Operanden erhalten Sie in z/OS DFSMS Macro Instructions for Data Sets.

### DDNAME={name-in-DSCNAME | name}

Dieser Operand gibt den Namen der Datendefinitionsanweisung an, die einer bestimmten Datei (Liniengruppe) zugeordnet ist. Wenn dieser Operand nicht angegeben wird, dann wird DSCNAME als DDNAME festgelegt.

### DEVICE=einheit

Sie können einen der folgenden Werte angeben:

- Für Kartenleser: {1442 | 2501 | 2520 | 2540 | 2560 | 2596 | 3505 | 3525 | 5425}
- Für Zeilendrucker: {1403 | 1404 | 1443 | 1445 | 3203 | 3211 | 5203}
- Für Platten (DASD): {2314 | 3330 | 3340 | 3350 | DASD | DISK}
- Für Bänder: TAPE.

Die Angabe TAPE generiert Arbeitsdateien für Bänder, die sowohl für die Eingabe- als auch die Ausgabedateien verwendet werden können. Beachten Sie hierbei, dass bei Verwendung eines Eingabebands mit abgelaufener Bezeichnung möglicherweise der Header neu geschrieben werden muss. Dies führt dazu, dass die ersten Datensätze zerstört werden.

### DSCNAME=name

Dieser Operand gibt den Namen der Eingabe- oder Ausgabedatei an. Wenn Sie die Eingabedatei definieren, dann muss der Wert für ISADSCN im Makro DFHTCT TYPE=LINE mit dem Namen übereinstimmen, den Sie angeben. Wenn Sie die Ausgabedatei definieren, dann muss der Wert für OSADSCN im Makro DFHTCT TYPE=LINE mit dem angegebenen Namen übereinstimmen.

### MACRF=( [R] [,W] )

Geben Sie diesen Operanden mit der Vorgehensweise an, die zum Erhalten des Zugriffs auf die sequenzielle Einheit verwendet wird.

- R** Gibt das READ-Makro an.
- W** Gibt das WRITE-Makro an.

Der Standardwert ist MACRF=R (für einen Kartenleser) und MACRF=W (für einen Zeilendrucker). Für andere sequenzielle Einheiten muss MACRF=R oder MACRF=W angegeben werden.

**RECFM={U|F|V}**

Geben Sie diesen Operanden mit dem Datensatzformat für den DCB (Data Control Block; Datensteuerblock) an.

**U** Gibt nicht definierte Datensätze an. Geben Sie diese Option für DEVICE=1403 oder 3211 oder dann an, wenn Sie DASD für die sequenzielle Terminalausgabe verwenden (d. h., wenn DEVICE=DASD und MACRF=W definiert ist).

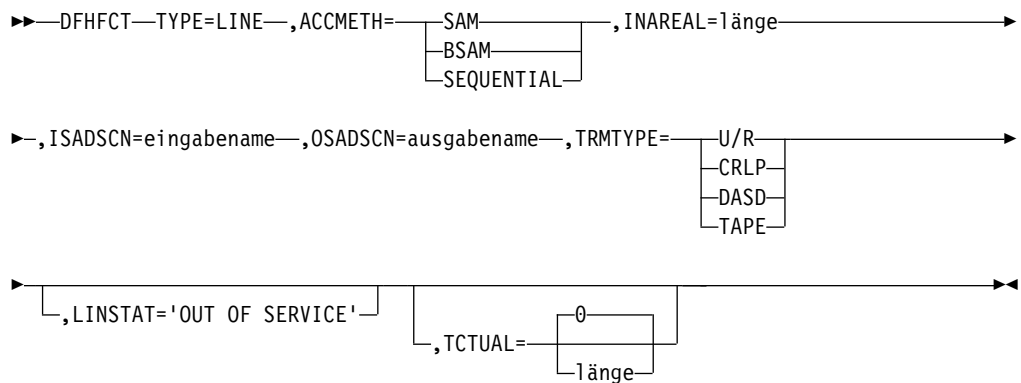
**F** Gibt Datensätze fester Länge an.

**V** Gibt Datensätze variabler Länge an.

Wenn Sie diesen Operanden nicht angeben, dann können Sie das Datensatzformat in der Datendefinitionsanweisung angeben, die der sequenziellen Datei zugeordnet ist.

### Sequenzielle Einheiten - DFHTCT TYPE=LINE

Das Makro DFHTCT TYPE=LINE gibt weiterführende Merkmale der sequenziellen Dateien an, die ein CICS-Terminal simulieren.



**ACCMETH={SAM|BSAM|SEQUENTIAL}**

Geben Sie die Optionen SAM, BSAM oder SEQUENTIAL an, die in CICS gleichwertig sind.

**INAREAL=länge**

Geben Sie diesen Operanden mit der Länge des Nachrichteneingabebereichs an. Der Wert muss gleich der Länge des längsten logischen Anfangsdatsatzes einer Transaktion sein, die mehrere physische Datensätze umfassen kann.

**ISADSCN=name**

Der Name der Eingabedatei. Der Operand TYPE=SDSCI DSCNAME für die Eingabedatei muss mit diesem Wert übereinstimmen.

**LINSTAT='OUT OF SERVICE'**

Die Zeile muss mit dem Status 'out of service' beginnen.

Der Standardwert ist 'in service'.

**OSADSCN=name**

Der Name der Ausgabedatei. Der Operand TYPE=SDSCI DSCNAME für die Ausgabedatei muss mit diesem Wert übereinstimmen.

**TCTUAL={0|länge}**

Gibt die Länge in Byte (0 bis 255) des Benutzerbereichs (Feld für die Prozesssteuerungsinformationen oder PCI) für alle Terminaleinträge (TCTTEs) an, die dieser Zeile zugeordnet sind. Die Länge sollte auf ein Minimum begrenzt wer-

den. Der TCT-Benutzerbereich wird bei der Systeminitialisierung mit Nullen initialisiert. Wenn Sie Felder unterschiedlicher (variabler) Länge verwenden möchten, dann können Sie den Wert für TCTUAL in einzelnen oder mehreren TYPE=TERMINAL-Makroinstruktionen für die Terminals angeben, die dieser Zeile zugeordnet sind.

#### TRMTYPE=(U/RvCRLPvDASDvTAPE)

Gibt den Typ der sequenziellen Einheit an:

**U/R** Beliebiges Lesegerät oder beliebiger Drucker.

**CRLP** Ein Kartenleser und ein Zeilendrucker.

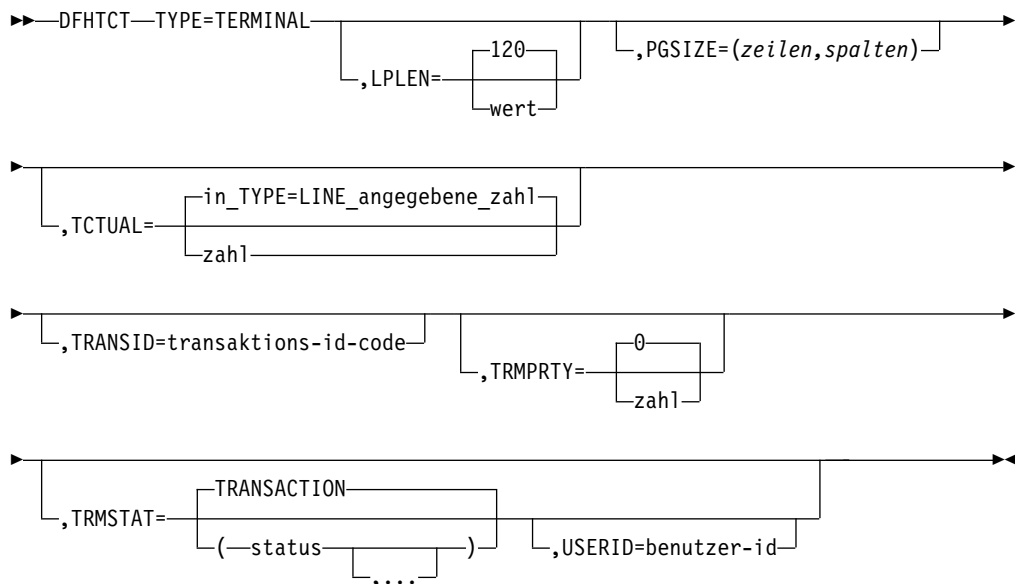
**DASD**

Eine DASD-Einheit (DASD = Direct Access Storage Device).

**TAPE** Eine Magnetbandeinheit.

#### Sequenzielle Einheiten: DFHTCT TYPE=TERMINAL

Im Makro DFHTCT TYPE=TERMINAL können Sie den Terminalnamen und andere Merkmale eines CICS-Terminals angeben, das mithilfe eines Paares sequenzieller Dateien simuliert wird.



#### LPLEN={120vwert}

Steuert die Länge der Druckzeile für SAM-Ausgabezeilendrucker. Wird in einer segmentierten Schreiboperation kein Zeilenvorschubzeichen gefunden, dann entspricht die Länge der Druckzeile dem Wert für LPLEN. Der Standardwert ist LPLEN=120.

#### PGESIZE=(zeilen, spalten)

Die Standardseitengröße für ein 1403- oder CRLP-Terminal ist (12,80). Geben Sie PGESIZE an, wenn für eine Einheit, für die TRMTYPE=DASD angegeben wurde, BMS benötigt wird. Geben Sie die Anzahl der Zeilen und Spalten an, die verwendet werden sollen. Die Multiplikation dieser beiden Werte muss gleich dem Wert sein, der für INAREAL angegeben wurde. Das Produkt von *zeilen* und *spalten* darf maximal 32767 betragen.

#### TCTUAL={in\_TYPE=LINE\_angegebene\_zahl vzahl}

Gibt die Länge in Byte (0 bis 255) des Benutzerbereichs (Feld für die Prozesssteuerungsinformationen oder PCI) für den Terminaleintrag (TCTTE) an, der

diesem Terminal zugeordnet ist. Die Länge sollte auf ein Minimum begrenzt werden. Der TCT-Benutzerbereich wird bei der Systeminitialisierung mit Nullen initialisiert.

Verwenden Sie den Operanden TCTUAL des Makros DFHTCT TYPE=TERMINAL, wenn Sie Felder unterschiedlicher (variabler) Länge für die Terminals verwenden möchten, die dieser Zeile zugeordnet sind. In jedem Fall wird das PCI-Feld für jedes Terminal nach dem letzten Terminaleintrag der letzten Zeile generiert. Die Adresse des PCI-Felds befindet sich in TCTTECIA, die Länge hingegen in TCTTECIL.

#### **TRANSID=transaktions-id-code**

Geben Sie diesen Operanden mit einem aus einem bis vier Zeichen bestehenden Transaktionscode an. Dieser Code gibt eine Transaktion an, die immer dann initialisiert werden soll, wenn Eingabedaten von dem Terminal empfangen werden und keine aktive Task vorhanden ist.

Wenn im TCTTE keine Transaktions-ID (TRANSID) angegeben wurde, dann wird der Wert für TRANSID aus einem Befehl RETURN einer vorangegangenen Transaktion verwendet. Andernfalls werden die Zeichen eins bis vier der Daten, die in TIOA übergeben wurden, als Transaktionscode verwendet. Für Transaktionskennungen mit weniger als vier Zeichen ist ein Begrenzer erforderlich.

#### **TRMIDNT=name**

Geben Sie diesen Operanden mit einer eindeutigen, aus vier Zeichen bestehenden, symbolischen Kennung für jedes Terminal an. Die angegebene Kennung ist linksbündig ausgerichtet und wird mit Leerzeichen aufgefüllt, wenn weniger als vier Zeichen angegeben werden.

Der Wert CERR ist reserviert, da es sich hierbei um die Kennung handelt, die für die Fehlerkonsole generiert wurde.

#### **TRMPRTY={0vzahl}**

Mit diesem Operanden wird die Terminalpriorität eingerichtet. Dieser Dezimalwert (0 bis 255) wird verwendet, um die übergreifende Priorität für die Transaktionsverarbeitung einzurichten. (Die Priorität für die Transaktionsverarbeitung entspricht der Summe der Terminalpriorität, Transaktionspriorität und Bedienerpriorität und darf den Wert 255 nicht überschreiten.)

#### **TRMSTAT={TRANSACTIONv(status,...)}**

Geben Sie diesen Operanden mit den Aktivitätstypen an, die auf einem bestimmten Terminal auftreten können. Dieser Terminalstatus wird zu Beginn im TCTTE gesetzt und stellt eine Kombination des Verarbeitungstatus und des Servicestatus dar.

#### **TRANSACTION**

Ein Terminal mit TRANSACTION-Status wird zur Verarbeitung von Transaktionen wie z. B. Abfragen oder für die Auftragserfassung verwendet. Ein Datensichtgerät oder ein Hardcopy-Terminal, an das keine Nachrichten ohne Terminalanforderung gesendet werden und über das Transaktionen eingegeben werden, wird als TRANSACTION-Terminal bezeichnet.

#### **INPUT**

Gibt ein Terminal an, das Nachrichten an CICS senden kann, das jedoch keine Nachrichten von CICS empfangen kann.

**Anmerkung:** Die Systemnachrichten können an ein Eingabeterminal weitergeleitet werden, wenn Bedingungen wie beispielsweise eine ungültige Transaktionskennung und ein ATP-Stapelzähler vorliegen. Dies führt dazu,

dass DFHTACP terminiert wird. Zur Behandlung dieser Situation müssen Sie DFHTEP angeben, sodass alle Aktionen ausgeführt werden, die der Benutzer benötigt.

#### **'OUT OF SERVICE'**

Gibt ein Terminal an, das weder Nachrichten empfangen noch Eingabedaten senden kann. Solche Terminals werden von CICS nicht abgefragt. Der Parameter 'OUT OF SERVICE' kann zusammen mit einer beliebigen Stauseinstellung verwendet werden.

Alle Terminals mit Ausnahme des Hauptterminals können mit der Option 'OUT OF SERVICE' markiert werden. Wenn dies erforderlich ist, kann das Terminal vom Hauptterminal in Betrieb genommen und die Abfrage kann wiederaufgenommen werden.

#### **RECEIVE**

Gibt ein Terminal an, an das Nachrichten gesendet werden, über das jedoch keine Eingabe zulässig ist. Ein Beispiel für diesen Terminaltyp stellt ein Terminal dar, das sich an einem fernen Standort (z. B. in einem Warehouse) befindet und dort nicht überwacht wird, jedoch Nachrichten empfangen kann. Die automatische Transaktionsinitialisierung wird wie für TRANSCEIVE implementiert.

#### **TRANSCEIVE**

Ein Terminal mit TRANSCEIVE-Status ist ein TRANSACTION-Terminal, an das Nachrichten automatisch gesendet werden. Die automatische Transaktionsinitialisierung anhand der Steuerung transienter Daten oder der Intervallsteuerung, setzt eine Bedingung in einem entsprechenden Terminaleintrag einer Terminalsteuertabelle. Wenn der Terminalstatus TRANSCEIVE lautet und wenn keine Transaktion in dem Terminal stattfindet, dann leitet die Terminalsteuerung die benutzerdefinierte Task ein. Diese Task sendet erwartungsgemäß Nachrichten an das Terminal.

#### **USERID=benutzer-id**

Geben Sie diesen Operanden an, um eine Benutzer-ID für Einheiten wie beispielsweise Drucker anzugeben, die keine Anmeldung mit CESN durchführen können. (Sie können für eine Bildschirmeinheit auch USERID angeben. In diesem Fall bleibt der Bildschirm permanent angemeldet. Bediener können sich nicht anmelden.) Sie müssen diesen Operanden angeben, wenn Sie die voreingestellte Sicherheit für diese Einheit benutzen möchten. Der gesamte Zugriff auf geschützte Ressourcen ist vom Operanden USERID abhängig.

Auf die Benutzer-ID wird in Nachrichten zu Sicherheitsfehlern und Sicherheitsverstößen sowie im Prüfprotokoll verwiesen. Sie muss für den Sicherheitsmanager definiert sein.

Die Benutzer-ID muss eine eindeutige Benutzerkennung aus einem bis acht Zeichen sein. (Die folgenden Zeichen sind zulässig: A-Z 0-9 # \$ und @.)

## **Ferne Terminals für Transaktionsrouting**

In Bezug auf die übergreifende CICS-Kommunikation stellt das System, auf dem die Terminalsteuertabelle (TCT = Terminal Control Table) installiert ist, in Hinblick auf die Ressourcendefinition das **lokale** System dar. Das System, das in der Terminalsteuertabelle definiert wird, ist das **ferne** System. Das *Transaktionsrouting* ermöglicht Terminals auf einem CICS-System den Aufruf von Transaktionen auf einem anderen CICS-System.

Sie können das Transaktionsrouting zwischen Systemen verwenden, die über MRO oder über eine LUTYPE 6.2-Verbindung verbunden sind.



## Ferne Definitionen für Terminals für das Transaktionsrouting

Es gibt zwei Möglichkeiten, um die Terminals mit Makros zu definieren.

Die folgenden beiden Methoden stehen zur Verfügung:

- **Methode 1:**

DFHTCT TYPE=REGION, ... (eine für jede Region)

DFHTCT TYPE=SDSCI, ... (nur für Nicht-SNA-LU;  
wird für ferne Definitionen ignoriert)

DFHTCT TYPE=LINE, ... (nur für Nicht-SNA-LU)

DFHTCT TYPE=TERMINAL, ... (für Nicht-SNA-Einheiten: eine für jede LU)

- **Methode 2:**

DFHTCT TYPE=REMOTE, ... (eine für jedes Terminal)

**Tip:** Eine andere Methode, der sog. Versand von Terminaldefinitionen, kann mit Hilfe von RDO angewendet werden. Weitere Informationen hierzu finden Sie in „Terminals für Transaktionsrouting“ auf Seite 216.

Bei beiden Methoden können dieselben Terminaldefinitionen verwendet werden, um die erforderlichen Einträge auf dem lokalen und dem fernen System zu generieren.

**Methode 1:**

Sie können Copybooks verwenden, um denselben Quellcode in die TCTs für lokale und ferne Systeme einzuschließen. Die Informationen, die nicht benötigt werden (das vollständige Makro TYPE=SDSCI und einige der TYPE=LINE- und TYPE=TERMINAL-Makros), werden für ferne Einträge verworfen.

CICS entscheidet auf Basis des Operanden SYSIDNT im Makro TYPE=REGION, ob eine ferne oder eine lokale Definition erstellt werden soll. Dabei wird ein Vergleich mit dem Operanden SYSIDNT in DFHTCT TYPE=INITIAL durchgeführt. Wird dabei eine Übereinstimmung festgestellt, dann handelt es sich um lokale Definitionen. Wenn sie sich unterscheiden, dann handelt es sich um ferne Definitionen.

**Methode 2:**

Verwendet ein einzelnes Makro DFHTCT TYPE=REMOTE.

CICS entscheidet auf Basis des Operanden SYSIDNT im Makro TYPE=REMOTE, ob eine ferne oder eine lokale Definition erstellt werden soll. Dabei wird ein Vergleich mit dem Operanden SYSIDNT in DFHTCT TYPE=INITIAL durchgeführt. Wird dabei eine Übereinstimmung festgestellt, dann handelt es sich um lokale Definitionen. Wenn sie sich unterscheiden, dann handelt es sich um ferne Definitionen.

Diese Terminals können das Transaktionsrouting nicht verwenden und können daher nicht als fern definiert werden:

- Terminals IBM 7770 oder 2260
- MVS-Systemkonsolen
- Logische 3600- oder 3650-Pipelineeinheiten im Pool

### Ferne Terminals - Methode 1 - DFHTCT TYPE=REGION

Mit dem Makro DFHTCT TYPE=REGION werden Informationen über die angegebene Region eingeführt. Die Informationen bestehen aus den Makros DFHTCT TYPE=LINE und TYPE=TERMINAL.

Diese Makros müssen nach dem Makro DFHTCT TYPE=REGION angegeben werden. Für eine ferne Region wird vom Makro DFHTCT TYPE=LINE kein TCT-Zeileintrag (TCTLE) generiert. Jedes Terminal, das am Transaktionsrouting teilnimmt, muss definiert werden. Nur bestimmte DFHTCT-Makrotypen und -Operanden sind in den Definitionen der fernen Regionen von Bedeutung, wohingegen alle anderen ignoriert werden. Die Operanden, die von Bedeutung sind, werden in „Ferne Terminals - Methode 2 - DFHTCT TYPE=REMOTE“ auf Seite 473 aufgelistet.

►►DFHTCT—TYPE=REGION—,SYSIDNT=name—————►  
                                   └LOCAL┐

#### **SYSIDNT={name┐LOCAL}**

Gibt den aus vier Zeichen bestehenden Namen des Systems oder der Region an, dessen bzw. deren Informationen hier beginnen oder fortgesetzt werden. SYSIDNT=LOCAL kann angegeben werden, um festzulegen, dass die nachfolgenden TYPE=TERMINAL-Definitionen auf die Ausgangsregion verweisen. Dies ist auch bei allen Definitionen vor dem ersten Makro DFHTCT TYPE=REGION der Fall. Der Name der Ausgangsregion (d. h. der Region, in der diese Terminalsteuertabelle verwendet wird) stellt den Wert des Operanden SYSIDNT des Makros DFHTCT TYPE=INITIAL dar. Der Name kann stattdessen auch der einer zuvor definierten MRO- oder ISC-Verbindung sein.

### **Ferne Terminals - Methode 1 - DFHTCT TYPE=TERMINAL**

TCT-Makrodefinitionen (TCT = Terminal Control Table; Terminalsteuertabelle) zum Definieren ferner Terminals unter Verwendung von Methode 1.

**Anmerkung:** Das Makro DFHTCT TYPE=LINE und die zusätzlichen Operanden des Makros DFHTCT TYPE=TERMINAL sind gültig, werden jedoch ignoriert, wenn der Operand SYSIDNT im vorangegangenen Makro DFHTCT TYPE=REGION eine ferne Region angibt. (Detaillierte Informationen zu den Makros DFHTCT TYPE=LINE und DFHTCT TYPE=TERMINAL finden Sie in „Sequenzielle Einheiten“ auf Seite 465.)

►►DFHTCT—TYPE=TERMINAL—,ACCMETH=zugriffsmethode—,SYSIDNT=name—————►  
 ►,TRMIDNT=name—,TRMTYPE=terminaltyp—————►  
 ►  
   └,RMTNAME=in\_TRMIDNT\_angegebener\_name┐  
   └name┐

#### **ACCMETH=zugriffsmethode**

Geben Sie diesen Operanden mit der Zugriffsmethode des fernen Terminals an.

#### **RMTNAME={in\_TRMIDNT\_angegebener\_name┐name}**

Gibt den aus einem bis vier Zeichen bestehenden Namen an, unter dem das Terminal auf dem System oder in der Region bekannt ist, das bzw. die das Terminal verwaltet (d. h. in der Terminalsteuertabelle des jeweils **anderen** Systems). Wenn dieser Operand nicht angegeben wird, wird der Name im Operanden TRMIDNT verwendet.

#### **SYSIDNT=name**

Gibt den aus vier Zeichen bestehenden Namen des Systems oder der Region an, das bzw. die dieses Terminal verwaltet. Hierbei kann es sich um das lokale System oder die lokale Region (d. h., den im Makro TYPE=INITIAL definierten

Namen) handeln. In diesem Fall handelt es sich bei dem erstellten TCT-Eintrag um eine lokale Definition. Es kann sich auch um den Namen eines anderen Systems oder einer anderen Region handeln. In diesem Fall handelt es sich bei dem erstellten TCT-Eintrag um eine ferne Definition. Diese SYSIDNT muss mit der SYSIDNT im Makro TYPE=REGION übereinstimmen, das diesem Makro vorangestellt ist.

**TRMIDNT=name**

Gibt den aus einem bis vier Zeichen bestehenden Namen an, unter dem das Terminal auf **diesem** System bekannt ist (d. h. auf dem lokalen System, das Eigner dieser Terminalsteuertabelle ist und ebenfalls Eigner der Transaktionen).

**TRMTYPE=terminaltyp**

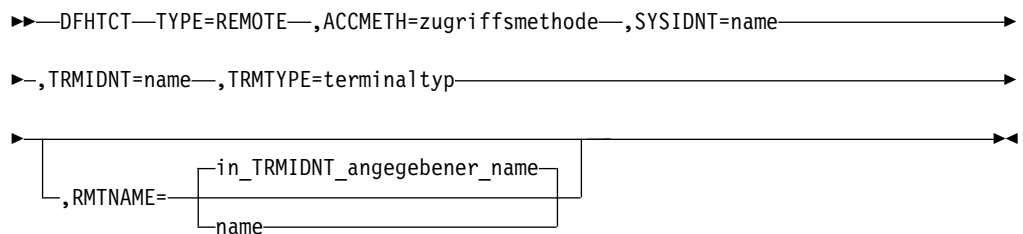
Geben Sie diesen Operanden mit dem Terminaltyp an. Detaillierte Informationen hierzu finden Sie in „Sequenzielle Einheiten“ auf Seite 465.

**Ferne Terminals - Methode 2 - DFHTCT TYPE=REMOTE**

Terminaleinträge für ferne Systeme oder Regionen können in CICS mithilfe des Makros DFHTCT TYPE=REMOTE definiert werden. Diese Vorgehensweise kann alternativ zum Definieren dieser Einträge mit den DFHTCT TYPE=TERMINAL-Makroinstruktionen in Verbindung mit dem Makro DFHTCT TYPE=REGION angewendet werden.

Die Erweiterung des Makros DFHTCT TYPE=REMOTE ist unabhängig von der derzeit referenzierten Region.

**Anmerkung:** Wenn der Operand SYSIDNT angibt, dass die **Ausgangsregion** das Terminal verwaltet, dann werden alle Operanden des Makros DFHTCT TYPE=TERMINAL im Makro DFHTCT TYPE=REMOTE gültig und haben dieselbe Bedeutung wie für TYPE=TERMINAL. Wenn allerdings der Operand SYSIDNT (wie üblich) eine ferne Region angibt, dann sind die zusätzlichen Operanden von DFHTCT TYPE=TERMINAL zwar für das Makro DFHTCT TYPE=REMOTE gültig, werden jedoch ignoriert.



**ACCMETH=zugriffsmethode**

Geben Sie diesen Operanden mit der Zugriffsmethode des fernen Terminals an.

**RMTNAME={in\_TRMIDNT\_angegebener\_name|name}**

Gibt den aus einem bis vier Zeichen bestehenden Namen an, unter dem das Terminal auf dem System oder in der Region bekannt ist, das bzw. die das Terminal verwaltet (d. h. in der Terminalsteuertabelle des jeweils **anderen** Systems). Wenn dieser Operand nicht angegeben wird, wird der Name im Operanden TRMIDNT verwendet.

**SYSIDNT=name**

Gibt den Namen des Systems oder der Region an, das bzw. die dieses Terminal verwaltet. Der Name muss mit dem Namen übereinstimmen, der im Operanden SYSIDNT eines vorherigen Makros TYPE=SYSTEM oder im Makro TYPE=INITIAL verwendet wird.

**TRMIDNT=name**

Gibt den aus einem bis vier Zeichen bestehenden Namen an, unter dem das Terminal auf **diesem** System bekannt ist (d. h. auf dem lokalen System, das Eigner dieser Terminalsteuertabelle ist und ebenfalls Eigner der Transaktionen).

**TRMYPE=terminaltyp**

Geben Sie diesen Operanden mit dem Terminaltyp an. Detaillierte Informationen hierzu finden Sie in „Sequenzielle Einheiten“ auf Seite 465.

## DFHTCT: CICS-Terminalliste

Dieses Release von CICS ist in der Lage, mit einer umfangreichen Palette von Terminals entweder direkt oder indirekt zu kommunizieren.

Neue oder aktuelle Terminals werden direkt von CICS Transaction Server for z/OS unterstützt, wenn sie mit der z/OS Communications Server-Schnittstelle kompatibel sind.

In Tabelle 20 wird zusammengefasst, in welcher Weise die Terminals in CICS unterstützt werden.

*Tabelle 20. Von CICS Transaction Server for z/OS unterstützte IBM Terminals und Systemtypen.*

### Direkte Unterstützung durch CICS Transaction Server for z/OS über z/OS Communications Server

3101-Datensichtgerät  
 3230-Drucker  
 3268-Drucker  
 3270 Information Display System  
 3270-PC  
 3270-PC/G  
 3270-PC/GX  
 3287-Drucker  
 3600 Finance Communication System  
 3630 Plant Communication System  
 3640 Plant Communication System  
 3650 Retail Store System  
 3680 Programmable Store System  
 3730 Distributed Office Communication System  
 3767 Communication Terminal  
 3770 Data Communication System  
 3790 Communication System  
 4300-Prozessoren  
 4700 Finance Communication System  
 5280 Distributed Data System  
 5520 Administrative System  
 5550 Administrative System  
 5937 Rugged Terminal  
 6670 Information Distributor  
 8100 Information System  
 8775 Display Terminal  
 8815 Scanmaster  
 Displaywriter  
 Personal Computer, PS/2, PS/55  
 System/32  
 System/34  
 System/36

*Tabelle 20. Von CICS Transaction Server for z/OS unterstützte IBM Terminals und Systemtypen. (Forts.)*

**Direkte Unterstützung durch CICS Transaction Server for z/OS über z/OS Communications Server**

System/38

iSeries

z Systems

Teletypewriter Exchange Service (TWX 33/35)

World Trade Typewriter Terminal (WTTY)

## **Logische z/OS Communications Server-Einheiten**

Ein detaillierte Liste der von z/OS Communications Server for SNA unterstützten LUs und Angaben zur Definition dieser Einheiten in CICS finden Sie in „Unterstützte Einheiten“ auf Seite 274.

---

## **TLT: Terminallistentabelle**

Eine Terminallistentabelle (TLT = Terminal List Table), die mit der DFHTLT-Makroinstruktion generiert wird, ermöglicht die logische Gruppierung von Terminal- und Bedieneridentifikationen.

Für eine Terminallistentabelle gilt Folgendes:

- Sie ist **zwingend erforderlich** für CEST (Supervisor-Terminal-Transaktion), um den geltenden Bereich der Operation zu definieren und einzuschränken. Beispiel:  
CEST SET TERMINAL(\*) SUPRID(CG) OUTSERVICE

Diese Angabe setzt alle in DFHTLTG definierten Terminals außer Betrieb ('OUT OF SERVICE').

- Sie kann von CEST oder CEMT (Hauptterminaltransaktion) verwendet werden, um eine Operation auf eine vordefinierte Gruppe von Terminals anzuwenden. (Für eine CEST-Operation muss diese Terminallistentabelle eine Untergruppe der Terminallistentabelle definieren, die in SUPRID angegeben ist.) Beispiel: Jeder der folgenden Befehle

```
CEST SET TERMINAL(*) SUPRID(CG) CLASS(EM) INSERVICE
CEMT SET TERMINAL(*) CLASS(EM) INSERVICE
```

setzt alle in DFHTLTEM definierten Terminals in Betrieb.

- Sie kann alleine oder in Kombination mit anderen TLTs verwendet werden, um vordefinierte Ziele für die Nachrichtenverteilung bereitzustellen. Beispiel:

```
MSG ROUTE=PG,'PRODUCTION MEETING AT 11.00 IN
ROOM 2124',SEND
```

Diese Angabe sendet eine Nachricht an alle Terminals oder Bediener, die in DFHTLTPG definiert sind.

Eine bestimmte Terminallistentabelle kann für die Nachrichtenverteilung und für die Funktionen der Supervisor-Terminals und Hauptterminals verwendet werden. Eine Terminallistentabelle, die zum Definieren der Terminals dient, die sich unter der Steuerung eines Supervisor-Terminals befinden, kann auch als Zieladressenliste für das Senden von Nachrichten an diese Terminals benutzt werden.

Für bestimmte logische Einheiten werden mnemonische LDC-Zeichen (LDC = Logical Device Code; Code für logische Einheiten) verwendet, die den einzelnen Ta-

belleneinträgen zugeordnet sein können. Sie werden für die Nachrichtenverteilung benutzt und für Operationen von Hauptterminals und Supervisor-Terminals ignoriert.

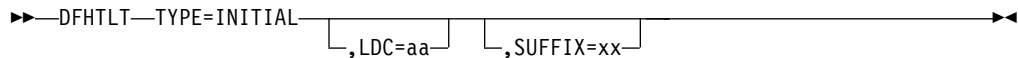
In einem Netz mit übergreifender Kommunikation müssen alle Terminals in einer Terminallistentabelle dem System zugeordnet sein, auf dem die Tabelle verwendet wird.

