

CICS Transaction Server for z/OS



# CICSplex SM API-Referenz

*Version 5 Release 5*



CICS Transaction Server for z/OS



# CICSplex SM API-Referenz

*Version 5 Release 5*

**Hinweis**

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die Informationen unter „Bemerkungen“ auf Seite 195 gelesen werden.

---

# Inhaltsverzeichnis

|                                       |   |
|---------------------------------------|---|
| Informationen zu dieser PDF . . . . . | v |
|---------------------------------------|---|

## Kapitel 1. CICSplex SM-Befehlsschnittstelle verwenden . . . . . 1

|  |    |
|--|----|
| CICSplex SM-Befehlsformat . . . . .                | 1  |
| CICSplex SM-Argumentwerte . . . . .                | 1  |
| COBOL-Argumentwerte . . . . .                      | 2  |
| C-Argumentwerte . . . . .                          | 3  |
| PL/I-Argumentwerte . . . . .                       | 4  |
| Argumentwerte für Assembler-Sprache . . . . .      | 6  |
| CICSplex SM mit REXX verwenden . . . . .           | 6  |
| Befehlsformat . . . . .                            | 6  |
| Argumentwerte . . . . .                            | 7  |
| MVS-Einschränkungen . . . . .                      | 8  |
| CICS- und CICSplex SM-Wertedatenbereiche . . . . . | 9  |
| Sprachaspekte . . . . .                            | 10 |
| Längenoptionen . . . . .                           | 10 |
| RESPONSE- und REASON-Optionen . . . . .            | 10 |

## Kapitel 2. REXX-Funktionen und -Befehle . . . . . 13

|                      |    |
|----------------------|----|
| Funktionen . . . . . | 13 |
| EYUAPI() . . . . .   | 13 |
| EYUINIT() . . . . .  | 14 |
| EYUREAS() . . . . .  | 14 |
| EYURESP() . . . . .  | 15 |
| EYUTERM() . . . . .  | 15 |
| Befehle . . . . .    | 16 |
| TBUILD . . . . .     | 17 |
| TPARSE . . . . .     | 18 |

## Kapitel 3. CICSplex SM-API-Befehle . . . . . 21

|                      |    |
|----------------------|----|
| ADDRESS . . . . .    | 21 |
| CANCEL . . . . .     | 23 |
| CONNECT . . . . .    | 25 |
| COPY . . . . .       | 29 |
| CREATE . . . . .     | 34 |
| DELETE . . . . .     | 38 |
| DISCARD . . . . .    | 41 |
| DISCONNECT . . . . . | 44 |
| EXPAND . . . . .     | 46 |
| FEEDBACK . . . . .   | 50 |
| FETCH . . . . .      | 54 |
| GET . . . . .        | 61 |
| GETDEF . . . . .     | 67 |
| GROUP . . . . .      | 73 |

|                          |     |
|--------------------------|-----|
| LISTEN . . . . .         | 78  |
| LOCATE . . . . .         | 81  |
| MARK . . . . .           | 86  |
| ORDER . . . . .          | 90  |
| PERFORM OBJECT . . . . . | 93  |
| PERFORM SET . . . . .    | 100 |
| QUALIFY . . . . .        | 106 |
| QUERY . . . . .          | 109 |
| RECEIVE . . . . .        | 112 |
| REFRESH . . . . .        | 115 |
| REMOVE . . . . .         | 120 |
| SET . . . . .            | 124 |
| SPECIFY FILTER . . . . . | 130 |
| SPECIFY VIEW . . . . .   | 133 |
| TERMINATE . . . . .      | 136 |
| TRANSLATE . . . . .      | 137 |
| UNMARK . . . . .         | 140 |
| UPDATE . . . . .         | 145 |

## Kapitel 4. Liste der CICSplex SM-API-Befehlsargumente . . . . . 151

## Kapitel 5. Funktionscodewerte für CICSplex SM-API-Befehle . . . . . 153

## Kapitel 6. RESPONSE- und REASON-Werte . . . . . 155

## Kapitel 7. EYUDA-Werte . . . . . 161

|  |     |
|--|-----|
| Allgemeine EYUDA-Werte in numerischer Reihenfolge . . . . .            | 161 |
| Allgemeine Werte vom Typ EYUDA in alphabetischer Reihenfolge . . . . . | 175 |
| RESPONSE-Werte vom Typ EYUDA in numerischer Reihenfolge . . . . .      | 187 |
| RESPONSE-Werte vom Typ EYUDA in alphabetischer Reihenfolge . . . . .   | 188 |
| REASON-Werte vom Typ EYUDA in numerischer Reihenfolge . . . . .        | 189 |
| REASON-Werte vom Typ EYUDA in alphabetischer Reihenfolge . . . . .     | 191 |

## Bemerkungen . . . . . 195

## Index . . . . . 201



---

## Informationen zu dieser PDF

Diese PDF ist eine Referenz der Befehle der Anwendungsprogrammierschnittstelle für das CICSplex SM-Bauteil von CICS Transaction Server for z/OS. Diese Dokumentation richtet sich an Anwendungsprogrammierer, die Anwendungen für die Interaktion mit CICSplex SM schreiben.

Informationen zum Schreiben von Anwendungen für CICS unter Verwendung dieser API finden Sie in der Veröffentlichung *CICSplex SM Application Programming Guide*.

Ausführliche Informationen zu den in diesem Handbuch verwendeten Begriffen sowie zur Schreibweise finden Sie in Conventions and terminology used in the CICS documentation im IBM Knowledge Center.

### Datum dieser PDF

Diese PDF wurde am 14. Dezember 2018 erstellt.





---

# Kapitel 1. CICSPlex SM-Befehlsschnittstelle verwenden

In diesem Abschnitt wird beschrieben, wie die CICSPlex SM-Befehlsschnittstelle verwendet wird.

---

## CICSPlex SM-Befehlsformat

Das Format eines API-Befehls bei der Ausgabe über die CICSPlex SM-Befehlsschnittstelle ist EXECUTE CPSM (oder EXEC CPSM) gefolgt vom Namen des erforderlichen Befehls und möglicherweise von mindestens einer Option.

Die Syntax lautet wie folgt:

```
EXEC CPSM Befehlsoption(Arg)....
```

Dabei gilt Folgendes:

**Befehl** Beschreibt die erforderliche Operation (z. B. CONNECT).

**Option** Beschreibt eine der erforderlichen oder optionalen Funktionen, die mit jedem Befehl verfügbar sind. Auf einige Optionen folgt ein Argument in runden Klammern. Sie können Optionen (einschließlich der Optionen, die Argumente erfordern) in beliebiger Reihenfolge schreiben.

**Arg** Dies ist die Abkürzung für Argument; dies wiederum ist ein Wert wie *Datenwert* oder *Datenreferenz*. Ein *Datenwert* kann eine Konstante sein. Dies bedeutet, dass ein Argument, über das Daten an CICSPlex SM gesendet werden, im Allgemeinen ein *Datenwert* ist. Ein Argument, über das Daten aus CICSPlex SM empfangen werden, muss eine *Datenreferenz* sein.

Im Folgenden sehen Sie ein Beispiel für einen Befehl des Typs EXEC CPSM:

```
EXEC CPSM CONNECT  
          USER(JONES) VERSION(0310)  
          CONTEXT(EYUPLX01) SCOPE(EYUCSG01)  
          THREAD(THRDTKN)  
          RESPONSE(RESVAR) REASON(REASVAR)
```

Sie müssen einen Begrenzer für das Ende des Befehls hinzufügen, der für die von Ihnen verwendete Programmiersprache gültig ist. In COBOL-Programmen ist z. B. der Begrenzer für das Ende des Befehls eine Anweisung des Typs END-EXEC. In PL/I- und C-Programmen ist der Begrenzer ein Semikolon (;).

---

## CICSPlex SM-Argumentwerte

Sie müssen die in Klammern angegebenen Argumentwerte angeben, die Optionen in einem API-Befehl folgen.

Die Werte lauten wie folgt:

*Datenwert*

Ein Sendeargument, das verwendet wird, um Daten von Ihrem Programm an CICSPlex SM zu übergeben.

Die Daten, die Sie übergeben, können Vollwort-Binärdaten, Zeichendaten mit fester oder variabler Länge oder nicht spezifizierte Daten sein. Wenn der Datentyp nicht spezifiziert ist, nimmt CICSPlex SM eine zusammenge-

setzte Datenstruktur an, die aus mehreren Feldern mit unterschiedlichen Datentypen besteht. Das Argument kann eins der folgenden Formate aufweisen:

- Variablenname
- Selbstdefinierender Begriff
- Ausdruck

*Datenwert* umfasst *Datenreferenz* als Untergruppe.

#### *Datenreferenz*

Ein Empfangsargument (oder ein Sende- und Empfangsargument), das hauptsächlich zum Übergeben von Daten von CICSplex SM an Ihr Programm verwendet wird.

Bei dem Datentyp kann es sich um einen beliebigen Typ handeln, der für die Argumente des Typs *Datenwert* zulässig ist. Das Argument muss jedoch eine benannte Variable sein.

In einigen Fällen können Sie ein Argument des Typs *Datenreferenz* verwenden, um CICSplex SM Eingabedaten zur Verfügung zu stellen, bevor CICSplex SM Ihnen seine Ausgabe zurückgibt; Sie können beispielsweise ein Argument des Typs *Datenreferenz* bei der Option COUNT im Befehl FETCH angeben.

#### *Datenbereich*

Ein Sende- oder Empfangsargument, das zum Ermitteln eines Puffers mit Daten verwendet wird. Ein Argument des Typs *Datenbereich* kann als Argument des Typs *Datenreferenz* mit einem nicht spezifizierten Datentyp betrachtet werden. Ein Argument des Typs *Datenbereich* kann nicht durch einen selbstdefinierenden Begriff oder Ausdruck definiert werden; es muss sich um eine benannte Variable handeln.

#### *Zeigerreferenz*

Ein Empfangsargument, das verwendet wird, um Zeigerwerte von CICSplex SM an Ihr Programm zu übergeben.

Ein Argument des Typs *Zeigerreferenz* ist ein Sonderformat des Arguments *Datenreferenz*. Bei den übergebenen Daten handelt es sich um einen Adresszeiger und nicht um Binär- oder Zeichendaten.

#### *CPSM-Token*

Ein Sende- oder Empfangsargument, das verwendet wird, um identifizierende Tokens zu übergeben, die von CICSplex SM generiert werden. Ein Argument des Typs *CPSM-Token* kann als Argument des Typs *Datenreferenz* mit einem nicht spezifizierten Datentyp betrachtet werden.

Tokens werden von CICSplex SM erstellt, um API-Verarbeitungsthreads, Ergebnislisten, Filter und Benachrichtigungen zu ermitteln.

Da Tokenwerte von CICSplex SM erstellt werden, muss Ihr Programm ein Token in eine Variable aufnehmen, bevor es dieses Token bei nachfolgenden Befehlen angeben kann. Ein Token kann nicht durch einen selbstdefinierenden Begriff oder Ausdruck definiert werden; es muss sich um eine benannte Variable handeln.

## **COBOL-Argumentwerte**

In COBOL können Sie die Argumentwerte der CICSplex SM-API durch verschiedene Datentypen ersetzen. Der Datentyp, den Sie verwenden können, richtet sich nach dem Typ des Argumentwerts.

### *Datenwert*

Kann durch einen beliebigen COBOL-Datennamen des korrekten Datentyps für das Argument oder durch eine Konstante ersetzt werden, die in den richtigen Typ für das Argument konvertiert werden kann. In der folgenden Tabelle wird angegeben, wie der richtige Datentyp definiert wird:

| Datentyp             | COBOL-Definition   |
|----------------------|--|
| Halbwort-Binärzahl   | PIC S9(4) USAGE BINARY   |
| Vollwort-Binärzahl   | PIC S9(8) USAGE BINARY   |
| Doppelwort-Binärzahl | PIC S9(18) COMP  |
| Zeiger               | USAGE IS POINTER   |
| Zeichenfolge         | PIC X( <i>n</i> ); dabei ist <i>n</i> die Anzahl an Byte                   |
| UTF-8-Zeichenfolge   | PIC X( <i>n</i> ); dabei ist <i>n</i> die Anzahl an Byte                   |
| Gepackte Dezimalzahl | PIC S9( <i>n</i> ) COMP-3; dabei ist <i>n</i> die Anzahl an Dezimalstellen |

*Datenwert* umfasst *Datenreferenz* als Untergruppe.

### *Datenreferenz*

Kann durch einen beliebigen COBOL-Datennamen des richtigen Datentyps für das Argument ersetzt werden. In der folgenden Tabelle wird angegeben, wie der richtige Datentyp definiert wird:

| Datentyp             | COBOL-Definition   |
|----------------------|--|
| Halbwort-Binärzahl   | PIC S9(4) USAGE BINARY   |
| Vollwort-Binärzahl   | PIC S9(8) USAGE BINARY   |
| Doppelwort-Binärzahl | PIC S9(18) COMP  |
| Zeiger               | USAGE IS POINTER   |
| Zeichenfolge         | PIC X( <i>n</i> ); dabei ist <i>n</i> die Anzahl an Byte                   |
| UTF-8-Zeichenfolge   | PIC X( <i>n</i> ); dabei ist <i>n</i> die Anzahl an Byte                   |
| Gepackte Dezimalzahl | PIC S9( <i>n</i> ) COMP-3; dabei ist <i>n</i> die Anzahl an Dezimalstellen |

Ist der Datentyp nicht angegeben, kann sich *Datenreferenz* auf ein Datenelement oder eine Datengruppe beziehen.

### *Datenbereich*

Kann durch einen beliebigen COBOL-Datennamen mit dem Datentyp 'Halbwort-Binärzahl' (PIC S9(4) COMP), 'Vollwort-Binärzahl' (PIC S9(8) COMP) oder 'Zeichenfolge' (PIC X(*n*)) ersetzt werden.

### *Zeigerreferenz*

Kann durch eine Zeigervariable oder ein Sonderregister des Typs ADDRESS ersetzt werden.

### *CPSM-Token*

Kann durch einen beliebigen COBOL-Datennamen mit dem Datentyp 'Vollwort-Binärzahl' (PIC S9(8) COMP) ersetzt werden.

## C-Argumentwerte

In C können Sie die Argumentwerte der CICSplex SM-API durch verschiedene Datentypen ersetzen. Sie können eine beliebige Datenreferenz mit dem korrekten Datentyp für *Datenwert*, *Datenreferenz* oder *Datenbereich* verwenden, vorausgesetzt, die Referenz zeigt auf einen zusammenhängenden Speicher.

### *Datenwert*

Kann durch einen beliebigen C-Ausdruck ersetzt werden, der in den richtigen Datentyp für das Argument konvertiert werden kann. In der folgenden Tabelle wird angegeben, wie der richtige Datentyp definiert wird:

| Datentyp             | C-Definition  |
|----------------------|---|
| Halbwort-Binärzahl   | short int   |
| Vollwort-Binärzahl   | long int  |
| Doppelwort-Binärzahl | long long int   |
| Zeichenbereich       | char[n]; dabei ist <i>n</i> die Anzahl an Byte in dem Feld (das Feld muss mit Leerzeichen gefüllt sein) |
| UTF-8-Zeichenfolge   | char[n]; dabei ist <i>n</i> die Anzahl an Byte in dem Feld (das Feld ist mit ASCII-Leerzeichen gefüllt) |

*Datenwert* umfasst *Datenreferenz* als Untergruppe.

### *Datenreferenz*

Kann durch eine beliebige C-Datenreferenz ersetzt werden, die den richtigen Datentyp für das Argument aufweist. In der folgenden Tabelle wird angegeben, wie der richtige Datentyp definiert wird:

| Datentyp             | C-Definition  |
|----------------------|---|
| Halbwort-Binärzahl   | short int   |
| Vollwort-Binärzahl   | long int  |
| Doppelwort-Binärzahl | long long int   |
| Zeichenbereich       | char[n]; dabei ist <i>n</i> die Anzahl an Byte in dem Feld (das Feld muss mit Leerzeichen gefüllt sein) |
| UTF-8-Zeichenfolge   | char[n]; dabei ist <i>n</i> die Anzahl an Byte in dem Feld (das Feld ist mit ASCII-Leerzeichen gefüllt) |

Ist der Datentyp nicht angegeben, kann sich *Datenreferenz* auf einen skalaren Datentyp, ein skalares Array oder eine skalare Struktur beziehen. Die Referenz muss ein zusammenhängender Speicher sein.

### *Datenbereich*

Kann durch eine beliebige benannte Variable mit dem Datentyp 'Halbwort-Binärzahl' (short int), 'Vollwort-Binärzahl' (long int) oder 'Zeichenbereich' (char[n]) ersetzt werden.

### *Zeigerreferenz*

Kann durch eine beliebige C-Zeigertypreferenz ersetzt werden.

### *CPSM-Token*

Kann durch eine beliebige benannte Variable mit dem Datentyp 'Vollwort-Binärzahl' (long int) ersetzt werden.

## PL/I-Argumentwerte

In PL/I können Sie die Argumentwerte der CICSplex SM-API durch eine beliebige PL/I-Datenreferenz des richtigen Datentyps ersetzen, sofern auf den verbundenen Speicher verwiesen wird.

### Datenwert

Kann durch einen beliebigen PL/I-Ausdruck ersetzt werden, der in den richtigen Datentyp für das Argument konvertiert werden kann. In der folgenden Tabelle wird angegeben, wie der richtige Datentyp definiert wird:

| Datentyp                                   | PL/I-Definition                               |
|--|---|
| Halbwort-Binärzahl                         | FIXED BIN(15)                                 |
| Vollwort-Binärzahl                         | FIXED BIN(31)                                 |
| Doppelwort-Binärzahl                       | FIXED BIN(63)                                 |
| Zeiger                                     | POINTER                                       |
| Zeichenfolge                               | CHAR( $n$ ); dabei ist $n$ die Anzahl an Byte |
| UTF-8-Zeichenfolge                         | CHAR( $n$ ); dabei ist $n$ die Anzahl an Byte |
| Gepackte Dezimalzahl ( $n$ Dezimalstellen) | FIXED DEC( $n,0$ )                            |

*Datenwert* umfasst *Datenreferenz* als Untergruppe.

### Datenreferenz

Kann durch eine beliebige PL/I-Datenreferenz ersetzt werden, die den richtigen Datentyp für das Argument aufweist. In der folgenden Tabelle wird angegeben, wie der richtige Datentyp definiert wird:

| Datentyp                                   | PL/I-Definition                               |
|--|---|
| Halbwort-Binärzahl                         | FIXED BIN(15)                                 |
| Vollwort-Binärzahl                         | FIXED BIN(31)                                 |
| Doppelwort-Binärzahl                       | FIXED BIN(63)                                 |
| Zeiger                                     | POINTER                                       |
| Zeichenfolge                               | CHAR( $n$ ); dabei ist $n$ die Anzahl an Byte |
| UTF-8-Zeichenfolge                         | CHAR( $n$ ); dabei ist $n$ die Anzahl an Byte |
| Gepackte Dezimalzahl ( $n$ Dezimalstellen) | FIXED DEC( $n,0$ )                            |

Ist der Datentyp nicht angegeben, kann sich *Datenreferenz* auf ein Element, ein Array oder eine Struktur beziehen, z. B. FROM(P->STRUCTURE) LENGTH(LNG). Die Referenz muss ein verbundener Speicher sein.

Der Datenbereich muss außerdem das richtige PL/I-Ausrichtungsattribut aufweisen: ALIGNED für binäre Datenelemente und UNALIGNED für Zeichenfolgen.

Wenn Sie eine variable Datenzeichenfolge ohne explizite Länge verwenden, beginnen die übergebenen Daten mit zwei Byte langen Feldern und die Länge ist die maximale Länge, die für die Zeichenfolge deklariert ist. Wenn Sie explizit eine Länge im Befehl angeben, haben die übergebenen Daten diese Länge, d. h. die Felder mit einer Länge von zwei Byte gefolgt von Daten bis zu der von Ihnen angegebenen Länge.

### Datenbereich

Kann durch eine beliebige benannte Variable mit dem Datentyp 'Halbwort-Binärzahl' (FIXED BIN(15)), 'Vollwort-Binärzahl' (FIXED BIN(31)) oder 'Zeichenfolge' (CHAR( $n$ )) ersetzt werden.

### Zeigerreferenz

Kann durch eine beliebige PL/I-Referenz des Typs POINTER ALIGNED ersetzt werden.

#### *CPSM-Token*

Kann durch eine beliebige benannte Variable mit dem Datentyp 'Vollwort-Binärzahl' (FIXED BIN(31)) ersetzt werden.

## **Argumentwerte für Assembler-Sprache**

Im Allgemeinen kann es sich bei einem Argument entweder um die Adresse der Daten oder um die Daten selbst handeln (in Assembler-Sprache ausgedrückt: entweder einen verschiebbaren Ausdruck oder um einen absoluten Ausdruck).