Die folgenden Makros definieren die TLT-Einträge:

- Programmabschnitt: DFHTLT TYPE=INITIAL
- Einträge in Terminallistentabelle: DFHTLT TYPE=ENTRY
- Ende der Terminallistentabelle: DFHTLT TYPE=FINAL (siehe hierzu TYPE=FINAL (end of table))

## Programmabschnitt: DFHTLT TYPE=INITIAL

Mit dem Makro DFHTLT TYPE=INITIAL werden der Einstiegspunkt und die Adresse für den Beginn der momentan definierten Terminallistentabelle (TLT = Terminal List Table) eingerichtet.



**Anmerkung:** Allgemeine Informationen zu den TYPE=INITIAL-Makros finden Sie in TYPE=INITIAL (control section).

### LDC=aa

Geben Sie diesen Operanden mit einem zweistelligen mnemonischen LDC-Zeichen (LDC = Logical Device Code; Code für logische Einheiten) an. Diese Komponente ist jeder Identifikation einer logischen Einheit zugeordnet. Eine Ausnahme bilden hierbei lediglich diejenigen Elemente, für die ein mnemonisches LDC-Zeichen im Makro DFHTLT TYPE=ENTRY angegeben ist.

### SUFFIX=xx

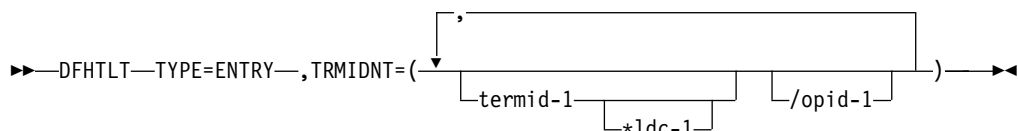
Der Modulname der Terminallistentabelle lautet DFHTLTxx. Hierbei steht xx für ein aus einem bis zwei Zeichen bestehendes Suffix. Auf diese Weise kann jede verwendete Terminallistentabelle eindeutig identifiziert werden. Da die Namen TLTBa, TLTBb, TLTBc und TLTEa in der Terminallistentabelle verwendet werden, dürfen die Suffixe BA, BB, BC und EA nicht verwendet werden.

Eine Terminallistentabelle muss über ein Suffix verfügen, das von der Nachrichtenverteilungstransaktion (CMSG) verwendet werden kann.

## Einträge in Terminallistentabelle: DFHTLT TYPE=ENTRY

Mit DFHTLT TYPE=ENTRY können Sie einen Eintrag in der Terminallistentabelle (TLT = Terminal List Table) definieren.

In der Terminallistentabelle werden Einträge wie folgt angegeben:



**TYPE=ENTRY**

Geben Sie diese Option an, wenn einzelne oder mehrere Einträge in dieser Tabelle generiert werden sollen. Es können maximal 1000 Einträge erstellt werden.

**TRMIDNT=([termid-1[\*1dc-1]] [/opid-1][, termid-2[\*1dc-2] [/opid-2],...])**

Geben Sie diesen Operanden mit einer Start-Stopp-Liste und dem BSC-Terminal, der logischen Einheit und den Bedieneridentifikationen an. Die Kennung einer logischen Einheit kann anhand eines mnemonischen LDC-Zeichens qualifiziert werden.

**termid**

Gibt die aus einem bis vier Zeichen bestehende Kennung der Start-Stopp-Komponente, des BSC-Terminals oder der logischen Einheit an.

**Anmerkung:** Eine 3614-Einheit, die mit einem DFV-Controller verbunden ist, kann in Hauptterminal- oder Supervisor-Terminal-Operationen verwendet werden, sollte jedoch nicht in Nachrichtenverteilungsoperationen verwendet werden. (Eine 3614-Einheit kann nicht als Nachrichtenziel angegeben werden.)

**1dc**

Gibt ein zweistelliges mnemonisches LDC-Zeichen an, dem ein Stern (\*) vorangestellt werden muss und das nur nach dem Parameter 'termid' angegeben werden kann, dem es zugeordnet ist.

**opid**

Gibt die aus einem bis drei Zeichen bestehende Bedieneridentifikation an, der ein Schrägstrich (/) vorangestellt werden muss.

Jede Terminal- oder Bedieneridentifikation, die angegeben wird, muss auch im Operanden TRMIDNT des Makros DFHTCT und in Ihrem externen Sicherheitsmanager angegeben werden. (Wenn Sie mit RACF arbeiten, dann verwenden Sie den Operanden OPIDENT des Befehls ADDUSER, um die Identifikation für jeden Bediener aufzuzeichnen.) Jedes angegebene mnemonische LDC-Zeichen muss auch im Operanden LDC der Makros DFHTCT TYPE=LDC und DFHTCT TYPE=TERMINAL angegeben werden.

Supervisor-Terminal-Funktionen und Hauptterminalfunktionen verwenden die Identifikationen von Terminals und logischen Einheiten, die in der Terminallistentabelle (TLT = Terminal List Table) angegeben sind, ignorieren jedoch alle Verweise auf mnemonische LDC-Zeichen und Bedieneridentifikationen.

**DFHTLT-Beispiel**

Ein Beispiel für die Erstellung einer Terminallistentabelle (TLT = Terminal List Table).

**Beispiel 1**

```

DFHTLT TYPE=INITIAL,          *
      SUFFIX=AA
DFHTLT TYPE=ENTRY,           *
      TRMIDNT=(NYC,CHI,LA,WDC)
DFHTLT TYPE=ENTRY,           *
      TRMIDNT=SF
DFHTLT TYPE=ENTRY,           *
      TRMIDNT=(BSTN/OP1,ATL/OP5,/OP9,DNVR)
DFHTLT TYPE=ENTRY,           *
      TRMIDNT=/OP6
DFHTLT TYPE=FINAL
END

```

**Beispiel 2**

```

DFHTLT TYPE=INITIAL,          *
      SUFFIX=XX
DFHTLT TYPE=ENTRY,           *
      TRMIDNT=(NYC,T361*LP,T362*LP/OP1)
DFHTLT TYPE=ENTRY,           *
      TRMIDNT=(T363/OP2,T364/OP5,T365)
DFHTLT TYPE=FINAL
END

```

*Abbildung 16. Terminallistentabelle: Beispiel*

## TST: Tabelle für temporären Speicher

Die Tabelle für temporären Speicher (TST = Temporary Storage Table) enthält eine Liste generischer Namen (oder Präfixe), die zur Identifikation von Gruppen mit Warteschlangen für temporären Speicher benutzt werden. Jede eindeutige Kennung für temporären Speicher, die in einem Anwendungsprogramm dynamisch erstellt wird und die mit den gleichen Zeichen beginnt wie die generischen Namen, erhält automatisch die gleichen Eigenschaften wie die TST-Einträge.

CICS bietet weiterhin Unterstützung für die Verwendung des Makros DFHTST in Kombination mit oder anstelle von TSMODEL-Ressourcendefinitionen. Sie müssen eine Tabelle für temporären Speicher (TST = Temporary Storage Table) verwenden, wenn die folgenden Bedingungen gelten:

- Sie arbeiten mit Anwendungsprogrammen, die auf Warteschlangen für temporären Speicher mit gemeinsamer Datennutzung verweisen, indem in den EXEC CICS-Befehlen für temporären Speicher explizit der Operand SYSID angegeben wird.
- Eine SYSID wird für EXEC CICS-Befehle für temporären Speicher über ein globales XTSEREQ-Benutzerexitprogramm hinzugefügt.
- Sie benötigen das Attribut TSAGE.

Für Warteschlangen für temporären Speicher, in denen diese Funktionen nicht benötigt werden, können Sie die TSMODEL-Ressourcendefinitionen verwenden, die alle anderen Funktionen der TST und einige zusätzliche Funktionen bereitstellen.

Der standardmäßige Systeminitialisierungsparameter TST=NO bedeutet, dass CICS nur mit RDO-Unterstützung für Warteschlangen für temporären Speicher (TS-Warteschlangen) initialisiert wird. Zur Verwendung einer TST in Kombination mit TSMODEL-Ressourcendefinitionen müssen Sie mithilfe des Systeminitialisierungsparameters TST ein TST-Suffix angeben. Darüber hinaus müssen Sie das TST-Lademodul mit der Option MIGRATE assemblieren. Wenn die TST nicht mit der



Option MIGRATE assembliert wird, dann lädt CICS die TST nur und stellt keine RDO-Unterstützung für Warteschlangen für temporären Speicher (TS-Warteschlangen) zur Verfügung. Jeder Versuch, TSMODEL-Ressourcendefinitionen zu installieren, wird zurückgewiesen.

Wenn Sie sowohl eine TST- als auch eine TSMODEL-Ressourcendefinition verwenden, dann unterliegt der Einsatz der TST folgenden Einschränkungen:

- Unterstützung der TS-Warteschlangen mit gemeinsamer Datennutzung, auf die explizit mit der Option SYSID in einem TS-API-Befehl verwiesen wird.
- Das Attribut TSAGE.

Wenn Sie eine TST alleine verwenden, dann werden alle Funktionen der TST benutzt.

## Generische Namen

In einer TST werden generische Namen aus den führenden Zeichen der entsprechenden Warteschlangennamen gebildet. Sie können bis zu sieben Zeichen lang sein.

- Die generischen Namen, die im Makro DFHTST TYPE=RECOVERY angegeben werden, geben Warteschlangen an, für die CICS das Zurücksetzen von Änderungen im Falle eines Transaktionsfehlers oder als Schutz gegen Systemfehler bereitstellt.
- Der generische Name in einem Makro DFHTST TYPE=REMOTE gibt Warteschlangen an, für die CICS die Anforderung für temporären Speicher an eine ferne CICS-Region oder einen TS-Server weiterleitet, es sei denn, der Name des fernen Systems (SYSIDNT) ist identisch mit dem Namen des lokalen CICS-Systems. Wenn in SYSIDNT der gleiche Name angegeben wird, der auch dem lokalen CICS-System zugeordnet wurde, dann werden die in der Option DATAID angegebenen Warteschlangen von CICS als lokale Warteschlangen eingestuft.
- Der generische Name, der im Makro DFHTST TYPE=LOCAL angegeben ist, identifiziert Warteschlangen als lokale Warteschlangen, die in der CICS-Region gespeichert sind, in der die TST installiert ist.
- Der generische Name, der im Makro DFHTST TYPE=SECURITY angegeben ist, identifiziert Warteschlangen, für die die Ressourcensicherheitsprüfung erforderlich ist.

Wenn Sie einen aus acht Zeichen bestehenden Namen angeben, definiert dies einen eindeutigen Namen für eine Warteschlange für temporären Speicher.

Wählen Sie eine Namenskonvention für Warteschlangennamen aus, die es Ihnen ermöglicht, viele Warteschlangen mit nur wenigen generischen Namen zu definieren. Dadurch kann der Aufwand für die TST-Definition erheblich reduziert werden. Berücksichtigen Sie hierbei, dass CICS in der TST nach dem ersten Präfix sucht, das die angegebenen Suchkriterien erfüllt. Beispiel: Wenn CICS nach der Warteschlange für temporären Speicher mit dem Namen ABCDEFGH sucht und die TST das Präfix A, gefolgt vom Präfix AB enthält, dann wird A ausgewählt. Um Probleme zu vermeiden, sollten Sie weniger generische Einträge in der TST vor den generischeren Einträgen definieren, sodass der erste gefundene Eintrag der am wenigsten generische Eintrag aller möglichen Übereinstimmungen ist.

Wenn CICS nach generischen Namen sucht, um diese mit dem Namen einer TS-Warteschlange abzugleichen, dann sucht das Programm nur die Eintragstypen, die in Bezug auf die jeweilige Suche von Interesse sind. CICS sucht die folgenden Einträge:

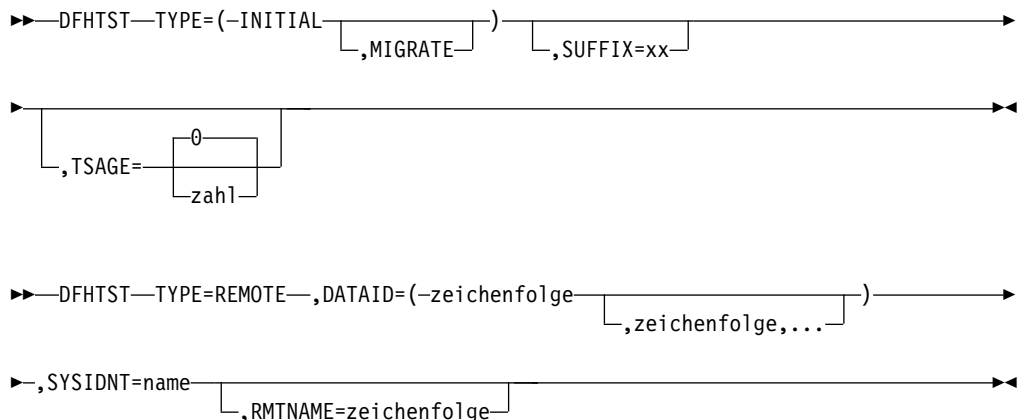
- Lokale *und* ferne Einträge, um zu ermitteln, ob es sich um eine ferne Warteschlange handelt. Auf diese Weise werden lokale und ferne Einträge als eine Suchkategorie eingestuft, wenn CICS einen Warteschlangennamen mit generischen Namen abgleicht.
- Wiederherstellungseinträge und ferne Einträge, wenn ermittelt werden soll, ob eine Warteschlange wiederherstellbar ist. Wenn die führenden Zeichen des Warteschlangennamens mit den generischen Namen von TYPE=RECOVERY **und** TYPE=REMOTE übereinstimmen, dann hat die Angabe für TYPE=REMOTE Vorrang und die Wiederherstellungsoption muss in der lokalen Region, in der die Warteschlange gespeichert ist, erneut definiert werden. (Warteschlangen in einem gemeinsam genutzten TS-Pool dürfen nicht wiederherstellbar sein.)
- Sicherheitseinträge werden nur gesucht, wenn ermittelt werden soll, ob für eine Warteschlange die Sicherheitsfunktion verwendet wird.

Verwenden Sie diese Makros zum Definieren der TST-Einträge:

- Programmabschnitt: DFHTST TYPE=INITIAL
- Wiederherstellbarer temporärer Speicher: DFHTST TYPE=RECOVERY
- Lokaler temporärer Speicher: DFHTST TYPE=LOCAL
- Ferner temporärer Speicher: DFHTST TYPE=REMOTE
- Sicherheitsprüfung für temporären Speicher: DFHTST TYPE=SECURITY
- Gemeinsame Datennutzung für temporären Speicher: DFHTST TYPE=SHARED
- Ende der Tabelle für temporären Speicher: DFHTST TYPE=FINAL (siehe hierzu TYPE=FINAL (end of table))

## Programmabschnitt: DFHTST TYPE=INITIAL

Der Einstiegspunkt und die Startadresse für die momentan definierte Tabelle für temporären Speicher (TST = Temporary Storage Table) werden mithilfe des Makros DFHTST TYPE=INITIAL eingerichtet.



Allgemeine Informationen zu Makros vom Typ TYPE=INITIAL einschließlich der Verwendung des Operanden SUFFIX finden Sie in TYPE=INITIAL (control section).

### MIGRATE

Geben Sie MIGRATE an, wenn Sie Ihre Tabelle für temporären Speicher (TST = Temporary Storage Table) für die Migration auf eine CSD-Datei assemblieren oder wenn Sie eine TST in Kombination mit TSMODEL-Ressourcendefinitionen verwenden. Wenn Sie mit dem Systeminitialisierungsparameter TST ein TST-Suffix angeben und wenn die TST mit der Option MIGRATE assembliert wird,

**GE={0..zahl}**  
Definiert den Grenzwert für die Prioritätssteuerung nach Verweildauer von Daten im temporären Speicher, die von der Domäne für temporären Speicher während eines Wiederanlaufs von CICS nach einem Systemabsturz verwendet werden. Daten, auf die während des angegebenen Intervalls nicht verwiesen wurde, werden nicht wiederhergestellt. Der Wert wird in Tagen angegeben. Der maximal zulässige Wert beträgt 512. Der Wert null gibt an, dass keine Daten bereinigt werden.

Im Makro DFHTST TYPE=RECOVERY werden die generischen Namen angegeben, die für Warteschlange für temporären Speicher verwendet werden, für die eine Wiederherstellung durchgeführt werden kann.

**TYPE=RECOVERY**

**Anmerkung:** TYPE=ENTRY wird aus Gründen der Kompatibilität mit früheren Releases weiterhin beibehalten und ist gleichbedeutend mit TYPE=RECOVERY.

Geben Sie diesen Operanden mit mindestens einem alphanumerischen TS-Warteschlangennamen an, der als wiederherstellbar festgelegt werden soll. Hierbei kann jeder Name bis zu acht Zeichen lang sein. (Informationen zu generischen Namen und den Kriterien für die Übereinstimmung finden Sie in „TST: Tabelle für temporären Speicher“ auf Seite 478.)

Jede Zeichenfolge kann einen generischen Warteschlangennamen oder einen eindeutigen TS-Warteschlangennamen darstellen. Generische Namen werden im ersten bis siebten Zeichen von TS-Warteschlangennamen angegeben. DATAIDs, die alle acht Zeichen verwenden, definieren eindeutige Warteschlangennamen.

Kapitel 5. Ressourcendefinitionsmakros 481

- **DF** bezieht sich auf Warteschlangen für temporären Speicher, die von der CICS-Intervallsteuerung für START-Befehle mit Daten verwendet werden, die jedoch keine REQID angeben.
  - **\*\*** bezieht sich auf Warteschlangen für temporären Speicher, die vom BMS-Befehl ROUTE und von den Befehlen verwendet werden, in denen der Operand PAGING verwendet wird.
  - **\$\$** bezieht sich auf Warteschlangen für temporären Speicher, die von der BMS CMSG-Transaktion verwendet werden, wenn die Option PROTECT=YES im Befehl START TRANSID angegeben ist.
- ( ) Dieser spezielle Operand (null) ohne einen Wert zwischen den Klammern wird von CICS als Angabe einer beliebigen Warteschlange interpretiert, die in anderen DATAIDs nicht expliziter angegeben ist.

#### Anmerkung:

1. Wenn eine TST ohne Einträge für TYPE=RECOVERY generiert wird, dann wird keine Wiederherstellung durchgeführt. Wenn ein Befehl EXEC CICS START mit einem der Parameter FROM, RTRANSID, RTERMID oder QUEUE angegeben wird und wenn keine REQID angegeben wurde, dann generiert CICS Anforderungskennungen, die mit dem Präfix DF beginnen. Wenn die Wiederherstellung für diese Anforderungen erforderlich ist, dann sollte die TST mit dem entsprechenden generischen Namen generiert werden.
2. Alle Warteschlangen für temporären Speicher, die für wiederanlauffähige Transaktionen (definiert mit RESTART=YES in der Transaktionsressourcendefinition) verwendet werden, sollten als wiederherstellbar definiert werden (einschließlich der Einheiten, die das Standardpräfix DF verwenden).
3. Nur Daten im Zusatzspeicher können als wiederherstellbar definiert werden. Daten, die in den Hauptspeicher geschrieben werden, sind nicht wiederherstellbar, und zwar unabhängig von eventuell angegebenen Wiederherstellungsoptionen.
4. Wenn eine Task Daten im temporären Speicher ändert, die als wiederherstellbar definiert sind, dann werden die Daten gegen die Änderungen durch eine gleichzeitig ausgeführte Task geschützt, indem eine Einreihung in die Warteschlange mit dem betreffenden Namen durchgeführt wird. Der Warteschlangennamen wird nicht aus der Warteschlange entfernt, bis die Task abgeschlossen ist oder bis eine Synchronisationspunktanforderung für die Task abgesetzt wird, um das Ende einer logischen Arbeitseinheit anzugeben. Zu diesem Zeitpunkt wird ein Protokolldatensatz in die Systemprotokolldatei geschrieben, um die benötigten externen Informationen für die Wiederherstellung der Daten bereitzustellen, wenn das System nachfolgend abnormal beendet wird.

## Beispiel

Dieses Makro DFHTST TYPE=RECOVERY definiert wiederherstellbare Warteschlangen für temporären Speicher:

```
DFHTST TYPE=RECOVERY,
      DATAID=(DF,**,
               $$(<zeichenfolge>)...)
```

- Mit der DATAID DF werden die Warteschlangen für temporären Speicher, die in CICS-Startanforderungen benutzt werden, als wiederherstellbar definiert.
- Mit den DATAIDs \*\* und \$\$ werden die standardmäßigen Warteschlangen für temporären Speicher, die von BMS benutzt werden, als wiederherstellbar definiert.



gibt das TST-Makro eine Meldung zu einem Assemblierungsfehler aus. Wenn Sie also DATAID=() in Einträgen der lokalen TS-Warteschlange angeben, dann müssen die TYPE=LOCAL-Makros nach allen TYPE=REMOTE-Makros angegeben werden.

## Ferner temporärer Speicher: DFHTST TYPE=REMOTE

Mit dem Makro DFHTST TYPE=REMOTE können Sie die Namen von Warteschlangen für temporären Speicher definieren, die in fernen CICS-Regionen gespeichert sind, wenn die CICS-Einrichtungen für die übergreifende Kommunikation verwendet werden.

Mit diesem Makro können Sie auch Warteschlangen definieren, die sich in einem Pool gemeinsam genutzter Warteschlangen befinden, der wie eine ferne Region behandelt wird. Eine Ausnahme bildet hierbei lediglich, dass der Name des fernen Systems mit dem Systemnamen in einem Makro DFHTST TYPE=SHARED übereinstimmt.

```

▶▶—DFHTST—TYPE=REMOTE—,DATAID=(—zeichenfolge—
                                   |,zeichenfolge,...|)————▶
▶—,SYSIDNT=name—
               |,RMTNAME=zeichenfolge|————▶▶

```

### TYPE=REMOTE

Gibt an, dass dieser TST-Eintrag eine Gruppe ferner Warteschlangen für temporären Speicher definiert, die entweder in einer fernen CICS-Region oder in einem gemeinsam genutzten TS-Pool in einer Coupling-Facility resident sein können.

### DATAID=(zeichenfolge[,zeichenfolge,...])|()

Geben Sie diesen Operanden mit mindestens einem alphanumerischen TS-Warteschlangennamen an. Hierbei können die Namen maximal acht Zeichen lang sein. Verwenden Sie eines bis sieben der führenden Zeichen der Namen der TS-Warteschlangen zur Bildung der generischen Namen der Warteschlangen, für die Anforderungen an eine ferne Region oder einen TS-Server weitergeleitet werden sollen. (Informationen zu generischen Namen und den Kriterien für die Übereinstimmung finden Sie in „TST: Tabelle für temporären Speicher“ auf Seite 478.)

**Anmerkung:** Wenn RMTNAME angegeben ist, dann darf die Listenform des Operanden DATAID nicht verwendet werden. Wenn Sie den Parameter RMTNAME angeben, dann hat DATAID folgende Syntax: DATAID=zeichenfolge.

### zeichenfolge

Jede Zeichenfolge kann einen generischen Warteschlangennamen oder einen eindeutigen TS-Warteschlangennamen darstellen. Gewöhnlich werden generische Namen im ersten bis siebten Zeichen von TS-Warteschlangennamen angegeben. Die generischen Namen werden von Anwendungsprogrammen in der Region verwendet, in der diese Tabelle für temporären Speicher (TST = Temporary Storage Table) installiert ist.

Werden mehrere Namen angegeben, dann müssen diese in Klammern eingeschlossen und durch ein Komma von den anderen Namen getrennt werden. Sie können die Klammern weglassen, wenn Sie nur einen Namen angeben.

( ) Dieser spezielle Operand ohne einen Wert zwischen den Klammern wird

von CICS als Angabe einer beliebigen Warteschlange interpretiert, die in anderen DATAIDs nicht expliziter angegeben ist. Dieser Operand kann wie folgt verwendet werden, um alle Warteschlangen abzudecken:

- Wenn die Warteschlangen mit Namen, die mit den Buchstaben L, M und N beginnen, sich auf einem lokalen System befinden und wenn sie in einem Makro TYPE=LOCAL mit passenden generischen DATAIDs angegeben sind, dann können Sie alle anderen Warteschlangen als fern definieren, indem Sie DATAID=() im Makro TYPE=REMOTE wie folgt angeben:

```

      DFHTST TYPE=LOCAL,      *
      DATAID=(L,M,N)
*
      DFHTST TYPE=REMOTE,    *
      DATAID=()
```

Die Option DATAID=() im Makro TYPE=REMOTE und die Angabe von DATAID=() im Makro TYPE=LOCAL schließen sich gegenseitig aus. Wird diese Angabe sowohl im lokalen als auch im fernen Eintrag definiert, dann gibt das TST-Makro eine Meldung zu einem Assemblierungsfehler aus.

DATAID=() muss der letzte Eintrag in einer Gruppe lokaler und ferner Einträge sein. Wenn Sie also DATAID=() in Einträgen der fernen TS-Warteschlange angeben, dann müssen die TYPE=REMOTE-Makros nach allen TYPE=LOCAL-Makros angegeben werden.

#### **SYSIDNT=name**

Gibt die Region oder den Server an, in der bzw. dem die fernen oder gemeinsam genutzten Warteschlangen für temporären Speicher gespeichert ist. Für eine ferne Warteschlange, deren Eigner eine andere CICS-Region ist, muss der aus vier Zeichen bestehende alphanumerische Name mit dem in der Option REMOTENAME festgelegten Wert, der in der CONNECTION-Definition angegeben ist, mit den ersten vier Zeichen des IPCONN-Namens in einer IPCONN-Definition oder mit dem SYSIDNT-Namen identisch sein, der in einem DFHTST TYPE=SHARED-Eintrag angegeben wurde.

Sie können diesen Parameter verwenden, um den Namen der lokalen Region anzugeben, in der die TST installiert ist. Wenn der Operand SYSIDNT mit dem Wert für SYSIDNT übereinstimmt, der im Systeminitialisierungsparameter angegeben wurde, dann werden die TS-Warteschlangen, die mit den DATAIDs übereinstimmen, als lokale Warteschlangen eingestuft.

#### **RMTNAME=zeichenfolge**

Geben Sie diesen Operanden mit dem aus einem bis acht Zeichen bestehenden Präfix an, das von CICS verwendet wird, um den entsprechenden Wert zu ersetzen, der im Operanden DATAID angegeben wird, wenn ein Verweis auf die Warteschlange für temporären Speicher an ein fernes System oder eine ferne Region übertragen wird. Dieser Operand gibt standardmäßig die Zeichenfolge an, die im Operanden DATAID festgelegt ist. Die Länge der in diesem Operanden angegebenen Zeichenfolge muss mit der Länge der Zeichenfolge im Operanden DATAID identisch sein. Dieser Mechanismus ermöglicht den Zugriff auf eine Warteschlange für temporären Speicher auf dem fernen System mit dem gleichen Namen wie der einer Warteschlange auf dem lokalen System.

## **Sicherheitsprüfung für temporären Speicher: DFHTST TYPE=SECURITY**

Das Makro DFHTST TYPE=SECURITY gibt an, dass die Sicherheitsprüfung für die in der TST (Temporary Storage Table; Tabelle für temporären Speicher) angegebenen Warteschlangen für temporären Speicher erforderlich ist.

►►—DFHTST—TYPE=SECURITY—,DATAID=(—zeichenfolge—  
,zeichenfolge,...)—►►

#### TYPE=SECURITY

Gibt an, dass dieser TST-Eintrag eine Gruppe von Warteschlangen für temporären Speicher definiert, für die die Sicherheitsprüfung erforderlich ist.

**DATAID=(zeichenfolge[,zeichenfolge,...])|()**

Geben Sie diesen Operanden mit mindestens einem alphanumerischen TS-Warteschlangennamen an. Hierbei können die Namen maximal acht Zeichen lang sein. Verwenden Sie eines bis sieben der führenden Zeichen eines Warteschlangennamens, um generische Namen dieser Warteschlangen zu bilden, die der Sicherheitsprüfung unterliegen. (Informationen zu generischen Namen und den Kriterien für die Übereinstimmung finden Sie in „TST: Tabelle für temporären Speicher“ auf Seite 478.)

#### Anmerkung:

1. Wird dieses Makro verwendet, dann muss ein passendes Profil für den externen Sicherheitsmanager definiert werden, um den Zugriff auf die Warteschlange für temporären Speicher (TSQ = Temporary Storage Queue) zu steuern. Andernfalls hat das Makro nicht den gewünschten Effekt. Informationen zu Profilen finden Sie in Refreshing resource profiles in main storage.
2. Der vollständige TSQ-Name wird an den Sicherheitsmanager übergeben.

#### zeichenfolge

Jede Zeichenfolge kann einen generischen Warteschlangennamen oder einen eindeutigen TS-Warteschlangennamen darstellen. Gewöhnlich werden generische Namen im ersten bis siebten Zeichen von TS-Warteschlangennamen angegeben. Die generischen Namen werden von Anwendungsprogrammen in der Region verwendet, in der diese Tabelle für temporären Speicher (TST = Temporary Storage Table) installiert ist.

Werden mehrere Namen angegeben, dann müssen diese in Klammern eingeschlossen und durch ein Komma von den anderen Namen getrennt werden. Sie können die Klammern weglassen, wenn Sie nur einen Namen angeben.

- ( ) Dieser Nulloperand ohne einen Wert zwischen den Klammern wird von CICS als Angabe einer beliebigen Warteschlange interpretiert, die in anderen DATAIDs nicht expliziter angegeben ist.

## Gemeinsame Datennutzung für temporären Speicher: DFHTST TYPE=SHARED

Das Makro DFHTST TYPE=SHARED gibt den Namen des fernen Systems an, anhand dessen CICS einen Pool für temporären Speicher in der Coupling-Facility identifiziert.

►►—DFHTST—TYPE=SHARED—,SYSIDNT=systemname—,POOL=poolname—►►

#### TYPE=SHARED

Gibt an, dass dieser TST-Eintrag eine Zuordnung zwischen einer System-ID (SYSIDNT), die in einem TYPE=REMOTE-Eintrag angegeben ist, und einem Pool mit TS-Warteschlangen mit gemeinsamer Datennutzung definiert.



**SYSIDNT=systemname**

Gibt den aus einem bis vier Zeichen bestehenden Systemnamen an, der dem TS-Poolnamen entspricht.

CICS verwendet diese SYSIDNT für die Zuordnung ferner Warteschlangen (im Eintrag TYPE=REMOTE oder in einer expliziten SYSID eines API-Befehls definiert) zu einem TS-Server. Dabei wird wie folgt vorgegangen:

- Wenn in einem API-Befehl für temporären Speicher eine ferne Warteschlange explizit (in der Option SYSID) angegeben wird, dann ordnet CICS die SYSID einer passenden SYSIDNT in einem TYPE=SHARED-Eintrag zu:
  - Wenn eine übereinstimmende SYSIDNT gefunden wird, dann verwendet CICS den entsprechenden POOL-Namen, um den TS-Server anzugeben, der die gemeinsam genutzte TS-Warteschlange verwaltet.
  - Wenn die SYSID mit keinem der TYPE=SHARED-Einträge übereinstimmt, dann wird die Anforderung über eine Funktionsverlagerung an die ferne Warteschlangenverwaltungsregion (QOR = Queue-Owning Region) übertragen, die in SYSID angegeben ist.
- Wenn ein API-Befehl für temporären Speicher auf eine ferne Warteschlange verweist, die in einem TYPE=REMOTE-Eintrag angegeben ist, dann überprüft CICS das System auf eine übereinstimmende SYSIDNT in den TYPE=SHARED-Einträgen:
  - Wenn ein TYPE=SHARED-Eintrag mit einer übereinstimmenden SYSIDNT gefunden wird, dann verwendet CICS den entsprechenden POOL-Namen, um den TS-Server anzugeben, der die gemeinsam genutzte TS-Warteschlange verwaltet.
  - Wenn kein TYPE=SHARED-Eintrag gefunden wird, dann handelt es sich um eine ferne Warteschlange und die Anforderung wird über eine Funktionsverlagerung an die Warteschlangenverwaltungsregion übertragen.

Sie können mehrere TYPE=SHARED-Einträge mit unterschiedlichen SYSIDNT-Namen erstellen, die auf den gleichen POOL-Namen verweisen. In diesem Fall beziehen sich Verweise auf den gleichen Warteschlangennamen unabhängig davon, welche SYSID in der API verwendet wird, auf diesen gleichen Warteschlangennamen.