Ein verschiebbarer Ausdruck darf keine nicht übereinstimmenden eckigen Klammern (außerhalb von Anführungszeichen) oder nicht übereinstimmende Anführungszeichen (abgesehen von Längenattributreferenzen) enthalten. Wenn diese Regel eingehalten wird, kann ein beliebiger Ausdruck verwendet werden, einschließlich Literalkonstanten, z. B. = AL2(100), Formaten wie 20(0,R11) und Formaten, bei denen die Makroersatzfunktionen verwendet werden.

Ein absoluter Ausdruck muss ein einzelner Begriff sein, bei dem es sich entweder um eine Längenattributreferenz oder um eine selbstdefinierende Konstante handelt.

Bei gleichgesetzten Symbolen ist Vorsicht geboten; sie sollten nur beim Verweis auf Register (Zeigerreferenzen) verwendet werden. Wenn ein gleichgesetztes Symbol für eine Länge verwendet wird, wird es z. B. als Adresse der Länge behandelt und ein unvorhersehbarer Fehler tritt auf.

Die Argumentwerte können wie folgt ersetzt werden:

#### *Datenwert*

Kann durch einen verschiebbaren Ausdruck ersetzt werden, der eine Assemblersprachreferenz auf Daten des richtigen Typs für das Argument ist, oder durch eine Konstante des richtigen Typs für das Argument.

#### *Datenreferenz*

Kann durch einen verschiebbaren Ausdruck ersetzt werden, der eine Assemblersprachreferenz auf Daten des richtigen Typs für das Argument ist.

#### *Datenbereich*

Kann durch einen verschiebbaren Ausdruck ersetzt werden, der eine Assemblersprachreferenz auf Daten des Typs Halbwort (DS H), Vollwort (DS F) oder Zeichenfolge (CLn) ist.

#### *Zeigerreferenz*

Kann durch einen beliebigen absoluten Ausdruck ersetzt werden, der eine Assemblersprachreferenz auf ein Register ist.

#### *CPSM-Token*

Kann durch einen verschiebbaren Ausdruck ersetzt werden, der eine Assemblersprachreferenz auf Daten des Typs Vollwort (DS F) ist.

---

## **CICSplex SM mit REXX verwenden**

Sie können CICSplex SM-API-Befehle über ein REXX-Programm aufrufen.

### **Befehlsformat**

Ein API-Befehl kann von REXX an CICSplex SM übergeben werden, und zwar entweder über den REXX-Befehl ADDRESS oder über die Funktion EYUAPI().

Das Format des REXX-Befehls ADDRESS lautet wie folgt:

```
ADDRESS CPSM 'Befehlsoption(Arg)...'
```

Mit dieser Methode zum Aufrufen der API wird eine CICSplex SM-Hostunterbefehls Umgebung aufgerufen.

Alternativ dazu können Sie die Funktion EYUAPI() verwenden, die von CICSplex SM bereitgestellt wird:

```
var = EYUAPI('Befehlsoption(Arg)...')
```

Durch diese Methode wird das CICSplex SM-REXX-Funktionspaket aufgerufen.

Beachten Sie, dass Sie mit beiden Methoden Text sowohl in Groß- als auch in Kleinbuchstaben eingeben können.

Im Folgenden sehen Sie ein Beispiel für einen API-Befehl wie er von einem REXX-Programm abgesetzt werden würde:

```
var = EYUAPI('CONNECT'
             'CONTEXT('WCCONTEXT')' ,
             'SCOPE('WScope')' ,
             'VERSION(0310)' ,
             'THREAD(THRDTKN)' ,
             'RESPONSE(RESVAR)' ,
             'REASON(REASVAR)')
.
.
```

## Argumentwerte

Die CICSplex SM-API nutzt die REXX-Standardvariablenschnittstelle im vollen Umfang.

Ein Argument des Typs *Datenwert* wird als Zeicheneingabe behandelt. Binärdaten (einschließlich der EYUDA- und CVDA-Werte) werden in das entsprechende interne Format umgesetzt. Benutzertokens werden nicht umgesetzt.

### *Datenreferenz*

Ein Empfangsargument (oder ein Sende- und Empfangsargument), das hauptsächlich zum Übergeben von Daten von CICSplex SM an Ihr Programm verwendet wird.

Ein Argument des Typs *Datenreferenz* muss eine benannte Variable sein, die für den Empfang der resultierenden Ausgabe verwendet werden kann. Die Ausgabedaten können wie folgt umgesetzt werden:

- Zeichendaten werden nicht umgesetzt; die Daten werden unverändert in die Variable eingesetzt.
- Binärdaten werden in das Anzeigeformat (Dezimalformat) umgesetzt und in die Variable eingesetzt.
- Benutzertokens werden nicht umgesetzt; der Tokenwert wird unverändert in die Variable eingesetzt.
- Adresswerte werden nicht umgesetzt; der angegebene Speicherpuffer wird direkt in eine oder mehrere Variablen eingesetzt.

In einigen Fällen können Sie ein Argument des Typs *Datenreferenz* verwenden, um CICSplex SM Eingabedaten zur Verfügung zu stellen, bevor CICSplex SM Ihnen seine Ausgabe zurückgibt (die Option COUNT für den Befehl FETCH ist hierfür ein Beispiel). Wenn ein Argument des Typs *Daten-*

*referenz* als Eingabe bereitgestellt werden kann, müssen Sie für dieses Argument eine Variable angeben. Wenn Sie keinen Eingabewert angeben wollen, müssen Sie die Variable initialisieren.

#### *Datenbereich*

Ein Sende- oder Empfangsargument, das zum Ermitteln eines Puffers mit Daten verwendet wird. Ein Argument des Typs *Datenbereich* muss eine benannte Variable sein.

Für Ausgabepuffer, die mehrere Ressourcentabellendatensätze empfangen können, erstellt CICSplex SM zur Aufnahme der Daten Stammvariablen (oder füllt diese). Der Nulleintrag des Stammbereichs gibt die Anzahl der Einträge in dem Bereich an.

In der Stammvariablen `W_INT0_EVALDEF` beispielsweise enthält der Eintrag `W_INT0_EVALDEF.0` die Anzahl der zurückgegebenen `EVALDEF`-Ressourcentabellendatensätze. Die Einträge `W_INT0_EVALDEF.1` bis `W_INT0_EVALDEF.n` enthalten die tatsächlichen Ressourcentabellendatensätze.

Es wird eine Stammvariable erstellt, und zwar unabhängig davon, ob es sich bei der tatsächlichen Ausgabe um einen einzelnen Datensatz oder um mehrere Datensätze handelt.

#### *Zeigerreferenz*

Ein Empfangsargument, das verwendet wird, um Zeigerwerte von CICSplex SM an Ihr Programm zu übergeben.

Ein Argument des Typs *Zeigerreferenz* muss eine benannte Variable sein, die für den Empfang der resultierenden Ausgabe verwendet werden kann. Bei den übergebenen Daten handelt es sich um eine Zeichendarstellung einer Hexadezimaladresse.

#### *CPSM-Token*

Ein Sende- oder Empfangsargument, das verwendet wird, um identifizierende Tokens zu übergeben, die von CICSplex SM generiert werden.

Ein Argument des Typs *CPSM-Token* muss eine benannte Variable sein. Tokens werden nicht umgesetzt; der Tokenwert wird unverändert in die Variable eingesetzt.

**Anmerkung:** Jede Variable (oder Stammvariable), die von CICSplex SM zurückgegeben wird, enthält einen ganzen Ressourcentabellendatensatz. Sie können den Befehl `TPARSE` verwenden, um einen Datensatz in einzelne Felder zu zerlegen. Eine Beschreibung dieses Befehls finden Sie in Kapitel 2, „REXX-Funktionen und -Befehle“, auf Seite 13.

---

## MVS-Einschränkungen

Für alle CICSplex SM-API-Befehle gilt eine Reihe allgemeiner Einschränkungen.

- Das Programm muss sich im primären Adressierungsmodus befinden, wenn ein beliebiger CICSplex SM-Service aufgerufen wird. Der Primäradressraum muss der Speicherbereich für die Ausgangsadresse sein. Alle Parameter, die an CICSplex SM übergeben werden, müssen sich im Primäradressraum befinden.
- CICSplex SM bewahrt Zugriffsregister nicht immer befehlsübergreifend bei. Wenn Ihr Programm Zugriffsregister nutzt, sollte es vor dem Aufruf eines CICSplex SM-Service diese speichern und vor der erneuten Verwendung wiederherstellen.



---

## CICS- und CICSplex SM-Wertedatenbereiche

Die Werte für einige CICSplex SM-Ressourcentabellenattribute werden in verschlüsselter Form verwaltet.

Bei diesen Werten kann es sich um CICSplex SM-Wertedatenbereiche (EYUDAs) oder CICS-Wertedatenbereiche (CVDAs) handeln. Zur Umsetzung dieser Werte können Sie eine der beiden integrierten Umsetzerfunktionen verwenden:

### EYUDAs

Verwenden Sie die CICSplex SM-Umsetzerfunktion EYUVALUE.

Die Funktion EYUVALUE steht für Programme, die in REXX geschrieben sind, nicht zur Verfügung. Sie können den Befehl **TPARSE** verwenden, der speziell für REXX-Programme bereitgestellt wird, um auf die Attributwerte in einer Ressourcentabelle zuzugreifen und sie umzusetzen. Eine Beschreibung dieses Befehls finden Sie in Kapitel 2, „REXX-Funktionen und -Befehle“, auf Seite 13.

### CVDAs

Verwenden Sie die CICS-Umsetzerfunktion DFHVALUE.

In einigen CICS-Umgebungen gibt die Funktion DFHVALUE möglicherweise inkompatible CVDA-Werte zurück. Da diese CVDA-Werte in Konflikt mit Werten stehen, die in anderen CICS-Umgebungen verwendet werden, muss CICSplex SM sie modifizieren, damit sie ihre Eindeutigkeit beibehalten. Für alle diese CVDA-Attribute fügt CICSplex SM entweder 8000 oder 9000 zum Wert hinzu, der von DFHVALUE zurückgegeben wird. Weitere Informationen zum Umsetzen von CVDA-Werten finden Sie in TRANSLATE command.

Betrachten Sie beispielsweise die folgende COBOL-Anweisung:

```
MOVE EYUVALUE(QUIESCING) TO EYUDATA
```

Bei dieser Anweisung wird bei der Programmumsetzung der EYUDA-Zeichenwert QUIESCING in seine numerische Entsprechung 48 umgesetzt. CICSplex SM stellt außerdem den Befehl **TRANSLATE** zur Umsetzung von EYUDA- und CVDA-Werten zur Laufzeit bereit. Mithilfe des Befehls **TRANSLATE** können Sie einen EYUDA- oder CVDA-Wert konvertieren, der einer bestimmten Ressourcentabelle und einem bestimmten Attribut zugeordnet ist:

```
EXEC CPSM TRANSLATE OBJECT(WLMAWAOR)
                     ATTRIBUTE(STATUS)
                     FROMCV(48)
                     TOCHAR(EYUCHAR)
                     RESPONSE(RESPDATA)
                     REASON(READDATA)
```

Mit diesem Befehl wird der EYUDA-Wert für das Attribut STATUS der Ressourcentabelle WLMAWAOR während der Programmausführung in seinen Zeichenwert umgesetzt.

Eine Beschreibung des Befehls **TRANSLATE** finden Sie in TRANSLATE command. Eine Liste der EYUDA-Werte, die von CICSplex SM verwendet werden, finden Sie in Kapitel 7, „EYUDA-Werte“, auf Seite 161.

---

## Sprachaspekte

Alle Sprachaspekte, die für die unterschiedlichen Umgebungen gelten (CICS, MVS Batch, TSO und NetView), gelten auch für CICSplex SM-Programme, die für die Ausführung in diesen Umgebungen geschrieben wurden.

---

## Längenoptionen

In vielen API-Befehlen ist der Transfer von Daten zwischen dem Anwendungsprogramm und CICSplex SM einbezogen.

Bei der COBOL-, PL/I- und Assembler-Sprache kann der Umsetzer bestimmte Längenoptionen auf einen Standard einstellen; dies bedeutet, dass sie in Programmen, die Datenbereiche angeben, optional sein können. In C und REXX müssen alle Längenoptionen angegeben werden.

Bei der CICSplex SM-API ist für die meisten Argumente des Typs Datenwert ein Standard möglich; sie werden lediglich von Ihrem Programm an CICSplex SM übergeben. Ausnahme ist die Option LENGTH in den folgenden Befehlen:

- CREATE
- REMOVE
- UPDATE

Argumente des Typs Datenreferenz, die zwischen Ihrem Programm und CICSplex SM hin- und herübergeben werden können, müssen hingegen immer angegeben werden.

Wenn ein API-Befehl eine Längenoption bietet, wird er immer als Vollwort-Binärwert mit Vorzeichen ausgedrückt. Damit wird eine theoretische Obergrenze von 2.147.483.647 Byte hinsichtlich der Länge gesetzt. Die erreichbare Obergrenze variiert von Befehl zu Befehl und mit verschiedenen Sprachcompilern; die maximale Begrenzung für alle Eingabedatenbereiche in einem API-Befehl beträgt jedoch in der Regel 16.124 Byte. Wenn dieser Grenzwert überschritten wird, schlägt der API-Befehl mit der Antwort INVALIDCMD und der Ursache LENGTH fehl.

---

## RESPONSE- und REASON-Optionen

Sobald ein API-Befehl die Verarbeitung abgeschlossen hat, gibt er eine Antwort und, sofern erforderlich, eine Ursache zurück. Sie müssen die RESPONSE- und REASON-Optionen für jeden Befehl angeben, um die Antwort- und Ursachenwerte zu empfangen, die von diesem Befehl zurückgegeben werden.

**Anmerkung:** Die Befehle TBUILD und TPARSE, die nur mit der REXX-Laufzeitschnittstelle verwendet werden können, verwenden die RESPONSE- und REASON-Optionen nicht. Das Ergebnis dieser REXX-spezifischen Prozesse wird durch deren entsprechende Option STATUS zurückgegeben. Weitere Informationen finden Sie in den Beschreibungen zu den Befehlen TBUILD und TPARSE in Kapitel 2, „REXX-Funktionen und -Befehle“, auf Seite 13.

### RESPONSE (Datenreferenz)

*Datenreferenz* ist eine benutzerdefinierte Variable. Bei der Rückgabe des Befehls ist ein Zeichenwert enthalten, der das Ergebnis der Befehlsverarbeitung beschreibt. RESPONSE-Werte werden in der Beschreibung der einzelnen Befehle angegeben.

**REASON**(*Datenreferenz*)

*Datenreferenz* ist eine benutzerdefinierte Variable. Bei der Rückgabe des Befehls ist ein Wert enthalten, der die Antwort auf bestimmte Befehle weiter differenziert. REASON-Werte werden mit den RESPONSE-Werten angegeben, und zwar für die Antworten, die sie verwenden.

Weitere Informationen zu den RESPONSE- und REASON-Optionen finden Sie in *Developing CICSplex SM applications*. Eine Übersicht der RESPONSE- und REASON-Werte nach Befehl finden Sie in Kapitel 6, „RESPONSE- und REASON-Werte“, auf Seite 155.



---

## Kapitel 2. REXX-Funktionen und -Befehle

Dieser Abschnitt enthält ausführliche Beschreibungen der REXX-Funktionen und -Befehle, die mit CICSplex SM bereitgestellt werden. Diese Funktionen und Befehle können nur in Verbindung mit der REXX-Laufzeitschnittstelle verwendet werden.

Jede Beschreibung enthält die folgenden Informationen:

- Eine Beschreibung des Befehls
- Zweck
- Syntax des Befehls (dabei steht *Variable* für eine Variable)
- Verfügbare Optionen für den Befehl
- Vom Befehl zurückgegebene REXX-Antwortcodes

Die Funktionen werden in alphabetischer Reihenfolge angezeigt:

- „EYUAPI()“
- „EYUINIT()“ auf Seite 14
- „EYUREAS()“ auf Seite 14
- „EYURESP()“ auf Seite 15
- „EYUTERM()“ auf Seite 15

---

### Funktionen

Für die REXX-Funktionen, die mit CICSplex SM bereitgestellt werden, werden die standardmäßigen REXX-Variablensubstitutionsregeln angewendet.

Zusätzlich zu den REXX-Rückgabecodes können mit diesen Funktionen EYU-ARnnnn-Nachrichten erzeugt werden. Beschreibungen zu diesen Nachrichten finden Sie in CICSplex SM messages.

#### EYUAPI()

Übergeben eines API-Befehls an CICSplex SM.

```
var = EYUAPI(Befehlsfolge)

OR

var = EYUAPI('Befehlsfolge')
```

#### Beschreibung

Mit dieser Funktion wird ein API-Befehl an CICSplex SM übergeben. Sie müssen eine Funktion des Typs EYUAPI oder EYUINIT absetzen, bevor Sie den Befehl ADDRESS CPSM verwenden können, um API-Befehle an REXX zu übergeben.

#### Optionen

*Befehlsfolge*

Gibt den API-Befehl und die Optionen an, die übergeben werden sollen.

## Rückgabecodes

Im Folgenden finden Sie eine Liste der REXX-Rückgabecodes, die von der Funktion EYUAPI in der zugeordneten Variablen (*Variable*) zurückgegeben werden können.

Diese Rückgabecodes geben an, was REXX mit der EYUAPI-Funktion verarbeitet hat. Sie geben nicht an, ob der übergebene API-Befehl erfolgreich von CICSplex SM verarbeitet wurde. Diese Informationen erhalten Sie über die RESPONSE- und REASON-Werte, die durch den Befehl zurückgegeben werden.

- 0 Die Funktion EYUAPI war erfolgreich.
- 1 Die Funktion EYUAPI war nicht erfolgreich.

## EYUINIT()

Initialisieren der CICSplex SM-API-Umgebung und Zuordnen der erforderlichen Ressourcen.

```
var = EYUINIT()
```

### Beschreibung

Mit diesem Befehl wird die CICSplex SM-API-Umgebung initialisiert und es werden die erforderlichen Ressourcen zugeordnet. EYUINIT muss die erste Funktion sein, die in einem REXX-Programm abgesetzt wird.

**Anmerkung:** Sie müssen eine Funktion des Typs EYUINIT oder EYUAPI absetzen, bevor Sie den Befehl ADDRESS CPSM verwenden können, um API-Befehle an REXX zu übergeben.

## Rückgabecodes

Im Folgenden finden Sie eine Liste der REXX-Rückgabecodes, die von der Funktion EYUINIT in der zugeordneten Variablen (*Variable*) zurückgegeben werden können.

- 0 Die Funktion EYUINIT war erfolgreich.
- 1 Die Funktion EYUINIT war nicht erfolgreich.

## EYUREAS()

Umsetzen des numerischen Werts, der durch die Option REASON eines API-Befehls zurückgegeben wird.

```
var = EYUREAS(Ursache)
```

### Beschreibung

Durch diesen Befehl wird der numerische Wert, der durch die Option REASON eines API-Befehls zurückgegeben wird, in sein Zeichenäquivalent umgesetzt und umgekehrt.

## Optionen

*Ursache*

Ist der umzusetzende REASON-Wert.

## Rückgabecodes

Im Folgenden finden Sie eine Liste der REXX-Rückgabecodes, die von der Funktion EYUREAS in der zugeordneten Variablen (*Variable*) zurückgegeben werden können.

- nnnn** Der numerische Wert oder das Zeichenäquivalent des angegebenen REASON-Werts.
- 1** Der angegebene REASON-Wert ist ungültig und konnte nicht umgesetzt werden.

## EYURESP()

Umsetzen des numerischen Werts, der durch die Option RESPONSE eines API-Befehls zurückgegeben wird.

```
var = EYURESP(Antwort)
```

## Beschreibung

Durch diesen Befehl wird der numerische Wert, der durch die RESPONSE-Option eines API-Befehls zurückgegeben wird, in sein Zeichenäquivalent umgesetzt und umgekehrt.

## Optionen

*Antwort*

Ist der umzusetzende RESPONSE-Wert.

## Rückgabecodes

Im Folgenden finden Sie eine Liste der REXX-Rückgabecodes, die von der Funktion EYURESP in der zugeordneten Variablen (*Variable*) zurückgegeben werden können.

- nnnn** Der numerische Wert oder das Zeichenäquivalent des angegebenen RESPONSE-Werts.
- 1** Der angegebene RESPONSE-Wert ist ungültig und konnte nicht umgesetzt werden.

## EYUTERM()

Beenden der CICSplex SM-API-Umgebung und Freigeben sämtlicher zugeordneter Ressourcen.

```
var = EYUTERM()
```

## Beschreibung

Durch diesen Befehl wird die CICSplex SM-API-Umgebung beendet und es werden sämtliche zugeordneten Ressourcen freigegeben. EYUTERM muss die letzte Funktion sein, die in einem REXX-Programm abgesetzt wird.

**Anmerkung:** Wenn die CICSplex SM-Host-Unterbefehls Umgebung in Ihrem Unternehmen installiert ist (und nicht über das Funktionspaket aufgerufen wird), müssen Sie EYUTERM möglicherweise nicht am Ende jedes Programms verwenden. Die Ressourcen, die zugeordnet bleiben, können vom nächsten Programm, das auf die Host-Unterbefehls Umgebung zugreift, wiederverwendet werden.

## Rückgabecodes

Im Folgenden finden Sie eine Liste der REXX-Rückgabecodes, die von der Funktion EYUTERM in der zugeordneten Variablen (*Variable*) zurückgegeben werden können.

- |   |   |
|---|---|
| 0 | Die Funktion EYUTERM war erfolgreich.       |
| 1 | Die Funktion EYUTERM war nicht erfolgreich. |

---

## Befehle

Mit den im Lieferumfang von CICSplex SM enthaltenen REXX-spezifischen Befehlen wird intern eine Reihe von API-Befehlen durchgeführt und die Ergebnisse werden an REXX zurückgegeben.

Die Befehle werden hier in alphabetischer Reihenfolge dargestellt. Jede Beschreibung enthält den Zweck, die Syntax und die verfügbaren Optionen für den Befehl.

### Anmerkung:

1. Sie können diese Befehle nicht zum Verarbeiten benutzerdefinierter Ansichten einer Ressourcentabelle verwenden, die mit dem Befehl SPECIFY VIEW erstellt wurden. Wenn Sie eine Ansicht mit demselben Namen erstellen, den auch eine bereitgestellte Ressourcentabelle trägt, und dann diesen Namen in einem dieser Befehle angeben, schlägt der Befehl fehl.
2. Mit diesen Befehlen werden die Optionen RESPONSE und REASON nicht verwendet. Das Ergebnis dieser REXX-spezifischen Prozesse wird durch die Option STATUS zurückgegeben.
3. Mit diesen Befehlen werden keine nützlichen Feedback-Informationen (FEEDBACK) bereitgestellt. Die API-Befehle, die intern ausgegeben werden, verwenden denselben Feedback-Bereich wieder. Wenn also einer dieser Befehle mit der Verarbeitung fertig ist, stellt der Feedback-Bereich nicht die gesamte Folge von Ereignissen dar.

Die Befehle sind folgende:

- „TBUILD“ auf Seite 17
- „TPARSE“ auf Seite 18



## TBUILD

Erstellen eines Ressourcentabellendatensatzes aus einer Gruppe von Variablen.

```
►—TBUILD—OBJECT—(—Datenwert—)—PREFIX—(—Datenwert—)—►
►—STATUS—(—Datenreferenz—)—VAR—(—Datenbereich—)—  
└─ASIS─┘►
►—THREAD—(—CPSM-Token—)—►
```

### Beschreibung

Mit diesem Befehl wird ein Ressourcentabellendatensatz aus einer Gruppe von Variablen erstellt, die die einzelnen Attribute einer CICSplex SM- oder CICS-Definition darstellen. Eine Definition wird durch eine Ressourcentabelle mit einem Typ von CPSM-Definition oder CICS-Definition dargestellt.

Sie bilden die Attributvariablen, indem dem Attributnamen ein Präfix hinzugefügt wird, wie z. B. folgendes:

Präfix\_Attribut

Dabei ist Präfix eine Textzeichenfolge, die Sie angeben, und Attribut ist der Name eines Attributs in der Ressourcentabelle. Zwischen dem Präfix und dem Attributnamen muss ein Unterstreichungszeichen ( \_ ) eingefügt werden.

Der Ressourcentabellendatensatz kann in jeder gültigen REXX-Variablen platziert werden, einschließlich einer Stammvariablen.

TBUILD verwendet nur die Attribute, die Sie angeben. Es werden keine Standardwerte für optionale Attribute angenommen. Wenn Sie keine Variable für ein optionales Attribut angeben, wird das entsprechende Feld im Ressourcentabellendatensatz seinem Datentyp entsprechend initialisiert (d. h. Zeichenfelder werden auf 'leer' gesetzt, Binärdaten und EYUDA-Werte werden auf null gesetzt).

**Anmerkung:** Eine Liste der CICSplex SM-Ressourcentabellen nach Typ finden Sie in *Developing CICSplex SM applications*. Eine vollständige Beschreibung einer bestimmten Ressourcentabelle und ihrer Attribute finden Sie in den CICSplex SM-Ressourcentabellen in den Referenzinformationen.

### Optionen

#### ASIS

Gibt an, dass die Werte der Ressourcentabellenattribute bereits in ihrem internen Format vorhanden sind; sie sollen nicht umgesetzt, sondern unverändert verarbeitet werden.

Sie müssen die Option ASIS verwenden, um eine CICSplex SM- oder CICS-Definition, die Sie zuvor (mit dem Befehl TPARSE ASIS) syntaktisch analysiert haben, erneut zu erstellen.

#### OBJECT(*Datenwert*)

Gibt die Ressourcentabelle an, für die ein Datensatz erstellt werden soll. Bei diesem Wert muss es sich um einen aus 1 bis 8 Zeichen bestehenden Namen einer gültigen Ressourcentabelle einer CPSM-Definition oder CICS-Definition handeln.

**Anmerkung:** Sie können den Befehl SPECIFY VIEW nicht verwenden, um eine Ressourcentabellenansicht zu verarbeiten, die durch den Befehl SPECIFY VIEW erstellt wurde. Wenn Sie eine Ansicht mit demselben Namen erstellen, den auch eine bereitgestellte Ressourcentabelle trägt, und dann diesen Namen in einem Befehl des Typs TBUILD angeben, schlägt der Befehl fehl.

#### **PREFIX**(*Datenwert*)

Gibt das Präfix an, das Sie zum Benennen der Variablen verwendet haben, die die Ressourcentabellenattribute enthalten.

**Anmerkung:** Die maximal zulässige Länge für ein Präfix wird von REXX und der Umgebung bestimmt, in der das Programm ausgeführt wird.

#### **STATUS**(*Datenreferenz*)

Benennt eine Variable, um den für diesen Befehl zurückgegebenen REXX-Status zu empfangen. Der Status wird im Zeichenformat wie folgt zurückgegeben:

**OK** Die Verarbeitung des Befehls TBUILD wurde erfolgreich ausgeführt.

#### **SYNTAX ERROR**

Der Befehl TBUILD konnte aufgrund eines Syntaxfehlers nicht verarbeitet werden. Nachrichten des Typs EYUARnnnn, die den Fehler beschreiben, werden in das auf Ihrem System definierte Ziel für die Ausgabe von IRXSAY WRITEERR geschrieben.

#### **FAILURE**

Der Befehl TBUILD ist fehlgeschlagen, da einige Daten ungültig waren, für die eine Verarbeitung versucht wurde. Tracedaten werden in eine REXX-Stammvariable mit dem Namen EYUTRACE geschrieben. Nachrichten des Typs EYUARnnnn, die das Fehlschlagen beschreiben, werden möglicherweise ebenfalls in das auf Ihrem System definierte Ziel für die Ausgabe von IRXSAY WRITEERR geschrieben.

**Anmerkung:** Weitere Informationen zur Stammvariablen EYUTRACE finden Sie in Developing CICSplex SM applications.

#### **THREAD**(*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

#### **VAR**(*Datenbereich*)

Benennt eine Variable, um den Ressourcentabellendatensatz zu empfangen, der durch TBUILD erstellt wird.

## **TPARSE**

Durchführen einer Syntaxanalyse für einen Ressourcentabellendatensatz, von einer Variablen bis hinzu einer Gruppe von Variablen.

```

▶▶—TPARSE—OBJECT—(—Datenwert—)—PREFIX—(—Datenwert—)——————▶
▶—STATUS—(—Datenreferenz—)—VAR—(—Datenbereich—)—[ASIS]—————▶
▶—THREAD—(—CPSM-Token—)——————▶▶

```

## Beschreibung

Mit diesem Befehl wird eine Syntaxanalyse für einen Ressourcentabellendatensatz durchgeführt, von einer Variablen bis hinzu einer Gruppe von Variablen, die die einzelnen Attribute der Tabelle darstellen. Sie können TPARSE mit einem beliebigen CICSplex SM-Ressourcentabellentyp verwenden.

Die Variable der Ressourcentabelle kann jede gültige REXX-Variable sein, einschließlich einer Stammvariablen. Die Ausgabevariablen werden gebildet, indem dem Attributnamen ein Präfix hinzugefügt wird, wie z. B. folgendes:

Präfix\_Attribut

Dabei ist Präfix eine Textzeichenfolge, die Sie angeben, und Attribut ist der Name eines Attributs in der Ressourcentabelle. Zwischen dem Präfix und dem Attributnamen wird ein Unterstreichungszeichen ( \_ ) eingefügt.

**Anmerkung:** Vollständige Beschreibungen der Ressourcentabellen und ihrer Attribute finden Sie in den CICSplex SM-Ressourcentabellen in den Referenzinformationen.

## Optionen

### ASIS

Gibt an, dass die Werte der Ressourcentabellenattribute nicht in ihr externes Format umgesetzt werden sollen; sie sollen unverändert zurückgegeben werden. Attributwerte werden wie folgt dargestellt:

- Zeichenwerte weisen abschließende Leerzeichen auf.
- Binärwerte weisen führende Nullen auf und werden nicht in das Anzeigeformat konvertiert.
- EYUDA- und CVDA-Werte werden nicht ins Zeichenformat konvertiert.

Sie müssen die Option ASIS verwenden, um eine CPSM-Definitions- oder eine CICS-Definitionsressourcentabelle syntaktisch zu analysieren, die Sie erneut erstellen wollen (mit dem Befehl TBUILD ASIS).

**Anmerkung:** Wenn Sie die Option ASIS mit EYUDA- oder CVDA-Werten verwenden, können Sie den Befehl TRANSLATE verwenden, um den codierten numerischen Wert in einen Zeichenwert umzuwandeln.

### OBJECT(Datenwert)

Gibt die Ressourcentabelle an, die syntaktisch analysiert werden soll. Bei diesem Wert muss es sich um einen aus 1 bis 8 Zeichen bestehenden Namen einer gültigen Ressourcentabelle handeln.

**Anmerkung:** Sie können den Befehl TPARSE nicht verwenden, um eine Ressourcentabellenansicht zu verarbeiten, die durch den Befehl SPECIFY VIEW erstellt wurde. Wenn Sie eine Ansicht mit demselben Namen erstellen, den auch eine bereitgestellte Ressourcentabelle trägt, und dann diesen Namen in einem Befehl des Typs PARSE angeben, schlägt der Befehl fehl.

### PREFIX(Datenwert)

Gibt das Präfix an, das verwendet werden soll, um die von TPARSE zurückgegebenen Attributvariablen zu benennen.

**Anmerkung:** Die maximal zulässige Länge für ein Präfix wird von REXX und der Umgebung bestimmt, in der das Programm ausgeführt wird.

**STATUS**(*Datenreferenz*)

Benennt eine Variable, um den für diesen Befehl zurückgegebenen REXX-Status zu empfangen. Der Status wird im Zeichenformat wie folgt zurückgegeben:

**OK** Die Verarbeitung des Befehls TPARSE wurde erfolgreich ausgeführt.

**SYNTAX ERROR**

Der Befehl TPARSE konnte aufgrund eines Syntaxfehlers nicht verarbeitet werden. Nachrichten des Typs EYUARnnnn, die den Fehler beschreiben, werden in das auf Ihrem System definierte Ziel für die Ausgabe von IRXSAY WRITEERR geschrieben.

**FAILURE**

Der Befehl TPARSE ist fehlgeschlagen, da einige Daten ungültig sind, für die eine Verarbeitung versucht wurde. Tracedaten werden in eine REXX-Stammvariable mit dem Namen EYU\_TRACE geschrieben. Nachrichten des Typs EYUARnnnn, die das Fehlschlagen beschreiben, werden möglicherweise ebenfalls in das auf Ihrem System definierte Ziel für die Ausgabe von IRXSAY WRITEERR geschrieben.

**Anmerkung:** Weitere Informationen zur Stammvariablen EYUTRACE finden Sie in Developing CICSplex SM applications.

**THREAD**(*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

**VAR**(*Datenbereich*)

Gibt eine Variable an, die den Ressourcentabellendatensatz enthält, der syntaktisch analysiert werden soll.

---

## Kapitel 3. CICSplex SM-API-Befehle

Dieser Abschnitt enthält ausführliche Beschreibungen der API-Befehle. Alle diese Befehle können entweder mit der Befehlsschnittstelle oder mit der REXX-Laufzeitschnittstelle verwendet werden.

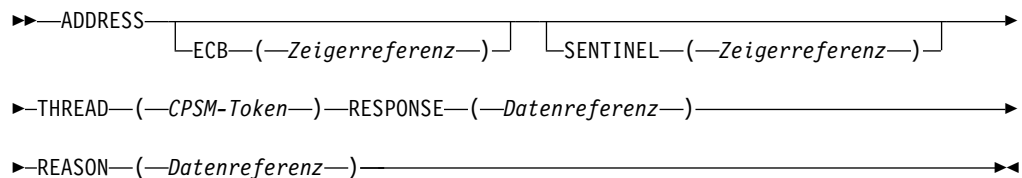
Jede Beschreibung enthält je nach Bedarf die folgenden Informationen:

- Eine Beschreibung des Befehls
- Hinweise zur Verwendung
- Zugehörige Befehle
- Syntax des Befehls
- Verfügbare Optionen für den Befehl
- Vom Befehl zurückgegebene Antworten

---

### ADDRESS

Gewähren von Zugriff auf CICSplex SM-Speicherbereiche.



#### Beschreibung

Mit dem Befehl ADDRESS wird Zugriff auf CICSplex SM-Speicherbereiche gewährt.

- Mit ADDRESS werden die Adressen zweier Steuerfelder zurückgegeben, die den einzelnen API-Threads zugeordnet sind:
  - Ereignissteuerblock (ECB)
  - Sentinel
- Wenn Ihr Programm in REXX geschrieben ist, werden die ECB- und Sentinel-Werte als Zeichendarstellungen der Hexadezimaladressen zurückgegeben. Für den Zugriff auf den Speicher bei diesen Adressen müssen Sie die REXX-Funktion STORAGE verwenden.

#### Zugehörige Befehle

LISTEN, RECEIVE

#### Optionen

##### ECB(*Zeigerreferenz*)

Benennt eine Variable, um die Adresse des Ereignissteuerblocks zu empfangen, die gepostet wird, sobald asynchrone Anforderungen, die diesem Thread zugeordnet sind, auf die Verarbeitung warten. Das ECB-Feld wird gelöscht, sobald der Zählerwert im Feld SENTINEL den Wert 0 erreicht.

**REASON**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

**RESPONSE**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**SENTINEL**(*Zeigerreferenz*)

Benennt eine Variable, um die Adresse eines 4 Byte großen Zählers abgeschlossener asynchroner Anforderungen zu empfangen, die diesem Thread zugeordnet sind.

Der Sentinel-Wert erhöht sich jedes Mal, wenn eine asynchrone Anforderung abgeschlossen wird. Beispiele für Anforderungen einer asynchronen Operation:

- Ein Befehl wird mit der Option NOWAIT ausgegeben
- Es tritt ein Ereignis auf, das in einem Befehl LISTEN angegeben wird

Der Sentinel-Wert verringert sich, wenn ein Befehl des Typs RECEIVE abgesetzt wird. Wenn der Zählerwert 0 ist, bedeutet dies, dass keine ausstehenden asynchronen Anforderungen für den Empfang bereitstehen.

**Anmerkung:** Jeder API-Verarbeitungsthread kann maximal 256 asynchrone Anforderungen gleichzeitig verarbeiten (wie durch den SENTINEL-Zähler angegeben).

**THREAD**(*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl ADDRESS zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

**ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

**NOSERVICE**

Das Anwendungsstubprogramm konnte das API-Service Modul nicht laden.

**NOSTORAGE**

Das Anwendungsstubprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

### INVALIDPARM

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- ECB
- SENTINEL
- THREAD.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

### NOTAVAILABLE

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

#### APITASK

Die API-Steuersubtask ist nicht aktiv.

#### CPSMAPI

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

### SERVERGONE

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

### VERSIONINVL

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

#### NOTSUPPORTED

Die Version des Anwendungsstubprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

#### NOTVSNCONN

Die Version des Anwendungsstubprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

---

## CANCEL

Abbrechen der Benachrichtigungsanforderung, die durch einen vorherigen Befehl des Typs LISTEN erzeugt wurde.

►►—CANCEL—NOTIFICATION—(—CPSM-Token—)—THREAD—(—CPSM-Token—)——————►

►—RESPONSE—(—Datenreferenz—)—REASON—(—Datenreferenz—)——————►◄

### Beschreibung

Mit diesem Befehl wird die Benachrichtigungsanforderung abgebrochen, die durch einen vorherigen Befehl des Typs LISTEN erzeugt wurde.

### Zugehörige Befehle

LISTEN

### Optionen

#### NOTIFICATION(CPSM-Token)

Gibt die Benachrichtigungsanforderung an, die abgebrochen werden soll. Der

Wert für *CPSM-Token*, der eine Benachrichtigungsanforderung angibt, wird durch den Befehl LISTEN zurückgegeben.

**REASON**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

**RESPONSE**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**THREAD**(*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl CANCEL zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

**NOSERVICE**

Das Anwendungsstubprogramm konnte das API-Servicemodul nicht laden.

**NOSTORAGE**

Das Anwendungsstubprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- NOTIFICATION
- THREAD.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

**NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':



#### APITASK

Die API-Steuersubtask ist nicht aktiv.

#### CPSMAPI

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

#### SERVERGONE

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

#### VERSIONINVL

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

#### NOTSUPPORTED

Die Version des Anwendungsstubprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

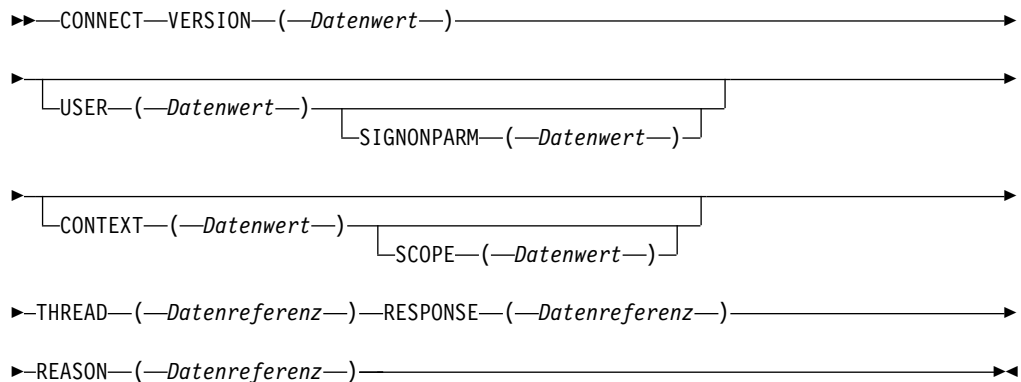
#### NOTVSNCONN

Die Version des Anwendungsstubprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

---

## CONNECT

Einrichten einer Verbindung mit CICSplex SM, Definieren eines API-Verarbeitungsthreads und Bereitstellen von Standardeinstellungen, die vom Thread verwendet werden sollen.



### Beschreibung

Die Besonderheiten des Verbindungsprozesses sind von der Umgebung abhängig, in der Ihr Programm ausgeführt wird. Eine vollständige Beschreibung des Verbindungsprozesses finden Sie in *Developing CICSplex SM applications*.

### Zugehörige Befehle

DISCONNECT, QUALIFY, TERMINATE

### Optionen

#### CONTEXT (Datenwert)

Gibt den Standardkontext für Befehle an, die für diesen Thread abgesetzt wer-

den. Bei dem Kontext muss es sich um einen aus 1 bis 8 Zeichen bestehenden Namen eines CMAS oder CICSplex handeln.

Der Standardkontext ist für alle Befehle, die für den Thread abgesetzt werden, wirksam, es sei denn, Sie überschreiben ihn für einen bestimmten Befehl oder ändern ihn durch Absetzen des Befehls QUALIFY. Als Alternative zur Angabe eines Standardkontexts für den Thread können Sie den Kontext für einzelne Befehle während ihrer Verarbeitung angeben.

Wenn Sie die Option CONTEXT nicht angeben, ist der Standardkontext für den Thread der CMAS, mit dem der Thread verbunden ist.

**REASON**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

**RESPONSE**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**SCOPE**(*Datenwert*)

Gibt den Standardbereich für Befehle an, die für diesen Thread abgesetzt werden.