**POOL=poolname**

Gibt den aus einem bis acht Zeichen bestehenden Namen des Pools mit den TS-Warteschlangen an, der in TS-Anforderungen verwendet wird, die implizit oder explizit den entsprechenden Systemnamen angeben. Der Poolname muss mit dem Namen übereinstimmen, der im Parameter POOL des TS-Servers angegeben ist, mit dem der TS-Pool verwaltet wird.

## DFHTST-Beispiel

Ein Beispiel für die Angabe einer Tabelle für temporären Speicher (TST = Temporary Storage Table).

```

        DFHTST TYPE=INITIAL,          LIST OF GENERIC NAMES OF QUEUES *
        SUFFIX=01                     THAT ARE RECOVERABLE, REMOTE,
*                                     SHARED, LOCAL, OR REQUIRE
*                                     SECURITY CHECKING.
*
* The following macro specifies that all LOCAL queues with
* names beginning with the letter 'R' are RECOVERABLE:
*
        DFHTST TYPE=RECOVERY,          *
        DATAID=R
*
* The following macro specifies that queues with names
* beginning with C,D,E, and X are local queues:
*
        DFHTST TYPE=LOCAL,             *
        DATAID=(C,D,E,X)
*
* The following macro specifies that queues with names
* beginning with AB,L,M,N are remote queues on system RSYS:
*
        DFHTST TYPE=REMOTE,            *
        DATAID=(AB,L,M,N),            *
        SYSIDNT=RSYS,                  *
*
* The next macro specifies that all queues not local as defined
* above, or remote in system RSYS as defined above, are remote
* queues that reside in a shared TS pool TYPE=SHARE macro.
*
        DFHTST TYPE=REMOTE,            *
        DATAID=(),                    *
        SYSIDNT=SHR1
*
* The next macro specifies that remote queues with SYSIDNT=SHR1
* are mapped to shared TS pool named TSQSHR1.
*
        DFHTST TYPE=SHARED,            *
        SYSIDNT=SHR1,                  *
        POOL=TSQSHR1
*
* The following macro specifies that queues with names
* beginning with SAQ require security checking.
* Note that the full TS queue name is passed to the ESM.
*
        DFHTST TYPE=SECURITY,          *
        DATAID=SAQ
*
        DFHTST TYPE=FINAL
        END

```

Abbildung 17. Tabelle für temporären Speicher: Beispiel

---

## XLT: Transaktionslistentabelle

Die Transaktionslistentabelle gibt Transaktionen, die während des Systemabschlusses über Terminals eingeleitet werden können, und Gruppen von Transaktionen an, die zusammen aktiviert oder inaktiviert werden sollen. Verwenden Sie eine TRANSACTION-Ressource als Vorgabe für XLT.

Die Transaktionslistentabelle (XLT = Transaction List Table) kann zum Definieren der folgenden Komponenten verwendet werden:

- Eine Liste mit Transaktionskennungen, die während der ersten Wartephase des Systemabschlusses über Terminals eingeleitet werden können. Wenn keine PLT-

Programme zur Ausführung verfügbar sind, dann kann die erste Wartezeit kurz sein, sodass wenig Zeit zum Aufruf eines XLT-Programms verbleibt, bevor die zweite Wartephase eingeleitet wird. Sie geben das Suffix der Tabelle an, die im Systeminitialisierungsparameter XLT verwendet wird. Der Hauptterminalbediener kann das Suffix beim Systemabschluss ändern, indem er die Option XLT des Befehls **CEMT PERFORM SHUTDOWN** verwendet.

**Anmerkung:** Alternativ hierzu können Sie auch eine PROGRAM-Ressource erstellen, um die Transaktionslistentabelle zu definieren. Die Definition als Programm bedeutet auch, dass die automatische Installation durchgeführt werden kann. Informationen zur automatischen Installation von Programmen finden Sie in *Autoinstalling programs, map sets, and partition sets in Configuring*.

- Eine Gruppe von Transaktionskennungen, die über das Hauptterminal inaktiviert oder aktiviert werden soll. Der Hauptterminalbediener gibt das Suffix der Tabelle an, die verwendet werden soll. Dazu wird die Option CLASS von CEMT SET TRANSACTION verwendet.

In Abb. 18 auf Seite 491 ist die zur Erstellung einer Transaktionslistentabelle erforderliche Codierung dargestellt.

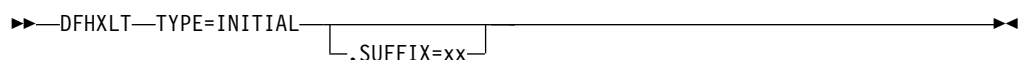
Zum Definieren der XLT-Einträge sind die folgenden Makros verfügbar:

- Programmabschnitt: DFHXLT TYPE=INITIAL
- Einträge in Transaktionslistentabelle: DFHXLT TYPE=ENTRY
- Ende der Transaktionslistentabelle: DFHXLT TYPE=FINAL (siehe hierzu TYPE=FINAL (end of table))

## Programmabschnitt: DFHXLT TYPE=INITIAL

Mit dem Makro DFHXLT TYPE=INITIAL wird der Einstiegspunkt und die Startadresse der zu definierenden XLT (Transaction List Table; Transaktionslistentabelle) eingerichtet.

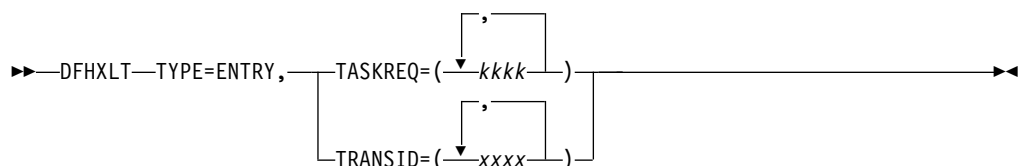
Mit dem Makro DFHXLT TYPE=INITIAL wird der Einstiegspunkt und die Startadresse der zu definierenden XLT (Transaction List Table; Transaktionslistentabelle) eingerichtet.



Allgemeine Informationen zu den TYPE=INITIAL-Makros sowie zur Verwendung des Operanden SUFFIX finden Sie in TYPE=INITIAL (control section).

## Einträge in Transaktionslistentabelle: DFHXLT TYPE=ENTRY

Im Makro DFHXLT TYPE=ENTRY wird eine Liste der Transaktionskennungen angegeben, die während der ersten Wartephase des Systemabschlusses über Terminals eingeleitet werden können.



**TYPE=ENTRY**

Geben Sie diese Option an, wenn einzelne oder mehrere Einträge in der XLT (Transaction List Table; Transaktionslistentabelle) generiert werden sollen.

**TASKREQ=(kkkk[,kkkk],...)**

Die Variable *kkkk* kann für eine der folgenden Angaben stehen:

- Die Tasten PA1 bis PA3 und PF1 bis PF24 geben eine der 3270-Sondertasten an, die zum Einleiten einer Task verwendet werden können.
- Eine LPA (Light Pen Attention; Lichtstiftunterbrechung) gibt an, dass eine Transaktion eingeleitet werden soll, wenn ein lichtstiftempfindliches Feld ausgewählt wird.
- Die OPID (Operator Identification Card Reader; Leser für Operator-ID-Karten) gibt an, dass eine Transaktion eingeleitet wird, nachdem der entsprechende Identitätsnachweis eines Bedieners (Operators) eingelesen wurde.
- MSRE gibt an, dass Transaktionen eingeleitet werden, wenn der 10/63-Zeichen-Magnetcodeleser verwendet wird.

Definieren Sie jeden TASKREQ-Wert in der CSD-Datei und installieren Sie ihn auf dem aktiven System. (Weitere Informationen zu diesem Thema finden Sie in der Beschreibung des Attributs TASKREQ in TRANSACTION-Attribute.)

**TRANSID=(xxxx[,xxxx],...)**

Stellt einen aus einem bis vier Zeichen bestehenden Transaktionscode dar. Definieren Sie jeden TRANSID-Wert in der CSD-Datei und installieren Sie ihn auf dem aktiven System. (Weitere Informationen zu diesem Thema finden Sie in der Beschreibung des Attributs TRANSACTION in TRANSACTION-Attribute.)

Wenn die TRANSID ein Sonderzeichen (z. B. ein Komma) enthält, dann darf die Anweisung TYPE=ENTRY nur eine TRANSID enthalten, die von Anführungszeichen als Begrenzer eingeschlossen wird.

**Anmerkung:** TASKREQ und TRANSID sind Parameter, die sich gegenseitig ausschließen.

## DFHXLT-Beispiel

Ein Beispiel für die Angabe einer Transaktionslistentabelle (XLT = Transaction List Table).

<pre> DFHXLT TYPE=INITIAL,       SUFFIX=IN * * * DFHXLT TYPE=ENTRY,TASKREQ=PF5 * * * * DFHXLT TYPE=ENTRY,TRANSID=(USR1,USR2) DFHXLT TYPE=ENTRY,TRANSID='AA,1' DFHXLT TYPE=ENTRY,TRANSID='AA,2' DFHXLT TYPE=FINAL END  DFHXLT TYPE=INITIAL,       SUFFIX=G1 * * DFHXLT TYPE=ENTRY,TRANSID=(TSSA,TSRA) DFHXLT TYPE=ENTRY,TRANSID=(TDSA,TDRA) DFHXLT TYPE=ENTRY,TRANSID=ICSA DFHXLT TYPE=FINAL END </pre>	<pre> LIST OF TRANSACTIONS      * THAT ARE ACCEPTED DURING THE FIRST QUIESCE PHASE OF SYSTEM TERMINATION. (TASKREQ MUST ALSO BE DEFINED IN THE CSD AND INSTALLED IN THE RUNNING CICS SYSTEM. AN ENTRY FOR THE XLT MUST BE MADE IN THE CSD.)  LIST OF LOGICALLY RELATED* TRANSIDS TO BE ENABLED OR DISABLED BY MASTER TERMINAL. (TRANSIDS MUST ALSO BE DEFINED IN THE CSD AND INSTALLED IN THE RUNNING CICS SYSTEM.) </pre>
--	--

Abbildung 18. Transaktionslistentabelle: Beispiel



---

## Kapitel 6. Zur Wahrung der Kompatibilität beibehaltene veraltete Attribute

Bestimmte Ressourcendefinitionsattribute gelten für die in CICS Transaction Server for z/OS, Version 5 Release 5 verwendeten Ressourcen nicht, werden jedoch unterstützt, um die CSD-Kompatibilität mit früheren Releases von CICS zu wahren, in denen sie weiterhin gültig sind.

Informationen zum Kompatibilitätsmodus finden Sie in *Sharing the CSD between different releases of CICS*.

Die folgenden Attribute gelten nicht für Ressourcen, die in diesem CICS-Release verwendet werden. Überprüfen Sie die Dokumentation zu den früheren CICS-Releases, die Sie verwenden, um festzustellen, ob das Attribut in diesem Release unterstützt wird.

*Tabelle 21. Veraltete Ressourcendefinitionsattribute*

Ressource	Attribut	Beschreibung
CONNECTION	BINDPASSWORD	Sicherheitskennwort für APPC-Bindung
FILE	RESSECCNUM	Wert für Ressourcensicherheit
FILE	PASSWORD	Dateizugriffsprüfung
LSRPOOL	LSRPOOLID	Numerische Kennung 1-8 für Pool
MAPSET	RSL	Wert für Ressourcensicherheit
PARTITIONSET	RSL	Wert für Ressourcensicherheit
PROFILE	PROTECT	Nachrichtenintegrität und Nachrichtenprotokollierung
PROGRAM	HOTPOOL	Java-Hotpooling
PROGRAM	JVMPROFILE	JVM-Profil für JVMs im Pool
PROGRAM	RSL	Wert für Ressourcensicherheit
SESSIONS	INSERVICE	Sitzung kann verwendet werden
SESSIONS	OPERID	Aus drei Zeichen bestehende Bediener-ID
SESSIONS	OPERPRIORITY	Bedienerprioritätscode
SESSIONS	OPERRSL	Ressourcensicherheitsschlüssel
SESSIONS	OPERSECURITY	Voreingestellte Transaktionssicherheitsschlüssel
SESSIONS	RECOVNOTIFY	Benachrichtigung zur XRF-Übernahme
SESSIONS	TRANSACTION	Nur angegebene Transaktion zulassen
TCPIPSERVICE	DNSGROUP	Gibt den Gruppennamen an, mit dem CICS die Registrierung beim Workload Manager durchführt
TCPIPSERVICE	GRPCRITICAL	Markiert den Service als kritisches Mitglied der DNS-Gruppe
TCPIPSERVICE	PRIVACY	Erforderliche Stufe der SSL-Verschlüsselung
TCPIPSERVICE	TSQPREFIX	Präfix der Warteschlange für temporären Speicher

Tabelle 21. Veraltete Ressourcendefinitionsattribute (Forts.)

Ressource	Attribut	Beschreibung
TERMINAL	BINDPASSWORD	Sicherheitskennwort für APPC-Bindung
TERMINAL	CONSOLE	Konsolidifikationsnummer
TERMINAL	OPERID	Aus drei Zeichen bestehende Bediener-ID
TERMINAL	OPERPRIORITY	Bedienerprioritätscode
TERMINAL	OPERRSL	Ressourcensicherheitsschlüssel
TERMINAL	OPERSECURITY	Voreingestellte Transaktionssicherheitsschlüssel
TERMINAL	XRFSIGNOFF	Abmeldung nach XRF-Übernahme erzwingen
TRANSACTION	EXTSEC	Externen Sicherheitsmanager verwenden
TRANSACTION	INDOUBT	Aktion bei Abbruch
TRANSACTION	PRIMEDSIZE	Vorbereitete Speicherzuordnungsgröße
TRANSACTION	RSL	Wert für Ressourcensicherheit
TRANSACTION	TCLASS	Zugeordnete Klasse der Task
TRANSACTION	TRANSEC	Wert für Transaktionssicherheit
TYPETERM	LOGMODECOM	LOGMODE-Kompatibilität
TYPETERM	XRFSIGNOFF	Abmeldung nach XRF-Übernahme erzwingen



---

## Kapitel 7. Von CICS bereitgestellte Ressourcendefinitionen, Gruppen und Listen

IBM stellt Definitionen für Ressourcen zur Verfügung, die in Ihrer CICS-Region installiert werden müssen. Außerdem werden Definitionen für Ressourcen bereitgestellt, die von den Beispielanwendungsprogrammen benutzt werden.

Detaillierte Informationen zu den von CICS bereitgestellten Transaktionen finden Sie in List of CICS transactions.

Die CSD-Datei wird mit dem DFHCSDUP-Befehl INITIALIZE initialisiert. Nach Abschluss der Initialisierung enthält die CSD-Datei zwei Kategorien von Ressourcendefinitionsgruppen:

1. In DFHLIST aufgeführte Gruppen, die für die Verwendung von RDO (Resource Definition Online; Onlineressourcendefinition) und anderer von CICS bereitgestellter Transaktionen von grundlegender Bedeutung sind.
2. Gruppen von Definitionen für die Beispielanwendungsprogramme. Detaillierte Informationen hierzu finden Sie in „Gruppen von Beispielanwendungsprogrammen“ auf Seite 507.

Die von IBM bereitgestellten CICS-Transaktionen, TYPETERM-Definitionen, TERMINAL-Modelldefinitionen und PROFILE-Definitionen sind in vier Gruppen untergliedert:

- DFHTYPE: TYPETERM-Definitionen
- DFHTERM: TERMINAL-Modelldefinitionen für die automatische Installation
- DFHISC: PROFILE-Definitionen für ISC-Sitzungen (ISC = Intersystem Communication; systemübergreifende Kommunikation)
- DFHSTAND: PROFILE-Definitionen

---

### DFHLIST-Definitionen

DFHLIST enthält Definitionen für Ressourcen, die erforderlich sind.

*Tabelle 22. DFHLIST-Ressourcendefinitionen*

Gruppenname	Beschreibung	Programme	Transaktionen	Sonstige Ressourcen
DFH\$SOT	Bereitgestellte TCPIPSERVICE-Ressourcen			TCP/IP-Services: ECI HTTPNSSL HTPSSL
DFHBMS	Basic Mapping Support	DFHTPQ DFHTPR DFHTPS	CSPG CSPQ CSPS	
DFHBR	Bridgeprogramme	DFHL3270 DFHBRMP DFHBRCV		

Tabelle 22. DFHLIST-Ressourcendefinitionen (Forts.)

Gruppenname	Beschreibung	Programme	Transaktionen	Sonstige Ressourcen
<b>DFHCBTS</b>	Lokale Anforderungswarteschlangendatei für BTS <b>Anmerkung:</b> Diese Gruppe ist nicht durch eine Sperre geschützt. Die Definitionen, die darin enthalten sind, können bei Bedarf geändert werden.			<b>Datei:</b> DFHLRQ
<b>DFHCFC</b>	Für CICS C++-Basisklassen benötigte Programme	ICCFCDLL		
<b>DFHCLNT</b>	CTIN und CCIN für CICS-Client	DFHZCT1 DFHZCN1	CTIN CCIN	<b>Transaktionsklasse:</b> DFHCOMCL
<b>DFHCONS</b>	In CPU-Konsole schreiben	DFHCWTO	CWTO	
<b>DFHDBCTL</b>	DBCTL-Transaktionen	DFHDBAT DFHDBCON DFHDBCT DFHDBDI DFHDBDSC DFHDBIQ DFHDBME DFHDBMP DFHDBUEX	CDBC CDBD CDBI CDBM CDBN CDBO CDBT	<b>Datei:</b> DFHDBFK  <b>Maskengruppen:</b>  DFHDBIE DFHDBNE
<b>DFHDB2</b>	Db2-Unterstützung	DFHD2CM0 DFHD2CM1 DFHD2CM2 DFHD2CM3 DFHD2EDF DFHD2EX1 DFHD2EX2 DFHD2INI DFHD2PXT DFHD2SPS DSNCUEXT DSNTIAC DSNTIA1	CDBF CDBQ CEX2 DSNC	

Tabelle 22. DFHLIST-Ressourcendefinitionen (Forts.)

Gruppenname	Beschreibung	Programme	Transaktionen	Sonstige Ressourcen
DFHDCTG	<p>Warteschlangen für transiente Daten für CICS-Basisfunktionen</p> <p><b>Anmerkung:</b> Diese Gruppe ist nicht durch eine Sperre geschützt. Die Definitionen, die darin enthalten sind, können bei Bedarf geändert werden.</p>			<p><b>Warteschlangen für transiente Daten:</b></p> <p>CADL CADO CADS CAIL CCPI CCSE CCSO CCZM CDB2 CDBC CDEP CDUL CECO CEJL CEPO CESE CESO CIEO CIIL CISL CISO CJRM CKQQ CMIG CMPO CMQM CMLO CPIO CRDI CRLO CRPO CSBA CSBR CSCC CSCS CSDH CSDL CSFL CSJE CSJO CSKL CSLB CSML CSMT CSNE CSOO CSPL CSQL CSRL CSSH CSSL CSTL CSZL CSZX CWBO CWBW</p>

Tabelle 22. DFHLIST-Ressourcendefinitionen (Forts.)

Gruppenname	Beschreibung	Programme	Transaktionen	Sonstige Ressourcen
DFHDOC	Leser für CICS-Dokumenthandlerschablonen	DFHDHEI		
DFHDP	3270-Schnittstelle des Profilmanagers für das Anwendungsdebugging	DFHDPLU DFHDPIN DFHDPCP	CADP CIDP	<b>Maskengruppe:</b> DFHDPMS
DFHDPWB	Webschnittstelle des Profilmanagers für das Anwendungsdebugging	DFHDPWB DFHDPWM0 DFHDPWM1 DFHDPWM2 DFHDPWM3 DFHDPWM4 DFHDPWM5 DFHDPWM6 DFHDPWT0 DFHDPWT1 DFHDPWT2 DFHDPWT3 DFHDPWT4 DFHDPWT5 DFHDPWT6 DFHDPWF0		
DFHEDF	Execution Diagnostic Facility	DFHDBMS DFHDBTI DFHEDFBR DFHEDFD DFHEDFP DFHEDFR DFHEDFX DFHEIGDS DFHEITAB	CEBR CEDF CEDG CEDX CEDY	<b>Maskengruppe:</b> DFHEDFM  <b>Transaktionsklasse:</b>  DFHEDFTC DFHEDFTO
DFHEDP	EXEC-DLI-Benutzerexit	DFHEDP		
DFHEP	Ereignisverarbeitung	DFHECEAH DFHECEAM DFHECEAS DFHECEAT	CEPH CEPQ CEPS CEPT	<b>Profil:</b> DFHECEPH
DFHFCRL	Datei im Bundle inaktivieren	DFHFCRN	CFCR	
DFHFE	FE-Terminaltestfunktion	DFHFEP DFHTRAP	CSFE	
DFHFEPI	Front-End-Programmierschnittstelle	DFHEITSZ DFHSZRMP	CSZI	
DFHHARDC	3270-Drucker - z/OS Communications Server	DFHP3270	CSPP	
DFHINDT	Testtool für unbestätigte Fehler	DFHINDAP DFHINDT DFHINTRU	CIND	<b>Transaktionsklasse:</b> DFHTCIND
DFHINQUI	Befehlsdefinition	DFHEITBS		
DFHINTER	Befehlsinterpreter	DFHECID DFHECIP DFHECSP	CECI CECS	
DFHIPECI	ECI over TCP/IP	DFHIEP	CIEP	

Tabelle 22. DFHLIST-Ressourcendefinitionen (Forts.)

Gruppenname	Beschreibung	Programme	Transaktionen	Sonstige Ressourcen
<b>DFHISC</b>	Intersystem Communication (ISC = systemübergreifende Kommunikation)	DFHCCNV DFHCHS DFHCLS3 DFHCLS4 DFHCLS5 DFHCNV DFHCRNP DFHCRQ DFHCRR DFHCRS DFHCRSP DFHCRT DFHDFST DFHDSRP DFHDYP DFHLUP DFHMIRS DFHMXP DFHRTC DFHRTE DFHSHRRP DFHSHRSP DFHUCNV DFHZLS1	CDFS CEHP CEHS CLQ2 CLR2 CLS1 CLS2 CLS3 CLS4 CMPX CPMI CQPI CQPO CRSQ CRSR CRTE CRTX CSHR CSMI CSM1 CSM2 CSM3 CSM5 CSNC CSSF CVM1 CXRT	<b>Profile:</b> DFHCICSF DFHCICSR DFHCICSS
<b>DFHISCIP</b>	IP-Interkonnektivitätsunterstützung	DFHCIS4 DFHISAIP DFHISCIP DFHISCOP DFHISDIP DFHISEMP DFHISLQP DFHISIP DFHISPHP DFHISPRP DFHISREU DFHISREX DFHISRRP DFHISRSP	CISB CISC CISC CISE CISM CISP CISR CISQ CISS CIST CISU CISX CIS1 CIS4	<b>Profil:</b> DFHCICSC <b>TS-Modell:</b> DFHISLQ
<b>DFHJAVA</b>	Für Java-Unterstützung erforderliche Programme	DFHSJJI DFHSJITL DFHSJTHP	CJSA CJSR CJSU	
<b>DFHLGMOD</b>	CICS-Protokollmanager			<b>Journalmodelle:</b> DFHLOG DFHSHUNT DFHLGLOG
<b>DFHLGQC</b>	Stilllegung des CICS-Protokollmanagers	DFHLGQC	CSQC	
<b>DFHMISC</b>	Sonstige Programme	DFHLETRU DFHPEP DFHREST		

Tabelle 22. DFHLIST-Ressourcendefinitionen (Forts.)

Gruppenname	Beschreibung	Programme	Transaktionen	Sonstige Ressourcen
DFHMQ	Für WebSphere MQ-Unterstützung erforderliche Programme, Transaktionen und Maskengruppen	CSQCAPX CSQAVICM CSQCBDCI CSQCBE30 CSQCBP10 CSQCBP53 CSQCBP00 CSQCBP10 CSQCBR00 CSQCBR53 CSQCCB CSQCCLOS CSQCCONN CSQCCONX CSQCCTL CSQCDISC CSQCDSC CSQCDSP CSQCQCON CSQCSSQ CSQCGET CSQCINQ CSQCOPEN CSQCPUT CSQCPUT1 CSQCRST CSQCSET CSQCSSQ CSQCSUB CSQCSUBR CSQFCTAB DFHMQBAS DFHMQBP0 DFHMQBP1 DFHMQBP2 DFHMQBP3 DFHMQBR0 DFHMQCOD DFHMQCON DFHMQCTL DFHMQDCI DFHMQDIS DFHMQDSC DFHMQDSL DFHMQMON DFHMQPLT DFHMQPOP DFHMQPRM DFHMQPUL DFHMQQC DFHMQRET DFHMQRS DFHMQSSQ DFHMQTRU DFHMQTSK IMQB23IC IMQS23IC	CKAM CKBC CKBM CKBP CKBR CKCN CKDL CKDP CKQC CKRS CKRT CKSD CKSQ CKTI	<b>Maskengruppen:</b>  DFHMQHC DFHMQHE DFHMQHK DFHMQHU DFHMQ1C DFHMQ1E DFHMQ1K DFHMQ1U DFHMQ2C DFHMQ2E DFHMQ2K DFHMQ2U

Tabelle 22. DFHLIST-Ressourcendefinitionen (Forts.)

Gruppenname	Beschreibung	Programme	Transaktionen	Sonstige Ressourcen
<b>DFHMSWIT</b>	Nachrichtenverteilungsprogramm	DFHMSP	CMSG	
<b>DFHOPCLS</b>	Programm zum dynamischen Öffnen und Schließen	DFHFCU	CSFU	
<b>DFHOPER</b>	Bedienerprogramme	DFHCEMNA DFHCEMNB DFHCEMNC DFHCEMND DFHCETRA DFHCETRB DFHCETRC DFHCETRD DFHCETRF DFHECBAM DFHEITMT DFHEITOT DFHEITST DFHEMTA DFHEMTD DFHEMTP DFHEOTP DFHESTP DFHLDMAP DFHSOCRL	CBAM CCRL CEMN CEMT CEOT CEST CETR CLDM	<b>Maskengruppen:</b>  DFHSO1M DFHCMNH DFHCMNM DFHCTRH DFHCTRM
<b>DFHOTS</b>	OTS-Resynchronisation	DFHOTR	CJTR	
<b>DFHPGAIP</b>	Automatische Installation für Programme	DFHPGADX DFHPGAHX DFHPGALX DFHPGAOX DFHPGAPG		<b>Maskengruppe:</b> DFHPGAMP  <b>Partitionsgruppe:</b>  DFHPGAPT

Tabelle 22. DFHLIST-Ressourcendefinitionen (Forts.)

Gruppenname	Beschreibung	Programme	Transaktionen	Sonstige Ressourcen
<b>DFHPIPE</b>	Web-Service-Verarbeitung	DFHJSON DFHMLBSJ DFHMLBST DFHPIAP DFHPIDSH DFHPIDSQ DFHPIEP DFHPIIR DFHPILSQ DFHPIR DFHPIRT DFHPISN1 DFHPISN2 DFHPITE DFHPITP DFHPITQ1 DFHWSADH DFHWSSE1 DFHWSXXX C128N IOSTREAM IXM4C57 IXMI33UC IXMI33DA IXMI33IN	CPIA CPIH CPIL CPIS	
<b>DFHPIVAL</b>	Prüfprogramm für SOAP-Nachrichten <b>Anmerkung:</b> Diese Gruppe ist nicht durch eine Sperre geschützt. Die Definitionen, die darin enthalten sind, können bei Bedarf geändert werden.	DFHPIVAL		
<b>DFHPSSGN</b>	Beibehalten von Anmeldungen für persistente Sitzungen	DFHZSGN DFHZPCT DFHZRPT	CPSS CPCT CRTP	
<b>DFHRL</b>	Verwalten des Lebenszyklus von Ressourcen	DFHRLMF DFHRLR DFHRLSC DFHRLVC	CRLR	
<b>DFHRMI</b>	Ressourcenmanagerschnittstelle	DFHRMSY	CRSY	
<b>DFHRQS</b>	Verknüpfungsprogramm für Anforderungsdatenströme	DFHRZJN		
<b>DFHRS</b>	Regionsstatus, Dateierstellung	DFHRSFDL		
<b>DFHRSEND</b>	z/OS Communications Server-Programm für erneutes Senden	DFHZRSP	CSRS	
<b>DFHSDAP</b>	SHUTDOWN ASSIST	DFHCESD	CESD	
<b>DFHSIGN</b>	An- und Abmeldeprogramme und Tabelle	DFHCESC DFHCEGN DFHSFP DFHSNP	CESC CEGN CESF CESN CESL	<b>Maskengruppen:</b>  DFHSNLE DFHSNPE DFHSNSE
<b>DFHSO</b>	Emulieren der CICS-Socketschnittstelle	DFHSOCI DFHSOLI		



Tabelle 22. DFHLIST-Ressourcendefinitionen (Forts.)

Gruppenname	Beschreibung	Programme	Transaktionen	Sonstige Ressourcen
<b>DFHSPI</b>	Onlineressourcendefinition	DFHAMP DFHDMP DFHEDAD DFHEDAP DFHEITSP DFHPUP DFHTBS DFHTOR DFHZATA DFHZATD DFHZATDX DFHZATMD DFHZATMF DFHZATR DFHZATS DFHZCQ DFHZCTDX DFHZDTDX DFHZPTDX	CATA CATD CATR CATS CDTS CEDA CEDA CEDB CEDC CFTS CITS CMTS CRMD CRMF	
<b>DFHSTAND</b>	CICS-Standardanwendungsprogramme	DFHACP DFHCXCU DFHPIITL DFHPIPA DFHPIXC DFHPSIP DFHQRY DFHSTP DFHTACP DFHTEP DFHTEPT DFHTFP DFHZXCU DFHZXRE DFHZXST	CPIR CQRY CSAC CSTE CXCX CXRE	<b>Profile:</b> DFHCICSA DFHCICSE DFHCICSP DFHCICST DFHCICSV DFHPPF01 DFHPPF02  <b>Maskengruppe:</b>  DFHXMSG
<b>DFHTCL</b>	TRANCLASS-Definitionen für Kompatibilität			<b>Transaktionsklassen:</b> DFHTCL00 DFHTCL00 DFHTCL01 DFHTCL03 DFHTCL04 DFHTCL05 DFHTCL06 DFHTCL07 DFHTCL08 DFHTCL09 DFHTCL10 DFHTCLQ2 DFHTSDEL

Tabelle 22. DFHLIST-Ressourcendefinitionen (Forts.)

Gruppenname	Beschreibung	Programme	Transaktionen	Sonstige Ressourcen
<b>DFHTERM</b>	TERMINAL-Modelldefinitionen			<b>Terminals:</b> LU2 LU3 APPC SCSP 3270 3284 L0E2 L0M2 L0M3 L0M4 L0M5 L2E2 L2M2 L2E3 L2M3 L2E4 L2M4 L2M5
<b>DFHTYPE</b>	TYPETERM-Definitionen			<b>Typeterms:</b> DFHCONS DFHLU2 DFHLU3 DFHLU62T DFHSCSP DFH3270 DFH3270P DFHLU0E2 DFHLU0M2 DFHLU0M3 DFHLU0M4 DFHLU0M5 DFHLU2E2 DFHLU2M2 DFHLU2E3 DFHLU2M3 DFHLU2E4 DFHLU2M4 DFHLU2M5
<b>DFHVTAM</b>	z/OS Communications Server-Programme	DFHGMM DFHZNAC DFHZNEP	CSGM CSNE	
<b>DFHVTAMP</b>	z/OS Communications Server-Drucktastenfunktion für Terminalsteuerung	DFHCPY DFHEXI DFHPRK DFHRKB	CSCY CSPK CSRK	

Tabelle 22. DFHLIST-Ressourcendefinitionen (Forts.)