Mit der Option SCOPE wird die Option CONTEXT qualifiziert. Wenn es sich bei dem Kontext um ein CICSplex handelt, kann für den Geltungsbereich Folgendes gelten:

- Es ist der aus 1 bis 8 Zeichen bestehende Name des CICSplex selbst
- Es ist ein CICS-System oder eine CICS-Systemgruppe innerhalb des CICSplex
- Es ist ein logischer Bereich wie er in einer CICSplex SM-Ressourcenbeschreibung (RESDESC) definiert wurde

Wenn es sich bei dem Kontext um einen CMAS handelt, ist diese Option bedeutungslos und wird ignoriert.

Der Standardbereich ist für alle Befehle, die für den Thread abgesetzt werden, wirksam, es sei denn, Sie überschreiben ihn für einen bestimmten Befehl oder ändern ihn durch Absetzen des Befehls QUALIFY. Wenn Sie die Option SCOPE nicht angeben, wird kein Standardbereich angenommen.

**Anmerkung:** Für bestimmte API-Befehle ist ein gültiger Bereich erforderlich, wenn der Kontext ein CICSplex ist. Wenn Sie keinen Bereich im Befehl CONNECT oder QUALIFY angeben, müssen Sie die Option SCOPE beim Ausgeben eines der folgenden Befehle für eine Ressourcentabelle angeben, die eine CICS-Ressource darstellt:

- GET
- PERFORM OBJECT
- PERFORM SET
- REFRESH
- SET

**SIGNONPARM**(*Datenwert*)

Gibt einen aus 1 bis 8 Zeichen bestehenden Anmeldeparameter an, der bei Ihrem Unternehmen an die API-Sicherheitsexitroutine übergeben werden soll (EYU9XESV). Kennwortphrasen werden nicht unterstützt.

Wenn die CMAS-Sicherheit aktiv ist und keine Sicherheit in der Umgebung definiert ist, in der das API-Programm ausgeführt wird, übergibt CICSplex SM

die Werte USER und SIGNONPARM des Befehls CONNECT an EYU9XESV. Weitere Informationen zur API-Sicherheit finden Sie in Developing CICSplex SM applications.

**THREAD** *(Datenreferenz)*

Benennt eine Variable, um das Vollwort-Token zu empfangen, das CICSplex SM diesem Verarbeitungsthread zuordnet.

Dieses identifizierende Token muss in allen nachfolgenden Befehlen angegeben werden, die für diesen Thread abgesetzt werden.

**USER** *(Datenwert)*

Gibt eine aus 1 bis 8 Zeichen bestehende Benutzer-ID an, die bei Ihrem Unternehmen an die API-Sicherheitsexitroutine übergeben werden soll (EYU9XESV).

Wenn die CMAS-Sicherheit aktiv ist und CICSplex SM keine definierte Sicherheit in der Umgebung findet, in der das API-Programm ausgeführt wird, werden die Werte USER und SIGNONPARM des Befehls CONNECT an EYU9XESV übergeben. Weitere Informationen zur API-Sicherheit finden Sie in Developing CICSplex SM applications.

**VERSION** *(Datenwert)*

Gibt das Release der CICSplex SM-Ressourcentabellendaten an, die für Ihr Programm verfügbar sein sollen. Bei dem Wert für VERSION muss es sich um eine aus 4 Zeichen bestehende Zahl eines gültigen CICSplex SM-Release handeln, z. B. 0410 für CICS TS 4.1.

**Hinweise:**

1. Der Wert für VERSION muss 0120 oder größer sein. Die API kann nicht auf Daten eines Release von CICSplex SM vor Release 2 zugreifen.
2. Der Wert für VERSION muss kleiner-gleich der Version der CICSplex SM-Laufzeitumgebung sein.
3. Es ist möglich, einen Wert für VERSION anzugeben, der größer als das Release ist, mit dem Ihr API-Programm ursprünglich geschrieben wurde; allerdings müssen die folgenden Voraussetzungen gelten:
  - Sie kompilieren Ihr Programm mit den entsprechenden Copybooks für die angegebene Version.
  - Ihr Programm ist mit den Copybooks für die angegebene Version kompatibel.

Ausführliche Informationen zu den bei der Ausführung unter einem anderen Release zu berücksichtigenden Aspekten finden Sie in Developing CICSplex SM applications.

## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl CONNECT zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

**APITASKERR**

Die API-Steuersubtask hat beim Start einen Fehler festgestellt.

**INVALIDTCB**

Ein Programm, das den Anwendungsstub für Nicht-CICS-Umgebungen verwendet (EYU9ABSI) versuchte, eine Verbindung herzustellen, als das Programm auf einem CICS TCB ausgeführt wurde.

**NOSERVICE**

Das Anwendungsstubprogramm konnte das API-Servicemodul nicht laden.

**NOSTORAGE**

Das Anwendungsstubprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

**SOERESOURCE**

Eine erforderliche Ressource, deren Eigner der ESSS-Adressraum (Environment Services System Services) ist, ist nicht verfügbar.

**SOLRESOURCE**

Eine erforderliche Ressource, deren Eigner ein lokaler Eigner ist (d. h. Eigner des Adressraums, in dem der Verarbeitungsthread ausgeführt wird), ist nicht verfügbar.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- CONTEXT
- SCOPE
- SIGNONPARM
- USRID
- VERSION

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

**NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

**APITASK**

Die API-Steuersubtask ist nicht aktiv.

**CPSMAPI**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

### CPSMSERVER

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread versucht hat, eine Verbindung herzustellen, ist nicht verfügbar.

### CPSMSYSTEM

Es sind keine CICSplex SM-Systeme verfügbar.

### CPSMVERSION

Es ist kein CICSplex SM-System in der angegebenen Version verfügbar.

### NOTPERMIT

Aus einem der folgenden Gründe kam es zu einer Unzulässigkeitsbedingung:

#### EXPIRED

Die Sicherheitsberechtigung der angegebenen Benutzer-ID ist abgelaufen.

#### SIGNONPARM

Der angegebene Anmeldeparameter ist für die Benutzer-ID nicht autorisiert.

#### USRID

Die angegebene Benutzer-ID verfügt nicht über die erforderliche Sicherheitsberechtigung.

### VERSIONINVL

Aus den folgenden Gründen kam es zu einem Versionskonflikt:

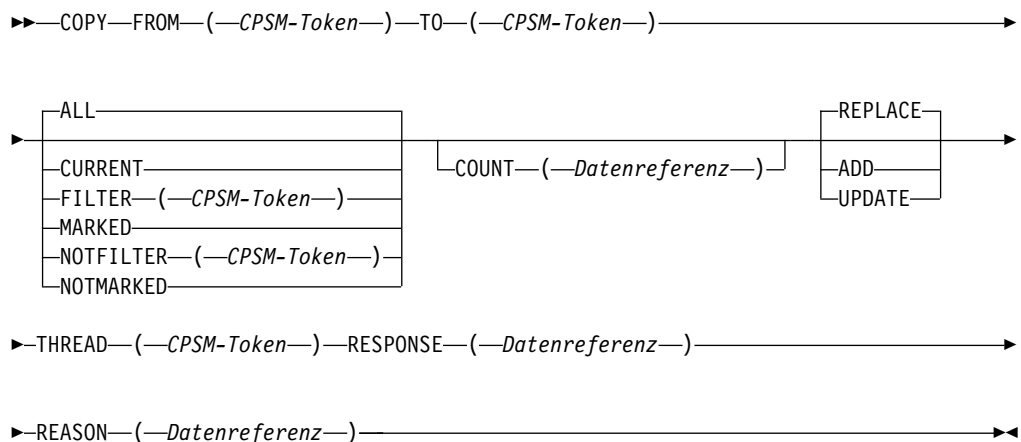
### NOTSUPPORTED

Die Version des Anwendungstreibprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

---

## COPY

Kopieren von Ressourcentabellendatensätzen.



## Beschreibung

Mit diesem Befehl werden einige oder alle Ressourcentabellendatensätze in einer Ergebnisliste in eine andere Ergebnisliste desselben Verarbeitungsthreads kopiert.

- Der Befehl COPY beginnt immer mit der Verarbeitung des letzten abgerufenen Datensatzes und nicht mit dem nächsten Datensatz in der Ergebnisliste.
- Bei der Zielergebnisliste kann es sich um eine vorhandene Ergebnisliste oder um eine neue Ergebnisliste handeln, die durch diesen Prozess erstellt wird. Wenn Sie eine vorhandene Ergebnisliste als Ziel angeben, können Sie die vorhandenen Datensätze überschreiben oder diesen Daten hinzufügen.
- Eine Ergebnisliste kann nur einen einzigen Datensatz für eine bestimmte Ressource enthalten. Wenn während des Kopierprozesses doppelte Datensätze gefunden werden, bestimmt die von Ihnen angegebene Option ADD, REPLACE oder UPDATE, welcher Datensatz aufbewahrt wird.
- Um ausgewählte Datensätze aus einer Quellenergebnisliste zu kopieren, können Sie die folgenden Befehle verwenden:
  - Befehl SPECIFY FILTER zum Definieren eines Filters für die Quellenergebnisliste.
  - Befehle MARK und UNMARK zum Markieren von Datensätzen in der Quellenergebnisliste. Sämtliche Markierungen, die Sie in Datensätzen in der Quellenergebnisliste platzieren, werden nicht beibehalten, wenn diese Datensätze in die Zielergebnisliste kopiert werden.
- Die relative Position der Datensätze in der Zielergebnisliste ist möglicherweise nicht identisch mit der relativen Position der Datensätze in der Quellenergebnisliste. Die Position kann wie folgt beeinflusst werden:
  - Durch in der Quellenergebnisliste verbliebene, gelöschte Datensätze (Angabe von COPY ALL) und andere Datensätze, die ihre Position in der Zielergebnisliste voraussetzen.
  - Durch die Sortierreihenfolge, die der Zielergebnisliste zugeordnet ist (falls vorhanden). Wenn die Zielergebnisliste nicht vorhanden ist, werden die Datensätze in derselben Reihenfolge kopiert, in der sie in der Quellenergebnisliste enthalten sind. Wird eine vorhandene Ergebnisliste als Ziel angegeben, werden Datensätze kopiert und anschließend der für diese Ergebnisliste geltenden Reihenfolge entsprechend sortiert.

## Zugehörige Befehle

DELETE, DISCARD, GET, GETDEF, LOCATE, MARK, ORDER, PERFORM OBJECT, QUERY, SPECIFY FILTER

## Optionen

### ADD

Fügt die Ressourcentabellendatensätze aus der Quellenergebnisliste zu einer bereits vorhandenen Zielergebnisliste hinzu. Falls doppelte Datensätze gefunden werden, wird der Datensatz in der Zielergebnisliste beibehalten.

Wenn keine vorhandene Ergebnisliste als Ziel angegeben ist, wird die Option ADD ignoriert.

### ALL

Kopiert alle Ressourcentabellendatensätze in der Quellenergebnisliste zur Zielergebnisliste.

Alle Datensätze, die aus der Quellenergebnisliste gelöscht wurden, werden nicht kopiert. Ist die Option ALL aktiviert, wird eine Ergebnisliste komprimiert, indem gelöschte Datensätze in der Quellenergebnisliste beibehalten und die verbliebenen Datensätze in eine neue Ergebnisliste kopiert werden.

**COUNT** (*Datenreferenz*)

Benennt eine Variable, um die Anzahl der Ressourcentabellendatensätze in der Zielergebnisliste zu empfangen, wenn der Kopierprozess abgeschlossen ist.

**CURRENT**

Kopiert nur den aktuellen Ressourcentabellendatensatz in der Quellenergebnisliste zur Zielergebnisliste.

**FILTER** (*CPSM-Token*)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option FILTER kopiert nur die Ressourcentabellendatensätze, die den angegebenen Filterkriterien entsprechen.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

**FROM** (*CPSM-Token*)

Gibt die Quellenergebnisliste für diese Operation an. Die Ergebnisliste kann durch einen der folgenden Befehle erzeugt werden:

- COPY
- GET
- GETDEF
- PERFORM OBJECT.

**MARKED**

Kopiert nur die Ressourcentabellendatensätze, die in der Quellenergebnisliste markiert sind. Ressourcentabellendatensätze können mit den Befehlen MARK und UNMARK markiert werden.

**NOTFILTER** (*CPSM-Token*)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option NOTFILTER kopiert nur die Ressourcentabellendatensätze, die den angegebenen Filterkriterien nicht entsprechen.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

**NOTMARKED**

Kopiert nur die Ressourcentabellendatensätze, die in der Quellenergebnisliste nicht markiert sind. Ressourcentabellendatensätze können mit den Befehlen MARK und UNMARK markiert werden.

**REASON** (*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

**REPLACE**

Löscht die Ressourcentabellendatensätze in einer vorhandenen Zielergebnisliste und ersetzt sie durch die Ergebnisse dieser Kopieroperation. Wenn die Kopieroperation nicht in kopierten Ressourcentabellendatensätzen resultiert, wird die Zielergebnisliste verworfen.

Wenn keine vorhandene Ergebnisliste als Ziel angegeben ist, wird die Option REPLACE ignoriert.

**RESPONSE** (*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**THREAD**(*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

**TO**(*CPSM-Token*)

Gibt die Zielergebnisliste für diese Operation an. Die Ergebnisliste kann durch einen der folgenden Befehle erzeugt werden:

- COPY
- GET
- GETDEF
- PERFORM OBJECT.

**Anmerkung:** Die Zielergebnisliste darf nicht mit der Quellenergebnisliste identisch sein, die Sie in der Option FROM angegeben haben.

CICSplex SM erstellt eine neue Ergebnisliste und gibt das identifizierende Token im selben Feld zurück, wenn für dieses Feld Folgendes gilt:

- Es ist auf eine binäre Null gesetzt (in COBOL, C, PL/I oder Assembler)
- Es ist eine nicht initialisierte Variable (in REXX)

**UPDATE**

Aktualisiert eine vorhandene Zielergebnisliste mit Ressourcentabellendatensätzen aus der Quellenergebnisliste. Falls doppelte Datensätze gefunden werden, ersetzt der Datensatz in der Quellenergebnisliste den Datensatz in der Zielergebnisliste.

Wenn keine vorhandene Ergebnisliste als Ziel angegeben ist, wird die Option UPDATE ignoriert.

## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl COPY zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

**NODATA**

Es wurden keine Datensätze gefunden, die mit den angegebenen Suchkriterien übereinstimmen.

**BUSY** Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Aktiv':

**FROM**

Die in der Option FROM angegebene Quellenergebnisliste wird von einem anderen Befehl verarbeitet.

**TO**

Die in der Option TO angegebene Zielergebnisliste wird von einem anderen Befehl verarbeitet. Zu dieser Bedingung kann es kommen, wenn Sie dieselbe Ergebnisliste bei den Optionen FROM und TO angegeben haben.

**ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

**NOSERVICE**

Das Anwendungsstubprogramm konnte das API-Servicemodul nicht laden.



**NOSTORAGE**

Das Anwendungsstubsprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**INCOMPATIBLE**

Aus einem der folgenden Gründe kam es zu einer Inkompatibilitätsbedingung:

**INVALIDOBJ**

Die in der Option TO angegebene Zielergebnisliste ist mit der in der Option FROM angegebenen Quellenergebnisliste nicht kompatibel. Die Ergebnislisten müssen denselben Typ von Ressourcentabellendatensätzen enthalten.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- FILTER
- FROM
- NOTFILTER
- THREAD
- TO.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

**NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

**APITASK**

Die API-Steuersubtask ist nicht aktiv.

**CPSMAPI**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

**SERVERGONE**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

**VERSIONINVL**

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

## NOTSUPPORTED

Die Version des Anwendungsstubprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

## NOTVSNNCONN

Die Version des Anwendungsstubprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

---

# CREATE

Erstellen einer neuen CICSplex SM- oder CICS-Definition.

```
►►—CREATE—OBJECT—(—Datenwert—)—FROM—(—Datenbereich—)—►
►—LENGTH—(—Datenwert—)—►
►
  └─PARM—(—Datenbereich—)—PARMLN—(—Datenwert—)—┘
►
  └─CONTEXT—(—Datenwert—)—┘ └─SCOPE—(—Datenwert—)—┘►
►—THREAD—(—CPSM-Token—)—RESPONSE—(—Datenreferenz—)—►
►—REASON—(—Datenreferenz—)—►◄
```

## Beschreibung

Durch diesen Befehl wird eine neue CICSplex SM- oder CICS-Definition erstellt, und zwar mithilfe der von Ihnen angegebenen Attributwerte. Die neue Definition wird im CICSplex SM-Datenrepository gespeichert. Für Definitionen mit einem CICSplex als Kontext (z. B. Workload-Management- oder Echtzeitanalysedefinitionen) wird die Definition automatisch an alle CMASs verteilt, die an der Verwaltung des CICSplex beteiligt sind.

## Zugehörige Befehle

REMOVE, UPDATE

## Optionen

### CONTEXT (Datenwert)

Gibt den Kontext für diesen Befehl an. Bei dem Kontext muss es sich um einen aus 1 bis 8 Zeichen bestehenden Namen eines CMAS oder CICSplex handeln.

Wenn Sie die Option CONTEXT nicht angeben, wird der Standardkontext für den Thread angenommen.

### FROM (Datenbereich)

Gibt einen Puffer an, der einen Ressourcentabellendatensatz enthält, der die zu erstellende Definition darstellt.

Der Datensatz muss alle Attribute für die Ressourcentabelle enthalten, die in der Option OBJECT angegeben ist. Für optionale Attribute, die Sie nicht angeben möchten, setzen Sie das Feld auf Nullwerte (d. h. 0).

Eine Liste aller zulässigen Nullwerte finden Sie in CICSplex SM resource tables.

**LENGTH**(*Datenwert*)

Ein Vollwortwert, der die Länge des FROM-Puffers angibt.

**OBJECT**(*Datenwert*)

Gibt die Ressourcentabelle an, die die zu erstellende Definition darstellt. Bei diesem Wert muss es sich um einen aus 1 bis 8 Zeichen bestehenden Namen einer gültigen Ressourcentabelle einer CICSplex SM-Definition oder CICS-Definition handeln. Eine Liste der CICSplex SM-Ressourcentabellen nach Typ finden Sie in *Developing CICSplex SM applications*.

**PARM**(*Datenbereich*)

Gibt einen Puffer mit dem Parameterausdruck an, der zum Erstellen der Definition verwendet werden soll.

Details zur Verwendung eines Parameterausdrucks mit dem Befehl CREATE finden Sie in *Developing CICSplex SM applications*. Eine Beschreibung der Parameter, die für eine angegebene Ressourcentabelle gültig sind, finden Sie in *Übersicht über die CICSplex SM-Ressourcentabellen in den Referenzinformationen* und in *Übersicht über die CICSplex SM-Ressourcentabellen in den Referenzinformationen*.

**PARMLN**(*Datenwert*)

Ein Vollwortwert, der die Länge des PARM-Puffers angibt.

**REASON**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

**RESPONSE**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**SCOPE**(*Datenwert*)

Gibt den Bereich für diesen Befehl an.

Wenn der aktuelle Kontext (definiert durch diesen Befehl oder einen früheren Befehl des Typs CONNECT oder QUALIFY) ein CICSplex ist und die Option OBJECT eine CICS-Definitionsressource angibt und die Option PARM den Parameter CSD beinhaltet, kann ein gültiger Bereich angegeben werden.

Der Bereich kann ein CICS-System innerhalb des CICSplex sein. Wenn es sich bei dem aktuellen Kontext um einen CMAS handelt oder die Option OBJECT einen anderen Typ von Ressourcentabelle angibt oder der Parameter CSD in keiner CICS-Definitionsressource angegeben ist, ist diese Option bedeutungslos und wird ignoriert.

Wenn SCOPE für den Befehl gültig ist und Sie die Option SCOPE nicht angeben, wird der Standardbereich für den Thread angenommen. Wenn der aktuelle Kontext ein CICSplex ist und kein Standardbereich für den Befehl CONNECT oder QUALIFY festgelegt wurde, erhalten Sie eine Antwort des Typs INVALIDPARM für die Option SCOPE.

**THREAD**(*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl CREATE zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

### ENVIRONERROR

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

#### NOSERVICE

Das Anwendungsstubsprogramm konnte das API-Servicemodul nicht laden.

#### NOSTORAGE

Das Anwendungsstubsprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

#### NOTPROCESSED

Einer der MASs, an den die Anforderung übertragen wurde, konnte die Anforderung nicht verarbeiten (nur bei CSD-Anforderungen).

#### REQTIMEOUT

Einer der CMASs, an die die Anforderung übertragen wurde, antwortete nicht.

#### SOCRESOURCE

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

### FAILED

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

#### ABENDED

Die Befehlsverarbeitung wurde abnormal beendet.

#### EXCEPTION

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

### INVALIDCMD

Aus dem folgenden Grund ist der Befehl ungültig:

#### LENGTH

Die Gesamtlänge aller Optionen im Befehl überschreitet den maximalen Grenzwert.

### INVALIDPARM

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- CONTEXT
- FROM
- LENGTH
- OBJECT
- PARM
- PARMLen
- THREAD.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

#### **NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

##### **APITASK**

Die API-Steuersubtask ist nicht aktiv.

##### **CMAS**

Ein CMAS (CICSplex SM-Adressraum), an den die Anforderung übertragen wurde, ist nicht verfügbar.

##### **CPSMAPI**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

##### **CSDAPI**

Es ist keine Unterstützung für die CICSplex SM-API für den Zugriff auf CICS CSD verfügbar.

##### **MAINTPOINT**

Der Wartungspunkt für den aktuellen Kontext ist nicht verfügbar.

#### **NOTPERMIT**

Aus einem der folgenden Gründe kam es zu einer Unzulässigkeitsbedingung:

##### **USRID**

Die Benutzer-ID, die dem Verarbeitungsthread zugeordnet ist, verfügt nicht über die erforderliche Sicherheitsberechtigung.

#### **SERVERGONE**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

#### **TABLEERROR**

Ein Ressourcentabellendatensatz ist aus einem der folgenden Gründe ungültig:

##### **DATAERROR**

Der Wert, der einem oder mehreren Ressourcentabellenattributen zugeordnet ist, ist ungültig. Dieser Fehler kann in folgenden Situationen auftreten:

- Wenn der Ressourcentabelle erforderliche Attribute fehlen, sie ein oder mehrere miteinander in Konflikt stehende Attribute enthält oder sie doppelt vorhanden ist.
- Wenn eine CICS-Ressourcendefinition Attribute enthält, die dazu führen würden, dass der Befehl EXEC CICS CREATE Warnungen ausgeben würde.

Verwenden Sie den Befehl FEEDBACK, um weitere Informationen zu diesem Fehler abzurufen.

##### **INVALIDATTR**

Eines der Ressourcentabellenattribute ist ungültig.

##### **INVALIDVER**

Die angegebene Version der Ressourcentabelle wird von CICSplex SM nicht unterstützt.

## VERSIONINVL

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

## NOTSUPPORTED

Die Version des Anwendungsstubprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

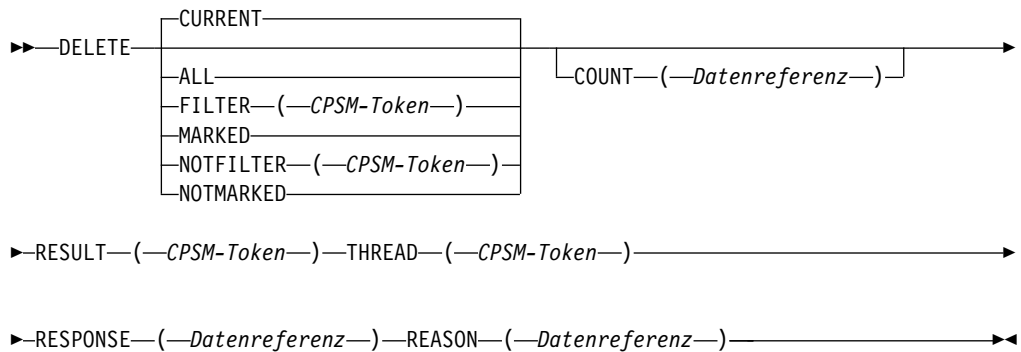
## NOTVSNCONN

Die Version des Anwendungsstubprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

---

# DELETE

Löschen von Ressourcentabellendatensätzen.



## Beschreibung

Mit diesem Befehl werden ein oder mehrere Ressourcentabellendatensätze aus einer Ergebnisliste gelöscht.

- Der Befehl DELETE beginnt immer mit der Verarbeitung des letzten abgerufenen Datensatzes und nicht mit dem nächsten Datensatz in der Ergebnisliste.
- Die von Ihnen gelöschten Datensätze werden als 'Gelöscht' markiert, bleiben aber in der Ergebnisliste an ihren Positionen. Die übrigen Datensätze behalten ebenfalls ihre Positionen bei; sie werden nicht neu nummeriert. Alle API-Befehle, die Sie nach einem Befehl des Typs DELETE absetzen, überspringen die gelöschten Datensätze in einer Ergebnisliste. Eine Ausnahme ist der Befehl ORDER, der alle Datensätze in einer Ergebnisliste sortiert, einschließlich gelöschter Datensätze. Wenn Sie versuchen, einen Befehl für einen gelöschten Datensatz abzusetzen, empfangen Sie den RESPONSE-Wert NODATA.
- Wenn Sie gelöschte Datensätze entfernen und eine Ergebnisliste komprimieren möchten, können Sie die verbleibenden Datensätze in eine neue Ergebnisliste kopieren. Verwenden Sie den Befehl COPY mit der Option ALL, um alle Datensätze in einer Ergebnisliste zu kopieren, mit Ausnahme der Datensätze, die gelöscht wurden.

**Anmerkung:** Gelöschte Datensätze werden ebenfalls entfernt und die verbleibenden Datensätze werden neu nummeriert, wenn Sie einen Befehl des Typs REFRESH absetzen.

## Zugehörige Befehle

COPY, DISCARD, GET, GROUP, LOCATE, MARK, ORDER, PERFORM OBJECT, REFRESH, SPECIFY FILTER

## Optionen

### ALL

Löscht alle Ressourcentabellendatensätze in der Ergebnisliste.

### COUNT (Datenreferenz)

Benennt eine Variable, um die Anzahl der Ressourcentabellendatensätze in der Ergebnisliste zu empfangen, wenn der Löschprozess abgeschlossen ist.

### CURRENT

Es wird nur der aktuelle Ressourcentabellendatensatz in der Ergebnisliste gelöscht.

**Anmerkung:** Der Datensatzzeiger bleibt beim gelöschten Datensatz positioniert. Wenn Sie einen weiteren API-Befehl mit der Option CURRENT absetzen, bevor Sie den Zeiger neu positionieren, empfangen Sie einen RESPONSE-Wert von NODATA.

### FILTER (CPSM-Token)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option FILTER löscht nur die Ressourcentabellendatensätze, die den angegebenen Filterkriterien entsprechen.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

### MARKED

Löscht nur die Ressourcentabellendatensätze, die in der Ergebnisliste markiert sind. Ressourcentabellendatensätze können mit den Befehlen MARK und UNMARK markiert werden.

### NOTFILTER (CPSM-Token)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option NOTFILTER löscht nur die Ressourcentabellendatensätze, die den angegebenen Filterkriterien nicht entsprechen.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

### NOTMARKED

Löscht nur die Ressourcentabellendatensätze, die in der Ergebnisliste nicht markiert sind. Ressourcentabellendatensätze können mit den Befehlen MARK und UNMARK markiert werden.

### REASON (Datenreferenz)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

### RESPONSE (Datenreferenz)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

### RESULT (CPSM-Token)

Gibt die API-Ergebnisliste an, die von dieser Operation verarbeitet werden soll. Die Ergebnisliste kann durch einen der folgenden Befehle erzeugt werden:

- COPY

- GET
- GROUP
- PERFORM OBJECT.

#### **THREAD** (*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

### **Bedingungen**

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl DELETE zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

#### **NODATA**

Es wurden keine Datensätze gefunden, die mit den angegebenen Suchkriterien übereinstimmen.

**BUSY** Aus den folgenden Gründen kam es zu einer Bedingung des Typs 'Aktiv':

#### **RESULT**

Die in der Option RESULT angegebene Ergebnisliste wird von einem anderen Befehl verarbeitet.

#### **ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

#### **NOSERVICE**

Das Anwendungsstubprogramm konnte das API-Servicemodul nicht laden.

#### **NOSTORAGE**

Das Anwendungsstubprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

#### **SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

#### **SOLRESOURCE**

Eine erforderliche Ressource, deren Eigner ein lokaler Eigner ist (d. h. Eigner des Adressraums, in dem der Verarbeitungsthread ausgeführt wird), ist nicht verfügbar.

#### **FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

#### **ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

#### **EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

#### **INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:



- FILTER
- NOTFILTER
- RESULT
- THREAD.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

#### NOTAVAILABLE

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

##### APITASK

Die API-Steuersubtask ist nicht aktiv.

##### CPSMAPI

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

#### SERVERGONE

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

#### VERSIONINVL

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

##### NOTSUPPORTED

Die Version des Anwendungsstubprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

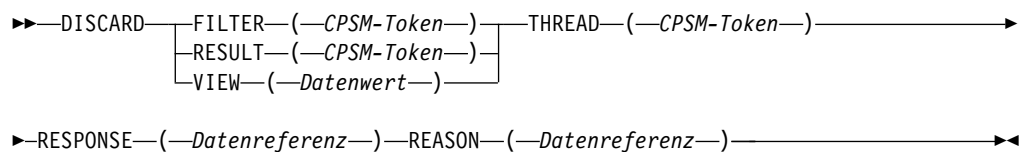
##### NOTVSNNCONN

Die Version des Anwendungsstubprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

---

## DISCARD

Eine Ergebnisliste, einen Filter oder eine Ansicht verwerfen.



### Beschreibung

Durch diesen Befehl wird eine Ergebnisliste, ein Filter oder eine Ansicht verworfen.

### Zugehörige Befehle

COPY, GET, GETDEF, GROUP, PERFORM OBJECT, SPECIFY FILTER, SPECIFY VIEW

### Optionen

#### **FILTER**(CPSM-Token)

Gibt den Filter an, der verworfen werden soll. Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

**REASON**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

**RESPONSE**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**RESULT**(*CPSM-Token*)

Gibt die API-Ergebnisliste an, die verworfen werden soll. Die Ergebnisliste kann durch einen der folgenden Befehle erzeugt werden:

- COPY
- GET
- GETDEF
- GROUP
- PERFORM OBJECT.

**Anmerkung:** Wenn Sie eine Ergebnisliste verwerfen, die durch den Befehl GROUP zusammengefasst wurde, werden alle zusammenfassenden Ergebnislisten ebenfalls verworfen.

**THREAD**(*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

**VIEW**(*Datenwert*)

Gibt die Ansicht an, die verworfen werden soll. Dieser Wert muss der Name einer Ansicht sein, der aus 1-8 Zeichen besteht und in einem Befehl des Typs SPECIFY VIEW definiert wurde.

## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl DISCARD zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

**BUSY** Aus den folgenden Gründen kam es zu einer Bedingung des Typs 'Aktiv':

**RESULT**

Die in der Option RESULT angegebene Ergebnisliste wird von einem anderen Befehl verarbeitet.

**ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

**NOSERVICE**

Das Anwendungsstubprogramm konnte das API-Servicemodul nicht laden.

**NOSTORAGE**

Das Anwendungsstubprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**INUSE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'In Gebrauch':

**FILTER**

Der angegebene Filter wird derzeit verwendet und kann nicht verworfen werden.

**VIEW** Die angegebene Ansicht wird derzeit verwendet und kann nicht verworfen werden.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- FILTER
- RESULT
- THREAD
- VIEW.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

**NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

**APITASK**

Die API-Steuersubtask ist nicht aktiv.

**CPSMAPI**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

**SERVERGONE**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

**VERSIONINVL**

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

**NOTSUPPORTED**

Die Version des Anwendungssubprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

## NOTVSNCONN

Die Version des Anwendungsstubprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

---

## DISCONNECT

Trennen einer Verbindung zwischen einem API-Verarbeitungsthread und CICSplex SM.

►—DISCONNECT—THREAD—(—*CPSM-Token*—)—RESPONSE—(—*Datenreferenz*—)—►  
►—REASON—(—*Datenreferenz*—)—►◀

### Beschreibung

Alle Ressourcen, die dem Thread zugeordnet sind, werden freigegeben, einschließlich Ergebnislisten, Filter, Ansichten, Diagnosedaten und ausstehenden asynchronen Anforderungen.

### Zugehörige Befehle

CONNECT, TERMINATE

### Optionen

#### REASON(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

#### RESPONSE(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

#### THREAD(*CPSM-Token*)

Gibt den API-Thread an, der getrennt werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

### Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl DISCONNECT zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

#### ENVIRONERROR

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

##### NOSERVICE

Das Anwendungsstubprogramm konnte das API-Servicemodul nicht laden.

##### NOSTORAGE

Das Anwendungsstubprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- THREAD.

Überprüfen Sie die Befehlsbeschreibung auf eine gültige Parametersyntax hin.

**NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

**APITASK**

Die API-Steuersubtask ist nicht aktiv.

**CPSMAPI**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

**SERVERGONE**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

**VERSIONINVL**

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

**NOTSUPPORTED**

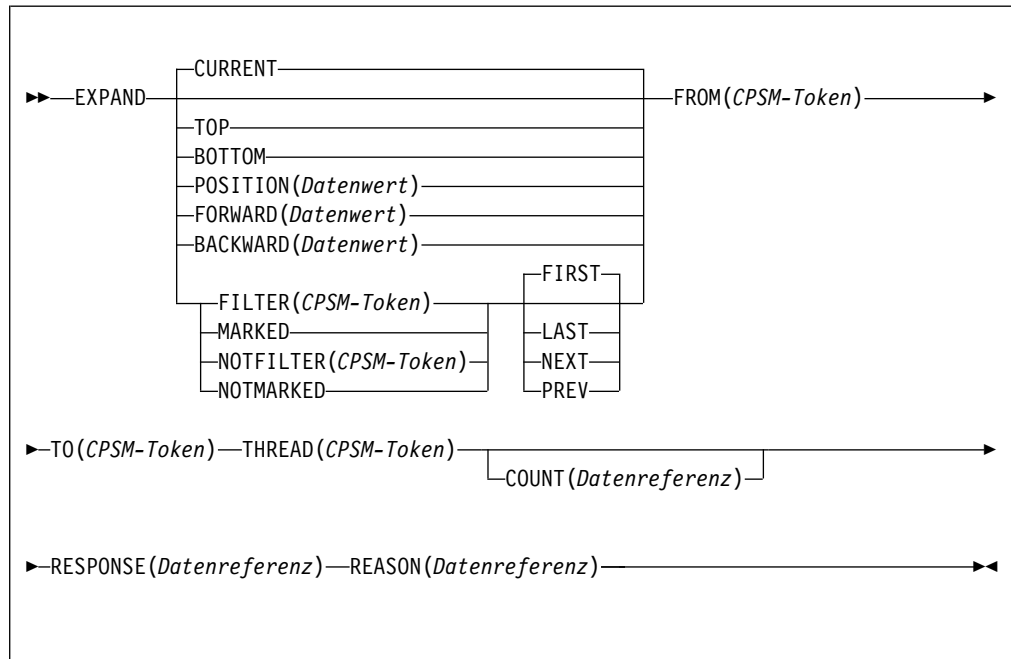
Die Version des Anwendungssubprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

**NOTVSNCONN**

Die Version des Anwendungssubprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

## EXPAND

Gibt eine Ergebnisliste zurück, die alle in einem Zusammenfassungsdatensatz zusammengefassten Datensätze enthält.



### Beschreibung

Durch diesen Befehl wird die Erweiterung von Zusammenfassungsergebnislisten unterstützt. Der Befehl akzeptiert ein Token aus einer zusammenfassenden Ergebnisliste, die durch den Befehl GROUP erzeugt wurde, sowie einen ausgewählten Datensatz, der durch die Position des Datensatzzeigers in der Ergebnisliste angegeben wird, die erweitert werden soll. Die Position des Datensatzzeigers ist von den Optionen abhängig, die Sie in dem Befehl angeben. Es wird eine neue Ergebnisliste mit allen Datensätzen erstellt, die in einem Zusammenfassungsdatensatz zusammengefasst sind.

### Zugehörige Befehle

FETCH, GET, GROUP, LOCATE, MARK, ORDER, QUERY, REFRESH, SPECIFY FILTER, UNMARK

### Optionen

#### BACKWARD(Datenwert)

Erweitert den Datensatz an der Position, an die er durch Rückwärtsverschieben von der aktuellen Zeigerposition für eine Anzahl an Datensätzen (Datenwert) gelangt ist. Wenn der Wert (Datenwert) größer als die verbleibende Anzahl von Datensätzen ist, wird der erste Datensatz in der Zusammenfassungsergebnisliste erweitert.

#### BOTTOM

Erweitert den letzten Datensatz in der Zusammenfassungsergebnisliste.

**COUNT** (*Datenreferenz*)

Die Anzahl der Ressourcentabellendatensätze in der TO-Ergebnisliste, nachdem diese Operation abgeschlossen ist. Dieser Parameter ist eingabegeschützt.

**CURRENT**

Erweitert den aktuellen Datensatz in der FROM-Ergebnisliste.

**FILTER** (*CPSM-Token*)

Gibt den Filter an, der für diese Operation verwendet werden soll, und führt eine EXPAND-Operation für den Datensatz oder die Datensätze aus, der/die mit den Filterkriterien übereinstimmt/übereinstimmen. Wird zusammen mit den Optionen FIRST, LAST, NEXT und PREV verwendet.

**FIRST**

Erweitert entweder den ersten markierten Datensatz in der Ergebnisliste oder den ersten Datensatz, der mit den Filterkriterien übereinstimmt. Wenn kein Datensatz gefunden wird, wird ein Code des Typs NODATA zurückgegeben.

**FORWARD** (*Datenwert*)

Erweitert den Datensatz an der Position, an die er durch Vorwärtsverschieben von der aktuellen Zeigerposition für eine Anzahl an Datensätzen (*Datenwert*) gelangt ist. Wenn der Wert (*Datenwert*) größer als die verbleibende Anzahl von Datensätzen in der Zusammenfassungsergebnisliste ist, wird der letzte Datensatz erweitert.

**FROM** (*CPSM-Token*)

Die Zusammenfassungsergebnisliste, für die der Befehl EXPAND ausgeführt werden soll. Wenn keine übereinstimmende Ergebnisliste gefunden werden kann, wird ein Rückgabecode des Typs INVALIDPARM mit einem Ursachen-code des Typs FROM ausgegeben.

**LAST**

Erweitert entweder den letzten markierten Datensatz in der Ergebnisliste oder den letzten Datensatz, der mit den Filterkriterien übereinstimmt. Wenn kein Datensatz gefunden wird, wird ein Code des Typs NODATA zurückgegeben.

**MARKED**

Erweitert mindestens einen Datensatz, der mit dem Befehl MARK ausgewählt wurde. Wird zusammen mit den Optionen FIRST, LAST, NEXT und PREV verwendet.

Ressourcentabellendatensätze können mit den Befehlen MARK und UNMARK markiert werden.

**NEXT**

NEXT beginnt mit dem aktuell ausgewählten Datensatz und setzt seine Bewegung durch die Ergebnisliste in Vorwärtsrichtung fort und erweitert dabei entweder den nächsten markierten Datensatz oder den nächsten Datensatz, der mit den Filterkriterien übereinstimmt. Wenn kein Datensatz gefunden wird, wird ein Code des Typs NODATA zurückgegeben.

**NOTFILTER** (*CPSM-Token*)

Gibt den Filter an, der für diese Operation verwendet werden soll, und führt eine EXPAND-Operation für den Datensatz oder die Datensätze aus, der/die mit den Filterkriterien nicht übereinstimmt/übereinstimmen. Wird zusammen mit den Optionen FIRST, LAST, NEXT und PREV verwendet.

**NOTMARKED**

Erweitert mindestens einen Datensatz, der nicht mit dem Befehl MARK ausgewählt wurde. Wird zusammen mit den Optionen FIRST, LAST, NEXT und PREV verwendet.

Ressourcentabellendatensätze können mit den Befehlen MARK und UNMARK markiert werden.

**POSITION**(*Datenwert*)

Erweitert den Datensatz an einer Position in der Zusammenfassungsergebnisliste, die durch den bereitgestellten Wert angegeben wird.

**PREVIOUS**

PREVIOUS beginnt mit dem aktuell ausgewählten Datensatz und setzt seine Bewegung durch die Ergebnisliste in Rückwärtsrichtung fort und erweitert dabei entweder den nächsten markierten Datensatz in der Ergebnisliste oder den nächsten Datensatz, der mit den Filterkriterien übereinstimmt. Wenn kein Datensatz gefunden wird, wird ein Code des Typs NODATA zurückgegeben.

**REASON**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

**RESPONSE**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**THREAD**(*CPSM-Token*)

Der API-Thread, der für die EXPAND-Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

**TO**(*CPSM-Token*)

Gibt die Zusammenfassungsergebnisliste an, die die erweiterten Datensätze enthalten soll, für die der Befehl EXPAND ausgeführt wird. Wenn diese Ergebnisliste bereits vorhanden ist, werden alle vorhandenen Ressourcentabellendatensätze, die sich auf diese Ergebnisliste beziehen, durch die Ressourcentabellendatensätze ersetzt, die durch diesen Befehl des Typs EXPAND erstellt werden.