Gruppenname	Beschreibung	Programme	Transaktionen	Sonstige Ressourcen
<b>DFHWEB</b>	CICS-Webunterstützungsdefinitionen	DFH\$WB1A DFHWBA DFHWBAAX DFHWBADX DFHWBAHX DFHWBALX DFHWBAOX DFHWBA1 DFHWBBLI DFHWBCLI DFHWBENV DFHWBEP DFHWBERX DFHWBGB DFHWBIMG DFHWBIP DFHWBLT DFHWBPA DFHWBPW DFHWBPW1 DFHWBPW2 DFHWBPW3 DFHWBPW4 DFHWBST DFHWBTC DFHWBTL DFHWBTRU DFHWBTTA DFHWBTTB DFHWBTTC DFHWBUN DFHWBXN	CWBA CWBC CWBG CWXN CW XU	<b>TS-Modelle:</b> DFHWEB  <b>Dokumentschablonen:</b>  DFHWBPW1 DFHWBPW2 DFHWBPW3 DFHWBPW4
<b>DFHWEB2</b>	Web 2.0-Definitionen	DFHW2A DFHW2FD DFHW2TS	CW2A	
<b>DFHWU</b>	CICS Management Client Interface	DFHWUIPG DFHWUIPI DFHWUIP1 DFHWUIP3 DFHWUIP4 DFHWUIP5	CWWU CWGQ	<b>Dokumentschablonen:</b> DFHWUIPI DFHWUIP1 DFHWUIP3 DFHWUIP4 DFHWUIP5
<b>Anmerkung:</b> Die DFHTYPE TYPETERM-Definitionen stimmen mit den von z/OS Communications Server bereitgestellten LOGMODE-Definitionen überein. Wenn Sie nicht mit den bereitgestellten z/OS Communications Server-LOGMODES arbeiten, dann müssen Sie möglicherweise die DFHTYPE TYPETERM-Definitionen ändern. Informationen zur Programmierung für die z/OS Communications Server-LOGMODE-Definitionen finden Sie in Coding entries in the VTAM LOGON mode table.				

## Von CICS bereitgestellte, nicht in DFHLIST enthaltene Gruppen

Nicht alle von CICS bereitgestellten Ressourcendefinitionsgruppen sind in der Liste DFHLIST definiert.

Tabelle 23. Nicht in DFHLIST enthaltene Ressourcendefinitionen

Gruppenname	Beschreibung	Programme	Transaktionen	Sonstige Ressourcen
DFHBRCF	Beispiel einer Coupling-Facility-Datentabelle für DFHBRNSF			<b>Datei:</b> DFHBRNSF
DFHBRUT	Beispiel einer benutzerverwalteten Datentabelle für DFHBRNSF			<b>Datei:</b> DFHBRNSF
DFHBRVR	VSAM RLS-Beispiel für DFHBRNSF			<b>Datei:</b> DFHBRNSF
DFHBRVSL	VSAM-Nicht-RLS-Beispiel (lokal) für DFHBRNSF			<b>Datei:</b> DFHBRNSF
DFHBRVSR	VSAM-Nicht-RLS-Beispiel (fern) für DFHBRNSF			<b>Datei:</b> DFHBRNSF
DFHCMAC	CICS-Onlinenachrichten und -Codes	DFHCMAC	CMAC (Alias CHLP)	<b>Maskengruppe:</b> DFHCMCM <b>Datei:</b> DFHCMACD
<b>Anmerkung:</b> DFHCMAC ist aus Gründen der Kompatibilität mit früheren Releases nicht in DFHLIST enthalten. Wenn Sie die Transaktion für die Onlinenachrichten und -Codes verwenden wollen, dann müssen Sie DFHCMAC zur Liste Ihrer Startgruppen hinzufügen.				
DFHMISC3	Sonstige Gruppen	DFHNET		
<b>Anmerkung:</b> DFHMISC3 ist aus Gründen der Kompatibilität mit früheren Releases nicht in DFHLIST enthalten. Wenn Sie dieses Element auf Ihrem System benötigen, dann fügen Sie es zur Liste Ihrer Startgruppen hinzu.				
DFHRPC	Prozedurfernauf Ruf	DFHRPAL DFHRPAS DFHRPC00 DFHRPMS DFHRPRP DFHRPTRU	CRPA CRPC CRPM	<b>Maskengruppe:</b> DFHRP0
DFHTERM C	TERMINAL-Modelldefinition für automatische Konsoleninstallation			<b>Terminals:</b> AUTC

## Von CICS bereitgestellte Kompatibilitätsgruppen

CICS stellt RDO-Gruppen bereit, um die Kompatibilität mit früheren Releases sicherzustellen.

Nachdem Sie ein Upgrade für eine CICS-Systemdefinitionsdatei (CICS System Definition Data Set) durchgeführt haben und wenn die CSD gemeinsam mit früheren Releases von CICS verwendet werden soll, dann müssen Sie die entsprechenden DFHCOMPx-Kompatibilitätsgruppen zur Liste Ihrer Startgruppen hinzufügen.

Weitere Informationen zur gemeinsamen Nutzung der CSD (CICS-Systemdefinition) in unterschiedlichen Releases von CICS und Angaben dazu, welche DFH-COMP<sub>x</sub>-Gruppen für die früheren Releases aufgenommen werden müssen, finden Sie in CSD compatibility between different CICS releases in Upgrading.

Die gemeinsame Nutzung einer CSD-Datei mit einer CICS-Region, die mit einem höheren Release-Level als diese CSD-Datei ausgeführt wird, ist nicht möglich.

## Gruppen von Beispielanwendungsprogrammen

Diese Ressourcendefinitionen sind zur Ausführung der Beispielanwendungsprogramme erforderlich, die zusammen mit CICS zur Verfügung gestellt werden. Die Gruppen werden in DFHLIST nicht aufgeführt.

Tabelle 24. CICS-Beispielanwendungen - Ressourcendefinitionen

Gruppenname	Sprache	Beschreibung	Definierte Ressourcen
DFH\$AFLA	Assembler	FILEA-Beispielanwendungen	<p><b>Maskengruppen:</b> DFH\$AGA DFH\$AGB DFH\$AGC DFH\$AGD DFH\$AGK DFH\$AGL</p> <p><b>Programme:</b> DFH\$AALL DFH\$ABRW DFH\$ACOM DFH\$AMNU DFH\$AREN DFH\$AREP</p> <p><b>Transaktionen:</b> AADD ABRW AINQ AMNU AORD AORQ AREP AUPD</p>
DFH\$BMSP	COBOL und PL/I	Unterstützungsanwendungen für BMS-Partitionen	<p><b>Partitionsgruppe:</b> DFH0PS</p> <p><b>Maskengruppen:</b> DFH0CGP DFH\$PGP</p> <p><b>Programme:</b> DFH0CPKO DFH0CPLA DFH\$PPKO DFH\$PPLA</p> <p><b>Transaktionen:</b> PPKO PPLA XPKO XPLA</p>
DFH\$CFLA	COBOL	FILEA-Beispielanwendungen	<p><b>Maskengruppen:</b> DFH0CGA DFH0CGB DFH0CGC DFH0CGD DFH0CGK DFH0CGL</p> <p><b>Programme:</b> DFH0CALL DFH0CBRW DFH0CCOM DFH0CMNU DFH0CKEN DFH0CREP</p> <p><b>Transaktionen:</b> ADDS BRWS INQY MENU OREN OREQ REPT UPDT</p>
DFH\$CNSL		Beispielkonsolendefinitionen	<p><b>Typeterms:</b> DFH\$JCLC DFH\$CONS</p> <p><b>Terminals:</b> CJCL CNSL CN02</p>

Tabelle 24. CICS-Beispielanwendungen - Ressourcendefinitionen (Forts.)

Gruppenname	Sprache	Beschreibung	Definierte Ressourcen
DFH\$CTXT	COBOL	CUA-Textmodellanwendung	<p><b>Dateien:</b> DFH0FCAI DFH0FCUS DFH0FHLP</p> <p><b>Maskengruppen:</b> DFH0AB DFH0ABT DFH0BRW DFH0DEL DFH0FPD DFH0HLP DFH0HP DFH0HPD DFH0LST DFH0NEW DFH0OPN DFH0PRT DFH0SAS DFH0T1 DFH0UPD</p> <p><b>Programme:</b> DFH0VAB DFH0VABT DFH0VBRW DFH0VDEL DFH0VDQ DFH0VHLP DFH0VHP DFH0VLIO DFH0VLST DFH0VNEW DFH0VOL DFH0VOPN DFH0VPRT DFH0VRIO DFH0VSAS DFH0VTBL DFH0VT1 DFH0VUPD</p> <p><b>Profil:</b> DFH\$CUA2</p> <p><b>Transaktionen:</b> AC2A AC2C AC2D AC2E AC2F AC20 AC21 AC22 AC23 AC24 AC25 AC26 AC27 AC28 DELQ</p>
DFH\$DFLA	C	FILEA-Beispielanwendungen	<p><b>Maskengruppen:</b> DFH\$DGA DFH\$DGB DFH\$DGC DFH\$DGD DFH\$DGK DFH\$DGL</p> <p><b>Programme:</b> DFH\$DALL DFH\$DBRW DFH\$DCOM DFH\$DMNU DFH\$DREN DFH\$DREP</p> <p><b>Transaktionen:</b> DADD DBRW DINQ DMNU DORD DORQ DREP DUPD</p>
DFH\$DLIV		IMS-Installationsprüfverfahren	<p><b>Programme:</b> DFH\$DLAC DFH\$DLAE DFH\$DLCC DFH\$DLCE DFH\$DLPC DFH\$DLPE</p> <p><b>Transaktionen:</b> ASMC ASME COBC COBE PLIC PLIE</p>
DFH\$EXBS		Kataloganwendungsbeispiel	<p><b>Dateien:</b> EXMPCAT EXMPCONF</p> <p><b>Maskengruppen:</b> DFH0XS1 DFH0XS2 DFH0XS3</p> <p><b>Programme:</b> DFH0XCMN DFH0XGUI DFH0XODE DFH0XSDS DFH0XSOD DFH0XSSM DFH0XVDS DFH0XWOD</p> <p><b>Transaktionen:</b> ECFG EGUI</p>

Tabelle 24. CICS-Beispielanwendungen - Ressourcendefinitionen (Forts.)

Gruppenname	Sprache	Beschreibung	Definierte Ressourcen
DFH\$EXCI		Beispiel für EXCI-Stapelaufchnittstelle	<b>Verbindungen:</b> EXCG EXCS <b>Programme:</b> DFH\$AXCS DFH\$AXVS <b>Sitzungen:</b> EXCG EXCS <b>Transaktionen:</b> EXCI HPJC
DFH\$EXWS		Kataloganwendungsbeispiel	<b>Programm:</b> DFH0XCUI <b>Transaktion:</b> ECLI <b>TCP/IP-Service:</b> EXMPPORT <b>Pipelines:</b> EXPIPE01 EXPIPE02
DFH\$FILA		FILEA-Beispiele	<b>Datei:</b> FILEA
DFH\$ICOM	Assembler	Intersystem Communication (ISC = systemübergreifende Kommunikation)	<b>Maskengruppen:</b> DFH\$IGB DFH\$IGC DFH\$IGS DFH\$IGX DFH\$IG1 DFH\$IG2  <b>Programme:</b> DFH\$ICIC DFH\$IFBL DFH\$IFBR DFH\$IMSN DFH\$IMSO DFH\$IQRD DFH\$IQRL DFH\$IQRR DFH\$IQXL DFH\$IQXR  <b>Transaktionen:</b> ICIC IFBL IFBR IMSN IMISO IQRD IQRL IQRR IQXL IQXR
DFH\$NACT		NACT-Beispielanwendung	<b>Dateien:</b> ACCTFILE ACCTNAM ACINUSE  <b>Maskengruppe:</b> DFH0MNA  <b>Programme:</b> DFH0CNA1 DFH0CNA2 DFH0CNA3 DFH0CNA4 DFH0CNA5  <b>Transaktionen:</b> NACP NACT
DFH\$OSGI		Beispielanwendungen für Java-Unterstützung über einen JVM-Server	<b>Bundle:</b> DFH\$OSGB <b>JVM-Server:</b> DFHJVMS  <b>Programme:</b> DFH\$JSAM DFH\$LCCA DFJ\$JHE1 DFJ\$JHE2 DFJ\$JPC1 DFJ\$JPC2 DFJ\$JTD1 DFJ\$JTS1 DFJ\$JTSC  <b>Warteschlange für transiente Daten:</b> JTD1  <b>Transaktionen:</b> JHE1 JHE2 JPC1 JPC2 JTD1 JTS1

Tabelle 24. CICS-Beispielanwendungen - Ressourcendefinitionen (Forts.)

Gruppenname	Sprache	Beschreibung	Definierte Ressourcen
DFH\$PFLA	PL/I	FILEA-Beispielanwendungen	<b>Maskengruppen:</b> DFH\$PGA DFH\$PGB DFH\$PGC DFH\$PGD DFH\$PGK DFH\$PGL  <b>Programme:</b> DFH\$PALL DFH\$PBRW DFH\$PCOM DFH\$PMNU DFH\$PREN DFH\$PREP  <b>Transaktionen:</b> PADD PBRW PINQ PMNU PORD PORQ PREP PUPD
DFH\$STAT		Statistiken	<b>Programme:</b> DFH\$STED
DFH\$SXP		Nachrichtendomänenexits	<b>Programme:</b> DFH\$SXP1 DFH\$SXP2 DFH\$SXP3 DFH\$SXP4 DFH\$SXP5 DFH\$SXP6
DFH\$UTIL	Assembler	Dynamische Zuordnung des Dienstprogramms für transiente Daten	<b>Programme:</b> DFH\$TDWT DFH99  <b>Transaktionen:</b> ADYN TDWT
DFH\$VTAM		Terminaldefinitionen	<b>Typeterms:</b> DFH\$L77 DFH\$L78 DFH\$L79 DFH\$L86  <b>Terminals:</b> L77C L77D L78A L79A L86A
DFH\$WLP		WebSphere Application Server Liberty-Beispieldefinitionen	<b>JVM-Server:</b> DFHWLP  <b>Bundles:</b> JDBC WLPJDBC WLPHELLO WLPTSQ
DFHAI62		APPC-Verbindungen für Starter	<b>Programm:</b> DFHZATDY  <b>Verbindungen:</b> CBPS CBSS CCPS  <b>Sitzungen:</b> CBPS CBSS CCPS
DFHMROAR		MRO-Systeme für Starter	<b>Verbindungen:</b> CICD CICT  <b>Sitzungen:</b> CICSRD CICSRT
DFHMRODR		MRO-Systeme für Starter	<b>Verbindung:</b> CICA  <b>Sitzung:</b> CICSRA



Tabelle 24. CICS-Beispielanwendungen - Ressourcendefinitionen (Forts.)

Gruppenname	Sprache	Beschreibung	Definierte Ressourcen
DFHMROFA		MRO-Systeme für Starter	<b>Datei:</b> FILEA  <b>Maskengruppen:</b> DFH\$AGA DFH\$AGB DFH\$AGC DFH\$AGD DFH\$AGK DFH\$AGL  <b>Programme:</b> DFH\$AALL DFH\$ABRW DFH\$ACOM DFH\$AMNU DFH\$AREN DFH\$AREP  <b>Transaktionen:</b> AADD ABRW AINQ AMNU AORD AORQ AREP AUPD
DFHMROFD		MRO-Systeme für Starter	<b>Datei:</b> FILEA
DFHMROFT		MRO-Systeme für Starter	<b>Transaktionen:</b> AADD ABRW AINQ AMNU AORD AORQ AREP AUPD
DFHMROTR		MRO-Systeme für Starter	<b>Verbindung:</b> CICA  <b>Sitzung:</b> CICSRA

## TYPETERM-Definitionen in Gruppe DFHTYPE

Die von CICS bereitgestellte CSD-Gruppe DFHTYPE enthält TYPETERM-Definitionen.

### DFHCONS

MVS-Konsole:

```
TYPETERM(DFHCONS)  GROUP(DFHTYPE)
DEVICE(CONSOLE)
PAGESIZE(1,124)      AUTOPAGE(NO)
BRACKET(YES)         BUILDCHAIN(YES)    ROUTEDMSG(S NONE)
UCTRAN(YES)
```

### DFHLU2

Logische SNA-Einheit Typ 2 (3270-Bildschirmseinheiten):

```
TYPETERM(DFHLU2)  GROUP(DFHTYPE)
DEVICE(LUTYPE2)    TERMMODEL(2)
DEFSCREEN(24,80)   PAGESIZE(24,80)    AUTOPAGE(NO)
BRACKET(YES)       BUILDCHAIN(YES)     ROUTEDMSG(S ALL)
AUDIBLEALARM(YES) EXTENDED(S YES)      UCTRAN(YES)
SENDSIZE(1536)     RECEIVESIZE(256)    IOAREALEN(256,4000)
ERRLASTLINE(YES)  ERRINTENSIFY(YES)    ATI(YES)  TTI(YES)
DISCREQ(YES)      RELREQ(YES)          AUTOCONNECT(YES)
LOGONMSG(YES)     QUERY(ALL)           CREATSESS(NO)
```

Diese Definition gilt für eine 3278 Model 2-Bildschirmseinheit. Sie eignet sich für die folgenden Einheiten: 3178, 3179, 3277, 3278, 3279, 3290, 3270PC, 3270PC/G, 3270PC/GX, 8775 und 5550.

### DFHLU3

Logische SNA-Einheit Typ 3 (3270-Drucker):

TYPETERM(DFHLU3)	GROUP(DFHTYPE)	
DEVICE(LUTYPE3)	TERMMODEL(2)	
DEFSCREEN(24,80)	PAGESIZE(24,80)	AUTOPAGE(YES)
BRACKET(YES)	BUILDCHAIN(NO)	ROUTEDMSGS(ALL)
SENDSIZE(256)	RECEIVESIZE(256)	IOAREALEN(512,0)
EXTENDEDDS(YES)	QUERY(ALL)	ATI(YES) TTI(YES)
DISCREQ(YES)	RELREQ(YES)	AUTOCONNECT(YES)
LOGONMSG(NO)		CREATESESS(NO)

Diese Definition gilt für einen 3287-Drucker. Sie eignet sich für die folgenden Einheiten: 3262, 3268, 3284, 3286, 3287, 3288, 3289 und 5550.

#### DFHSCSP

Logische SNA-Einheit Typ 1 (3270-SCS-Drucker):

TYPETERM(DFHSCSP)	GROUP(DFHTYPE)	
DEVICE(SCSPRINT)		
PAGESIZE(24,80)	AUTOPAGE(YES)	
BRACKET(YES)	BUILDCHAIN(NO)	ROUTEDMSGS(ALL)
SENDSIZE(256)	RECEIVESIZE(256)	IOAREALEN(512,0)
EXTENDEDDS(YES)	QUERY(ALL)	ATI(YES) TTI(YES)
DISCREQ(YES)	RELREQ(YES)	AUTOCONNECT(YES)
LOGONMSG(NO)		CREATESESS(NO)

Diese Definition gilt für einen 3287-Drucker. Sie eignet sich für die folgenden Einheiten: 3262, 3268, 3287, 3289 und 5550.

#### DFH3270

Lokal angeschlossene 3270-Bildschirmeinheiten (nicht SNA):

TYPETERM(DFH3270)	GROUP(DFHTYPE)	
DEVICE(3270)	TERMMODEL(2)	
DEFSCREEN(24,80)	PAGESIZE(24,80)	AUTOPAGE(NO)
BRACKET(YES)	BUILDCHAIN(NO)	ROUTEDMSGS(ALL)
AUDIBLEALARM(YES)	EXTENDEDDS(YES)	UCTRAN(YES)
SENDSIZE(0)	RECEIVESIZE(0)	IOAREALEN(512,0)
ERRLASTLINE(YES)	ERRINTENSIFY(YES)	ATI(YES) TTI(YES)
DISCREQ(YES)	RELREQ(YES)	AUTOCONNECT(YES)
LOGONMSG(YES)	QUERY(ALL)	CREATESESS(NO)

Diese Definition gilt für eine 3278 Model 2-Bildschirmeinheit. Sie eignet sich für die folgenden Einheiten: 3178, 3179, 3277, 3278, 3279 und 3290.

#### DFH3270P

Lokal angeschlossene 3270-Drucker (nicht SNA):

TYPETERM(DFH3270P)	GROUP(DFHTYPE)	
DEVICE(3270P)	TERMMODEL(2)	
DEFSCREEN(24,80)	PAGESIZE(24,80)	AUTOPAGE(YES)
BRACKET(YES)	BUILDCHAIN(NO)	ROUTEDMSGS(ALL)
SENDSIZE(0)	RECEIVESIZE(0)	IOAREALEN(512,0)
EXTENDEDDS(YES)	QUERY(ALL)	ATI(YES) TTI(YES)
DISCREQ(YES)	RELREQ(YES)	AUTOCONNECT(YES)
LOGONMSG(NO)		CREATESESS(NO)

Diese Definition gilt für einen 3284 Model 2-Drucker. Sie eignet sich für die folgenden Einheiten: 3262, 3268, 3284, 3287, 3288, 3289 und 5550.

#### DFHLU62T

Logische SNA-Einheit Typ 6.2 (APPC) - Einzelsitzungsterminal:

TYPETERM(DFHLU62T)	GROUP(DFHTYPE)	
DEVICE(APPC)		
	PAGESIZE(1,40)	AUTOPAGE(YES)
BRACKET(YES)	BUILDCHAIN(YES)	ROUTEDMSGS(NONE)

SENDSIZE(2048)	RECEIVESIZE(2048)	IOAREALEN(0,0)
		ATI(YES) TTI(YES)
LOGONMSG(NO)		CREATESESS(NO)

Diese Definition gilt für ein APPC-Einzelsitzungsterminal und eignet sich auch für die folgenden Einheiten: DISPLAYWRITER, SCANMASTER und SYSTEM/38.

#### DFHLU0E2

Model 2-Einheit mit erweitertem Datenstrom (nicht SNA; Abfrage):

TYPETERM(DFHLU0E2)	GROUP(DFHTYPE)	
DEVICE(3270)	TERMMODEL(2)	SHIPPABLE(YES)
DEFSCREEN(24,80)	PAGESIZE(24,80)	AUTOPAGE(NO)
ALTSCREEN(0,0)	ALTPAGE(0,0)	
BRACKET(YES)	BUILDCHAIN(NO)	ROUTEDMSGS(ALL)
AUDIBLEALARM(YES)	EXTENDEDDS(YES)	UCTRAN(YES)
RECEIVESIZE(0)	SENDSIZE(0)	IOAREALEN(512,0)
ERRASTLINE(YES)	ERRINTENSIFY(YES)	QUERY(ALL)
DISCREQ(YES)	RELREQ(YES)	AUTOCONNECT(NO)
LOGONMSG(YES)	CREATESESS(NO)	
ATI(YES)	TTI(YES)	

Model 2-Einheit mit erweitertem Datenstrom (nicht SNA; Abfrage). Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE NSX32702 überein.

#### DFHLU0M2

Model 2-Einheit (nicht SNA):

TYPETERM(DFHLU0M2)	GROUP(DFHTYPE)	
DEVICE(3270)	TERMMODEL(2)	SHIPPABLE(YES)
DEFSCREEN(24,80)	PAGESIZE(24,80)	AUTOPAGE(NO)
ALTSCREEN(0,0)	ALTPAGE(0,0)	
BRACKET(YES)	BUILDCHAIN(NO)	ROUTEDMSGS(ALL)
AUDIBLEALARM(YES)	EXTENDEDDS(NO)	UCTRAN(YES)
RECEIVESIZE(0)	SENDSIZE(0)	IOAREALEN(512,0)
ERRLASTLINE(YES)	ERRINTENSIFY(YES)	QUERY(NO)
DISCREQ(YES)	RELREQ(YES)	AUTOCONNECT(NO)
LOGONMSG(YES)	CREATESESS(NO)	
ATI(YES)	TTI(YES)	

Model 2-Einheit mit erweitertem Datenstrom (nicht SNA; Abfrage). Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE D4B32782 überein.

#### DFHLU0M3

Model 3-Einheit (nicht SNA):

TYPETERM(DFHLU0M3)	GROUP(DFHTYPE)	
DEVICE(3270)	TERMMODEL(2)	SHIPPABLE(YES)
DEFSCREEN(24,80)	PAGESIZE(24,80)	AUTOPAGE(NO)
ALTSCREEN(32,80)	ALTPAGE(32,80)	
BRACKET(YES)	BUILDCHAIN(NO)	ROUTEDMSGS(ALL)
AUDIBLEALARM(YES)	EXTENDEDDS(NO)	UCTRAN(YES)
RECEIVESIZE(0)	SENDSIZE(0)	IOAREALEN(512,0)
ERRLASTLINE(YES)	ERRINTENSIFY(YES)	QUERY(NO)
DISCREQ(YES)	RELREQ(YES)	AUTOCONNECT(NO)
LOGONMSG(YES)	CREATESESS(NO)	
ATI(YES)	TTI(YES)	

Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE D4B32783 überein.

#### DFHLU0M4

Model 4-Einheit (nicht SNA):

TYPETERM(DFHLU0M4)	GROUP(DFHTYPE)	
DEVICE(3270)	TERMMODEL(2)	SHIPPABLE(YES)
DEFSCREEN(24,80)	PAGESIZE(24,80)	AUTOPAGE(NO)
ALTSCREEN(43,80)	ALTPAGE(43,80)	
BRACKET(YES)	BUILDCHAIN(NO)	ROUTEDMSGS(ALL)
AUDIBLEALARM(YES)	EXTENDEDDS(NO)	UCTRAN(YES)
RECEIVESIZE(0)	SENDSIZE(0)	IOAREALEN(512,0)
ERRLASTLINE(YES)	ERRINTENSIFY(YES)	QUERY(NO)
DISCREQ(YES)	RELREQ(YES)	AUTOCONNECT(NO)
LOGONMSG(YES)	CREATESESS(NO)	
ATI(YES)	TTI(YES)	

Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE D4B32784 überein.

#### DFHLU0M5

Model 5-Einheit (nicht SNA):

TYPETERM(DFHLU0M5)	GROUP(DFHTYPE)	
DEVICE(3270)	TERMMODEL(2)	SHIPPABLE(YES)
DEFSCREEN(24,80)	PAGESIZE(24,80)	AUTOPAGE(NO)
ALTSCREEN(27,132)	ALTPAGE(27,132)	
BRACKET(YES)	BUILDCHAIN(NO)	ROUTEDMSGS(ALL)
AUDIBLEALARM(YES)	EXTENDEDDS(NO)	UCTRAN(YES)
RECEIVESIZE(0)	SENDSIZE(0)	IOAREALEN(512,0)
ERRLASTLINE(YES)	ERRINTENSIFY(YES)	QUERY(NO)
DISCREQ(YES)	RELREQ(YES)	AUTOCONNECT(NO)
LOGONMSG(YES)	CREATESESS(NO)	
ATI(YES)	TTI(YES)	

Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE D4B32785 überein.

#### DFHLU2E2

SNA LU Typ 2 Model 2-Einheit mit erweitertem Datenstrom (Abfrage):

TYPETERM(DFHLU2E2)	GROUP(DFHTYPE)	
DEVICE(LUTYPE2)	TERMMODEL(2)	SHIPPABLE(YES)
DEFSCREEN(24,80)	PAGESIZE(24,80)	AUTOPAGE(NO)
ALTSCREEN(0,0)	ALTPAGE(0,0)	
BRACKET(YES)	BUILDCHAIN(YES)	ROUTEDMSGS(ALL)
AUDIBLEALARM(YES)	EXTENDEDDS(YES)	UCTRAN(YES)
RECEIVESIZE(1024)	SENDSIZE(3840)	IOAREALEN(256,4000)
ERRLASTLINE(YES)	ERRINTENSIFY(YES)	QUERY(ALL)
DISCREQ(YES)	RELREQ(YES)	AUTOCONNECT(NO)
LOGONMSG(YES)	CREATESESS(NO)	
ATI(YES)	TTI(YES)	

SNA LU Typ 2 Model 2-Einheit mit erweitertem Datenstrom (Abfrage).

Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE SNX32702 überein.

#### DFHLU2M2

SNA LU Typ 2 Model 2-Einheit:

TYPETERM(DFHLU2M2)	GROUP(DFHTYPE)	
DEVICE(LUTYPE2)	TERMMODEL(2)	SHIPPABLE(YES)
DEFSCREEN(24,80)	PAGESIZE(24,80)	AUTOPAGE(NO)
ALTSCREEN(0,0)	ALTPAGE(0,0)	
BRACKET(YES)	BUILDCHAIN(YES)	ROUTEDMSGS(ALL)
AUDIBLEALARM(YES)	EXTENDEDDS(NO)	UCTRAN(YES)
RECEIVESIZE(1024)	SENDSIZE(1536)	IOAREALEN(256,4000)
ERRLASTLINE(YES)	ERRINTENSIFY(YES)	QUERY(NO)
DISCREQ(YES)	RELREQ(YES)	AUTOCONNECT(NO)
LOGONMSG(YES)	CREATESESS(NO)	
ATI(YES)	TTI(YES)	

Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE D4A32782 überein.

### DFHLU2E3

SNA LU Typ 2 Model 3-Einheit mit erweitertem Datenstrom (Abfrage):

TYPETERM(DFHLU2E3)	GROUP(DFHTYPE)	
DEVICE(LUTYPE2)	TERMMODEL(2)	SHIPPABLE(YES)
DEFSCREEN(24,80)	PAGESIZE(24,80)	AUTOPAGE(NO)
ALTSCREEN(32,80)	ALTPAGE(32,80)	
BRACKET(YES)	BUILDCHAIN(YES)	ROUTEDMSGS(ALL)
AUDIBLEALARM(YES)	EXTENDED(YES)	UCTRAN(YES)
RECEIVESIZE(1024)	SENDSIZE(3840)	IOAREALEN(256,4000)
ERRLASTLINE(YES)	ERRINTENSIFY(YES)	QUERY(ALL)
DISCREQ(YES)	RELREQ(YES)	AUTOCONNECT(NO)
LOGONMSG(YES)	CREATESESS(NO)	
ATI(YES)	TTI(YES)	

SNA LU Typ 2 Model 3-Einheit mit erweitertem Datenstrom (Abfrage).

Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE SNX32703 überein.

### DFHLU2M3

SNA LU Typ 2 Model 3-Einheit:

TYPETERM(DFHLU2M3)	GROUP(DFHTYPE)	
DEVICE(LUTYPE2)	TERMMODEL(2)	SHIPPABLE(YES)
DEFSCREEN(24,80)	PAGESIZE(24,80)	AUTOPAGE(NO)
ALTSCREEN(32,80)	ALTPAGE(32,80)	
BRACKET(YES)	BUILDCHAIN(YES)	ROUTEDMSGS(ALL)
AUDIBLEALARM(YES)	EXTENDED(YES)	UCTRAN(YES)
RECEIVESIZE(1024)	SENDSIZE(1536)	IOAREALEN(256,4000)
ERRLASTLINE(YES)	ERRINTENSIFY(YES)	QUERY(NO)
DISCREQ(YES)	RELREQ(YES)	AUTOCONNECT(NO)
LOGONMSG(YES)	CREATESESS(NO)	
ATI(YES)	TTI(YES)	

Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE D4A32783 überein.

### DFHLU2E4

SNA LU Typ 2 Model 4-Einheit mit erweitertem Datenstrom (Abfrage):

TYPETERM(DFHLU2E4)	GROUP(DFHTYPE)	
DEVICE(LUTYPE2)	TERMMODEL(2)	SHIPPABLE(YES)
DEFSCREEN(24,80)	PAGESIZE(24,80)	AUTOPAGE(NO)
ALTSCREEN(43,80)	ALTPAGE(43,80)	
BRACKET(YES)	BUILDCHAIN(YES)	ROUTEDMSGS(ALL)
AUDIBLEALARM(YES)	EXTENDED(YES)	UCTRAN(YES)
RECEIVESIZE(1024)	SENDSIZE(3840)	IOAREALEN(256,4000)
ERRLASTLINE(YES)	ERRINTENSIFY(YES)	QUERY(ALL)
DISCREQ(YES)	RELREQ(YES)	AUTOCONNECT(NO)
LOGONMSG(YES)	CREATESESS(NO)	
ATI(YES)	TTI(YES)	

SNA LU Typ 2 Model 4-Einheit mit erweitertem Datenstrom (Abfrage).

Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE SNX32704 überein.

### DFHLU2M4

SNA LU Typ 2 Model 4-Einheit:

TYPETERM(DFHLU2M4)	GROUP(DFHTYPE)	
DEVICE(LUTYPE2)	TERMMODEL(2)	SHIPPABLE(YES)
DEFSCREEN(24,80)	PAGESIZE(24,80)	AUTOPAGE(NO)
ALTSCREEN(43,80)	ALTPAGE(43,80)	
BRACKET(YES)	BUILDCHAIN(YES)	ROUTEDMSGS(ALL)

AUDIBLEALARM(YES)	EXTENDEDSS(NO)	UCTRAN(YES)
RECEIVESIZE(1024)	SENDSIZE(1536)	IOAREALEN(256,4000)
ERRLASTLINE(YES)	ERRINTENSIFY(YES)	QUERY(NO)
DISCREQ(YES)	RELREQ(YES)	AUTOCONNECT(NO)
LOGONMSG(YES)	CREATESESS(NO)	
ATI(YES)	TTI(YES)	

Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE D4A32784 überein.

#### DFHLU2M5

SNA LU Typ 2 Model 5-Einheit:

TYPETERM(DFHLU2M5)	GROUP(DFHTYPE)	
DEVICE(LUTYPE2)	TERMMODEL(2)	SHIPPABLE(YES)
DEFSCREEN(24,80)	PAGESIZE(24,80)	AUTOPAGE(NO)
ALTSCREEN(27,132)	ALTPAGE(27,132)	
BRACKET(YES)	BUILDCHAIN(YES)	ROUTEDMSGS(ALL)
AUDIBLEALARM(YES)	EXTENDEDSS(NO)	UCTRAN(YES)
RECEIVESIZE(1024)	SENDSIZE(1536)	IOAREALEN(256,4000)
ERRLASTLINE(YES)	ERRINTENSIFY(YES)	QUERY(NO)
DISCREQ(YES)	RELREQ(YES)	AUTOCONNECT(NO)
LOGONMSG(YES)	CREATESESS(NO)	
ATI(YES)	TTI(YES)	

Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE D4A32785 überein.

#### DFHLU2E5

SNA LU Typ 2 Model 5-Einheit mit erweitertem Datenstrom (Abfrage):

TYPETERM(DFHLU2E5)	GROUP(DFHTYPE)	
DEVICE(LUTYPE2)	TERMMODEL(2)	SHIPPABLE(YES)
DEFSCREEN(24,80)	PAGESIZE(24,80)	AUTOPAGE(NO)
ALTSCREEN(27,132)	ALTPAGE(27,132)	
BRACKET(YES)	BUILDCHAIN(YES)	ROUTEDMSGS(ALL)
AUDIBLEALARM(YES)	EXTENDEDSS(NO)	UCTRAN(YES)
RECEIVESIZE(1024)	SENDSIZE(3480)	IOAREALEN(256,4000)
ERRLASTLINE(YES)	ERRINTENSIFY(YES)	QUERY(ALL)
DISCREQ(YES)	RELREQ(YES)	AUTOCONNECT(NO)
LOGONMSG(YES)	CREATESESS(NO)	
ATI(YES)	TTI(YES)	

SNA LU Typ 2 Model 5-Einheit mit erweitertem Datenstrom (Abfrage).

Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE LSX32705 überein.

## TERMINAL-Modelldefinitionen in Gruppe DFHTERM

Die von CICS bereitgestellte CSD-Gruppe DFHTERM enthält TERMINAL-Modelldefinitionen für die automatische Installation.

**LU2** SNA 3270 Model 2-Bildschirmeinheit mit TYPETERM DFHLU2:

TERMINAL(LU2)	GROUP(DFHTERM)	TYPETERM(DFHLU2)
AUTINSTMODEL(ONLY)	AUTINSTNAME(DFHLU2)	

Diese Definition gilt für eine 3278 Model 2-Bildschirmeinheit. Sie eignet sich für die folgenden Einheiten: 3178, 3179, 3277, 3278, 3279, 3290, 3270PC, 3270PC/G, 3270PC/GX, 8775 und 5550.

**LU3** SNA 3270 Model 2-Drucker mit TYPETERM DFHLU3:

TERMINAL(LU3)	GROUP(DFHTERM)	TYPETERM(DFHLU3)
AUTINSTMODEL(ONLY)	AUTINSTNAME(DFHLU3)	

Diese Definition gilt für einen 3287-Drucker. Sie eignet sich für die folgenden Einheiten: 3262, 3268, 3284, 3286, 3287, 3288, 3289 und 5550.

**SCSP** SNA 3278 Model 2-Drucker mit TYPETERM DFHSCSP:

TERMINAL(SCSP)	GROUP(DFHTERM)	TYPETERM(DFHSCSP)
AUTINSTMODEL(ONLY)	AUTINSTNAME(DFHSCSP)	

Diese Definition gilt für einen 3287-Drucker. Sie eignet sich für die folgenden Einheiten: 3262, 3268, 3287, 3289 und 5550.

**3270** 3270 Model 2-Bildschirmeinheit mit TYPETERM DFH3270 (nicht SNA):

TERMINAL(3270)	GROUP(DFHTERM)	TYPETERM(DFH3270)
AUTINSTMODEL(ONLY)	AUTINSTNAME(DFH3270)	

Diese Definition gilt für eine 3278 Model 2-Bildschirmeinheit. Sie eignet sich für die folgenden Einheiten: 3178, 3179, 3277, 3278, 3279 und 3290.

**3284** 3270 Model 2-Drucker mit TYPETERM DFH3270P (nicht SNA):

TERMINAL(3284)	GROUP(DFHTERM)	TYPETERM(DFH3270P)
AUTINSTMODEL(ONLY)	AUTINSTNAME(DFH3270P)	

Diese Definition gilt für einen 3284 Model 2-Drucker. Sie eignet sich für die folgenden Einheiten: 3262, 3268, 3284, 3287, 3288, 3289 und 5550.

**LU62** APPC-Einzelsitzungsterminal (LU6.2) mit TYPETERM DFHLU62T:

TERMINAL(LU62)	GROUP(DFHTERM)	TYPETERM(DFHLU62T)
AUTINSTMODEL(ONLY)	AUTINSTNAME(DFHLU62T)	

Diese Definition gilt für ein APPC-Einzelsitzungsterminal und eignet sich auch für die folgenden Einheiten: DISPLAYWRITER, SCANMASTER und SYSTEM/38.

**L0E2** Model 2-Bildschirmeinheit mit TYPETERM DFHLU0E2 (nicht SNA):

TERMINAL(L0E2)	GROUP(DFHTERM)	TYPETERM(DFHLU0E2)
AUTINSTMODEL(ONLY)	AUTINSTNAME(DFHLU0E2)	

Model 2-Einheit mit erweitertem Datenstrom (nicht SNA; Abfrage). Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE NSX32702 überein.

**L0M2** Model 2-Bildschirmeinheit mit TYPETERM DFHLU0M2 (nicht SNA):

TERMINAL(L0M2)	GROUP(DFHTERM)	TYPETERM(DFHLU0M2)
AUTINSTMODEL(ONLY)	AUTINSTNAME(DFHLU0M2)	

Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE D4B32782 überein.

**L0M3** Model 3-Bildschirmeinheit mit TYPETERM DFHLU0M3 (nicht SNA):

TERMINAL(L0M3)	GROUP(DFHTERM)	TYPETERM(DFHLU0M3)
AUTINSTMODEL(ONLY)	AUTINSTNAME(DFHLU0M3)	

Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE D4B32783 überein.

**L0M4** Model 4-Bildschirmeinheit mit TYPETERM DFHLU0M4 (nicht SNA):

TERMINAL(L0M4)	GROUP(DFHTERM)	TYPETERM(DFHLU0M4)
AUTINSTMODEL(ONLY)	AUTINSTNAME(DFHLU0M4)	

Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE D4B32784 überein.

**L0M5** Model 5-Bildschirmeinheit mit TYPETERM DFHLU0M5 (nicht SNA):

TERMINAL(L0M5)      GROUP(DFHTERM)      TYPETERM(DFHLU0M5)  
AUTINSTMODEL(ONLY)    AUTINSTNAME(DFHLU0M5)

Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE D4B32785 überein.

**L2E2** SNA LU2 Model 2-Bildschirmeinheit mit TYPETERM DFHLU2E2:

TERMINAL(L2E2)      GROUP(DFHTERM)      TYPETERM(DFHLU2E2)  
AUTINSTMODEL(ONLY)    AUTINSTNAME(DFHLU2E2)

SNA LU Typ 2 Model 2-Einheit mit erweitertem Datenstrom (Abfrage).  
Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE SNX32702 überein.

**L2M2** SNA LU2 Model 2-Bildschirmeinheit mit TYPETERM DFHLU2M2:

TERMINAL(L2M2)      GROUP(DFHTERM)      TYPETERM(DFHLU2M2)  
AUTINSTMODEL(ONLY)    AUTINSTNAME(DFHLU2M2)

Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE D4A32782 überein.

**L2E3** SNA LU2 Model 3-Bildschirmeinheit mit TYPETERM DFHLU2E3:

TERMINAL(L2E3)      GROUP(DFHTERM)      TYPETERM(DFHLU2E3)  
AUTINSTMODEL(ONLY)    AUTINSTNAME(DFHLU2E3)

SNA LU Typ 2 Model 3-Einheit mit erweitertem Datenstrom (Abfrage).  
Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE SNX32703 überein.

**L2M3** SNA LU2 Model 3-Bildschirmeinheit mit TYPETERM DFHLU2M3:

TERMINAL(L2M3)      GROUP(DFHTERM)      TYPETERM(DFHLU2M3)  
AUTINSTMODEL(ONLY)    AUTINSTNAME(DFHLU2M3)

Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE D4A32783 überein.

**L2E4** SNA LU2 Model 4-Bildschirmeinheit mit TYPETERM DFHLU2E4:

TERMINAL(L2E4)      GROUP(DFHTERM)      TYPETERM(DFHLU2E4)  
AUTINSTMODEL(ONLY)    AUTINSTNAME(DFHLU2E4)

SNA LU Typ 2 Model 4-Einheit mit erweitertem Datenstrom (Abfrage).  
Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE SNX32704 überein.

**L2M4** SNA LU2 Model 4-Bildschirmeinheit mit TYPETERM DFHLU2M4:

TERMINAL(L2M4)      GROUP(DFHTERM)      TYPETERM(DFHLU2M4)  
AUTINSTMODEL(ONLY)    AUTINSTNAME(DFHLU2M4)

Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE D4A32784 überein.

**L2E5** SNA LU2 Model 5-Bildschirmeinheit mit TYPETERM DFHLU2E5:

TERMINAL(L2E5)      GROUP(DFHTERM)      TYPETERM(DFHLU2E5)  
AUTINSTMODEL(ONLY)    AUTINSTNAME(DFHLU2E5)

SNA LU Typ 2 Model 5-Einheit mit erweitertem Datenstrom (Abfrage).  
Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE LSX32705 überein.



**L2M5** SNA LU2 Model 5-Bildschirmeinheit mit TYPETERM DFHLU2M5:

TERMINAL(L2M5)	GROUP(DFHTERM)	TYPETERM(DFHLU2M5)
AUTINSTMODEL(ONLY)	AUTINSTNAME(DFHLU2M5)	

Diese Definition stimmt mit dem von z/OS Communications Server bereitgestellten LOGMODE D4A32785 überein.

**CBRF** Standardschablonenterminal zur Verwendung mit 3270-Brücke:

TERMINAL(CBRF)	GROUP(DFHTERM)	TYPETERM(DFHLU2)
NETNAME(CBRF)	REMOTESYSTEM(CBR)	REMOTENAME(CBRF)

---

## PROFILE-Definitionen in Gruppe DFHEP

Die von CICS bereitgestellte CSD-Gruppe DFHEP enthält PROFILE-Definitionen für die Ereignisverarbeitung.

### DFHECEPH

CICS verwendet dieses Profil für die CEPH-Transaktion (HTTP-Ereignisverarbeitungsadapter), um Werte für Zeitlimitüberschreitungen zu verwalten, die mit dem Parameter RTIMOUT festgelegt wurden.

Im Folgenden wird die Definition aufgeführt:

```
PROFILE(DFHECEPH)
GROUP(DFHEP)
SCRNSIZE(DEFAULT)
MSGJRNL(NO)
MSGINTEG(NO)
ONEWTE(NO)
DVSUPRT(ALL)
INBFMH(NO)
RAQ(NO)
LOGREC(NO)
NEPCLASS(000)
RTIMOUT(0005)
```

---

## PROFILE-Definitionen in Gruppe DFHISC

Die bereitgestellte CSD-Gruppe DFHISC enthält PROFILE-Definitionen für ISC-Sitzungen (ISC = Intersystem Communication; systemübergreifende Kommunikation).

### DFHCICSF

CICS verwendet dieses Profil für die Sitzungen mit fernen Systemen oder Regionen, wenn ein CICS-Anwendungsprogramm eine Funktionsverlagerungsanforderung absetzt.

Die Definition lautet wie folgt:

```
PROFILE(DFHCICSF)
GROUP(DFHISC)
SCRNSIZE(DEFAULT)
MSGJRNL(NO)
MSGINTEG(NO)
ONEWTE(NO)
DVSUPRT(ALL)
INBFMH(ALL)
RAQ(NO)
LOGREC(NO)
NEPCLASS(000)
```

### DFHCICSR

CICS verwendet dieses Profil beim Transaktionsrouting für die Kommuni-

kation zwischen der (in der Anwendungsverwaltungsregion ausgeführten) Benutzertransaktion und der regionsübergreifenden Verbindung oder der APPC-Verbindung.

Die Definition lautet wie folgt:

```
PROFILE(DFHCICSR)
GROUP(DFHISC)
SCRNSIZE(DEFAULT)
MSGJRNL(NO)
MSGINTEG(NO)
ONEWTE(NO)
DVSUPRT(ALL)
INBFMH(ALL)
RAQ(NO)
LOGREC(NO)
NEPCLASS(000)
```

### **DFHCICSS**

CICS verwendet dieses Profil beim Transaktionsrouting für die Kommunikation zwischen der (in der Terminalverwaltungsregion ausgeführten) Relay-Transaktion und der regionsübergreifenden Verbindung oder der APPC-Verbindung. Sie können ein anderes Profil angeben, indem Sie in der TRANSACTION-Definition die Option TRPROF angeben.

Die Definition lautet wie folgt:

```
PROFILE(DFHCICSS)
GROUP(DFHISC)
SCRNSIZE(DEFAULT)
MSGJRNL(NO)
MSGINTEG(NO)
ONEWTE(NO)
DVSUPRT(ALL)
INBFMH(ALL)
RAQ(NO)
LOGREC(NO)
NEPCLASS(000)
```

---

## **PROFILE-Definitionen in Gruppe DFHSTAND**

Die von CICS bereitgestellte CSD-Gruppe DFHSTAND enthält PROFILE-Definitionen.

Die von CICS bereitgestellte CSD-Gruppe DFHSTAND enthält die folgenden PROFILE-Definitionen:

### **DFHCICSA**

Dies ist das Standardprofil für alternative Funktionen, das mit dem Anwendungsprogrammbefehl ALLOCATE angefordert wird. Im Befehl ALLOCATE kann explizit ein anderes Profil angegeben werden.

Die Definition lautet wie folgt:

```
PROFILE(DFHCICSA)
GROUP(DFHSTAND)
SCRNSIZE(DEFAULT)
MSGJRNL(NO)
MSGINTEG(NO)
ONEWTE(NO)
PROTECT(NO)
DVSUPRT(ALL)
INBFMH(ALL)
RAQ(NO)
LOGREC(NO)
NEPCLASS(000)
```

## DFHCICSE

Dies ist das Fehlerprofil für Hauptfunktionen. CICS verwendet dieses Profil zum Übergeben einer Fehlermeldung an die Hauptfunktion, wenn das erforderliche Profil nicht gefunden werden kann.

Die Definition lautet wie folgt:

```
PROFILE(DFHCICSE)
GROUP(DFHSTAND)
SCRNSIZE(DEFAULT)
MSGJRNL(NO)
MSGINTEG(NO)
ONEWTE(NO)
PROTECT(NO)
DVSUPRT(ALL)
INBFMH(NO)
RAQ(NO)
LOGREC(NO)
NEPCLASS(000)
```

## DFHCICSP

Dies ist das Standardprofil für die CSPG-Seitenabruftransaktion. Sie können ein anderes Profil für eine bestimmte Transaktion angeben, indem Sie in der TRANSACTION-Definition die Option PROFILE angeben.

Die Definition lautet wie folgt:

```
PROFILE(DFHCICSP)
GROUP(DFHSTAND)
SCRNSIZE(DEFAULT)
UCTRAN(YES)MSGJRNL(NO)
MSGINTEG(NO)
ONEWTE(NO)
PROTECT(NO)
DVSUPRT(ALL)
INBFMH(NO)
RAQ(NO)
LOGREC(NO)
NEPCLASS(000)
```

## DFHCICST

Dies ist das Standardprofil für Hauptfunktionen. Sie können ein anderes Profil für eine bestimmte Transaktion angeben, indem Sie in der TRANSACTION-Definition die Option PROFILE angeben.

Die Definition lautet wie folgt:

```
PROFILE(DFHCICST)
GROUP(DFHSTAND)
SCRNSIZE(DEFAULT)
MSGJRNL(NO)
MSGINTEG(NO)
ONEWTE(NO)
PROTECT(NO)
DVSUPRT(ALL)
INBFMH(NO)
RAQ(NO)
LOGREC(NO)
NEPCLASS(000)
```

## DFHCICSV

Dies ist das Profil für Hauptfunktionen, das verwendet wird, wenn die Transaktion ausschließlich z/OS Communications Server-Einheiten unterstützt.

Die Definition lautet wie folgt:

```
PROFILE(DFHCICSV)
GROUP(DFHSTAND)
SCRNSIZE(DEFAULT)
MSGJRNL(NO)
MSGINTEG(NO)
ONEWTE(NO)
PROTECT(NO)
DVSUPRT(VTAM)
INBFMH(NO)
RAQ(NO)
LOGREC(NO)
NEPCLASS(000)
```

**Anmerkung:** VTAM wird jetzt als z/OS Communications Server bezeichnet.

#### **DFHPPF01**

Das Profil DFHPPF01 wird während der Initialisierung von CICS verwendet, um Tasks auszuführen, die vor der Installation der CSD-Dateideinitionen hinzugefügt wurden.

Die Definitionen lauten wie folgt:

```
PROFILE(DFHPPF01)
GROUP(DFHSTAND)
DESCRIPTION(VTAM-ONLY PROFILE)
SCRNSIZE(DEFAULT)
MSGJRNL(NO)
MSGINTEG(NO)
ONEWTE(NO)
PROTECT(NO)
DVSUPRT(VTAM)
INBFMH(NO)
RAQ(NO)
LOGREC(NO)
NEPCLASS(000)
```

#### **DFHPPF02**

Das Profil DFHPPF02 wird während der Initialisierung von CICS verwendet, um Tasks auszuführen, die vor der Installation der CSD-Dateideinitionen hinzugefügt wurden.

Die Definitionen lauten wie folgt:

```
PROFILE(DFHPPF02)
GROUP(DFHSTAND)
DESCRIPTION(ALL-NULLS PROFILE)
SCRNSIZE(DEFAULT)
MSGJRNL(NO)
MSGINTEG(NO)
ONEWTE(NO)
PROTECT(NO)
DVSUPRT(ALL)
INBFMH(NO)
RAQ(NO)
LOGREC(NO)
NEPCLASS(000)
```

---

## **Modelldefinitionen in Gruppe DFHPGAIP**

CICS stellt verschiedene Modelldefinitionen in der Gruppe DFHPGAIP bereit, um die automatische Installation von Programmen zu unterstützen.

#### **DFHPGAPG**

Dies ist die standardmäßige PROGRAM-Definition für die automatische Installation von Programmen.

Die Definition lautet wie folgt:

```
PROGRAM(DFHPGAPG) GROUP(DFHPGAIP)
DESCRIPTION (Standardprogramm für automatische Installation von Programmen)
                                RELOAD(NO)          RESIDENT(NO)
USAGE(NORMAL)                USELPACOPY(NO)        STATUS(ENABLED)
RSL(00)                      CEDF(YES)             DATALOCATION(ANY)
EXECKEY(USER)                EXECUTIONSET(FULLAPI)
```

Wenn die Angabe DATALOCATION(ANY) einen nicht passenden Standardwert darstellt, dann können Sie verhindern, dass für DATALOCATION automatisch der Standardwert ANY verwendet wird. Dies ist in folgender Weise möglich:

- Kopieren Sie die Definition für DFHPGAPG in Ihre eigene Gruppe und ändern Sie die Einstellung für DATALOCATION. Installieren Sie die Definition unbedingt nach der Gruppe DFHPGAIP.
- Geben Sie den Namen der eigenen Programmdefinition als eine Modelldefinition an, die als Ausgabeparameter des Exits für die automatische Installation von Programmen verwendet werden soll.

## DFHPGAMP

Dies ist die standardmäßige MAPSET-Definition für die automatische Installation von Programmen.

Die Definition lautet wie folgt:

```
MAPSET(DFHPGAMP) GROUP(DFHPGAIP)
DESCRIPTION (Standardmaskengruppe für automatische Installation von Programmen)
RESIDENT(NO)          USAGE(NORMAL)
USELPACOPY(NO)        STATUS(ENABLED)
RSL(00)
```

## DFHPGAPT

Dies ist die standardmäßige PARTITION-Definition für die automatische Installation von Programmen.

Die Definition lautet wie folgt:

```
PARTITIONSET(DFHPGAPT) GROUP(DFHPGAIP)
DESCRIPTION (Standardpartitionsgruppe für automatische Installation von Programmen)
RESIDENT(NO)          USAGE(NORMAL)
USELPACOPY(NO)        STATUS(ENABLED)
RSL(00)
```



---

## Bemerkungen

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden. IBM stellt dieses Material möglicherweise auch in anderen Sprachen zur Verfügung. Für den Zugriff auf das Material in einer anderen Sprache kann eine Kopie des Produkts oder der Produktversion in der jeweiligen Sprache erforderlich sein.

Möglicherweise bietet IBM die in dieser Dokumentation beschriebenen Produkte, Services oder Funktionen in anderen Ländern nicht an. Informationen über die gegenwärtig im jeweiligen Land verfügbaren Produkte und Services sind beim zuständigen IBM Ansprechpartner erhältlich. Hinweise auf IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Services von IBM verwendet werden können. Anstelle der IBM Produkte, Programme oder Services können auch andere, ihnen äquivalente Produkte, Programme oder Services verwendet werden, solange diese keine gewerblichen oder anderen Schutzrechte von IBM verletzen. Die Verantwortung für den Betrieb von Produkten, Programmen und Services anderer Anbieter liegt beim Kunden.

Für in diesem Handbuch beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieses Handbuchs ist keine Lizenzierung dieser Patente verbunden. Lizenzanforderungen sind schriftlich an folgende Adresse zu richten (Anfragen an diese Adresse müssen auf Englisch formuliert werden):

*IBM Director of Licensing  
IBM Europe, Middle East & Africa  
Tour Descartes 2, avenue Gambetta  
92066 Paris La Defense  
France*

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in dieser Veröffentlichung nicht ausgeschlossen werden. Die hier enthaltenen Informationen werden in regelmäßigen Zeitabständen aktualisiert und als Neuauflage veröffentlicht. IBM kann ohne weitere Mitteilung jederzeit Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen.

Verweise in diesen Informationen auf Websites anderer Anbieter werden lediglich als Service für den Kunden bereitgestellt und stellen keinerlei Billigung des Inhalts dieser Websites dar. Das über diese Websites verfügbare Material ist nicht Bestandteil des Materials für dieses IBM Produkt. Die Verwendung dieser Websites geschieht auf eigene Verantwortung.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängig voneinander erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der ausgetauschten Informationen zu ermöglichen, wenden sich an folgende Adresse:

*IBM Director of Licensing  
IBM Corporation  
North Castle Drive, MD-NC119  
Armonk, NY 10504-1785  
United States of America*

Die Bereitstellung dieser Informationen kann unter Umständen von bestimmten Bedingungen - in einigen Fällen auch von der Zahlung einer Gebühr - abhängig sein.

Die Lieferung des in diesen Informationen beschriebenen Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt auf der Basis der IBM Rahmenvereinbarung bzw. der Allgemeinen Geschäftsbedingungen von IBM, der IBM Internationalen Nutzungsbedingungen für Programmpakete oder einer äquivalenten Vereinbarung.

Alle Informationen zu Produkten anderer Anbieter stammen von den Anbietern der aufgeführten Produkte, deren veröffentlichten Ankündigungen oder anderen allgemein verfügbaren Quellen. IBM hat diese Produkte nicht getestet und kann daher keine Aussagen zu Leistung, Kompatibilität oder anderen Merkmalen machen. Fragen zu den Leistungsmerkmalen von Produkten anderer Anbieter sind an den jeweiligen Anbieter zu richten.

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufs. Sie sollen nur die Funktionen des Lizenzprogramms illustrieren und können Namen von Personen, Firmen, Marken oder Produkten enthalten. Alle diese Namen sind frei erfunden und jede Ähnlichkeit mit Namen und Adressen tatsächlicher Personen oder Unternehmen ist rein zufällig.

#### **COPYRIGHTLIZENZ:**

Diese Veröffentlichung enthält Beispielanwendungsprogramme, die in Quellsprache geschrieben sind und Programmiertechniken in verschiedenen Betriebsumgebungen veranschaulichen. Sie dürfen diese Beispielprogramme kostenlos kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, zu verwenden, zu vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle für die Betriebsumgebung konform sind, für die diese Beispielprogramme geschrieben werden. Diese Beispiele wurden nicht unter allen denkbaren Bedingungen getestet. Daher kann IBM die Zuverlässigkeit, Wartungsfreundlichkeit oder Funktion dieser Programme weder zusagen noch gewährleisten. Die Beispielprogramme werden ohne Wartung (auf "as-is"-Basis) und ohne jegliche Gewährleistung zur Verfügung gestellt. IBM übernimmt keine Haftung für Schäden, die durch die Verwendung der Beispielprogramme entstehen.

### **Informationen zu Programmierschnittstellen**

Die von CICS zur Verfügung gestellte Dokumentation kann teilweise als Programmierschnittstelle betrachtet werden und zum Teil nicht.

Programmierschnittstellen, die es dem Kunden ermöglichen, Programme zur Nutzung der Services von CICS Transaction Server for z/OS, Version 5 Release 5 zu schreiben, sind in folgenden Abschnitten der Online-Produktdokumentation enthalten:

- Developing applications



- Developing system programs
- Securing overview
- Developing for external interfaces
- Reference: application development
- Reference: system programming
- Reference: connectivity

Informationen, die NICHT zur Verwendung als Programmierschnittstelle von CICS Transaction Server for z/OS, Version 5 Release 5 bestimmt sind, die aber als Programmierschnittstelle missverstanden werden können, sind in folgenden Abschnitten der Online-Produktdokumentation enthalten:

- Troubleshooting and support
- Reference: diagnostics

Wenn Sie auf die CICS-Dokumentation in Handbüchern im PDF-Format zugreifen, sind Programmierschnittstellen, die es dem Kunden ermöglichen, Programme zur Nutzung der Services von CICS Transaction Server for z/OS, Version 5 Release 5 zu schreiben, in den folgenden Handbüchern enthalten:

- Application Programming Guide und Application Programming Reference
- Business Transaction Services
- Customization Guide
- C++ OO Class Libraries
- Debugging Tools Interfaces Reference
- Distributed Transaction Programming Guide
- External Interfaces Guide
- Front End Programming Interface Guide
- IMS Database Control Guide
- Installation Guide
- Security Guide
- Supplied Transactions
- CICSplex SM Managing Workloads
- CICSplex SM Managing Resource Usage
- CICSplex SM Application Programming Guide and Application Programming Reference
- Java Applications in CICS

Wenn Sie auf die CICS-Dokumentation in Handbüchern im PDF-Format zugreifen, sind Informationen, die NICHT zur Verwendung als Programmierschnittstelle von CICS Transaction Server for z/OS, Version 5 Release 5 bestimmt sind, die aber als Programmierschnittstelle missverstanden werden können, in den folgenden Handbüchern enthalten:

- Data Areas
- Diagnosis Reference
- Problem Determination Guide
- CICSplex SM Problem Determination Guide

## Marken

IBM, das IBM Logo und ibm.com sind Marken oder eingetragene Marken der International Business Machines Corporation in den USA und/oder anderen Ländern. Weitere Produkt- und Servicenamen können Marken von IBM oder anderen Unternehmen sein. Eine aktuelle Liste der IBM Marken finden Sie auf der Webseite Copyright and trademark information unter [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Adobe, das Adobe-Logo, PostScript und das PostScript-Logo sind Marken oder eingetragene Marken von Adobe Systems Incorporated in den USA und/oder anderen Ländern.

Intel, das Intel-Logo, Intel Inside, das Intel Inside-Logo, Intel Centrino, das Intel Centrino-Logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium und Pentium sind Marken oder eingetragene Marken der Intel Corporation oder ihrer Tochtergesellschaften in den USA oder anderen Ländern.

Java und alle auf Java basierenden Marken und Logos sind Marken oder eingetragene Marken der Oracle Corporation und/oder ihrer verbundenen Unternehmen.

Linux ist eine eingetragene Marke von Linus Torvalds in den USA und/oder anderen Ländern.

Microsoft, Windows, Windows NT und das Windows-Logo sind Marken der Microsoft Corporation in den USA und/oder anderen Ländern.

UNIX ist eine eingetragene Marke von The Open Group in den USA und anderen Ländern.

## Nutzungsbedingungen für die Produktdokumentation

Die Berechtigungen zur Nutzung dieser Veröffentlichungen werden Ihnen auf der Basis der folgenden Bedingungen gewährt.

### Anwendbarkeit

Diese Bedingungen sind eine Ergänzung der Nutzungsbedingungen auf der IBM Website.

### Persönliche Nutzung

Sie dürfen diese Veröffentlichungen für Ihre persönliche, nicht kommerzielle Nutzung unter der Voraussetzung vervielfältigen, dass alle Eigentumsvermerke erhalten bleiben. Sie dürfen diese Veröffentlichungen oder Teile der Veröffentlichungen ohne ausdrückliche Genehmigung von IBM nicht weitergeben, anzeigen oder abgeleitete Werke davon erstellen.

### Kommerzielle Nutzung

Sie dürfen diese Veröffentlichungen nur innerhalb Ihres Unternehmens und unter der Voraussetzung, dass alle Eigentumsvermerke erhalten bleiben, vervielfältigen, weitergeben und anzeigen. Sie dürfen diese Veröffentlichungen oder Teile der Veröffentlichungen ohne ausdrückliche Genehmigung von IBM außerhalb Ihres Unternehmens weder vervielfältigen, weitergeben oder anzeigen noch abgeleitete Werke davon erstellen.

### Rechte

Abgesehen von den hier gewährten Berechtigungen werden keine weiteren Berechtigungen, Lizenzen oder Rechte (veröffentlicht oder stillschweigend) in Bezug auf die Veröffentlichungen oder darin enthaltene Informationen, Daten, Software oder geistiges Eigentum gewährt.

IBM behält sich das Recht vor, die hierin gewährten Berechtigungen nach eigenem Ermessen zurückzuziehen, wenn sich die Nutzung der Veröffentlichungen für IBM als nachteilig erweist oder wenn die obigen Nutzungsbestimmungen nicht genau befolgt werden.

Sie dürfen diese Informationen nur in Übereinstimmung mit allen anwendbaren Gesetzen und Vorschriften, einschließlich aller US-amerikanischen Exportgesetze und Verordnungen, herunterladen und exportieren.

IBM übernimmt keine Gewährleistung für den Inhalt dieser Veröffentlichungen. Diese Veröffentlichungen werden auf der Grundlage des gegenwärtigen Zustands (auf "as-is"-Basis) und ohne eine ausdrückliche oder stillschweigende Gewährleistung für die Handelsüblichkeit, die Verwendbarkeit für einen bestimmten Zweck oder die Freiheit von Rechten Dritter zur Verfügung gestellt.

## **IBM Online-Datenschutzerklärung**

IBM Softwareprodukte, einschließlich Software as a Service-Lösungen („Softwareangebote“), können Cookies oder andere Technologien verwenden, um Informationen zur Produktnutzung zu erfassen, die Endbenutzererfahrung zu verbessern und Interaktionen mit dem Endbenutzer anzupassen oder zu anderen Zwecken. In vielen Fällen werden von den Softwareangeboten keine personenbezogenen Daten erfasst. Einige der IBM Softwareangebote können Sie jedoch bei der Erfassung personenbezogener Daten unterstützen. Wenn dieses Softwareangebot Cookies zur Erfassung personenbezogener Daten verwendet, sind nachfolgend nähere Informationen über die Verwendung von Cookies durch dieses Angebot zu finden:

### **Für die Webbenutzerschnittstelle von CICSplex System Manager (Hauptschnittstelle):**

Abhängig von den bereitgestellten Konfigurationen kann dieses Softwareangebot Sitzungscookies und persistente Cookies zum Erfassen der Benutzernamen und anderer personenbezogener Daten einzelner Benutzer für das Sitzungsmanagement, die Authentifizierung, einen besseren Bedienungskomfort, zur Nutzungsüberwachung und für andere funktionale Zwecke verwenden. Diese Cookies können nicht inaktiviert werden.