**TOP**

Erweitert den ersten Datensatz in der Zusammenfassungsergebnisliste.

## Bedingungen

Die folgenden RESPONSE-Werte können mit dem Befehl EXPAND zurückgegeben werden. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

**NODATA**

Es wurden keine Datensätze gefunden, die mit den angegebenen Kriterien übereinstimmen, und zwar aus einem der folgenden Gründe:

**BACKWARD**

Es sind keine weiteren Datensätze vorhanden, die die Suchkriterien in Rückwärtsrichtung erfüllen.

**FORWARD**

Es sind keine weiteren Datensätze vorhanden, die die Suchkriterien in Vorwärtsrichtung erfüllen.

**BUSY** Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Aktiv':



**FROM**

Die in der Option FROM angegebene Ergebnisliste wird von einem anderen Befehl verarbeitet.

**TO**

Die in der Option TO angegebene Ergebnisliste wird von einem anderen Befehl verarbeitet.

**ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

**NOSERVICE**

Das Anwendungsstubsprogramm konnte das API-Service-Modul nicht laden.

**NOSTORAGE**

Das Anwendungsstubsprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- BACKWARD
- FORWARD
- POSITION
- FILTER
- NOTFILTER
- FROM
- TO
- THREAD
- COUNT

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

**NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

**APITASK**

Die API-Steuersubtask ist nicht aktiv.

**CPSMAPI**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

**SERVERGONE**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

Die Version des Anwendungstubsprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

Die Version des Anwendungsstubsprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

Abrufen von Diagnosedaten.



- Die Diagnosedaten werden als FEEDBACK-Ressourcentabellendatensätze zurückgegeben.
- Wenn durch den vorherigen Befehl eine Ergebnisliste verarbeitet wurde und ein anderer RESPONSE-Wert als OK zurückgegeben wurde, wird an das Ende jedes Ressourcentabellendatensatzes in der Ergebnisliste ein FEEDBACK-Ressourcentabellendatensatz angehängt; bei dieser Ergebnisliste handelt es sich um eine Ergebnisliste, der ein Fehler zugeordnet war, der zum Senden des Nicht-OK-RESPONSE-Werts geführt hat. Die Diagnosedaten stehen dem Befehl FEEDBACK zur Verfügung, bis ein anderer Befehl die gleiche Ergebnisliste verarbeitet. An diesem Punkt werden die Daten durch FEEDBACK-Datensätze für den nachfolgenden Befehl ersetzt.

- Wenn der vorherige Befehl eine Ergebnisliste nicht verarbeitet hat, werden die FEEDBACK-Ressourcentabellendatensätze in einem separaten Feedback-Bereich zurückgegeben. Die Datensätze in diesem Feedback-Bereich werden für jeden Befehl, der nicht auf die Ergebnisliste ausgerichtet ist, gelöscht und aktualisiert. Bei Befehlen, bei denen die Diagnosedaten im Feedback-Bereich und nicht in einer Ergebnisliste platziert werden, können mit FEEDBACK nur Daten für den zuletzt abgesetzten Befehl abgerufen werden.
- Nachdem Sie den Befehl FEEDBACK abgesetzt haben, um Diagnosedaten für einen Befehl abzurufen, wird der Feedback-Datensatz oder -Bereich gelöscht. Es ist nicht möglich, mehr als einmal die gleichen FEEDBACK-Ressourcentabellendatensätze anzufordern.

- Wenn ein Befehl asynchron verarbeitet wird (d. h. durch Angabe der Option NOWAIT), werden die Diagnosedaten für diesen Befehl in der ASYNCREQ-Benachrichtigungsressourcentabelle zurückgegeben. Es werden keine FEEDBACK-Ressourcentabellendatensätze für eine asynchrone Anforderung erstellt.
- Für die folgenden Befehle sind keine Diagnosedaten verfügbar:
  - DISCONNECT
  - FEEDBACK
  - TERMINATE
- Die Befehle TBUILD und TPARSE, die für die Verwendung in REXX-Programmen bereitgestellt werden, stellen keine nützlichen FEEDBACK-Informationen zur Verfügung.

Eine vollständige Beschreibung finden Sie in Ressourcentabelle FEEDBACK in den Referenzinformationen.

## Optionen

### **COUNT** (*Datenreferenz*)

Gibt die Anzahl der Feedback-Datensätze an, die aus der Ergebnisliste abgerufen werden sollen, die in der Option RESULT angegeben ist. Wenn Sie die Option COUNT nicht angeben, wird nur ein einziger Feedback-Datensatz abgerufen.

Wenn Sie mehrere Feedback-Datensätze abrufen, werden diese nacheinander in den INTO-Puffer gestellt. Der INTO-Puffer muss lang genug sein, um alle abgerufenen Feedback-Datensätze aufnehmen zu können.

Der Wert, den CICSplex SM in diesem Feld zurückgibt, ist von dem RESPONSE-Wert für den Befehl FEEDBACK abhängig:

**OK** Die tatsächliche Anzahl der Datensätze, die im INTO-Puffer zurückgegeben werden.

### **WARNING AREATOOSMALL**

Die Anzahl der Datensätze, die im INTO-Puffer zurückgegeben werden; dies ist nicht die Gesamtzahl der angeforderten Datensätze.

### **INVALIDPARM LENGTH**

Das Feld ist nicht festgelegt, da der INTO-Puffer nicht lang genug war, um auch nur einen Ressourcentabellendatensatz aufnehmen zu können.

### **FIRST**

Ruft den ersten Feedback-Datensatz aus der Ergebnisliste ab, die in der Option RESULT angegeben ist.

Wenn Sie die Option COUNT angeben, wird mit FIRST die angegebene Anzahl an Datensätzen abgerufen; dabei wird mit dem ersten Datensatz in der Ergebnisliste begonnen.

### **INTO** (*Datenbereich*)

Gibt einen Puffer (bzw. eine Stammvariable in REXX) an, um die Feedback-Daten zu empfangen. Dieser Puffer muss lang genug sein, um die gesamten abgerufenen Feedback-Daten aufnehmen zu können.

### **LENGTH** (*Datenreferenz*)

Ein Vollwortwert, der die Länge des INTO-Puffers angibt.

Der Wert, den CICSplex SM in diesem Feld zurückgibt, ist von dem RESPONSE-Wert für den Befehl FEEDBACK abhängig:

**OK** Die tatsächliche Länge der Daten, die im INTO-Puffer zurückgegeben werden.

**WARNING AREATOOSMALL**

Die Pufferlänge, die erforderlich wäre, um alle angeforderten Datensätze aufzunehmen.

**INVALIDPARM LENGTH**

Das Feld ist nicht festgelegt, da der INTO-Puffer nicht lang genug war, um auch nur einen Ressourcentabellensatz aufnehmen zu können.

**NEXT**

Ruft den nächsten verfügbaren Feedback-Datensatz aus der Ergebnisliste ab, die in der Option RESULT angegeben ist.

Wenn Sie die Option COUNT angeben, wird mit NEXT die angegebene Anzahl an Datensätzen abgerufen; dabei wird mit dem nächsten Datensatz in der Ergebnisliste begonnen.

**REASON**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

**RESPONSE**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**RESULT**(*CPSM-Token*)

Gibt eine API-Ergebnisliste an, die von dieser Operation verarbeitet werden soll. Die Ergebnisliste kann durch einen der folgenden Befehle erzeugt werden:

- COPY
- GET
- GROUP
- PERFORM OBJECT.

Verwenden Sie die Option RESULT, um Feedback-Daten zu einem zuvor ausgegebenen Befehl abzurufen, mit dem eine Ergebnisliste verarbeitet wurde. Verwenden Sie FEEDBACK ohne die Option RESULT, um Daten zu dem zuletzt ausgegebenen Befehl abzurufen, mit dem keine Ergebnisliste verarbeitet wurde.

**THREAD**(*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl FEEDBACK zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

**NODATA**

Es wurden keine Datensätze gefunden, die mit den angegebenen Suchkriterien übereinstimmen, oder ein Befehl, der eine Ergebnisliste verarbeitet hat, hat den RESPONSE-Wert OK zurückgegeben.

**WARNING**

Der Befehl hat die Verarbeitung aus folgendem Grund mit einer Warnung beendet:

**AREATOOSMALL**

Der INTO-Puffer reicht nicht aus, um die Anzahl der angeforderten und verfügbaren Datensätze aufzunehmen.

**BUSY** Aus den folgenden Gründen kam es zu einer Bedingung des Typs 'Aktiv':

**RESULT**

Die in der Option RESULT angegebene Ergebnisliste wird von einem anderen Befehl verarbeitet.

**ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

**NOSERVICE**

Das Anwendungsstubsprogramm konnte das API-Service-Modul nicht laden.

**NOSTORAGE**

Das Anwendungsstubsprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- COUNT
- INTO
- LENGTH
- RESULT
- THREAD.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

**NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

**APITASK**

Die API-Steuersubtask ist nicht aktiv.

**CPSMAPI**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

**SERVERGONE**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

## VERSIONINVL

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

### NOTSUPPORTED

Die Version des Anwendungsstubprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

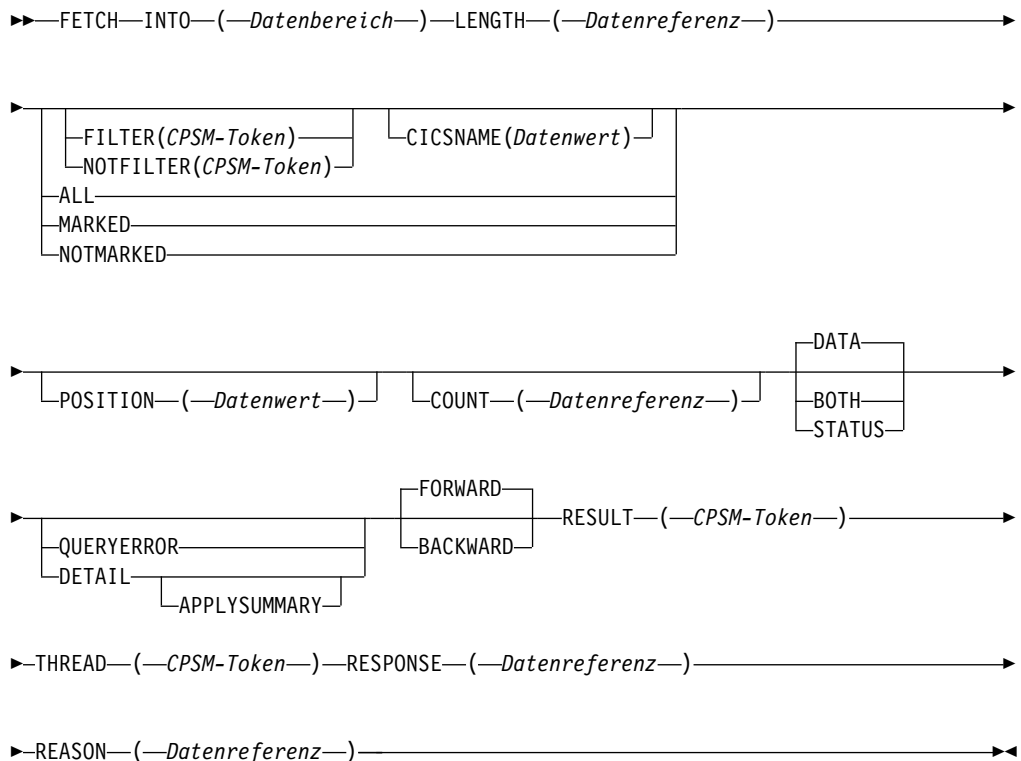
### NOTVSNNCONN

Die Version des Anwendungsstubprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

---

## FETCH

Abrufen von Daten und Statusinformationen für Ressourcentabellendatensätze.



## Beschreibung

Mit diesem Befehl werden Daten und Statusinformationen für einen oder mehrere Ressourcentabellendatensätze in einer Ergebnisliste abgerufen.

- Nach einem Befehl des Typs **FETCH** wird der Datensatzzeiger in der Regel auf den nächsten Datensatz in der Ergebnisliste positioniert (d. h. auf den Satz nach dem letzten Datensatz, der in der jeweiligen Richtung, in der sich der Zeiger bewegte, abgerufen wurde; dies kann vorwärts oder rückwärts sein). Die folgenden API-Befehle jedoch arbeiten immer mit dem letzten abgerufenen Datensatz (d. h. der Datensatzzeiger wird nicht erweitert):
  - COPY
  - DELETE
  - MARK

- UNMARK
- PERFORM SET CURRENT
- REFRESH CURRENT
- SET CURRENT
- Wenn keine Datensätze abgerufen wurden (da keine Datensätze mit den angegebenen Kriterien übereinstimmten), wird der Zeiger an den Anfang oder das Ende der Ergebnisliste positioniert, in Abhängigkeit davon, in welche Richtung er sich bewegt hat.

## Zugehörige Befehle

COPY, GET, GETDEF, GROUP, LOCATE, MARK, ORDER, PERFORM OBJECT, QUERY, SPECIFY FILTER

## Optionen

### ALL

Ruft alle Ressourcentabellendatensätze in der Ergebnisliste ab. Wenn Sie ALL angeben, werden die Optionen POSITION und COUNT ignoriert.

### APPLYSUMMARY

Es werden beliebige, einige oder alle der folgenden Optionen auf die Zusammenfassungsdatensätze angewendet und die den Zusammenfassungsdatensätzen zugeordneten Detaildatensätze werden abgerufen.

- MARKED
- NOTMARKED
- FILTER
- NOTFILTER

Die Option APPLYSUMMARY ist nur gültig, wenn auch die Option DETAIL angegeben wird.

Wenn die Option DETAIL ohne die Option APPLYSUMMARY angegeben wird, wird das Ergebnis wie unter der Option DETAIL beschrieben sein.

Wenn weder die Option DETAIL noch die Option APPLYSUMMARY angegeben ist, sondern eine beliebige Kombination einiger oder aller der Datenauswahloptionen MARKED, NOTMARKED, FILTER und NOTFILTER für eine Zusammenfassungsergebnisliste ausgegeben wird, werden diese Datenauswahloptionen auf die Zusammenfassungsergebnisliste angewendet und die ausgewählten Zusammenfassungsdatensätze werden abgerufen.

### BACKWARD

Beginnt den Abfrageprozess mit dem letzten abgerufenen Datensatz und setzt die Verarbeitung in Rückwärtsrichtung durch die angegebene Ergebnisliste fort.

### BOTH

Ruft sowohl die Daten der Ressourcentabelle als auch die OBJSTAT-Statusinformationen zu der letzten Aktion ab, die für die Ressourcentabelle ausgeführt wird. Jeder Datensatz enthält Informationen zu OBJSTAT sowie Ressourcentabellendaten.

### CICSNAME(*Datenwert*)

Gibt einen aus 1 bis 8 Zeichen bestehenden spezifischen oder generischen Namen für das CICS-System an.

Mit der Option CICSNAME wird angegeben, dass nur die Ressourcentabellendatensätze für die Abfrage infrage kommen sollen, die aus CICS-Systemen

stammen, die mit dem angegebenen Namensmuster übereinstimmen. Wenn CICSNAME zusammen mit FILTER oder NOTFILTER angegeben wird, werden nur Datensätze berücksichtigt, die die Anforderungen des Typs FILTER oder NOTFILTER erfüllen und auch mit dem CICSNAME-Muster übereinstimmen. Die Anzahl der abgerufenen Datensätze wird durch die Option COUNT bestimmt.

Bei der Angabe von CICSNAME darf es sich bei der in der Option RESULT angegebenen Ergebnisliste nicht um eine zusammenfassende Ergebnisliste handeln; sie muss Ressourcentabellendatensätze enthalten, die ein Attribut des Typs EYU\_CICSNAME aufweisen. Wenn die durch RESULT angegebene Ergebnisliste zusammenfassende Datensätze oder Ressourcentabellendatensätze enthält, für die kein Attribut des Typs EYU\_CICSNAME vorhanden ist, erhalten Sie für die Option CICSNAME eine Antwort des Typs INVALIDPARM.

#### **COUNT** (*Datenreferenz*)

Gibt die Anzahl der zu verarbeitenden Ressourcentabellendatensätze an.

Die Option COUNT gilt für die Ergebnisliste, die in der Option RESULT angegeben ist. Wenn Sie auch die Option DETAIL angeben, stellt COUNT die Anzahl der Zusammenfassungskdatensätze in der zusammenfassenden Ergebnisliste in RESULT zur Verfügung, für die Quelldatensätze zurückgegeben werden. Die OBJSTAT-Tabelle für jeden Zusammenfassungskdatensatz enthält die Anzahl der Quelldatensätze, die für diesen Datensatz zurückgegeben werden, wenn die Option DETAIL angegeben wird.

Wenn Sie die Option COUNT nicht angeben, ist der Standardwert 1.

Ist die Option COUNT angegeben, enthält COUNT die Anzahl der verarbeiteten Datensätze. In den meisten Fällen ist dies auch die Anzahl der zurückgegebenen Datensätze. Wenn Sie jedoch auch die Option DETAIL angeben, werden alle Quelldatensätze abgerufen, die der angeforderten Anzahl an Zusammenfassungskdatensätzen zugeordnet sind. Diese Zahl ist in der Regel größer als die in der Option COUNT angegebene Zahl.

Wenn Sie mehrere Datensätze abrufen, werden diese nacheinander in den INTO-Puffer gestellt. Der INTO-Puffer muss lang genug sein, um alle abgerufenen Datensätze aufnehmen zu können.

Der Wert, den CICSplex SM in diesem Feld zurückgibt, ist von dem RESPONSE-Wert für den Befehl FETCH wie folgt abhängig:

**OK** Die tatsächliche Anzahl der Datensätze, die im INTO-Puffer zurückgegeben werden.

#### **WARNING AREATOOSMALL**

Die Anzahl der Datensätze, die im INTO-Puffer zurückgegeben werden; dies ist nicht die Gesamtzahl der angeforderten Datensätze.

#### **INVALIDPARM LENGTH**

Das Feld ist nicht festgelegt, da der INTO-Puffer nicht lang genug war, um auch nur einen Ressourcentabellensatz aufnehmen zu können.

#### **DATA**

Ruft nur die angegebenen Daten der Ressourcentabelle ab. Die Datensätze enthalten keine OBJSTAT-Statusinformationen zu der letzten Aktion, die für die Ressourcentabelle ausgeführt wurde.

**Anmerkung:** Die OBJSTAT-Informationen beinhalten ein Zusammenfassungszählerfeld, das festgelegt wird, wenn Ressourcentabellendatensätze mit dem Befehl GROUP zusammengefasst werden. Wenn Sie die Ressourcentabellenda-



tensätze gruppieren (GROUP) und wissen möchten, wie viele Datensätze zur Bildung eines Zusammenfassungsdatensatzes gruppiert werden, müssen Sie BOTH angeben, um sowohl die Dateninformationen als auch die OBJSTAT-Informationen zu erhalten, wenn die Datensätze abgerufen werden.

#### **DETAIL**

Ruft die Quelldatensätze ab, die bestimmten Zusammenfassungsressourcentabellendatensätzen zugeordnet sind.

Bei der Angabe von DETAIL muss es sich bei der in der Option RESULT angegebenen Ergebnisliste um eine zusammenfassende Ergebnisliste handeln. Mit DETAIL wird der Zusammenfassungsdatensatz erweitert, und zwar durch Abrufen der ihm zugeordneten Ressourcentabellendatensätze aus der Quellenergebnisliste. Wenn Sie bei der Verarbeitung einer zusammenfassenden Ergebnisliste die Option DETAIL nicht angeben, werden die Zusammenfassungsdatensätze selbst abgerufen. Wenn es sich bei der Ergebnisliste nicht um eine zusammenfassende Ergebnisliste handelt, hat diese Option keine Bedeutung und wird ignoriert.

Sie können die Optionen FORWARD oder BACKWARD zusammen mit DETAIL verwenden, um den Zusammenfassungsdatensatz auszuwählen, der erweitert werden soll. Mit den Optionen FORWARD und BACKWARD wird auch die Richtung gesteuert, in der Datensätze aus der Quellenergebnisliste abgerufen werden.

Standardmäßig werden alle Quelldatensätze abgerufen, die dem Zusammenfassungsdatensatz oder den Zusammenfassungsdatensätzen zugeordnet sind. Sie können jedoch die Option FILTER oder NOTFILTER verwenden, um die aus der Quellenergebnisliste abgerufenen Datensätze zu begrenzen. Sie können auch die Option MARKED oder NOTMARKED verwenden, um nur die Datensätze abzurufen, die dem Zusammenfassungsdatensatz zugeordnet sind und die in der Quellenergebnisliste markiert (oder nicht markiert) sind.