### **Für die Webbenutzerschnittstelle von CICSplex System Manager (Datenschnittstelle):**

Abhängig von den bereitgestellten Konfigurationen kann dieses Softwareangebot Sitzungscookies und persistente Cookies zum Erfassen der Benutzernamen und anderer personenbezogener Daten einzelner Benutzer für das Sitzungsmanagement, die Authentifizierung, einen besseren Bedienungskomfort, zur Nutzungsüberwachung und für andere funktionale Zwecke verwenden. Diese Cookies können nicht inaktiviert werden.

### **Für die Webbenutzerschnittstelle von CICSplex System Manager ("hello world"-Seite):**

Abhängig von den bereitgestellten Konfigurationen kann dieses Softwareangebot Sitzungscookies verwenden, die keine personenbezogenen Daten erfassen. Diese Cookies können nicht inaktiviert werden.

### **Für CICS Explorer:**

Abhängig von den bereitgestellten Konfigurationen kann dieses Softwareangebot persistente Vorgaben und Sitzungsvorgaben zum Erfassen der Benutzernamen und Kennwörter von Benutzern für das Sitzungsmanagement, die Authentifizierung und zur Single Sign-on-Konfiguration (einmalige Anmeldung) verwenden. Diese Vorgaben können nicht inaktiviert werden, auch wenn die Speicherung eines Benutzerkennworts auf ei-

nem Datenträger in verschlüsselter Form nur aktiviert werden kann, indem der Benutzer bei der Anmeldung explizit ein Kontrollkästchen aktiviert.

Wenn es die für dieses Softwareangebot bereitgestellten Konfigurationen Ihnen als Kunde ermöglichen, personenbezogene Daten von Endbenutzern über Cookies und andere Technologien zu erfassen, müssen Sie sich zu allen gesetzlichen Bestimmungen in Bezug auf eine solche Datenerfassung, einschließlich aller Mitteilungspflichten und Zustimmungsanforderungen, rechtlich beraten lassen.

Weitere Informationen zur Nutzung verschiedener Technologien, einschließlich Cookies, für diese Zwecke finden Sie unter IBM Privacy Policy und in der IBM Online Privacy Statement im Abschnitt „Cookies, Web-Beacons und sonstige Technologien“ sowie auf der Seite IBM Software Products and Software-as-a-Service Privacy Statement.

---

# Index

## Numerische Stichwörter

3290-Terminal 277  
3600, Option  
    Operand LDC 462  
3770-Datenübertragungssystem  
    LDCs für logische Stapelverarbeitungseinheit  
    TCT-Beispiel 465

## A

Abbruchcodes  
    DFHSRT TYPE=SYSTEM 450  
    DFHSRT TYPE=USER 450  
ABCODE, Operand  
    DFHSRT TYPE=SYSTEM 450  
    DFHSRT TYPE=USER 450  
Abhängige Standardwerte 267  
ACCESSMETHOD, Attribut  
    CONNECTION-Definition 12  
ACCMETH, Operand  
    DFHFCT TYPE=FILE 408  
    DFHTCT TYPE=INITIAL 454  
    DFHTCT TYPE=LINE 467  
    DFHTCT TYPE=REMOTE 473  
    DFHTCT TYPE=TERMINAL 472  
    ferne Terminals 472, 473  
    sequenzielle Einheiten 467  
ACCOUNTREC, Attribut  
    DB2CONN-Definition 31  
    DB2ENTRY-Definition 38  
ACTION, Attribut  
    TRANSACTION-Definition 240  
ADD, Attribut  
    FILE-Definition 55  
ADD, Befehl  
    CEDA 337  
ADD, Befehl, Dienstprogramm DFHCS-DUP 373  
ADD, Option  
    Operand SERVREQ 414  
ADDCNT, Option  
    Operand PERFORM 425  
ALIAS, Attribut  
    TRANSACTION-Definition 241  
ALTER, Befehl  
    CEDA 339  
ALTER, Befehl, Dienstprogramm DFHCS-DUP 374  
    generische Benennung in 375  
ALTPAGE, Attribut  
    TYPETERM-Definition 282  
ALTPRINTCOPY, Attribut  
    TERMINAL-Definition 223  
ALTPRINTER, Attribut  
    TERMINAL-Definition 223  
ALTSCREEN, Attribut  
    TYPETERM-Definition 283

ALTSUFFIX, Attribut  
    TYPETERM-Definition 284  
ANALYZER, Attribut  
    URIMAP-Definition 316  
APF (Authorized Program Facility) 401, 447  
APLKYBD, Attribut  
    TYPETERM-Definition 284  
APLTEXT, Attribut  
    TYPETERM-Definition 284  
APPC-Einheiten für Transaktionsrouting 221  
APPC-Einzelsitzungsterminal (LUTYPE6.2) 216  
APPC-Verbindungen (LUTYPE6.2) und Parallelsitzungen 158  
APPEND, Befehl  
    CEDA 341  
APPEND, Befehl, Dienstprogramm  
    DFHCSDUP 376  
    Beispiele 377  
APPENDCRLF, Attribut  
    DOCTEMPLATE-Definition 45  
APPLID, Attribut  
    IPCONN-Definition 77  
APPLID, Operand  
    Makro DFHSIT 10  
APPLID des fernen Systems 18  
ASCII, Attribut  
    TYPETERM-Definition 285  
AT-TLS 185  
ATI, Attribut  
    TYPETERM-Definition 286  
ATIFACILITY, Attribut  
    TDQUEUE-Definition 191  
ATOMSERVICE, Attribut  
    ATOMSERVICE-Definition 2  
    URIMAP-Definition 317  
ATOMSERVICE-Definition  
    Attribut ATOMSERVICE 2  
    Attribut ATOMTYPE 2  
    Attribut BINDFILE 2  
    Attribut CONFIGFILE 3  
    Attribut RESOURCENAME 3  
    Attribut RESOURCETYPE 4  
    Attribut STATUS 4  
ATOMSERVICE-Ressourcendefinition 1  
ATOMTYPE, Attribut  
    ATOMSERVICE-Definition 2  
ATTACHSEC, Attribut  
    CONNECTION-Definition 13  
    TCPIPService-Definition 173  
    TERMINAL-Definition 224  
ATTLASWARE 185  
Attribute  
    CONNECTION-Definition 10  
    DB2CONN-Definition 24  
    DB2ENTRY-Definition 37  
    DB2TRAN-Definition 42  
    DOCTEMPLATE-Definition 45  
    ENQMODEL-Definition 50

Attribute (*Forts.*)  
    FILE-Definition 55  
    JOURNALMODEL-Definition 89  
    LIBRARY-Definition 101  
    LSRPOOL-Definition 106  
    MAPSET-Definition 112  
    MQCONN-Definition 115  
    PARTITIONSET-Definition 124  
    PARTNER-Definition 127  
    PROCESSTYPE-Definition 134  
    PROFILE-Definition 137  
    SESSIONS-Definition 159  
    TCPIPService-Definition 173  
    TDQUEUE-Definition 189  
    TERMINAL-Definition 222  
    TRANCLASS-Definition 235  
    TRANSACTION-Definition 239  
    TSMODEL-Definition 259  
    TYPETERM-Definition 279  
AUDIBLEALARM, Attribut  
    TYPETERM-Definition 286  
AUDITLEVEL, Attribut  
    PROCESSTYPE-Definition 135  
AUDITLOG, Attribut  
    PROCESSTYPE-Definition 135  
AUTHENTICATE, Attribut  
    TCPIPService-Definition 173  
AUTHID, Attribut  
    DB2CONN-Definition 31  
    DB2ENTRY-Definition 39  
Authorized Program Facility (APF) 401, 447  
AUTHTYPE, Attribut  
    DB2CONN-Definition 31  
    DB2ENTRY-Definition 39  
AUTINSTMODEL, Attribut  
    TERMINAL-Definition 224  
AUTINSTMODEL, TERMINAL-Definition  
    von CICS bereitgestellt 516  
AUTINSTNAME, Attribut  
    TERMINAL-Definition 225  
AUTOCONNECT, Attribut  
    CONNECTION-Definition 14  
    IPCONN-Definition 77  
    SESSIONS-Definition 161  
    TYPETERM-Definition 286  
AUTOPAGE, Attribut  
    TYPETERM-Definition 287

## B

BACKLOG, Attribut  
    TCPIPService-Definition 173  
BACKTRANS, Attribut  
    TYPETERM-Definition 287  
BACKUPTYPE, Attribut  
    FILE-Definition 55, 56  
Basic Mapping Support (BMS)  
    Nachrichten mit mehreren Seiten an Drucker 463

BCHLU, Option  
 Operand LDC 462  
 Befehlslistentabelle (CLT = Command List Table) 401  
 CLT (Command List Table; Befehlslistentabelle) 401  
 DFHCLT-Makros 401  
 DFHCLT TYPE=COMMAND 403  
 DFHCLT TYPE=INITIAL 402  
 DFHCLT TYPE=LISTEND 405  
 DFHCLT TYPE=LISTSTART 403  
 DFHCLT TYPE=WTO 404  
 Befehlsthreadattribute  
 DB2CONN-Definition 34  
 Beispiele  
 FCT 416  
 MCT 441  
 PLT 445  
 RST 448  
 SRT 451  
 TLT 477  
 XLT 491  
 Bildschirmseinheiten  
 Drucker zuordnen 213  
 BINDFILE, Attribut  
 ATOMSERVICE-Definition 2  
 BINDPASSWORD, Attribut  
 CONNECTION 493  
 BINDSECURITY, Attribut  
 Attribut CONNECTION 15  
 TERMINAL-Definition 225  
 BLKKEYL, Operand  
 DFHFCT TYPE=FILE 408  
 BLKSIZE, Operand  
 DFHFCT TYPE=FILE 408  
 DFHTCT TYPE=SDSCI 466  
 sequenzielle Einheiten 466  
 BLOCKED, Option  
 Operand RECFORM 413  
 BLOCKFORMAT, Attribut  
 TDQUEUE-Definition 192  
 BLOCKSIZE, Attribut  
 TDQUEUE-Definition 192  
 BMS (Basic Mapping Support)  
 Nachrichten mit mehreren Seiten an Drucker 463  
 BRACKET, Attribut  
 TYPETERM-Definition 287  
 BREXIT, Attribut  
 TRANSACTION-Definition 241  
 BROWSE, Attribut  
 FILE-Definition 55  
 BROWSE, Option  
 Operand SERVREQ 414  
 BUILDCHAIN, Attribut  
 SESSIONS-Definition 162  
 TYPETERM-Definition 287  
 BUNDLE-Ressourcen 4, 51

## C

CADL, Ziel für transiente Daten 204  
 CADO, Ziel für transiente Daten 205  
 CADS, Ziel für transiente Daten 205  
 CAIL, Ziel für transiente Daten 205  
 CCPI, Ziel für transiente Daten 205  
 CCSE, Ziel für transiente Daten 205

CCSI, Ziel für transiente Daten 205  
 CCSO, Ziel für transiente Daten 205  
 CCZM, Ziel für transiente Daten 205  
 CDB2, Ziel für transiente Daten 206  
 CDBC, Ziel für transiente Daten 205  
 CDEP, Ziel für transiente Daten 206  
 CDUL, Ziel für transiente Daten 205  
 CECO, Ziel für transiente Daten 206  
 CEDA  
 Befehl ADD 337  
 Befehl ALTER 339  
 Befehl APPEND 341  
 Befehl CHECK 342  
 Befehl COPY 344  
 Befehl DEFINE 347  
 Befehl DELETE 349  
 Befehl DISPLAY 351  
 Befehl DISPLAY GROUP 352  
 Befehl DISPLAY LIST 352  
 Befehl EXPAND 353  
 Befehl EXPAND GROUP 354  
 Befehl EXPAND LIST 355  
 Befehl INSTALL 356  
 Befehl LOCK 358  
 Befehl MOVE 360  
 Befehl REMOVE 362  
 Befehl RENAME 364  
 Befehl UNLOCK 366  
 Befehl USERDEFINE 367  
 Befehl VIEW 371  
 CEJL, Ziel für transiente Daten 206  
 CEPO, Ziel für transiente Daten 206  
 CERTIFICATE, Attribut  
 IPCONN-Definition 77  
 TCPIPService-Definition 173  
 URIMAP-Definition 317  
 CESE, Ziel für transiente Daten 206  
 CFDTPOOL, Attribut  
 FILE-Definition 55  
 CGCSGID, Attribut  
 TYPETERM-Definition 287  
 CHAINCONTROL, Attribut  
 PROFILE-Definition 138  
 CHARACTERSET, Attribut  
 URIMAP-Definition 318  
 CHECK, Befehl  
 CEDA 342  
 CIEO, Ziel für transiente Daten 206  
 CIPHERS, Attribut  
 IPCONN-Definition 77  
 TCPIPService-Definition 173  
 URIMAP-Definition 318  
 CISL, Ziel für transiente Daten 206  
 CISO, Ziel für transiente Daten 206  
 CJRM, Ziel für transiente Daten 206  
 CKQQ, Ziel für transiente Daten 207  
 CLASS, Operand  
 DFHMCT TYPE=EMP 423  
 DFHMCT TYPE=RECORD 427  
 CLOCK, Operand  
 DFHMCT TYPE=EMP 423  
 CLT (Command List Table; Befehlslistentabelle) 401  
 CLT (Command List Table; Befehlslistentabelle) 401  
 DFHCLT-Makros 401  
 DFHCLT TYPE=COMMAND 403

CLT (Command List Table; Befehlslistentabelle) (Forts.)  
 DFHCLT TYPE=INITIAL 402  
 DFHCLT TYPE=LISTEND 405  
 DFHCLT TYPE=LISTSTART 403  
 DFHCLT TYPE=WTO 404  
 CMDSEC, Attribut  
 TRANSACTION-Definition 241  
 CMLO, Ziel für transiente Daten 207  
 CMPO, Ziel für transiente Daten 207  
 CMQM, Ziel für transiente Daten 207  
 Code für logische Einheiten (LDC = Logical Device Code) 456, 464  
 Codierungsbeispiele  
 FCT 416  
 MCT 441  
 PLT 445  
 RST 448  
 SRT 451  
 TLT 477  
 XLT 491  
 COLOR, Attribut  
 TYPETERM-Definition 288  
 COMAUTHID, Attribut  
 DB2CONN-Definition 34  
 COMAUTHTYPE, Attribut  
 DB2CONN-Definition 35  
 COMMAND, Operand  
 DFHCLT TYPE=COMMAND 403  
 COMPLETE, Option  
 Operand MIGRATE 455  
 COMTHREADLIMIT, Attribut  
 DB2CONN-Definition 36  
 CONFDATA, Attribut  
 TRANSACTION-Definition 241  
 CONFIGFILE, Attribut  
 ATOMSERVICE-Definition 3  
 CONNECTERROR, Attribut  
 DB2CONN-Definition 25  
 CONNECTION, Attribut  
 CONNECTION-Definition 15  
 SESSIONS-Definition 162  
 CONNECTION-Definition  
 Attribut ACCESSMETHOD 10, 12  
 Attribut ATTACHSEC 13  
 Attribut AUTOCONNECT 14  
 Attribut BINDPASSWORD 493  
 Attribut BINDSECURITY 15  
 Attribut CONNECTION 15  
 Attribut CONNTYPE 15  
 Attribut DATASTREAM 15  
 Attribut INDSYS 10, 16  
 Attribut INSERVICE 16  
 Attribut MAXQTIME 17  
 Attribut NETNAME 10, 18  
 Attribut PROTOCOL 10, 19  
 Attribut PSRECOVERY 20  
 Attribut QUEUELIMIT 20  
 Attribut RECORDFORMAT 21  
 Attribut REMOTENAME 21  
 Attribut REMOTESYSNET 21  
 Attribut REMOTESYSTEM 22  
 Attribut SECURITYNAME 22  
 Attribut SINGLESESS 22  
 Attribut USEDFLTUSER 23  
 Attribut XLNACTON 23  
 CONNECTION-Ressourcen 6

CONNTYPE, Attribut  
 CONNECTION-Definition 15  
 CONSNAME, Attribut  
 TERMINAL-Definition 225  
 CONSOLE, Attribut  
 TERMINAL-Definition 493  
 CONVERTER, Attribut  
 URIMAP-Definition 319  
 COPY, Attribut  
 TYPETERM-Definition 288  
 COPY, Befehl  
 CEDA 344  
 COPY, Befehl, Dienstprogramm DFHCS-  
 DUP 377  
 Beispiele 380  
 generische Benennung in 379  
 Option FROMCSD 379  
 Option MERGE 379  
 Option REPLACE 379  
 COUNT, Operand  
 DFHMCT TYPE=EMP 424  
 Coupling-Facility-Datentabellen 54  
 CPIO, Ziel für transiente Daten 207  
 CRDI, Ziel für transiente Daten 207  
 CREATESESS, Attribut  
 TYPETERM-Definition 288  
 CRITICAL, Attribut  
 LIBRARY-Definition 101  
 CRLO, Ziel für transiente Daten 207  
 CRPO, Ziel für transiente Daten 207  
 CSBA, Ziel für transiente Daten 207  
 CSBR, Ziel für transiente Daten 207  
 CSCC, Ziel für transiente Daten 207  
 CSCS, Ziel für transiente Daten 207  
 CSD  
 gemeinsame Nutzung in Relea-  
 ses 506  
 CSD (CICS-Systemdefinitionen), Datei  
 Gruppen von Ressourcendefinitionen  
 DFHISC 495  
 DFHSTAND 495  
 DFHTERM 495  
 DFHTYPE 495  
 Inhalt nach Initialisierung 495  
 CSDH, Ziel für transiente Daten 208  
 CSDL, Ziel für transiente Daten 208  
 CSFL, Ziel für transiente Daten 208  
 CSFU-Transaktion 410  
 CSJE, Ziel für transiente Daten 208  
 CSJO, Ziel für transiente Daten 208  
 CSKL, Ziel für transiente Daten 208  
 CSLB, Ziel für transiente Daten 209  
 CSML, Ziel für transiente Daten 209  
 CSNE, Ziel für transiente Daten 209  
 CSOO, Ziel für transiente Daten 209  
 CSPL, Ziel für transiente Daten 209  
 CSQL, Ziel für transiente Daten 209  
 CSRL, Ziel für transiente Daten 209  
 CSSH, Ziel für transiente Daten 209  
 CSSL, Ziel für transiente Daten 209  
 CSTL, Ziel für transiente Daten 209  
 CSZL, Ziel für transiente Daten 210  
 CSZX, Ziel für transiente Daten 210  
 CTLAPPLS, Operand  
 DFHRST TYPE=RSE 448  
 CWBO, Ziel für transiente Daten 210  
 CWBW, Ziel für transiente Daten 210

## D

DATA12K, Attribut  
 LSRPOOL-Definition 107  
 DATA16K, Attribut  
 LSRPOOL-Definition 107  
 DATA1K, Attribut  
 LSRPOOL-Definition 107  
 DATA20K, Attribut  
 LSRPOOL-Definition 108  
 DATA24K, Attribut  
 LSRPOOL-Definition 108  
 DATA28K, Attribut  
 LSRPOOL-Definition 108  
 DATA2K, Attribut  
 LSRPOOL-Definition 107  
 DATA32K, Attribut  
 LSRPOOL-Definition 108  
 DATA4K, Attribut  
 LSRPOOL-Definition 107  
 DATA512, Attribut  
 LSRPOOL-Definition 107  
 DATA8K, Attribut  
 LSRPOOL-Definition 107  
 DATABUFFERS, Attribut  
 FILE-Definition 55  
 TDQUEUE-Definition 193  
 DATAID, Operand  
 DFHTST TYPE=LOCAL 483  
 DFHTST TYPE=RECOVERY 481  
 DFHTST TYPE=REMOTE 484  
 DFHTST TYPE=SECURITY 486  
 DATASTREAM, Attribut  
 CONNECTION-Definition 15  
 Dateien  
 Beziehung zu Dateien 406  
 Dateisteuertabelle (FCT = File Control Ta-  
 ble) 406  
 Codierungsbeispiel 416  
 DFHFCT-Makros 406  
 DFHFCT TYPE=FILE 407  
 DFHFCT TYPE=INITIAL 407  
 Zusammenfassungstabelle 415  
 Datenbanksteuerung (DBCTL = Database  
 Control) 447  
 Db2-Eintragsressourcen 36  
 Db2-Transaktionsressourcen 42  
 DB2CONN, Attribut  
 DB2CONN-Definition 25  
 DB2CONN-Definition  
 Attribut ACCOUNTREC 31  
 Attribut AUTHID 31  
 Attribut AUTHTYPE 31  
 Attribut COMAUTHID 34  
 Attribut COMAUTHTYPE 35  
 Attribut COMTHREADLIMIT 36  
 Attribut CONNECTERROR 25  
 Attribut DB2CONN 25  
 Attribut DB2GROUPID 25  
 Attribut DB2ID 25  
 Attribut DROLLBACK 31  
 Attribut MSGQUEUE1 25  
 Attribut MSGQUEUE2 25  
 Attribut MSGQUEUE3 25  
 Attribut NONTERMREL 25  
 Attribut PLAN 31  
 Attribut PLANEXITNAME 31  
 Attribut PURGECYCLE 25

DB2CONN-Definition (Forts.)  
 Attribut RESYNCMEMBER 25  
 Attribut SIGNID 25  
 Attribut STANDBYMODE 25  
 Attribut STATSQUEUE 25  
 Attribut TCBLIMIT 25  
 Attribut THREADERROR 25  
 Attribut THREADLIMIT 31  
 Attribut THREADWAIT 31  
 Befehlsthreadattribute 34  
 COMAUTHID 34  
 COMAUTHTYPE 35  
 COMTHREADLIMIT 36  
 Pool-Thread-Attribute  
 ACCOUNTREC 31  
 AUTHID 31  
 AUTHTYPE 31  
 DROLLBACK 31  
 PLAN 31  
 PLANEXITNAME 31  
 THREADLIMIT 31  
 THREADWAIT 31  
 Verbindungsattribute  
 CONNECTERROR 25  
 DB2GROUPID 25  
 DB2ID 25  
 MSGQUEUE1 25  
 MSGQUEUE2 25  
 MSGQUEUE3 25  
 NONTERMREL 25  
 PURGECYCLE 25  
 RESYNCMEMBER 25  
 SIGNID 25  
 STANDBYMODE 25  
 STATSQUEUE 25  
 TCBLIMIT 25  
 THREADERROR 25  
 DB2CONN-Ressourcen 24  
 DB2ENTRY, Attribut  
 DB2ENTRY-Definition 37  
 DB2ENTRY-Definition  
 Attribut ACCOUNTREC 38  
 Attribut AUTHID 39  
 Attribut AUTHTYPE 39  
 Attribut DB2ENTRY 37  
 Attribut DROLLBACK 41  
 Attribut PLAN 41  
 Attribut PLANEXITNAME 41  
 Attribut PRIORITY 41  
 Attribut PROTECTNUM 41  
 Attribut THREADLIMIT 42  
 Attribut THREADWAIT 42  
 Attribut TRANSID 38  
 Attribute für Threadauswahl  
 TRANSID 38  
 Attribute für Threadoperationen  
 ACCOUNTREC 38  
 AUTHID 39  
 AUTHTYPE 39  
 DROLLBACK 41  
 PLAN 41  
 PLANEXITNAME 41  
 PRIORITY 41  
 PROTECTNUM 41  
 THREADLIMIT 42  
 THREADWAIT 42

- DB2GROUPID, Attribut
    - DB2CONN-Definition 25
  - DB2ID, Attribut
    - DB2CONN-Definition 25
  - DB2TRAN, Attribut
    - DB2TRAN-Definition 43
  - DB2TRAN-Definition
    - Attribut DB2TRAN 43
    - Attribut ENTRY 43
    - Attribut TRANSID 43
    - RDO-Befehl 42
  - DBCTL (Database Control; Datenbanksteuerung) 447
  - DDName, Attribut
    - DOCTEMPLATE-Definition 45
  - DDNAME, Attribut
    - TDQUEUE-Definition 193
  - DDNAME, Operand
    - DFHTCT TYPE=SDSCI 466
    - sequenzielle Einheiten 466
  - DEFINE, Befehl
    - CEDA 347
  - DEFINE, Befehl, Dienstprogramm
    - DFHCSDUP 380
    - Beispiele 382
  - DEFSCREEN, Attribut
    - TYPETERM-Definition 289
  - DELETE, Attribut
    - FILE-Definition 55
  - DELETE, Befehl
    - CEDA 349
  - DELETE, Befehl, Dienstprogramm
    - DFHCSDUP 382
    - Beispiele 384
  - DELIVER, Option
    - Operand PERFORM 427
  - DESC, Operand
    - DFHCLT TYPE=WTO 404
  - DESCRIPTION, Attribut
    - DOCTEMPLATE-Definition 46
    - ENQMODEL-Definition 50
    - FILE-Definition 55
    - IPCONN-Definition 77
    - PROCESSTYPE-Definition 135
    - SESSIONS 162
    - TDQUEUE-Definition 193
    - TRANSACTION-Definition 242
    - TYPETERM-Definition 289
  - DEVICE, Operand
    - DFHTCT TYPE=SCSCI 466
    - sequenzielle Einheiten 466
  - DFHCLT-Operanden
    - TYPE=COMMAND 403
    - TYPE=INITIAL 402
    - TYPE=LISTEND 405
    - TYPE=LISTSTART 403
    - TYPE=WTO 404
  - DFHCSDUP, Dienstprogramm zur Systemdefinition
    - Befehl ADD 373
    - Befehl ALTER 374
    - Befehl APPEND 376
    - Befehl COPY 377
    - Befehl DEFINE 380
    - Befehl DELETE 382
    - Befehl EXTRACT 385
    - Befehl INITIALIZE 386
  - DFHCSDUP, Dienstprogramm zur Systemdefinition (*Fortsetz.*)
    - Befehl LIST 387
    - Befehl PROCESS 389
    - Befehl REMOVE 390
    - Befehl SCAN 390
    - Befehl SERVICE 393
    - Befehl UPGRADE 393
    - Befehl USERDEFINE 394
    - Befehl VERIFY 396
  - DFHDLPSB-Operanden
    - TYPE=ENTRY 405
    - TYPE=INITIAL 405
  - DFHFCT-Operanden
    - TYPE=FILE 407
    - TYPE=INITIAL 407
  - DFHMCT-Operanden
    - TYPE=EMP 422
    - TYPE=INITIAL 417
    - TYPE=RECORD 427
  - DFHPLT-Operanden
    - TYPE=ENTRY 444
    - TYPE=INITIAL 443
  - DFHRDT, für RDO auswählbare TCT-Einträge 453
  - DFHRST-Operanden
    - TYPE=INITIAL 447
    - TYPE=RSE 447
    - TYPE=SUBSYS 448
  - DFHSIT, Makro
    - Operand APPLID 10
  - DFHSRT-Operanden
    - TYPE=INITIAL 449
    - TYPE=SYSTEM 450
    - TYPE=USER 450
  - DFHTCT-Operanden
    - ferne Terminals 472, 473
    - sequenzielle Einheiten 466, 467, 468
    - TYPE=GROUP 456
    - TYPE=INITIAL 454
    - TYPE=LINE 467
    - TYPE=REGION 472
    - TYPE=REMOTE 473
    - TYPE=SDSCI 466
    - TYPE=TERMINAL 468, 472
  - DFHTLT-Operanden
    - TYPE=ENTRY 476
    - TYPE=INITIAL 476
  - DFHTST-Operanden
    - TYPE=ENTRY 481
    - TYPE=INITIAL 480
    - TYPE=RECOVERY 481, 483, 486
    - TYPE=SHARED 486
  - DFHXLT-Operanden
    - TYPE=INITIAL 489
  - DISCREQ, Attribut
    - SESSIONS-Definition 162
    - TYPETERM-Definition 290
  - DISP, Operand
    - DFHFCT TYPE=FILE 409
  - DISPLAY, Befehl
    - CEDA 351
  - DISPLAY GROUP, Befehl
    - CEDA 352
  - DISPLAY LIST, Befehl
    - CEDA 352
  - DISPOSITION, Attribut
    - FILE-Definition 55
    - TDQUEUE-Definition 193
  - DL/I (Data Language/I)
    - DL/I-PSB-Verzeichnis (PDIR) 405
  - DOCTEMPLATE-Definition
    - Attribut APPENDCRLF 45
    - Attribut DDName 45
    - Attribut DESCRIPTION 46
    - Attribut EXITPGM 46
    - Attribut FILE 46
    - Attribut HFSFILE 46
    - Attribut MEMBERNAME 47
    - Attribut NAME 46
    - Attribut PROGRAM 47
    - Attribut TDQUEUE 47
    - Attribut TEMPLATENAME 48
    - Attribut TSQUEUE 48
    - Attribut TYPE 48
  - DOCTEMPLATE-Ressourcen 44
  - DROLLBACK, Attribut
    - DB2CONN-Definition 31
    - DB2ENTRY-Definition 41
  - Druck, fortlaufend (Autopaging) 463
  - Drucken
    - Terminals zum 212
  - Drucker
    - Bildschirmeinheit zuordnen 213
    - Operand PGESTAT 463
    - Terminaldefinitionen 212
  - DSCNAME, Operand
    - DFHTCT TYPE=SDSCI 466
    - sequenzielle Einheiten 466
  - DSN, Operand
    - DFHTCT TYPE=LDC 461
  - DSNAME, Attribut
    - FILE-Definition 55
    - LIBRARY-Definition 102
    - TDQUEUE-Definition 194
  - DSNAME, Operand
    - DFHFCT TYPE=FILE 409
  - DSNSHARING, Attribut
    - FILE-Definition 55
  - DTIMOUT, Attribut
    - TRANSACTION-Definition 242
  - DUALCASEKEYBD, Attribut
    - TYPETERM-Definition 290
  - DUMP, Attribut
    - TRANSACTION-Definition 243
  - DVC, Operand
    - DFHTCT TYPE=LDC 461
  - DVSUPRT, Attribut
    - PROFILE-Definition 138
  - DYNAMIC, Attribut
    - TRANSACTION-Definition 243
- ## E
- Einheiten mit LDC-Listen 215
  - Einzelsitzungsterminals
    - APPC
      - Ressourcendefinitionen, CONNECTION- und SESSIONS-Methode 8
      - Vorteile der TERMINAL-TYPE-TERM-Methode 8



ENQMODEL, Attribut  
 ENQMODEL-Definition 50  
 ENQMODEL-Definition  
 Attribut DESCRIPTION 50  
 Attribut ENQMODEL 50  
 Attribut ENQNAME 50  
 Attribut ENQSCOPE 50  
 Attribut STATUS 51  
 ENQMODEL-Ressourcen 48  
 ENQNAME, Attribut  
 ENQMODEL 50  
 ENQSCOPE, Attribut  
 ENQMODEL-Definition 50  
 ENTRY, Attribut  
 DB2TRAN-Definition 43  
 Ereignis, Überwachungspunkt  
 DFHMCT TYPE=EMP 422  
 ERRATT, Operand  
 DFHTCT TYPE=INITIAL 454  
 ERRCOLOR, Attribut  
 TYPETERM-Definition 290  
 ERRHIGHLIGHT, Attribut  
 TYPETERM-Definition 291  
 ERRINTENSIFY, Attribut  
 TYPETERM-Definition 291  
 ERRLASTLINE, Attribut  
 TYPETERM-Definition 291  
 ERROPTION, Attribut  
 TDQUEUE-Definition 194  
 Erweiterte lokale LDC-Liste und System-  
 LDC-Tabelle  
 DFHTCT TYPE=LDC 460  
 Erweiterte Wiederherstellungsfunktion  
 (XRF = Extended Recovery Facility)  
 Befehlslistentabelle (CLT = Command  
 List Table) 401  
 EXCLUDE, Operand  
 DFHMCT TYPE=RECORD 427  
 EXCNT, Option  
 Operand PERFORM 425  
 EXITPGM, Attribut  
 DOCTEMPLATE-Definition 46  
 EXPAND, Befehl  
 CEDA 353  
 EXPAND GROUP, Befehl  
 CEDA 354  
 EXPAND LIST, Befehl  
 CEDA 355  
 EXPIRYINTMIN, Attribut  
 TSMODEL-Definition 260  
 EXTENDEDDES, Attribut  
 TYPETERM-Definition 291  
 EXTRACT, Befehl, Dienstprogramm  
 DFHCSDUP 385  
 Beispiele 386  
 Option OBJECTS 385  
 Option USERPROGRAM 385  
 EXTSEC, Attribut  
 TRANSACTION-Definition 493

## F

FACILITYID, Attribut  
 TDQUEUE-Definition 194  
 FACILITYLIKE, Attribut  
 PROFILE-Definition 138

FCT (File Control Table; Dateisteuertabel-  
 le) 406  
 Codierungsbeispiel 416  
 DFHFCT-Makros 406  
 DFHFCT TYPE=FILE 407  
 DFHFCT TYPE=INITIAL 407  
 Zusammenfassungstabelle 415  
 Ferne Dateien 53  
 Ferne Definition 217  
 Ferne Terminals für Transaktionsrouting  
 DFHTCT-Makros 471  
 DFHTCT TYPE=REGION-Operan-  
 den 472  
 DFHTCT TYPE=REMOTE-Operan-  
 den 473  
 DFHTCT TYPE=TERMINAL-Operan-  
 den 472  
 Ferner temporärer Speicher, DATAIDs  
 DFHTST TYPE=REMOTE 484  
 FIELD, Operand  
 DFHMCT TYPE=EMP 424  
 FILE, Attribut  
 DOCTEMPLATE-Definition 46  
 FILE-Definition 55  
 PROCESSTYPE-Definition 135  
 FILE, DFHFCT TYPE=, Makro 407  
 FILE, Operand  
 DFHFCT TYPE=FILE 409  
 FILE-Definition  
 Attribut ADD 55  
 Attribut BACKUPTYPE 55, 56  
 Attribut BROWSE 55  
 Attribut CFDTPOOL 55  
 Attribut DATABUFFERS 55  
 Attribut DELETE 55  
 Attribut DESCRIPTION 55  
 Attribut DISPOSITION 55  
 Attribut DSNNAME 55  
 Attribut DSNSHARING 55  
 Attribut FILE 55  
 Attribut FWDRECOVLOG 55  
 Attribut INDEXBUFFERS 55  
 Attribut JNLADD 55  
 Attribut JNLREAD 55  
 Attribut JNLSYNCREAD 55  
 Attribut JNLSYNCSWRITE 55  
 Attribut JNLUPDATE 55  
 Attribut JOURNAL 55  
 Attribut KEYLENGTH 55  
 Attribut LOAD 55  
 Attribut LSRPOOLID 55  
 Attribut LSRPOOLNUM 55  
 Attribut MAXNUMRECS 55  
 Attribut NSRGROUP 55  
 Attribut OPENTIME 55  
 Attribut PASSWORD 55  
 Attribut READINTEG 55  
 Attribut RECORDFORMAT 55  
 Attribut RECORDSIZE 55  
 Attribut RECOVERY 55  
 Attribut REMOTENAME 71  
 Attribut REMOTESYSTEM 71  
 Attribut RESSECNUM 493  
 Attribut RLSACCESS 72  
 Attribut STATUS 73  
 Attribut STRINGS 74  
 Attribut TABLE 74

FILE-Definition (*Forts.*)  
 Attribut TABLENAME 75  
 Attribut UPDATE 75  
 Attribut UPDATEMODEL 76  
 FILE-Ressourcen 51  
 FILSTAT, Operand  
 DFHFCT TYPE=FILE 410  
 FINAL, Option  
 Operand LOCAL 463  
 FIXED, Option  
 Operand RECFORM 413  
 FMHPARM, Attribut  
 TYPETERM-Definition 291  
 FORALT, Operand  
 DFHCLT TYPE=LISTSTART 403  
 FORMFEED, Attribut  
 TYPETERM-Definition 292  
 Fortlaufender Druck (Autopaging) 463  
 FWDRECOVLOG, Attribut  
 FILE-Definition 55

## G

Gemeinsame Datennutzung  
 DFHTST TYPE=SHARED 486  
 temporärer Speicher 486  
 Gemeinsame Nutzung, CSD  
 in Releases 506  
 Generische Benennung im Befehl AL-  
 TER 375  
 Generische Benennung im Befehl  
 COPY 379  
 GROUP, Operand  
 DFHTCT TYPE=GROUP 456

## H

HFSFILE, Attribut  
 DOCTEMPLATE-Definition 46  
 URIMAP-Definition 320  
 HIGHLIGHT, Attribut  
 TYPETERM-Definition 293  
 HORIZFORM, Attribut  
 TYPETERM-Definition 293  
 HOST, Attribut  
 IPCONN-Definition 77  
 URIMAP-Definition 321  
 HOSTCODEPAGE, Attribut  
 URIMAP-Definition 322  
 HSDATA12K, Attribut  
 LSRPOOL-Definition 108  
 HSDATA16K, Attribut  
 LSRPOOL-Definition 109  
 HSDATA20K, Attribut  
 LSRPOOL-Definition 109  
 HSDATA24K, Attribut  
 LSRPOOL-Definition 109  
 HSDATA28K, Attribut  
 LSRPOOL-Definition 109  
 HSDATA32K, Attribut  
 LSRPOOL-Definition 109  
 HSDATA4K, Attribut  
 LSRPOOL-Definition 108  
 HSDATA8K, Attribut  
 LSRPOOL-Definition 108

HSINDEX12K, Attribut  
LSRPOOL-Definition 109

HSINDEX16K, Attribut  
LSRPOOL-Definition 109

HSINDEX20K, Attribut  
LSRPOOL-Definition 109

HSINDEX24K, Attribut  
LSRPOOL-Definition 109

HSINDEX28K, Attribut  
LSRPOOL-Definition 109

HSINDEX32K, Attribut  
LSRPOOL-Definition 109

HSINDEX4K, Attribut  
LSRPOOL-Definition 109

HSINDEX8K, Attribut  
LSRPOOL-Definition 109

**I**

ID, Operand  
DFHMCT TYPE=EMP 423

IDPROP, Attribut  
IPCONN-Definition 77

IMS  
CICS-IMS-Verbindungen und -Sitzungen definieren 9

INAREAL, Operand  
DFHTCT TYPE=LINE 467  
sequenzielle Einheiten 467

INBFMH, Attribut  
PROFILE-Definition 139

INCLUDE, Operand  
DFHMCT TYPE=RECORD 440

INDEX12K, Attribut  
LSRPOOL-Definition 110

INDEX16K, Attribut  
LSRPOOL-Definition 110

INDEX1K, Attribut  
LSRPOOL-Definition 110

INDEX20K, Attribut  
LSRPOOL-Definition 110

INDEX24K, Attribut  
LSRPOOL-Definition 110

INDEX28K, Attribut  
LSRPOOL-Definition 110

INDEX2K, Attribut  
LSRPOOL-Definition 110

INDEX32K, Attribut  
LSRPOOL-Definition 110

INDEX4K, Attribut  
LSRPOOL-Definition 110

INDEX512, Attribut  
LSRPOOL-Definition 110

INDEX8K, Attribut  
LSRPOOL-Definition 110

INDEXBUFFERS, Attribut  
FILE-Definition 55

INDIRECT-Verbindungen 159  
Ressourcendefinitionen 10

INDIRECTNAME, Attribut  
TDQUEUE-Definition 195

Indirekte Verbindungen definieren 10

INDOUBT, Attribut  
TRANSACTION-Definition 493

INDSYS, Attribut  
CONNECTION  
für INDIRECT-Verbindungen 10  
CONNECTION-Definition 16

INITIAL, Option  
Operand LOCAL 463

INITIALIZE, Befehl, Dienstprogramm  
DFHCSDUP 386

INITQNAME, Attribut  
MQCONN-Definition 115

INPUT, Option  
Operand TRMSTAT 469

INSERVICE, Attribut  
CONNECTION-Definition 16  
IPCONN-Definition 77  
SESSIONS-Definition 493  
TERMINAL-Definition 226

INSTALL, Befehl  
CEDA 356

INTENSIFY, Option  
Operand ERRATT 455

IOAREALEN, Attribut  
SESSIONS-Definition 163  
TYPETERM-Definition 293

IPADDRESS, Attribut  
TCPIPService-Definition 173

IPCONN, Attribut  
IPCONN-Definition 77

IPCONN-Definition  
Attribut APPLID 77  
Attribut AUTOCONNECT 77  
Attribut CERTIFICATE 77  
Attribut CIPHERS 77  
Attribut DESCRIPTION 77  
Attribut HOST 77  
Attribut IDPROP 77  
Attribut INSERVICE 77  
Attribut IPCONN 77  
Attribut MAXQTIME 77  
Attribut MIRRORLIFE 77  
Attribut NETWORKID 77  
Attribut PORT 77  
Attribut QUEUELIMIT 77  
Attribut RECEIVECOUNT 77  
Attribut SECURITYNAME 77  
Attribut SENDCOUNT 77  
Attribut SSL 77  
Attribut TCPIPService 77  
Attribut USERAUTH 77  
Attribut XLNAction 77

ISADSCN, Operand  
DFHTCT TYPE=LINE 467  
sequenzielle Einheiten 467

ISOLATE, Attribut  
TRANSACTION-Definition 244

**J**

JES-Operanden  
DFHCLT TYPE=INITIAL 402

JESCHAR-Operanden  
DFHCLT TYPE=INITIAL 402

JESID, Operand  
DFHCLT TYPE=INITIAL 402

JID, Operand  
DFHFCT TYPE=FILE 411

JNLADD, Attribut  
FILE-Definition 55

JNLREAD, Attribut  
FILE-Definition 55

JNLSYNCREAD, Attribut  
FILE-Definition 55

JNLSYNCWRITE, Attribut  
FILE-Definition 55

JNLUPDATE, Attribut  
FILE-Definition 55

JOBNAME, Operand  
DFHRST TYPE=SUBSYS 448

JOURNAL, Attribut  
FILE-Definition 55  
PROFILE-Definition 139

JOURNALMODEL, Attribut  
JOURNALMODEL-Definition 90

JOURNALMODEL-Definition  
Attribut JOURNALMODEL 90  
Attribut JOURNALNAME 90  
Attribut STREAMNAME 92  
Attribut TYPE 95

JOURNALMODEL-Ressourcen 89

JOURNALNAME, Attribut  
JOURNALMODEL-Definition 90

JREQ, Operand  
DFHFCT TYPE=FILE 411

JVMSEVER-Ressourcen 97

**K**

KATAKANA, Attribut  
TYPETERM-Definition 294

KEY, Option  
Operand SERVREQ 414

KEYLEN, Operand  
DFHFCT TYPE=FILE 412

KEYLENGTH, Attribut  
FILE-Definition 55

Konfiguratoren  
DFHFCT TYPE=FILE 415

Konsolen  
voreingestellte Sicherheit 234

**L**

LASTLINE, Option  
Operand ERRATT 455

LDC (Logical Device Code; Code für logische Einheiten) 456, 464

LDC, Operand  
DFHTCT TYPE=LDC 461  
DFHTCT TYPE=LDCLIST 464  
DFHTLT TYPE=INITIAL 476

LDC-Liste  
erweiterte lokale und System-LDC-Tabelle, DFHTCT TYPE=LDC 460  
lokal, DFHTCT TYPE=LDCLIST 464

LDC-Listen  
Einheiten mit 215

LDCLIST, Attribut  
TYPETERM-Definition 294

LDCs für logische Einheit mit 3770-Stapel  
TCT-Beispiel 465

LIBRARY-Definition  
Attribut CRITICAL 101

LIBRARY-Definition (*Forts.*)  
 Attribut DSNAME 102  
 Attribut RANKING 104  
 Attribut STATUS 105  
 LIGHTPEN, Attribut  
 TYPETERM-Definition 295  
 LINSTAT='OUT OF SERVICE', Operand  
 DFHTCT TYPE=LINE 467  
 sequenzielle Einheiten 467  
 LIST, Befehl, Dienstprogramm DFHCS-  
 DUP 387  
 Beispieler 388  
 Option OBJECTS 388  
 LOAD, Attribut  
 FILE-Definition 55  
 LOCAL, Operand  
 DFHTCT TYPE=LDC 463  
 LOCALQ, Attribut  
 TRANSACTION-Definition 245  
 LOCATION, Attribut  
 TSMODEL-Definition 261  
 URIMAP-Definition 322  
 LOCK, Befehl  
 CEDA 358  
 LOG, Operand  
 DFHFCT TYPE=FILE 412  
 LOGMODE, Attribut  
 TYPETERM-Definition 295  
 LOGMODECOM, Attribut  
 TRANSACTION-Definition 493  
 LOGONMSG, Attribut  
 TYPETERM-Definition 296  
 LOGREC, Attribut  
 PROFILE-Definition 139  
 Lokale Definition 217  
 Lokale LDC-Liste  
 DFHTCT TYPE=LDCLIST 464  
 Lokaler temporärer Speicher, DATAIDs  
 DFHTST TYPE=LOCAL 483  
 LPLEN, Operand  
 DFHTCT TYPE=TERMINAL 468  
 sequenzielle Einheiten 468  
 LRECL, Operand  
 DFHFCT TYPE=FILE 413  
 LSRPOOL, Attribut  
 LSRPOOL-Definition 110  
 LSRPOOL-Definition  
 Attribut DATA12K 107  
 Attribut DATA16K 107  
 Attribut DATA1K 107  
 Attribut DATA20K 108  
 Attribut DATA24K 108  
 Attribut DATA28K 108  
 Attribut DATA2K 107  
 Attribut DATA32K 108  
 Attribut DATA4K 107  
 Attribut DATA512 107  
 Attribut DATA8K 107  
 Attribut HSDATA12K 108  
 Attribut HSDATA16K 109  
 Attribut HSDATA20K 109  
 Attribut HSDATA24K 109  
 Attribut HSDATA28K 109  
 Attribut HSDATA32K 109  
 Attribut HSDATA4K 108  
 Attribut HSDATA8K 108  
 Attribut HSINDEX12K 109

LSRPOOL-Definition (*Forts.*)  
 Attribut HSINDEX16K 109  
 Attribut HSINDEX20K 109  
 Attribut HSINDEX24K 109  
 Attribut HSINDEX28K 109  
 Attribut HSINDEX32K 109  
 Attribut HSINDEX4K 109  
 Attribut HSINDEX8K 109  
 Attribut INDEX12K 110  
 Attribut INDEX16K 110  
 Attribut INDEX1K 110  
 Attribut INDEX20K 110  
 Attribut INDEX24K 110  
 Attribut INDEX28K 110  
 Attribut INDEX2K 110  
 Attribut INDEX32K 110  
 Attribut INDEX4K 110  
 Attribut INDEX512 110  
 Attribut INDEX8K 110  
 Attribut LSRPOOL 110  
 Attribut LSRPOOLID 111, 493  
 Attribut LSRPOOLNUM 111  
 Attribut MAXKEYLENGTH 111  
 Attribut SHARELIMIT 111  
 Attribut STRINGS 111  
 LSRPOOL-Ressourcen 105  
 LSRPOOLID, Attribut  
 FILE-Definition 55  
 LSRPOOL-Definition 111, 493  
 LSRPOOLNUM, Attribut  
 FILE-Definition 55  
 LSRPOOL-Definition 111  
 LUTYPE4, Option  
 Operand LDC 462  
 LUTYPE6.1  
 CICS-CICS-ISC-Verbindungen und  
 -Sitzungen  
 Ressourcendefinitionen 8  
 CICS-IMS-Verbindungen und -Sitzun-  
 gen  
 Ressourcendefinitionen 9  
 MRO-Sitzungen  
 Ressourcendefinitionen 7  
 LUTYPE6.1-CICS-CICS-ISC-Verbindungen  
 und -Sitzungen 158  
 LUTYPE6.1-CICS-IMS-Verbindungen und  
 -Sitzungen 158

## M

MACRF, Operand  
 DFHTCT TYPE=SDSCI 466  
 sequenzielle Einheiten 466  
 MAPSET-Definition  
 Attribut MAPSET 113  
 Attribut RESIDENT 113  
 Attribut RSL 493  
 Attribut USAGE 113  
 Attribut USELPACOPY 114  
 MAPSET-Ressourcen 111  
 MAXACTIVE, Attribut  
 TRANCLASS-Definition 236  
 MAXDATALEN, Attribut  
 TCPIPService-Definition 173  
 MAXIMUM, Attribut  
 für APPC-Parallelsitzungen 8

MAXIMUM, Attribut (*Forts.*)  
 SESSIONS  
 für APPC-Einzelsitzungstermi-  
 nals 8  
 SESSIONS-Definition 164  
 MAXKEYLENGTH, Attribut  
 LSRPOOL-Definition 111  
 MAXNUMRECS, Attribut  
 FILE-Definition 55  
 MAXPERSIST, Attribut  
 TCPIPService-Definition 173  
 MAXQTIME, Attribut  
 CONNECTION-Definition 17  
 IPCONN-Definition 77  
 MCT (Monitoring Control Table; Überwa-  
 chungssteuertabelle) 416  
 Codierungsbeispiele 441  
 DFHMCT-Makros 417  
 DFHMCT TYPE=EMP 422  
 DFHMCT TYPE=INITIAL 417  
 DFHMCT TYPE=RECORD 427  
 Ereignisüberwachungspunkte des Be-  
 nutzers 422  
 Steuerdatenaufzeichnung 427  
 MEDIATYPE, Attribut  
 URIMAP-Definition 323  
 Mehrregionenbetrieb (MRO)  
 Ressourcendefinitionen für Verbindun-  
 gen und Sitzungen 7  
 MEMBERNAME, Attribut  
 DOCTEMPLATE-Definition 47  
 MF, Operand  
 DFHCLT TYPE=WTO 404  
 MIGRATE, Operand  
 DFHTCT TYPE=INITIAL 455  
 DFHTST TYPE=INITIAL 480  
 Migrieren, TCT-Einträge in RDO 456  
 MIRRORLIFE, Attribut  
 IPCONN-Definition 77  
 Mit Doppelfunktion, Definition 218  
 Mit Doppelfunktion, Ressourcendefinition  
 für transiente Daten 188  
 MLTCNT, Option  
 Operand PERFORM 425  
 MODENAME, Attribut  
 PROFILE-Definition 140  
 SESSIONS  
 für APPC-Einzelsitzungstermi-  
 nals 8  
 für APPC-Parallelsitzungen 8  
 SESSIONS-Definition 164  
 TERMINAL-Definition 226  
 MOVE, Befehl  
 CEDA 360  
 MQCONN  
 installieren 114  
 MQCONN, Attribut  
 MQCONN-Definition 115  
 MQCONN-Definition  
 Attribut INITQNAME 115  
 Attribut MQCONN 115  
 Attribut MQNAME 115  
 Attribut RESYNCMEMBER 115  
 MQCONN-Ressourcen 114  
 MQNAME, Attribut  
 MQCONN-Definition 115  
 MRO-Verbindungen und -Sitzungen 158

MSGINTEG, Attribut  
 PROFILE-Definition 140  
 MSGJRNL, Attribut  
 PROFILE-Definition 140  
 MSGQUEUE1, Attribut  
 DB2CONN-Definition 25  
 MSGQUEUE2, Attribut  
 DB2CONN-Definition 25  
 MSGQUEUE3, Attribut  
 DB2CONN-Definition 25  
 MSRCONTROL, Attribut  
 TYPETERM-Definition 297  
 MXSSASZ, Operand  
 DFHDLPSB TYPE=ENTRY 406

## N

NACNT, Option  
 Operand PERFORM 426  
 NAME, Attribut  
 DOCTEMPLATE-Definition 46  
 NATLANG, Attribut  
 TERMINAL-Definition 227  
 NEPCLASS, Attribut  
 PROFILE-Definition 141  
 SESSIONS-Definition 164  
 TYPETERM-Definition 297  
 NETNAME, Attribut  
 CONNECTION 18  
 für INDIRECT-Verbindungen 10  
 CONNECTION-Definition 18  
 PARTNER-Definition 128  
 TERMINAL-Definition 227  
 NETNAMEQ, Attribut  
 SESSIONS  
 für LUTYPE6.1-CICS-IMS-Sitzungen 9  
 SESSIONS-Definition 165  
 NETWORK, Attribut  
 PARTNER-Definition 128  
 NETWORKID, Attribut  
 IPCONN-Definition 77  
 NOEXCTL, Option  
 Operand SERVREQ 414  
 NONTERMREL, Attribut  
 DB2CONN-Definition 25  
 NONVTAM, Option  
 Angabe von  
 ACCMETH=NONVTAM 454  
 NSRGROUP, Attribut  
 FILE-Definition 55

## O

OBFORMAT, Attribut  
 TYPETERM-Definition 297  
 OBJECTS, Option  
 Befehl LIST (DFHCSDUP) 388  
 OBOPERID, Attribut  
 TYPETERM-Definition 297  
 ONEWTE, Attribut  
 PROFILE-Definition 141  
 OPENTIME, Attribut  
 FILE-Definition 55  
 TDQUEUE-Definition 195

OPERID, Attribut  
 SESSIONS-Definition 493  
 OPERPRIORITY, Attribut  
 SESSIONS-Definition 493  
 OPERRSL, Attribut  
 SESSIONS-Definition 493  
 OPERSECURITY, Attribut  
 SESSIONS-Definition 493  
 ORCNT, Option  
 Operand PERFORM 426  
 OSADSCN, Operand  
 DFHTCT TYPE=LINE 467  
 sequenzielle Einheiten 467  
 OUT OF SERVICE, Option  
 Operand LINSTAT 467  
 Operand TRMSTAT 470  
 OUTLINE, Attribut  
 TYPETERM-Definition 297

## P

PAGESIZE, Attribut  
 TYPETERM-Definition 297  
 Parallel, APPC-Sitzungen 8  
 Parallelsitzungen und APPC-Verbindungen (LUTYPE6.2) 158  
 PARTITIONS, Attribut  
 TYPETERM-Definition 299  
 PARTITIONSET, Attribut  
 TRANSACTION-Definition 246  
 PARTITIONSET-Definition  
 Attribut PARTITIONSET 125  
 Attribut RESIDENT 125  
 Attribut RSL 493  
 Attribut STATUS 125  
 Attribut USAGE 126  
 Attribut USELPACOPY 126  
 PARTITIONSET-Ressourcen 123  
 PARTNER, Attribut  
 PARTNER-Definition 128  
 PARTNER-Definition  
 Attribut NETNAME 128  
 Attribut NETWORK 128  
 Attribut PARTNER 128  
 Attribut PROFILE 128  
 Attribut TPNAME 128  
 Attribut XTPNAME 129  
 PARTNER-Ressourcen 126  
 PASSWORD, Attribut  
 FILE-Definition 55  
 PATH, Attribut  
 URIMAP-Definition 323  
 PCLOCK, Option  
 Operand PERFORM 426  
 PCPUCLK, Option  
 Operand PERFORM 426  
 PDIR (Program Specification Block Directory; Verzeichnis für Programmspezifikationsblöcke)  
 DFHDLPSB-Makros 405  
 DFHDLPSB TYPE=ENTRY 405  
 DFHDLPSB TYPE=INITIAL 405  
 PERFORM, Operand  
 DFHMCT TYPE=EMP 424  
 PGESIZE, Operand  
 DFHTCT TYPE=LDC 463  
 DFHTCT TYPE=TERMINAL 468

PGESIZE, Operand (Forts.)  
 sequenzielle Einheiten 468  
 PGESTAT, Operand  
 DFHTCT TYPE=LDC 463  
 PIPELINE, Attribut  
 URIMAP-Definition 325  
 Pipeline, TERMINAL-Definitionen 214  
 PIPELINE-Definition  
 Attribut RESPWAIT 131  
 Attribut SHELF 132  
 Attribut WSDIR 132  
 PIPELINE-Ressourcen 129  
 Pipeline-Terminals 278  
 PLAN, Attribut  
 DB2CONN-Definition 31  
 DB2ENTRY-Definition 41  
 PLANEXITNAME, Attribut  
 DB2CONN-Definition 31  
 DB2ENTRY-Definition 41  
 Platzhalterzeichen als Transaktions-IDs 43  
 PLT (Program List Table; Programmlistentabelle) 442  
 Codierungsbeispiel 445  
 DFHPLT TYPE=ENTRY 444  
 DFHPLT TYPE=INITIAL 443  
 Programmabschnitt 443  
 POOL, Attribut  
 TERMINAL-Definition 228  
 POOLNAME, Attribut  
 TSMODEL-Definition 261  
 PORT, Attribut  
 IPCONN-Definition 77  
 URIMAP-Definition 325  
 PORTNUMBER, Attribut  
 TCIPSERVICE-Definition 173  
 PREFIX, Attribut  
 TSMODEL-Definition 262  
 PRIMEDSIZE, Attribut  
 TRANSACTION 493  
 PRINTADAPTER, Attribut  
 TYPETERM-Definition 299  
 PRINTCONTROL, Attribut  
 TDQUEUE-Definition 195  
 PRINTER, Attribut  
 TERMINAL-Definition 228  
 PRINTERCOMP, Attribut  
 PROFILE-Definition 141  
 PRINTERCOPY, Attribut  
 TERMINAL-Definition 229  
 PRIORITY, Attribut  
 DB2ENTRY-Definition 41  
 TRANSACTION-Definition 247  
 PRIVACY, Attribut  
 TCIPSERVICE-Definition 493  
 PROCESS, Befehl, Dienstprogramm  
 DFHCSDUP 389  
 PROCESSTYPE, Attribut  
 PROCESSTYPE-Definition 136  
 PROCESSTYPE-Definition  
 Attribut AUDITLEVEL 135  
 Attribut AUDITLOG 135  
 Attribut DESCRIPTION 135  
 Attribut FILE 135  
 Attribut PROCESSTYPE 136  
 PROCESSTYPE-Ressourcen 133

PROFILE, Attribut  
     PARTNER-Definition 128  
     PROFILE-Definition 141  
     TRANSACTION-Definition 247  
 Profile, von CICS bereitgestellte Standardwerte  
     DFHCICSA 520  
     DFHCICSE 521  
     DFHCICSF 519  
     DFHCICSP 521  
     DFHCICSR 519  
     DFHCICSS 520  
     DFHCICST 521  
     DFHCICSV 521  
     DFHECEPH 519  
     DFHPGAMP 523  
     DFHPGAPG 523  
     DFHPGAPT 523  
     DFHPPF01 522  
     DFHPPF02 522  
 PROFILE-Definition  
     Attribut CHAINCONTROL 138  
     Attribut DVSUPRT 138  
     Attribut FACILITYLIKE 138  
     Attribut INBFMH 139  
     Attribut JOURNAL 139  
     Attribut LOGREC 139  
     Attribut MODENAME 140  
     Attribut MSGINTEG 140  
     Attribut MSGJRNL 140  
     Attribut NEPCCLASS 141  
     Attribut ONEWTE 141  
     Attribut PRINTERCOMP 141  
     Attribut PROFILE 141  
     Attribut PROTECT 493  
     Attribut RAQ 142  
     Attribut RTIMOUT 142  
     Attribut SCRNSIZE 142  
     Attribut UCTRAN 143  
     von CICS bereitgestellt 519, 522  
 PROFILE-Ressourcen 136  
 PROGRAM, Attribut  
     DOCTEMPLATE-Definition 47  
     TRANSACTION-Definition 247  
     URIMAP-Definition 325  
 PROGRAM, Operand  
     DFHPLT TYPE=ENTRY 444  
 PROGRAM-Definition  
     Attribut LANGUAGE  
         RPG 493  
     Attribut RSL 493  
 PROGRAM-Ressourcen 144  
 Programmlistentabelle (PLT = Program List Table) 442  
     Codierungsbeispiel 445  
     DFHPLT TYPE=ENTRY 444  
     DFHPLT TYPE=INITIAL 443  
     Programmabschnitt 443  
 Programmspezifikationsblock (PSB = Program Specification Block) 405, 406  
 PROGSYMBOLS, Attribut  
     TYPETERM-Definition 300  
 PROTECT, Attribut  
     PROFILE 493  
 PROTECTNUM, Attribut  
     DB2ENTRY-Definition 41  
 PROTOCL(HTTP) TCPIP SERVICES 185

PROTOCOL, Attribut  
     CONNECTION  
         für APPC-Einzelsitzungsterminals 8  
         für APPC-Verbindungen 8  
         für LUTYPE6.1-CICS-IMS-Verbindungen 9  
         für LUTYPE6.1-Verbindungen 8  
         für MRO-Verbindungen 7  
     CONNECTION-Definition 19  
     SESSIONS  
         für APPC-Einzelsitzungsterminals 8  
         für APPC-Parallelsitzungen 8  
         für LUTYPE6.1-CICS-IMS-Sitzungen 9  
         für LUTYPE6.1-Sitzungen 8  
         für MRO-Sitzungen 7  
     SESSIONS-Definition 165  
     TCPIP SERVICE-Definition 173  
 PSB, Operand  
     DFHDLPSB TYPE=ENTRY 406  
 PSB (Program Specification Block; Programmspezifikationsblock) 405, 406  
 PSB-Verzeichnis (PDIR)  
     DFHDLPSB-Makros 405  
     DFHDLPSB TYPE=ENTRY 405  
     DFHDLPSB TYPE=INITIAL 405  
 Pseudo-TDQ 210  
 PSRECOVERY, Attribut  
     CONNECTION-Definition 20  
 PURGECYCLE, Attribut  
     DB2CONN-Definition 25  
 PURGETHRESH, Attribut  
     TRANCLASS-Definition 236

## Q

QUERY, Attribut  
     TYPETERM-Definition 300  
 QUEUELIMIT, Attribut  
     CONNECTION-Definition 20  
     IPCONN-Definition 77

## R

RANKING, Attribut  
     LIBRARY-Definition 104  
 RAQ, Attribut  
     PROFILE-Definition 142  
 RDO, auswählbar, TCT-Einträge, DFHRDT 453  
 READ, Option  
     Operand SERVREQ 415  
 READINTEG, Attribut  
     FILE-Definition 55  
 REALM, Attribut  
     TCPIP SERVICE-Definition 173  
 RECEIVE, Option  
     Operand TRMSTAT 470  
 RECEIVECOUNT, Attribut  
     IPCONN-Definition 77  
     SESSIONS  
         für LUTYPE6.1-CICS-IMS-Sitzungen 9  
         für LUTYPE6.1-Sitzungen 8

RECEIVECOUNT, Attribut (*Forts.*)  
     SESSIONS (*Forts.*)  
         für MRO-Sitzungen 7  
     SESSIONS-Definition 165  
 RECEIVEPF, Attribut  
     SESSIONS  
         für LUTYPE6.1-Sitzungen 8  
         für MRO-Sitzungen 7  
     SESSIONS-Definition 166  
 RECEIVESIZE, Attribut  
     SESSIONS-Definition 167  
     TYPETERM-Definition 300  
 RECFM, Operand  
     DFHTCT TYPE=SDSCI 467  
     sequenzielle Einheiten 467  
 RECFORM, Operand  
     DFHFCT TYPE=FILE 413  
 RECORDFORMAT, Attribut  
     CONNECTION-Definition 21  
     FILE-Definition 55  
     TDQUEUE-Definition 195  
 RECORDSIZE, Attribut  
     FILE-Definition 55  
     TDQUEUE-Definition 196  
 RECOVER, Operand  
     DFHSRT TYPE=SYSTEM 450  
     DFHSRT TYPE=USER 450  
 RECOVERY, Attribut  
     FILE-Definition 55  
 RECOVNOTIFY, Attribut  
     SESSIONS-Definition 493  
     TYPETERM-Definition 301  
 RECOVOPTION, Attribut  
     SESSIONS-Definition 167  
     TYPETERM-Definition 302  
 RECOVSTATUS, Attribut  
     TDQUEUE-Definition 196  
     TSMODEL-Definition 262  
 REDIRECTTYPE, Attribut  
     URIMAP-Definition 326  
 RELREQ, Attribut  
     SESSIONS-Definition 169  
     TYPETERM-Definition 304  
 RELTYPE, Operand  
     DFHFCT TYPE=FILE 414  
 REMOTELENGTH, Attribut  
     TDQUEUE-Definition 198  
 REMOTENAME, Attribut  
     CONNECTION-Definition 21  
     FILE-Definition 71  
     TDQUEUE-Definition 198  
     TERMINAL-Definition 229  
     TRANSACTION-Definition 248  
 REMOTEPREFIX, Attribut  
     TSMODEL-Definition 263  
 REMOTESYSNET, Attribut  
     CONNECTION-Definition 21  
     TERMINAL-Definition 229  
 REMOTESYSTEM, Attribut  
     CONNECTION-Definition 22  
     FILE-Definition 71  
     TDQUEUE-Definition 198  
     TERMINAL-Definition 230  
     TRANSACTION-Definition 248  
     TSMODEL-Definition 263  
 REMOVE, Befehl  
     CEDA 362