Sie können den Datensatzzeiger nicht explizit in der Quellenergebnisliste positionieren. Bei der Angabe von DETAIL bezieht sich die Option POSITION auf den Datensatz in der Zusammenfassungsergebnisliste. Wenn die Option APPLYSUMMARY angegeben ist, werden die Optionen FILTER, NOTFILTER, MARKED und NOTMARKED auf Datensätze in der Zusammenfassungsergebnisliste und nicht auf die Quelldatensätze angewendet.

Weitere Informationen zur Verarbeitung von zusammenfassenden Ergebnislisten finden Sie in *Developing CICSplex SM applications*. Eine Beschreibung des Befehls GROUP, durch den zusammenfassende Ergebnislisten erstellt werden, finden Sie in „GROUP“ auf Seite 73.

#### **FILTER** (*CPSM-Token*)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option FILTER gibt an, dass nur die Ressourcentabellendatensätze, die den angegebenen Filterkriterien entsprechen, beim Abrufen berücksichtigt werden sollen. Die Anzahl der abgerufenen Datensätze wird durch die Option COUNT bestimmt.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

#### **FORWARD**

Beginnt den Abfrageprozess mit dem nächsten Datensatz (d. h. mit dem Datensatz, der dem letzten abgerufenen Datensatz folgt) und setzt die Verarbeitung in Vorwärtsrichtung durch die angegebene Ergebnisliste fort.

**INTO**(*Datenbereich*)

Gibt einen Puffer (bzw. eine Stammvariable in REXX) an, um die Ressourcentabellendatensätze zu empfangen. Dieser Puffer muss lang genug sein, um die gesamten abgerufenen Datensätze aufnehmen zu können.

**LENGTH**(*Datenreferenz*)

Ein Vollwortwert, der die Länge des INTO-Puffers angibt.

Der Wert, den CICSplex SM in diesem Feld zurückgibt, ist von dem RESPONSE-Wert für den Befehl FETCH abhängig:

**OK** Die tatsächliche Länge der Daten, die im INTO-Puffer zurückgegeben werden.

**NODATA**

Die Länge wird auf null gesetzt.

**WARNING AREATOOSMALL**

Die Pufferlänge, die erforderlich wäre, um alle angeforderten Datensätze aufzunehmen.

**INVALIDPARM LENGTH**

Das Feld ist nicht festgelegt, da der INTO-Puffer nicht lang genug war, um auch nur einen Ressourcentabellensatz aufnehmen zu können.

**MARKED**

Gibt an, dass nur die Ressourcentabellendatensätze, die in der Ergebnisliste markiert sind, beim Abrufen berücksichtigt werden sollen. Die Anzahl der abgerufenen Datensätze wird durch die Option COUNT bestimmt.

Ressourcentabellendatensätze können mit den Befehlen MARK und UNMARK markiert werden.

**NOTFILTER**(*CPSM-Token*)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option NOTFILTER gibt an, dass nur die Ressourcentabellendatensätze, die den angegebenen Filterkriterien nicht entsprechen, beim Abrufen berücksichtigt werden sollen. Die Anzahl der abgerufenen Datensätze wird durch die Option COUNT bestimmt.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

**NOTMARKED**

Gibt an, dass nur die Ressourcentabellendatensätze, die in der Ergebnisliste nicht markiert sind, beim Abrufen berücksichtigt werden sollen. Die Anzahl der abgerufenen Datensätze wird durch die Option COUNT bestimmt.

Ressourcentabellendatensätze können mit den Befehlen MARK und UNMARK markiert werden.

**POSITION**(*Datenwert*)

Beginnt den Abrufprozess mit dem n-ten Ressourcentabellendatensatz in der Ergebnisliste.

Dieser Wert muss eine Zahl sein, die die relative Position des Datensatzes in der Ergebnisliste angibt. Der erste Datensatz in einer Ergebnisliste wird durch die Zahl 1 angegeben.

Wenn Sie beispielsweise den Abrufprozess mit dem fünften Ressourcentabellendatensatz in einer Ergebnisliste beginnen möchten, geben Sie POSITION(5) an.

**Anmerkung:** Wenn die Option POSITION zusammen mit der Option DETAIL verwendet wird, um Quelldatensätze für einen bestimmten zusammenfassenden Datensatz einer Ergebnisliste abzurufen, wird der Wert der Option COUNT auf eins (1) gezwungen. In diesem Fall ist der Wert, der durch die Option COUNT zurückgegeben wird, die Anzahl der Quelldatensätze, die in dem angegebenen Datensatz der Ergebnisliste zusammengefasst sind.

#### **QUERYERROR**

Gibt an, dass diese Anforderung die MASQRYER-Ressourcen zurückgeben soll, die durch den letzten Befehl des Typs GET, PERFORM oder SET generiert wurden, um mit der Ergebnisliste zu arbeiten.

**Anmerkung:** Die Datenauswahloptionen FILTER, NOFILTER, MARKED, NOT-MARKED und CICSNAME werden ignoriert, wenn Sie die Option QUERYERROR angeben.

#### **REASON**(Datenreferenz)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

#### **RESPONSE**(Datenreferenz)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

#### **RESULT**(CPSM-Token)

Gibt die API-Ergebnisliste an, die von dieser Operation verarbeitet werden soll. Die Ergebnisliste kann durch einen der folgenden Befehle erzeugt werden:

- COPY
- GET
- GETDEF
- GROUP
- PERFORM OBJECT.

#### **STATUS**

Ruft nur die OBJSTAT-Statusinformationen für die letzte Aktion ab, die für die Ressourcentabelle ausgeführt wurde. Die Datensätze enthalten keine Ressourcentabellendaten.

#### **THREAD**(CPSM-Token)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

## **Bedingungen**

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl FETCH zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

#### **NODATA**

Es wurden keine Datensätze gefunden, die mit den angegebenen Suchkriterien übereinstimmen, und zwar aus einem der folgenden Gründe:

#### **BACKWARD**

Es sind keine weiteren Datensätze vorhanden, die die Suchkriterien in Rückwärtsrichtung erfüllen.

**FORWARD**

Es sind keine weiteren Datensätze vorhanden, die die Suchkriterien in Vorwärtsrichtung erfüllen.

**WARNING**

Der Befehl hat die Verarbeitung aus folgendem Grund mit einer Warnung beendet:

**AREATOOSMALL**

Der INTO-Puffer reicht nicht mehr aus, um die Anzahl der angeforderten und verfügbaren Datensätze aufzunehmen.

**BUSY** Aus den folgenden Gründen kam es zu einer Bedingung des Typs 'Aktiv':

**RESULT**

Die in der Option RESULT angegebene Ergebnisliste wird von einem anderen Befehl verarbeitet.

**ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

**NOSERVICE**

Das Anwendungsstubprogramm konnte das API-Servicemodul nicht laden.

**NOSTORAGE**

Das Anwendungsstubprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

**SOLRESOURCE**

Eine erforderliche Ressource, deren Eigner ein lokaler Eigner ist (d. h. Eigner des Adressraums, in dem der Verarbeitungsthread ausgeführt wird), ist nicht verfügbar.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- COUNT
- FILTER
- INTO
- LENGTH
- NOTFILTER
- POSITION
- RESULT

- THREAD
- CICSNAME

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

#### NOTAVAILABLE

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

#### APITASK

Die API-Steuersubtask ist nicht aktiv.

#### CPSMAPI

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

#### SERVERGONE

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

#### VERSIONINVL

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

#### NOTSUPPORTED

Die Version des Anwendungstübprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

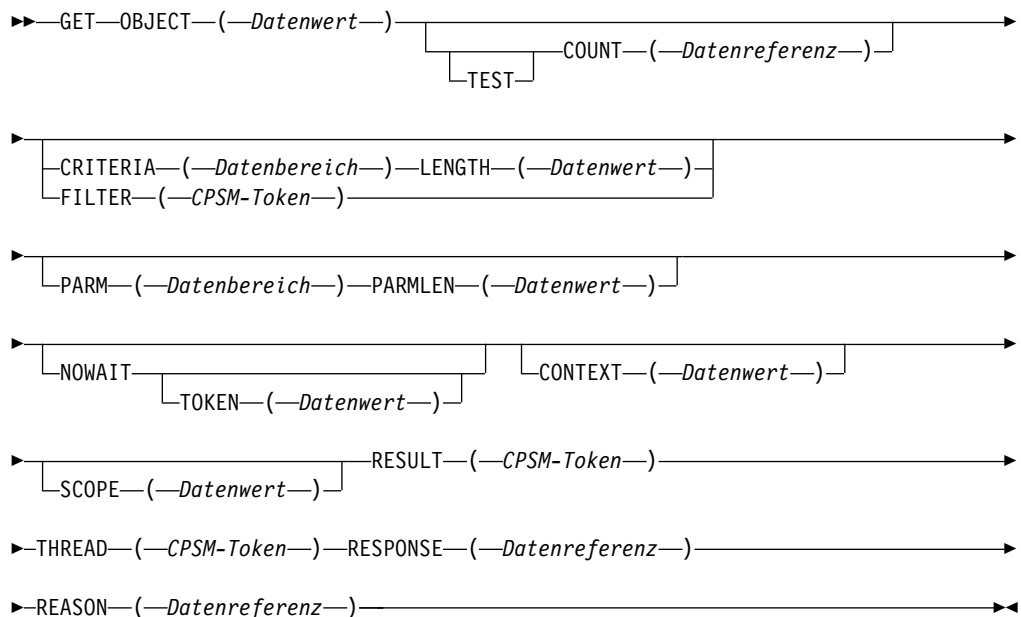
#### NOTVSNCONN

Die Version des Anwendungstübprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

---

## GET

Ermitteln und optionales Zurückgeben einer Ergebnisliste mit ausgewählten Ressourcentabellendatensätzen.



## Beschreibung

Mit diesem Befehl wird eine Ergebnisliste ermittelt, die ausgewählte Ressourcentabellendatensätze enthält. Der Befehl gibt die ermittelten Datensätze und/oder eine Anzahl der ermittelten Datensätze zurück.

- Bei der Ressourcentabelle kann es sich um eine Ressourcentabelle handeln, die eine CICS-Ressource, eine CICSplex SM- oder CICS-Definition oder ein CICSplex SM-Laufzeitobjekt darstellt.
- Nach einem Befehl des Typs GET wird der Datensatzzeiger an den Anfang der Ergebnisliste positioniert (d. h. auf den ersten Datensatz in der Ergebnisliste).
- Wenn der gültige Kontext und Bereich beim Absetzen des Befehls GET CICS-Systeme umfasst, die die angeforderte Ressourcentabelle nicht unterstützen, wird die Anforderung für diese CICS-Systeme ignoriert.
- In einigen CICS-Umgebungen stimmen die von CICSplex SM zurückgegebenen Ressourcentabellenattributwerte nicht mit den von CICS zurückgegebenen CVDA-Werten überein; diese sind in der folgenden Tabelle zu sehen:

| Ressourcentabelle | Attributwert      | CICS-Umgebung |
|-------------------|-------------------|---------------|
| LOCTRAN           | RESSEC(RESSECEXT) | CICS/MVS      |

Die von CICS zurückgegebenen Werte kollidieren mit den CVDA-Werten in anderen CICS-Umgebungen. Zur Wahrung der Eindeutigkeit der Attribute fügt CICSplex SM den Wert 9000 zu den von CICS zurückgegebenen Werten hinzu. Weitere Informationen zum Umsetzen von CVDA-Werten finden Sie in TRANS-LATE command.

## Zugehörige Befehle

DISCARD, FETCH, GETDEF, QUERY, RECEIVE, REFRESH, SPECIFY FILTER, SPECIFY VIEW

## Optionen

### CONTEXT (Datenwert)

Gibt den Kontext für diesen Befehl an. Bei dem Kontext muss es sich um einen aus 1 bis 8 Zeichen bestehenden Namen eines CMAS oder CICSplex handeln.

Wenn Sie die Option CONTEXT nicht angeben, wird der Standardkontext für den Thread angenommen.

### COUNT (Datenreferenz)

Benennt eine Variable, um die Anzahl der Ressourcentabellendatensätze in der Zielergebnisliste zu empfangen, wenn diese Operation abgeschlossen ist.

### CRITERIA (Datenbereich)

Gibt einen Puffer mit dem Filterausdruck an, der für diese Operation verwendet werden soll. Die Option CRITERIA ruft nur die Ressourcentabellendatensätze ab, die den angegebenen Filterkriterien entsprechen.

### FILTER (CPSM-Token)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option FILTER ruft nur die Ressourcentabellendatensätze ab, die den angegebenen Filterkriterien entsprechen.

Der Wert für CPSM-Token, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

### LENGTH (Datenwert)

Ein Vollwortwert, der die Länge des CRITERIA-Puffers angibt.

**Anmerkung:** Die von Ihnen angegebene Pufferlänge darf keine anderen Daten als einen Filterausdruck enthalten.

#### **NOWAIT**

Gibt die Steuerung an Ihr Programm zurück, sobald der Befehl GET akzeptiert wurde; dadurch kann der Befehl asynchron verarbeitet werden.

Wenn Sie die Option NOWAIT angeben, müssen Sie einen nachfolgenden Befehl des Typs RECEIVE verwenden, um zu testen, ob diese Anforderung ausgeführt wurde. Die Ergebnisse einer asynchronen Anforderung werden als Ressourcentabellendatensätze des Typs ASYNCREQ zurückgegeben.

**Anmerkung:** Wenn Sie die Option TOKEN angeben, wird die Option NOWAIT standardmäßig angenommen.

#### **OBJECT**(Datenwert)

Gibt die Ressourcentabelle an, für die Datensätze abzurufen sind. Bei diesem Wert muss es sich um einen aus 1 bis 8 Zeichen bestehenden Namen entweder einer gültigen Ressourcentabelle oder einer gültigen Ansicht handeln. Wenn Sie den Parameter TEST verwenden, müssen Sie einen CICS-Ressourcentabellennamen für **OBJECT** angeben. Die Angabe eines beliebigen anderen Werts, einschließlich eines Ansichtsnamens, führt zu einem Fehler.

#### **PARM**(Datenbereich)

Gibt einen Puffer mit dem Parameterausdruck an, der für die Vorauswahl von Ressourcentabellendatensätzen verwendet werden soll.

Details zur Verwendung eines Parameterausdrucks mit dem Befehl GET finden Sie in Using the PARM option. Eine Beschreibung der Parameter, die für eine angegebene Ressourcentabelle angegeben werden können, finden Sie in CICSplex SM resource tables.

#### **PARMLEN**(Datenwert)

Ein Vollwortwert, der die Länge des PARM-Puffers angibt.

#### **REASON**(Datenreferenz)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

#### **RESPONSE**(Datenreferenz)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

#### **RESULT**(CPSM-Token)

Gibt die API-Ergebnisliste an, die von dieser Operation verarbeitet werden soll. Die Ergebnisliste kann durch einen der folgenden Befehle erzeugt werden:

- COPY
- GET
- GROUP
- PERFORM OBJECT.

CICSplex SM ersetzt den Inhalt der vorhandenen Ergebnisliste durch die Ressourcentabellendatensätze, die durch diesen Befehl des Typs GET angefordert werden.

CICSplex SM erstellt eine neue Ergebnisliste und gibt das identifizierende Token im selben Feld zurück, wenn für dieses Feld Folgendes gilt:

- Es ist auf eine binäre Null gesetzt (in COBOL, C, PL/I oder Assembler)
- Es ist eine nicht initialisierte Variable (in REXX)

Wenn Sie den Parameter TEST verwenden, müssen Sie einen Nullwert für **RESULT** angeben. Die Angabe eines beliebigen anderen Werts führt zu einem Fehler.

#### **SCOPE** (*Datenwert*)

Gibt den Bereich für diesen Befehl an.

Wenn der aktuelle Kontext (definiert durch diesen Befehl oder einen früheren Befehl des Typs CONNECT oder QUALIFY) ein CICSplex ist und die Option OBJECT eine CICS-Ressource angibt, ist ein gültiger Bereich erforderlich. Für den Bereich kann Folgendes gelten:

- Es ist der aus 1 bis 8 Zeichen bestehende Name des CICSplex selbst
- Es ist ein CICS-System oder eine CICS-Systemgruppe innerhalb des CICSplex
- Es ist ein logischer Bereich wie er in einer CICSplex SM-Ressourcenbeschreibung (RESDESC) definiert wurde

Wenn OBJECT eine CICS-Definitionsressource angibt und die Option PARM den Parameter CSDGROUP beinhaltet, kann ein gültiger Bereich angegeben werden. Für den Bereich kann Folgendes gelten:

- Es ist ein CICS-System im CICSplex.

Wenn es sich bei dem aktuellen Kontext um einen CMAS handelt oder die Option OBJECT einen anderen Typ von Ressourcentabelle angibt, ist diese Option bedeutungslos und wird ignoriert.

Wenn Sie die Option SCOPE nicht angeben, wird der Standardbereich für den Thread angenommen. Wenn der aktuelle Kontext ein CICSplex ist und kein Standardbereich für den Befehl CONNECT oder QUALIFY festgelegt wurde, erhalten Sie eine Antwort des Typs INVALIDPARM für die Option SCOPE.

#### **TEST**

Gibt nur die Anzahl der Datensätze zurück, die mit der Anforderung übereinstimmen. Wenn Sie TEST angeben, gibt die Anforderung keine Datensätze zurück. Wenn Sie TEST angeben, müssen Sie einen CICS-Ressourcennamen für den Parameter OBJECT und einen Nullwert für den Parameter RESULT angeben.

#### **THREAD** (*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

#### **TOKEN** (*Datenwert*)

Definiert ein aus 1 bis 4 Zeichen bestehendes Token, das Sie für die Korrelation einer asynchronen Anforderung des Typs GET mit dem Ergebnis eines nachfolgenden Befehls des Typs RECEIVE auswählen. Dieses Token dient zur Verwendung durch Ihr Programm; CICSplex SM verwendet den Wert nicht. Das Token wird durch den Befehl RECEIVE zurückgegeben, wenn diese GET-Anforderung abgeschlossen ist.

## **Bedingungen**

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl GET zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.



**SCHEDULED**

Der Befehl wurde für die Verarbeitung geplant.

**NODATA**

Es wurden keine Datensätze gefunden, die mit den angegebenen Suchkriterien übereinstimmen.

**BUSY** Aus den folgenden Gründen kam es zu einer Bedingung des Typs 'Aktiv':

**RESULT**

Die in der Option RESULT angegebene Ergebnisliste wird von einem anderen Befehl verarbeitet.

**ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

**NOSERVICE**

Das Anwendungsstubsprogramm konnte das API-Servicemodul nicht laden.

**NOSTORAGE**

Das Anwendungsstubsprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**NOTPROCESSED**

Einer der MASs, an den die Anforderung übertragen wurde, konnte die Anforderung nicht verarbeiten.

**REQTIMEOUT**

Einer der CMASs oder MASs, an die die Anforderung übertragen wurde, antwortete nicht.

Wenn nur die Antwort einiger CMASs oder MASs ausblieb, ist es dennoch möglich, dass mit dem Befehl GET ein gültiges Ergebnis erzielt wird. Der Wert für COUNT kann größer als null sein, der Wert für RESULT kann ungleich null sein. Eine solche Ergebnisliste enthält Daten der CMASs und MASs, die nicht geantwortet haben.

**SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**INVALIDDATA**

Es wurden ungültige Daten gefunden. Der Parameter, der ungültige Daten enthält, wird als Ursachenwert zurückgegeben:

**CRITERIA**

Ein Attributwert, der im CRITERIA-Puffer aufgelistet ist, ist für das angegebene Attribut nicht gültig.

**INVALIDCMD**

Aus einem der folgenden Gründe ist der Befehl ungültig:

**FILTER**

Der für die Operation übergebene Filterausdruck ist zu umfangreich oder zu komplex.

**LENGTH**

Die Gesamtlänge aller Optionen im Befehl überschreitet den maximalen Grenzwert.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- CONTEXT
- CRITERIA
- FILTER
- LENGTH
- OBJECT
- PARM
- PARMLLEN
- RESULT
- SCOPE
- TOKEN
- THREAD.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

**NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

**APITASK**

Die API-Steuersubtask ist nicht aktiv.

**CMAS**

Ein CMAS (CICSplex SM-Adressraum), an den die Anforderung übertragen wurde, ist nicht verfügbar.

**CPSMAPI**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

**MAINTPOINT**

Der Wartungspunkt für den aktuellen Kontext ist nicht verfügbar.

**SCOPE**

Entweder ist keiner der MASs im angegebenen Bereich verfügbar, oder keiner von ihnen unterstützt die angeforderte Ressourcentabelle.

**WORKLOAD**

Die in der API-Anforderung angegebene Workload ist in dem lokalen CMAS nicht verfügbar.

**NOTFOUND**

Aus den folgenden Gründen kam es zu einer Bedingung des Typs 'Nicht gefunden':

**ATTRIBUTE**

Ein Attribut, das im CRITERIA-Puffer angegeben wurde, wurde für die angegebene Ressourcentabelle nicht gefunden.