REMOVE, Befehl, Dienstprogramm  
 DFHCSDUP 390  
 Beispiele 390  
 RENAME, Befehl  
 CEDA 364  
 RESOURCENAME, Attribut  
 ATOMSERVICE-Definition 3  
 RESOURCETYPE, Attribut  
 ATOMSERVICE-Definition 4  
 RESPWAIT, Attribut  
 PIPELINE-Definition 131  
 RESSEC, Attribut  
 TRANSACTION-Definition 249  
 RESSECNUM, Attribut  
 FILE-Definition 493  
 Ressourcendefinitionen  
 für APPC-Einzelsitzungsterminals  
 CONNECTION- und SESSIONS-  
 Methode 8  
 Vorteile der TERMINAL-TYPE-  
 TERM-Methode 8  
 für APPC-Verbindungen und -Parallel-  
 sitzungen 8  
 für INDIRECT-Verbindungen 10  
 für LUTYPE6.1-CICS-CICS-ISC-Verbin-  
 dungen und -Sitzungen 7  
 für LUTYPE6.1-CICS-IMS-Verbindun-  
 gen und -Sitzungen 9  
 für LUTYPE6.1-Verbindungen und  
 -Sitzungen 8  
 für MRO-Verbindungen und -Sitzun-  
 gen 7  
 RESTART, Attribut  
 TRANSACTION-Definition 249  
 RESYNCMEMBER, Attribut  
 DB2CONN-Definition 25  
 MQCONN-Definition 115  
 REWIND, Attribut  
 TDQUEUE-Definition 198  
 RKP, Operand  
 DFHFCT TYPE=FILE 414  
 RLSACCESS, Attribut  
 FILE-Definition 72  
 RLSACCESS(YES)  
 Auswirkungen auf andere FILE-Attri-  
 bute 72  
 RMTNAME, Operand  
 DFHDLPSB TYPE=ENTRY 406  
 DFHTCT TYPE=REMOTE 473  
 DFHTCT TYPE=TERMINAL 472  
 DFHTST TYPE=REMOTE 485  
 ferne Terminals 472, 473  
 ROUTABLE, Attribut  
 TRANSACTION-Definition 249  
 ROUTCDE, Operand  
 DFHCLT TYPE=WTO 404  
 ROUTEDMSG, Attribut  
 TYPETERM-Definition 304  
 RPG-Wert  
 für Attribut LANGUAGE  
 von PROGRAM-Definition 493  
 RSL, Attribut  
 MAPSET-Definition 493  
 PARTITIONSET-Definition 493  
 PROGRAM-Definition 493  
 TRANSACTION-Definition 493

RST (Recoverable Service Table; wieder-  
 herstellbare Servicetabelle) 447, 449  
 Codierungsbeispiel 448  
 DFHRST-Makrotypen 447  
 DFHRST TYPE=INITIAL 447  
 DFHRST TYPE=RSE 447  
 DFHRST TYPE=SUBSYS 448  
 RSTSIGNOFF, Attribut  
 TYPETERM-Definition 304  
 RTIMOUT, Attribut  
 PROFILE-Definition 142  
 RUNAWAY, Attribut  
 TRANSACTION-Definition 250

## S

SCAN, Befehl, Dienstprogramm DFHCS-  
 DUP 390  
 SCHEME, Attribut  
 URIMAP-Definition 326  
 SCLOCK, Option  
 Operand PERFORM 426  
 SCPUCLK, Option  
 Operand PERFORM 426  
 SCRNSIZE, Attribut  
 PROFILE-Definition 142  
 SECURITY, Attribut  
 TSMODEL-Definition 263  
 SECURITYNAME, Attribut  
 CONNECTION-Definition 22  
 IPCONN-Definition 77  
 TERMINAL-Definition 231  
 Segmentsuchargument, PSB 406  
 SENDCOUNT, Attribut  
 IPCONN-Definition 77  
 SESSIONS  
 für LUTYPE6.1-CICS-IMS-Sitzun-  
 gen 9  
 für LUTYPE6.1-Sitzungen 8  
 für MRO-Sitzungen 7  
 SESSIONS-Definition 169  
 SENDPFX, Attribut  
 SESSIONS  
 für LUTYPE6.1-Sitzungen 8  
 für MRO-Sitzungen 7  
 SESSIONS-Definition 169  
 SENDSIZE, Attribut  
 SESSIONS-Definition 170  
 TYPETERM-Definition 304  
 Sequenzielle Terminaleinheiten  
 DFHTCT TYPE=LINE 465  
 DFHTCT TYPE=SDSCI 465  
 DFHTCT TYPE=TERMINAL 465  
 SERVICE, Befehl, Dienstprogramm  
 DFHCSDUP 393  
 Operand FROMCSD 393  
 Operand LEVEL 393  
 SERVREQ, Operand  
 DFHFCT TYPE=FILE 414  
 SESSIONS, Attribut  
 SESSIONS-Definition 171  
 SESSIONS-Definition  
 Attribut AUTOCONNECT 161  
 Attribut BUILDCHAIN 162  
 Attribut CONNECTION 162  
 Attribut DESCRIPTION 162  
 Attribut DISCREQ 162

SESSIONS-Definition (Forts.)  
 Attribut INSERVICE 493  
 Attribut IOAREALEN 163  
 Attribut MAXIMUM 164  
 Attribut MODENAME 164  
 Attribut NEPCCLASS 164  
 Attribut NETNAMEQ 165  
 Attribut OPERID 493  
 Attribut OPERPRIORITY 493  
 Attribut OPERRSL 493  
 Attribut OPERSECURITY 493  
 Attribut PROTOCOL 165  
 Attribut RECEIVECOUNT 165  
 Attribut RECEIVEDPFX 166  
 Attribut RECEIVESIZE 167  
 Attribut RECOVNOTIFY 493  
 Attribut RECOVOPTION 167  
 Attribut RELREQ 169  
 Attribut SENDCOUNT 169  
 Attribut SENDPFX 169  
 Attribut SENDSIZE 170  
 Attribut SESSIONS 171  
 Attribut SESSNAME 171  
 Attribut SESSPRIORITY 171  
 Attribut TRANSACTION 493  
 Attribut USERAREALEN 171  
 Attribut USERID 171  
 Befehl RDO 159  
 Installation 158  
 SESSIONS-Ressourcen 158  
 SESSIONTYPE, Attribut  
 TYPETERM-Definition 305  
 SESSNAME, Attribut  
 SESSIONS  
 für LUTYPE6.1-CICS-IMS-Sitzun-  
 gen 9  
 SESSIONS-Definition 171  
 SESSPRIORITY, Attribut  
 SESSIONS-Definition 171  
 SHARELIMIT, Attribut  
 LSRPOOL-Definition 111  
 SHELF, Attribut  
 PIPELINE-Definition 132  
 SHIPPABLE, Attribut  
 TYPETERM-Definition 305  
 SHUTDOWN, Attribut  
 TRANSACTION-Definition 250  
 Sicherheit  
 voreingestellt 470, 493  
 Sicherheitsprüfung  
 DFHTST TYPE=SECURITY 486  
 temporärer Speicher 486  
 SIGNID, Attribut  
 DB2CONN-Definition 25  
 SIGNOFF, Attribut  
 TYPETERM-Definition 306  
 SINGLESESS, Attribut  
 CONNECTION  
 für APPC-Einzelsitzungstermi-  
 nals 8  
 CONNECTION-Definition 22  
 Sitzungen und LUTYPE6.1-CICS-CICS-  
 ISC-Verbindungen 158  
 Sitzungen und LUTYPE6.1-CICS-IMS-  
 Verbindungen 158  
 Sitzungen und MRO-Verbindungen 158  
 SOCKETCLOSE, Attribut 173

SOLICITED, Attribut  
 TERMINAL-Definition 231

SOSI, Attribut  
 TYPETERM-Definition 306

Speicherwarteschlangen, DFHTST  
 TYPE=LOCAL  
 lokal temporär 483

Speicherwarteschlangen, DFHTST  
 TYPE=REMOTE  
 ferne temporäre 484

SPURGE, Attribut  
 TRANSACTION-Definition 250

SRCHM, Operand  
 DFHFCT TYPE=FILE 415

SRT (System Recovery Table; Systemwiederherstellungstabelle)  
 Codierungsbeispiel 451  
 DFHSRT TYPE=INITIAL 449

SSL, Attribut  
 IPCONN-Definition 77

Standardwerte  
 TYPETERM-Attribute 267

STANDBYMODE, Attribut  
 DB2CONN-Definition 25

STATSQUEUE, Attribut  
 DB2CONN-Definition 25

STATUS, Attribut  
 ATOMSERVICE-Definition 4  
 ENQMODEL-Definition 51  
 FILE-Definition 73  
 LIBRARY-Definition 105  
 TRANSACTION-Definition 251  
 URIMAP-Definition 327

STORAGECLEAR, Attribut  
 TRANSACTION-Definition 251

STREAMNAME, Attribut  
 JOURNALMODEL-Definition 92

STRINGS, Attribut  
 FILE-Definition 74  
 LSRPOOL-Definition 111

SUBCNT, Option  
 Operand PERFORM 426

SUBSYSID, Operand  
 DFHRST TYPE=SUBSYS 448

SUFFIX, Operand  
 DFHTLT TYPE=INITIAL 476

SYSIDNT, Operand  
 DFHDLPSB TYPE=ENTRY 406  
 DFHSIT TYPE=INITIAL 456  
 DFHTCT TYPE=INITIAL 455  
 DFHTCT TYPE=REGION 472  
 DFHTCT TYPE=REMOTE 473  
 DFHTCT TYPE=TERMINAL 472  
 DFHTST TYPE=REMOTE 485  
 ferne Terminals 472, 473  
 Kleinschreibung in TCT 453

SYSOUTCLASS, Attribut  
 TDQUEUE-Definition 199

System, LDC-Tabelle und erweiterte lokale LDC-Liste  
 DFHTCT TYPE=LDC 460

SYSTEM, Option  
 Operand LDC 462

Systemwiederherstellungstabelle (SRT = System Recovery Table)  
 Codierungsbeispiel 451  
 DFHSRT TYPE=INITIAL 449

## T

Tabelle für temporären Speicher (TST = Temporary Storage Table) 478, 487

DATAIDs für fernen temporären Speicher, DFHTST TYPE=REMOTE 484

DATAIDs für lokalen temporären Speicher, DFHTST  
 TYPE=LOCAL 483

DFHTST TYPE=INITIAL 480

DFHTST TYPE=RECOVERY 481, 483, 486

DFHTST TYPE=SHARED 486

gemeinsame Datennutzung, temporärer Speicher 486

gemeinsame Datennutzung für temporären Speicher, DFHTST  
 TYPE=SHARED 486

Programmabschnitt 480

Sicherheitsprüfung für temporären Speicher, DFHTST  
 TYPE=SECURITY 486

wiederherstellbarer temporärer Speicher 481, 483, 486

wiederherstellbarer temporärer Speicher, DFHTST  
 TYPE=RECOVERY 481

TABLE, Attribut  
 FILE-Definition 74

TABlename, Attribut  
 FILE-Definition 75

TASKDATAKEY, Attribut  
 TRANSACTION-Definition 251

TASKDATALOC, Attribut  
 TRANSACTION-Definition 252

TASKLIMIT, Attribut  
 TERMINAL-Definition 231

TASKREQ, Attribut  
 TRANSACTION-Definition 253

TASKREQ, Operand  
 DFHXLTYPE=ENTRY 490

TCBLIMIT, Attribut  
 DB2CONN-Definition 25

TCLASS, Attribut  
 TRANSACTION-Definition 493

TCPIPSERVICE, Attribut  
 IPCONN-Definition 77  
 TCPIPSERVICE-Definition 173  
 URIMAP-Definition 327

TCPIPSERVICE-Definition  
 Attribut ATTACHSEC 173  
 Attribut AUTHENTICATE 173  
 Attribut BACKLOG 173  
 Attribut CERTIFICATE 173  
 Attribut CIPHERS 173  
 Attribut IPADDRESS 173  
 Attribut MAXDATALEN 173  
 Attribut MAXPERSIST 173  
 Attribut PORTNUMBER 173  
 Attribut PRIVACY 493  
 Attribut PROTOCOL 173  
 Attribut REALM 173  
 Attribut SOCKETCLOSE 173  
 Attribut SSL 173  
 Attribut STATUS 173  
 Attribut TCPIPSERVICE 173  
 Attribut TRANSACTION 173  
 Attribut TSQPREFIX 173

TCPIPSERVICE-Definition (Forts.)  
 Attribut URM 173

TCPIPSERVICE-Ressourcen 172

TCT (Terminal Control Table; Terminalsteuertabelle) 452, 475  
 DFHTCT-Makros, Zusammenfassung 452  
 DFHTCT TYPE=GROUP 456  
 Einführung 452  
 Einträge in RDO migrieren 456  
 Liste der CICS-Terminals 474  
 sequenzielle Einheiten 465

TCTUAL, Operand  
 DFHTCT TYPE=LINE 467  
 DFHTCT TYPE=TERMINAL 468  
 sequenzielle Einheiten 467, 468

TDQUEUE, Attribut  
 DOCTEMPLATE-Definition 47  
 TDQUEUE-Definition 200

TDQUEUE-Definition  
 Attribut ATIFACILITY 191  
 Attribut BLOCKFORMAT 192  
 Attribut BLOCKSIZE 192  
 Attribut DATABUFFERS 193  
 Attribut DDNAME 193  
 Attribut DESCRIPTION 193  
 Attribut DISPOSITION 193  
 Attribut DSNNAME 194  
 Attribut ERROPTION 194  
 Attribut FACILITYID 194  
 Attribut INDIRECTNAME 195  
 Attribut OPENTIME 195  
 Attribut PRINTCONTROL 195  
 Attribut RECORDFORMAT 195  
 Attribut RECORDSIZE 196  
 Attribut RECOVSTATUS 196  
 Attribut REMOTELENGTH 198  
 Attribut REMOTENAME 198  
 Attribut REMOTESYSTEM 198  
 Attribut REWIND 198  
 Attribut SYSOUTCLASS 199  
 Attribut TDQUEUE 200  
 Attribut TRANSID 200  
 Attribut TRIGGERLEVEL 200  
 Attribut TYPE 201  
 Attribut TYPEFILE 202  
 Attribut USERID 203  
 Attribut WAIT 203  
 Attribut WAITACTION 203

TDQUEUE-Ressourcen 188

TEMPLATENAME, Attribut  
 DOCTEMPLATE-Definition 48  
 URIMAP-Definition 328

Temporärer Speicher  
 gemeinsame Datennutzung, DFHTST  
 TYPE=SHARED 486  
 Sicherheitsprüfung, DFHTST  
 TYPE=SECURITY 486  
 wiederherstellbar, DFHTST  
 TYPE=RECOVERY 481

Temporärer Speicher, gemeinsame Datennutzung  
 DFHTST TYPE=SHARED 486

Temporärer Speicher, Sicherheitsprüfung  
 DFHTST TYPE=SECURITY 486

TERMINAL, Attribut  
 TERMINAL-Definition 231

TERMINAL-Definition	THREADWAIT, Attribut	TRANSACTION-Definition (Forts.)
Attribut ALTPRINTCOPY 223	DB2CONN-Definition 31	Attribut TRANCLASS 237, 254
Attribut ALTPRINTER 223	DB2ENTRY-Definition 42	Attribut TRANSACTION 254
Attribut ATTACHSEC 224	TLT (Terminal List Table; Terminallisten-tabelle) 478	Attribut TRANSEC 493
Attribut AUTINSTMODEL 224	Codierungsbeispiele 477	Attribut TRPROF 255
Attribut AUTINSTNAME 225	DFHTLT TYPE=ENTRY 476	Attribut TWASIZE 255
Attribut BINDSECURITY 225	DFHTLT TYPE=INITIAL 476	Attribut WAIT 255
Attribut CONSNAME 225	Programmabschnitt 476	Attribut WAITTIME 257
Attribut CONSOLE 493	TPNAME, Attribut	Attribut XTPNAME 258
Attribut INSERVICE 226	PARTNER-Definition 128	Attribut XTRANID 258
Attribut MODENAME 226	TRANSACTION-Definition 253	TRANSACTION-Ressourcen 237
Attribut NATLANG 227	TPURGE, Attribut	Transaktionslistentabelle (XLT = Transac-tion List Table) 488
Attribut NETNAME 227	TRANSACTION-Definition 253	Codierungsbeispiel 491
Attribut POOL 228	TRACE, Attribut	DFHXLT TYPE=INITIAL 489
Attribut PRINTER 228	TRANSACTION-Definition 254	Programmabschnitt 489
Attribut PRINTERCOPY 229	TRANCLASS, Attribut	Transaktionsrouting 470
Attribut REMOTENAME 229	TRANSACTION-Definition 237, 254	APPC-Einheiten für 221
Attribut REMOTESYSNET 229	TRANCLASS-Definition	DFHTCT-Makros für ferne Termi-nals 471
Attribut REMOTESYSTEM 230	Attribut MAXACTIVE 236	Makros für ferne Terminals 471
Attribut SECURITYNAME 231	Attribut PURGETHRESH 236	mit INDIRECT-Verbindungen 10
Attribut SOLICITED 231	TRANCLASS-Ressourcen 235	Terminals für 216
Attribut TASKLIMIT 231	TRANSACTION, Attribut	Zusammenfassung 222
Attribut TERMINAL 231	SESSIONS-Definition 493	TRANSCIEVE, Option
Attribut TERMPRIORITY 232	TERMINAL-Definition 233	Operand TRMSTAT 470
Attribut TRANSACTION 233	TRANSACTION-Definition 254	TRANSEC, Attribut
Attribut TYPETERM 233	URIMAP-Definition 329	TRANSACTION-Definition 493
Attribut USEDFLTUSER 233	TRANSACTION, Option	TRANSID, Attribut
Attribut USERID 234	Operand TRMSTAT 469	DB2ENTRY-Definition 38
TERMINAL-Definitionen	TRANSACTION-Definition	DB2TRAN-Definition 43
Pipeline 214	Attribut ACTION 240	TDQUEUE-Definition 200
TERMINAL-Ressourcen 211	Attribut ALIAS 241	TRANSID, Operand
Terminaldefinition	Attribut BREXIT 241	DFHTCT TYPE=TERMINAL 469
Zusammenfassung 266	Attribut CMDSEC 241	DFHXLT TYPE=ENTRY 490
Terminallistentabelle (TLT = Terminal List Table) 478	Attribut CONFDATA 241	sequenzielle Einheiten 469
Codierungsbeispiele 477	Attribut DESCRIPTION 242	Transiente Daten
DFHTLT TYPE=ENTRY 476	Attribut DTIMOUT 242	Ressourcendefinition mit Doppelfunk-tion 188
DFHTLT TYPE=INITIAL 476	Attribut DUMP 243	Transiente Daten, Warteschlangen
Programmabschnitt 476	Attribut DYNAMIC 243	Inaktivierung 210
Terminals	Attribut EXTSEC 493	Pseudo 210
unterstützt von RDO 274	Attribut INDOUBT 493	TRIGGERLEVEL, Attribut
zum Drucken 212	Attribut ISOLATE 244	TDQUEUE-Definition 200
Terminals, CICS, Liste 474	Attribut LOCALQ 245	TRMIDNT, Operand
Terminals für Transaktionsrouting 216	Attribut LOGMODECOM 493	DFHTCT TYPE=REMOTE 474
Terminalsteuertabelle (TCT = Terminal Control Table) 452, 475	Attribut PARTITIONSET 246	DFHTCT TYPE=TERMINAL 469, 473
DFHTCT-Makros, Zusammenfas-sung 452	Attribut PRIMEDSIZE 493	DFHTLT TYPE=ENTRY 477
DFHTCT TYPE=GROUP 456	Attribut PRIORITY 247	ferne Terminals 473, 474
Einführung 452	Attribut PROFILE 247	Kleinschreibung in TCT 453
Liste der CICS-Terminals 474	Attribut PROGRAM 247	sequenzielle Einheiten 469
Migrieren, Einträge in RDO 456	Attribut REMOTENAME 248	TRMPRTY, Operand
sequenzielle Einheiten 465	Attribut REMOTESYSTEM 248	DFHTCT TYPE=TERMINAL 469
TERMMODEL, Attribut	Attribut RESSEC 249	sequenzielle Einheiten 469
TYPETERM-Definition 307	Attribut RESTART 249	TRMSTAT, Operand
TERMPRIORITY, Attribut	Attribut ROUTABLE 249	DFHTCT TYPE=TERMINAL 469
TERMINAL-Definition 232	Attribut RSL 493	sequenzielle Einheiten 469
TEXTKYBD, Attribut	Attribut RUNAWAY 250	TRMTYPE, Operand
TYPETERM-Definition 307	Attribut SHUTDOWN 250	DFHTCT TYPE=LINE 468
TEXTPRINT, Attribut	Attribut SPURGE 250	DFHTCT TYPE=REMOTE 474
TYPETERM-Definition 307	Attribut STATUS 251	DFHTCT TYPE=TERMINAL 473
THREADERROR, Attribut	Attribut STORAGECLEAR 251	ferne Terminals 473, 474
DB2CONN-Definition 25	Attribut TASKDATAKEY 251	sequenzielle Einheiten 468
THREADLIMIT, Attribut	Attribut TASKDATALOC 252	TRPROF, Attribut
DB2CONN-Definition 31	Attribut TASKREQ 253	TRANSACTION-Definition 255
DB2ENTRY-Definition 42	Attribut TCLASS 493	
	Attribut TPNAME 253	
	Attribut TPURGE 253	
	Attribut TRACE 254	



TSAGE, Operand  
 DFHTST TYPE=INITIAL 481

TSMODEL, Attribut  
 TSMODEL-Definition 264

TSMODEL-Definition  
 Attribut EXPIRYINTMIN 260  
 Attribut POOLNAME 261  
 Attribut PREFIX 262  
 Attribut RECOVSTATUS 262  
 Attribut REMOTEPREFIX 263  
 Attribut REMOTESYSTEM 263  
 Attribut SECURITY 263  
 Attribut TSMODEL 264  
 Attribut XPREFIX 264  
 Attribut XREMOTEPEFX 264  
 LOCATION, Attribut 261

TSMODEL-Ressourcen 259

TSQPREFIX, Attribut  
 TCIPSERVICE-Definition 173

TSQUEUE, Attribut  
 DOCTEMPLATE-Definition 48

TST (Temporary Storage Table; Tabelle für temporären Speicher) 478, 487  
 DATAIDs für fernen temporären Speicher, DFHTST TYPE=REMOTE 484  
 DATAIDs für lokalen temporären Speicher, DFHTST TYPE=LOCAL 483  
 DFHTST TYPE=INITIAL 480  
 DFHTST TYPE=RECOVERY 481, 483, 486  
 DFHTST TYPE=SHARED 486  
 gemeinsame Datennutzung, temporärer Speicher 486  
 gemeinsame Datennutzung für temporären Speicher, DFHTST TYPE=SHARED 486  
 Programmabschnitt 480  
 Sicherheitsprüfung für temporären Speicher, DFHTST TYPE=SECURITY 486  
 wiederherstellbarer temporärer Speicher 481, 483, 486  
 wiederherstellbarer temporärer Speicher, DFHTST TYPE=RECOVERY 481

TTI, Attribut  
 TYPETERM-Definition 307

TWASIZE, Attribut  
 TRANSACTION-Definition 255

TYPE, Attribut  
 DOCTEMPLATE-Definition 48  
 JOURNALMODEL-Definition 95  
 TDQUEUE-Definition 201

TYPE=LOCAL, Makro DFHTST  
 DATAIDs für lokalen temporären Speicher, DFHTST 483

TYPE=REMOTE, Makro DFHTST  
 DATAIDs für fernen temporären Speicher, DFHTST 484

TYPE=SECURITY, Makro  
 Sicherheitsprüfung für temporären Speicher, DFHTST 486

TYPE=SHARED, Makro  
 gemeinsame Datennutzung für temporären Speicher, DFHTST 486

TYPEFILE, Attribut  
 TDQUEUE-Definition 202

TYPETERM  
 Standardwerte für Attribute 267

TYPETERM, Attribut  
 TERMINAL-Definition 233  
 TYPETERM-Definition 308

TYPETERM-Definition  
 Attribut ALTPAGE 282  
 Attribut ALTSCREEN 283  
 Attribut ALTSUFFIX 284  
 Attribut APLKYBD 284  
 Attribut APLTEXT 284  
 Attribut ASCII 285  
 Attribut ATI 286  
 Attribut AUDIBLEALARM 286  
 Attribut AUTOCONNECT 286  
 Attribut AUTOPAGE 287  
 Attribut BACKTRANS 287  
 Attribut BRACKET 287  
 Attribut BUILDCHAIN 287  
 Attribut CGCSGID 287  
 Attribut COLOR 288  
 Attribut COPY 288  
 Attribut CREATESESS 288  
 Attribut DEFSCREEN 289  
 Attribut DESCRIPTION 289  
 Attribut DISCREQ 290  
 Attribut DUALCASEKYBD 290  
 Attribut ERRCOLOR 290  
 Attribut ERRHIGHLIGHT 291  
 Attribut ERRINTENSIFY 291  
 Attribut ERRLASTLINE 291  
 Attribut EXTENDEDDES 291  
 Attribut FMHPARM 291  
 Attribut FORMFEED 292  
 Attribut HIGHLIGHT 293  
 Attribut HORIZFORM 293  
 Attribut IOAREALEN 293  
 Attribut KATAKANA 294  
 Attribut LDCLIST 294  
 Attribut LIGHTPEN 295  
 Attribut LOGMODE 295  
 Attribut LOGONMSG 296  
 Attribut MSRCONTROL 297  
 Attribut NEPCCLASS 297  
 Attribut OBFORMAT 297  
 Attribut OBOPERID 297  
 Attribut OUTLINE 297  
 Attribut PAGESIZE 297  
 Attribut PARTITIONS 299  
 Attribut PRINTADAPTER 299  
 Attribut PROGSYMBOLS 300  
 Attribut QUERY 300  
 Attribut RECEIVESIZE 300  
 Attribut RECOVNOTIFY 301  
 Attribut RECOVOPTION 302  
 Attribut RELREQ 304  
 Attribut ROUTEDMSGS 304  
 Attribut RSTSIGNOFF 304  
 Attribut SENDSIZE 304  
 Attribut SESSIONTYPE 305  
 Attribut SHIPPABLE 305  
 Attribut SIGNOFF 306  
 Attribut SOSI 306  
 Attribut TERMMODEL 307  
 Attribut TEXTKYBD 307

TYPETERM-Definition (Forts.)  
 Attribut TEXTPRINT 307  
 Attribut TTI 307  
 Attribut TYPETERM 308  
 Attribut UCTRAN 308  
 Attribut USERAREALEN 309  
 Attribut VALIDATION 309  
 Attribut VERTICALFORM 309

TYPETERM-Ressourcen 264

## U

Überwachungspunkt, Ereignis  
 DFHMCT TYPE=EMP 422

Überwachungssteuertabelle (MCT = Monitoring Control Table) 416  
 Codierungsbeispiele 441  
 DFHMCT-Makros 417  
 DFHMCT TYPE=EMP 422  
 DFHMCT TYPE=INITIAL 417  
 DFHMCT TYPE=RECORD 427  
 Ereignisüberwachungspunkte des Benutzers 422  
 Steuerdatenaufzeichnung 427

UCTRAN, Attribut  
 PROFILE-Definition 143  
 TYPETERM-Definition 308

UNBLOCKED, Option  
 Operand RECFORM 413

UNDEFINED, Option  
 Operand RECFORM 413

UNLOCK, Befehl  
 CEDA 366

UPDATE, Attribut  
 FILE-Definition 75

UPDATE, Option  
 Operand SERVREQ 415

UPDATEMODEL, Attribut  
 FILE-Definition 76

UPGRADE, Befehl, Dienstprogramm  
 DFHCSDUP 393  
 Operand REPLACE 394  
 Operand USING 394

URIMAP, Attribut  
 URIMAP-Definition 330

URIMAP-Definition  
 Attribut ANALYZER 316  
 Attribut ATOMSERVICE 317  
 Attribut CERTIFICATE 317  
 Attribut CHARACTERSET 318  
 Attribut CIPHERS 318  
 Attribut CONVERTER 319  
 Attribut HFSFILE 320  
 Attribut HOST 321  
 Attribut HOSTCODEPAGE 322  
 Attribut LOCATION 322  
 Attribut MEDIATYPE 323  
 Attribut PATH 323  
 Attribut PIPELINE 325  
 Attribut PORT 325  
 Attribut PROGRAM 325  
 Attribut REDIRECTTYPE 326  
 Attribut SCHEME 326  
 Attribut SOCKETCLOSE 327  
 Attribut STATUS 327  
 Attribut TCIPSERVICE 327  
 Attribut TEMPLATENAME 328

- URIMAP-Definition (*Forts.*)
  - Attribut TRANSACTION 329
  - Attribut URIMAP 330
  - Attribut USAGE 330
  - Attribut USERID 331
  - Attribut WEBSERVICE 332
- URIMAP-Ressourcen 310
- URM, Attribut
  - TCPIPService-Definition 173
- USAGE, Attribut
  - URIMAP-Definition 330
- USEDFLTUSER, Attribut
  - CONNECTION-Definition 23
  - TERMINAL-Definition 233
- USERAREALEN, Attribut
  - SESSIONS-Definition 171
  - TYPETERM-Definition 309
- USERAUTH, Attribut
  - IPCONN-Definition 77
- USERDEFINE, Befehl
  - CEDA 367
- USERDEFINE, Befehl, Dienstprogramm
  - DFHCSDUP 394
- USERID, Attribut
  - SESSIONS-Definition 171
  - TDQUEUE-Definition 203
  - TERMINAL-Definition 234
  - URIMAP-Definition 331
- USERID, Operand
  - DFHTCT TYPE=TERMINAL 470
  - sequenzielle Einheiten 470
- USERVAR des fernen Systems 18

## V

- VALIDATION, Attribut
  - TYPETERM-Definition 309
- VARIABLE, Option
  - Operand RECFORM 413
- Verbindungen, INDIRECT
  - Ressourcendefinitionen 10
- VERIFY, Befehl, Dienstprogramm
  - DFHCSDUP 396
- VERIFY, Operand
  - DFHFCT TYPE=FILE 415
- Versenden, Terminaldefinitionen an Anwendungsverwaltungsregionen 219
- VERTICALFORM, Attribut
  - TYPETERM-Definition 309
- VIEW, Befehl
  - CEDA 371
- Von CICS bereitgestellte TYPETERM-Definitionen
  - Attribut ALTPAGE 282
  - von CICS bereitgestellt 511
- Voreingestellte Sicherheit 470, 493
- Voreingestellte Terminalsicherheit
  - Konsolen 234
- VTAM, Option
  - Operand ACCMETH 454

## W

- WAIT, Attribut
  - TDQUEUE-Definition 203
  - TRANSACTION-Definition 255

- WAITACTION, Attribut
  - TDQUEUE-Definition 203
- WAITTIME, Attribut
  - TRANSACTION-Definition 257
- Warteschlange mit transienten Daten, Definitionen
  - Dateidefinitionen 188
- WEBSERVICE, Attribut
  - URIMAP-Definition 332
- WEBSERVICE-Ressourcen 332
- Wiederherstellbare Servicetabelle (RST = Recoverable Service Table) 447, 449
  - Codierungsbeispiel 448
  - DFHRST-Makrotypen 447
  - DFHRST TYPE=INITIAL 447
  - DFHRST TYPE=RSE 447
  - DFHRST TYPE=SUBSYS 448
- Wiederherstellbarer temporärer Speicher
  - DFHTST TYPE=RECOVERY 481
- WSDIR, Attribut
  - PIPELINE-Definition 132
- WTOL, Operand
  - DFHCLT TYPE=WTO 404

## X

- XLNAction, Attribut
  - CONNECTION-Definition 23
  - IPCONN-Definition 77
- XLT (Transaction List Table; Transaktionslistentabelle) 488
  - Codierungsbeispiel 491
  - DFHXLT TYPE=INITIAL 489
  - Programmabschnitt 489
- XPREFIX, Attribut
  - TSMODEL-Definition 264
- XREMOTEPEFX, Attribut
  - TSMODEL-Definition 264
- XTPNAME, Attribut
  - PARTNER-Definition 129
  - TRANSACTION-Definition 258
- XTRANID, Attribut
  - TRANSACTION-Definition 258

## Z

- z/OS Communications Server
  - Anwendungsname des fernen Systems 18