**NOTPERMIT**

Aus den folgenden Gründen kam es zu einer Unzulässigkeitsbedingung:

**USRID**

Die Benutzer-ID, die dem Verarbeitungsthread zugeordnet ist, verfügt nicht über die erforderliche Sicherheitsberechtigung.

**SERVERGONE**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

**TABLEERROR**

Ein Ressourcentabellendatensatz ist aus dem folgenden Grund ungültig:

**DATAERROR**

Der Wert, der einem oder mehreren Ressourcentabellenattributen zugeordnet ist, ist ungültig. Verwenden Sie den Befehl **FEEDBACK**, um weitere Informationen zu diesem Fehler abzurufen.

**VERSIONINVL**

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

**NOTSUPPORTED**

Die Version des Anwendungsstubsprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

**NOTVSNNCONN**

Die Version des Anwendungsstubsprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl **CONNECT** verwendet wurde.

**WARNING**

Der Befehl hat die Verarbeitung aus folgendem Grund mit einer Warnung beendet:

**MAXRECORDS**

Die Anzahl der von einem MAS zu der Ergebnisliste hinzugefügten Datensätze hätte den Wert von **MAXHISTRECS** für diesen MAS überschritten. Datensätze innerhalb des Grenzwerts von **MAXHISTRECS** wurden zur Ergebnisliste hinzugefügt. Ändern Sie die Parameterwerte **FILTER** oder **PARM**, um die Anzahl der Datensätze, die der MAS der Ergebnisliste hinzufügen soll, zu erhöhen oder zu reduzieren.

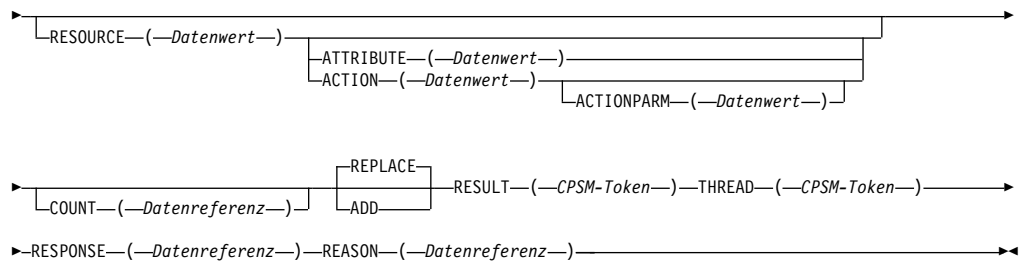
**Anmerkung:** Wenn ein Bereich angegeben wird, der mehr als einen MAS enthält, kann die Gesamtzahl der erfassten Datensätze den **MAXHISTRECS**-Wert für einen einzelnen MAS überschreiten.

---

## GETDEF

Zurückgeben einer Ergebnisliste mit ausgewählten beschreibenden Datensätzen für eine Ressourcentabelle.

►►—GETDEF—OBJECT—(—Datenwert—)—————►



## Beschreibung

Mit diesem Befehl wird eine Ergebnisliste mit ausgewählten beschreibenden Datensätzen für eine Ressourcentabelle zurückgegeben.

- GETDEF ist eine Variante des Befehls GET. Mit GET werden Datensätze für die Ressource abgerufen; die Darstellung findet über eine Tabelle statt. Mit GETDEF hingegen werden interne Daten abgerufen, die die Ressourcentabelle selbst beschreiben.
- Mit dem Befehl GETDEF werden die Daten, die sogenannten Metadaten, aus internen Ressourcentabellen abgerufen, die die einzelnen externen Ressourcentabellen beschreiben. Diese internen Ressourcentabellen werden CICSplex SM-Metadatentabellen genannt. Die Attribute einer CICSplex SM-Metadatentabelle sind die Merkmale der externen Tabelle und nicht der Ressource, die sie darstellt. Eine Liste der CICSplex SM-Metadatenressourcentabellen, die mit GETDEF abgerufen werden können, finden Sie bei der Beschreibung der Option OBJECT (Object).
- Mithilfe von GETDEF können Sie ermitteln, welche Ressourcentabellen für die Verarbeitung durch andere Befehle verfügbar sind. Darüber hinaus können Sie die Attribute einer Ressourcentabelle, die Werte, die für ihre änderbaren Attribute zulässig sind, und die Aktionen, die für diese Tabelle ausgeführt werden können, angeben. Sie können den Befehl GETDEF auch dazu verwenden, um Beschreibungen der CICSplex SM-Metadatenressourcentabellen selbst anzufordern.
- Sie können den Befehl GETDEF nur mit Ressourcentabellen verwenden, die von CICSplex SM bereitgestellt werden. GETDEF ist für benutzerdefinierte Ansichten einer Ressourcentabelle, die durch den Befehl SPECIFY VIEW erstellt wurden, nicht gültig.
- Der Befehl REFRESH kann nicht zum Aktualisieren der Datensätze verwendet werden, die mit GETDEF abgerufen werden.

## Zugehörige Befehle

DISCARD, FETCH, GET, LOCATE, QUERY

## Optionen

### **ACTION**(Datenwert)

Der aus 12 Zeichen bestehende Name einer Aktion für die Ressourcentabelle, für die CICSplex SM-Metadatendatensätze abgerufen werden sollen.

### **ACTIONPARM**(Datenwert)

Der aus 12 Zeichen bestehende Name eines Parameters für eine Aktion für die Ressourcentabelle, für die CICSplex SM-Metadatendatensätze abgerufen werden sollen, und zwar so wie er in der API-Parameterzeichenfolge dargestellt wird.

**ADD**

Fügt die abgerufenen CICSplex SM-Metadaten-Ressourcentabellendatensätze zu einer vorhandenen Zielergebnisliste hinzu. Wenn keine vorhandene Ergebnisliste als Ziel angegeben ist, wird die Option ADD ignoriert.

**ATTRIBUTE** (*Datenwert*)

Gibt mindestens ein Attribut der Ressourcentabelle an, die in der Option RESOURCE angegeben ist, für die CICSplex SM-Metadatendatensätze abgerufen werden sollen.

In Abhängigkeit davon, welche CICSplex SM-Metadatentabelle in der Option OBJECT benannt ist, kann es sich bei diesem Wert um einen aus 1 bis 12 Zeichen bestehenden Namen eines bestimmten Attributs oder einen Stern (\*) für alle Attribute in der Ressourcentabelle handeln. Wenn Sie die Option ATTRIBUTE für ein OBJECT nicht angeben, für das die Angabe nicht erforderlich ist, werden Daten für alle Attribute in der Ressourcentabelle abgerufen.

Details zu den CICSplex SM-Metadaten-Ressourcentabellen und den gültigen ATTRIBUTE-Werten für die einzelnen Tabellen finden Sie bei der Beschreibung zur Option OBJECT.

**COUNT** (*Datenreferenz*)

Benennt eine Variable, um die Anzahl der Ressourcentabellendatensätze in der Zielergebnisliste zu empfangen, wenn diese Operation abgeschlossen ist.

**OBJECT** (*Datenwert*)

Gibt den Typ der Metadaten an, die für die Ressourcentabelle abgerufen werden sollen, die in der Option RESOURCE angegeben ist. Dieser Wert muss einer der folgenden CICSplex SM-Metadaten-Ressourcentabellennamen sein:

**OBJECT**

Für jede Instanz der Ressourcentabelle, die in der Option RESOURCE angegeben ist, wird ein einziger Datensatz zurückgegeben. Der Datensatz beschreibt die allgemeinen Merkmale der Ressourcentabelle. Zugehörige Optionen und Einschränkungen:

- ACTION wird ignoriert.
- ATTRIBUTE wird ignoriert.
- ACTIONPARM wird ignoriert.
- RESOURCE muss ein bestimmter Ressourcentabellenname oder \* für alle Ressourcentabellen sein.

**OBJECT**

Für jede für die Ressourcentabelle verfügbare Aktion, die in der Option RESOURCE angegeben ist, wird ein einziger Datensatz zurückgegeben.

Zugehörige Optionen und Einschränkungen:

- ACTION wird normalerweise weggelassen. Falls dem nicht so ist, kann mit dieser Option der Name einer Aktion bzw. \* angegeben werden.
- ATTRIBUTE wird ignoriert.
- ACTIONPARM wird ignoriert.
- RESOURCE muss ein bestimmter Ressourcentabellenname sein; der Wert \* ist nicht zulässig.

**METADDESC**

Für jedes Attribut der Ressourcentabelle, die in der Option RESOURCE angegeben ist, wird ein einziger Datensatz zurückgegeben. Jeder Datensatz stellt nur die Basisstruktur des Attributs bereit; dazu gehören

Name, Datentyp, Länge und relative Position in der Ressourcentabelle. Diese Informationen können für den Zugriff auf die Attributfelder in einem Puffer, die vom Befehl FETCH zurückgegeben werden, nützlich sein.

Zugehörige Optionen und Einschränkungen:

- ACTION wird ignoriert.
- ATTRIBUTE kann ein bestimmter Attributname bzw. \* für alle Attribute in der Ressourcentabelle sein.
- ACTIONPARM wird ignoriert.
- RESOURCE muss ein bestimmter Ressourcentabellenname sein; der Wert \* ist nicht zulässig.

**ATTR** Für jedes Attribut der Ressourcentabelle, die in der Option RESOURCE angegeben ist, wird ein Datensatz zurückgegeben. Jeder Datensatz stellt vollständige Informationen zu dem Attribut bereit.

Zugehörige Optionen und Einschränkungen:

- ATTRIBUTE kann ein bestimmter Attributname bzw. \* für alle Attribute in der Ressourcentabelle sein.
- RESOURCE muss ein bestimmter Ressourcentabellenname sein; der Wert \* ist nicht zulässig.

#### **ATTRAVA**

Für jeden der EYUDA- oder CVDA-Werte, die für das angegebene Attribut gültig sind, wird ein einziger Datensatz zurückgegeben.

Zugehörige Optionen und Einschränkungen:

- ATTRIBUTE muss der Name eines bestimmten Attributs sein, das über den Datentyp EYUDA, CVDAS oder CVDAT verfügt.
- RESOURCE muss ein bestimmter Ressourcentabellenname sein; der Wert \* ist nicht zulässig.

**Anmerkung:** Das Attribut AVAAVAIL der internen ATTR-Ressourcentabelle gibt an, ob eine AVA-Liste für ein bestimmtes Attribut verfügbar ist.

#### **METANAME**

Für jeden CVDA und EYUDA wird ein einziger Datensatz zurückgegeben. Die Schlüsselwörter RESOURCE, ATTRIBUTE, ACTION und ACTIONPARM werden ignoriert.

#### **METAPARM**

Für jeden Parameter für die angegebene Ressource (RESOURCE) und Aktion (ACTION) wird ein einziger Datensatz zurückgegeben.

- ACTION muss ein bestimmter Aktionsname sein; der Wert \* ist nicht zulässig.
- ACTIONPARM wird ignoriert.
- RESOURCE muss eine bestimmte Tabelle sein; der Wert \* ist nicht zulässig.

#### **PARMAVA**

Für die angegebene Ressource (RESOURCE), Aktion (ACTION) und den Aktionsparameter (ACTIONPARM) wird ein einziger Datensatz zurückgegeben.

- ACTION muss ein bestimmter Aktionsname sein; der Wert \* ist nicht zulässig.

- ACTIONPARM muss ein bestimmter Parametername sein; der Wert \* ist nicht zulässig.
- RESOURCE muss eine bestimmte Tabelle sein; der Wert \* ist nicht zulässig.

#### **REASON**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

#### **REPLACE**

Löscht den Inhalt einer vorhandenen Zielergebnisliste und ersetzt ihn durch die Ergebnisse dieser Operation. Wenn die Operation nicht in ausgewählten CICSplex SM-Metadaten-Ressourcentabellendatensätzen resultiert, wird die Zielergebnisliste verworfen.

Wenn keine vorhandene Ergebnisliste als Ziel angegeben ist, wird die Option REPLACE ignoriert.

#### **RESOURCE**(*Datenwert*)

Gibt die Ressourcentabelle an, für die CICSplex SM-Metadatendatensätze abzurufen sind.

Bei der Angabe der Option ATTRIBUTE muss es sich bei diesem Wert um einen aus 1 bis 8 Zeichen bestehenden Namen einer bestimmten CICSplex SM-Ressourcentabelle handeln. Andernfalls können Sie \* angeben, um Daten für alle Ressourcentabellen abzurufen.

**Anmerkung:** Sie können den Befehl GETDEF nur mit Ressourcentabellen verwenden, die von CICSplex SM bereitgestellt werden. GETDEF ist für benutzerdefinierte Ansichten einer Ressourcentabelle, die durch den Befehl SPECIFY VIEW erstellt wurden, nicht gültig.

#### **RESPONSE**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

#### **RESULT**(*CPSM-Token*)

Gibt die API-Ergebnisliste an, die von dieser Operation verarbeitet werden soll. Die Ergebnisliste kann durch einen der folgenden Befehle erzeugt werden:

- COPY
- GET
- GROUP
- PERFORM OBJECT.

CICSplex SM ersetzt den Inhalt der vorhandenen Ergebnisliste durch die Ressourcentabellendatensätze, die durch diesen Befehl des Typs GETDEF angefordert werden. Wenn die Operation nicht in ausgewählten Ressourcentabellendatensätzen resultiert, wird die Zielergebnisliste verworfen.

CICSplex SM erstellt eine neue Ergebnisliste und gibt das identifizierende Token im selben Feld zurück, wenn für dieses Feld Folgendes gilt:

- Es ist auf eine binäre Null gesetzt (in COBOL, C, PL/I oder Assembler)
- Es ist eine nicht initialisierte Variable (in REXX)

#### **THREAD**(*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl GETDEF zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

### **NODATA**

OBJECT wurde mit der Option OBJECT angegeben, doch für die angegebene RESOURCE wurden keine Aktionen definiert.

**BUSY** Aus den folgenden Gründen kam es zu einer Bedingung des Typs 'Aktiv':

### **RESULT**

Die in der Option RESULT angegebene Ergebnisliste wird von einem anderen Befehl verarbeitet.

### **ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

### **NOSERVICE**

Das Anwendungsstubsprogramm konnte das API-Servicemodul nicht laden.

### **NOSTORAGE**

Das Anwendungsstubsprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

### **SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

### **FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

### **ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

### **EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

### **INCOMPATIBLE**

Aus den folgenden Gründen kam es zu einer Inkompatibilitätsbedingung:

### **INVALIDOBJ**

Die in der Option RESULT angegebene Zielergebnisliste ist mit der Ausgabe dieses Befehls nicht kompatibel. Die Ergebnisliste muss denselben Typ von Metadaten enthalten (angegeben durch die Option OBJECT), die durch den Befehl erzeugt werden.

### **INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- ACTION
- ACTIONPARM
- ATTRIBUTE
- OBJECT
- RESOURCE



- RESULT
- THREAD.

Überprüfen Sie die Befehlsbeschreibung auf eine gültige Parametersyntax hin.

#### NOTAVAILABLE

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

##### APITASK

Die API-Steuersubtask ist nicht aktiv.

##### CPSMAPI

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

#### SERVERGONE

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

#### TABLEERROR

Ein Ressourcentabellendatensatz ist aus einem der folgenden Gründe ungültig:

##### DATAERROR

Der Wert, der einem oder mehreren Ressourcentabellenattributen zugeordnet ist, ist ungültig. Verwenden Sie den Befehl FEEDBACK, um weitere Informationen zu diesem Fehler abzurufen.

##### INVALIDVER

Die angegebene Version der Ressourcentabelle wird von CICSplex SM nicht unterstützt.

#### VERSIONINVL

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

##### NOTSUPPORTED

Die Version des Anwendungsstubprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

##### NOTVSNCONN

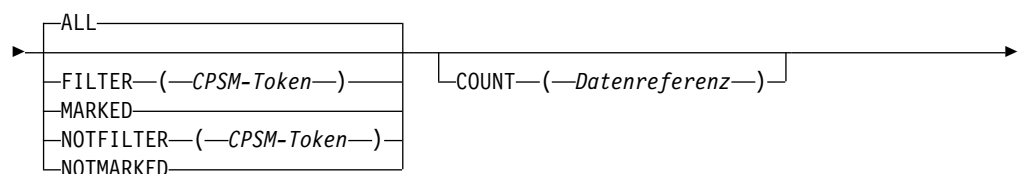
Die Version des Anwendungsstubprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

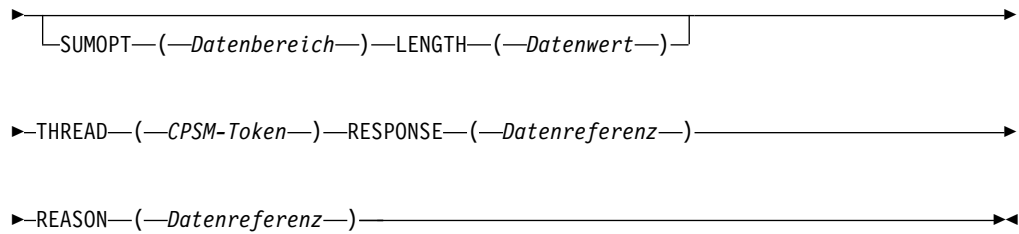
---

## GROUP

Zurückgeben einer zusammenfassenden Ergebnisliste.

►► GROUP BY (—Datenwert—) FROM (—CPSM-Token—) TO (—CPSM-Token—) ►





## Beschreibung

Durch diesen Befehl wird eine zusammenfassende Ergebnisliste einiger oder aller Ressourcentabellendatensätze in einer Ergebnisliste zurückgegeben.

- Bei der Zielergebnisliste kann es sich um eine vorhandene Ergebnisliste oder um eine neue Ergebnisliste handeln, die durch diesen Prozess erstellt wird. Wenn Sie eine vorhandene Ergebnisliste als Ziel eines Befehls des Typs GROUP angeben, muss Folgendes gelten:
  - Es muss sich um eine zusammenfassende Ergebnisliste handeln, die durch einen vorherigen Befehl des Typs GROUP für dieselbe Quellenergebnisliste erstellt wurde.
  - Sie muss denselben Typ von Ressourcentabellendatensätzen enthalten, die derzeit in der Quellenergebnisliste enthalten sind.
  - Die vorhandenen Datensätze in der Ergebnisliste werden überschrieben.
- Um eine zusammenfassende Ergebnisliste aus ausgewählten Datensätzen einer Quellenergebnisliste zu erstellen, können Sie die folgenden Befehle verwenden:
  - Befehl SPECIFY FILTER zum Definieren eines Filters für die Quellenergebnisliste.
  - Befehle MARK und UNMARK zum Markieren von Datensätzen in der Quellenergebnisliste.
- Der Befehl GROUP darf nur für Attribute mit einer Länge von 251 oder weniger verwendet werden. Für Attributlängen über 251 tritt ein Fehler des Typs RESPONSE(INVALIDPARM) REASON(BY) auf.
- Weitere Informationen zur Verarbeitung von zusammenfassenden Ergebnislisten finden Sie in Developing CICSplex SM applications.

## Zugehörige Befehle

DISCARD, EXPAND, FETCH, GET, LOCATE, MARK, ORDER, QUERY, SPECIFY FILTER

## Optionen

### ALL

Fasst alle Ressourcentabellendatensätze in der Quellenergebnisliste zusammen.

### BY(*Datenwert*)

Gibt das Ressourcentabellenattribut an, dessen Wert als Gruppierungsfaktor für diese Operation verwendet werden soll. Bei diesem Wert muss es sich um einen aus 1 bis 12 Zeichen bestehenden Namen eines gültigen Attributs für die Ressourcentabelle handeln.

### COUNT(*Datenreferenz*)

Benennt eine Variable, um die Anzahl der Ressourcentabellendatensätze in der Zielergebnisliste zu empfangen, wenn diese Operation abgeschlossen ist.

**FILTER**(*CPSM-Token*)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option FILTER fasst nur die Ressourcentabellendatensätze zusammen, die den angegebenen Filterkriterien entsprechen.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

**FROM**(*CPSM-Token*)

Gibt die Quellenergebnisliste für diese Operation an. Die Ergebnisliste kann durch einen der folgenden Befehle erzeugt werden:

- COPY
- GET
- PERFORM OBJECT.

**Anmerkung:** Wenn Sie die Quellenergebnisliste löschen, werden alle zusammenfassenden Ergebnislisten, die aus ihr erstellt wurden, ebenfalls verworfen.

**LENGTH**(*Datenwert*)

Ein Vollwortwert, der die Länge des SUMOPT-Puffers angibt.

**Anmerkung:** Die von Ihnen angegebene Pufferlänge darf keine anderen Daten als einen Zusammenfassungsausdruck enthalten.

**MARKED**

Fasst nur die Ressourcentabellendatensätze zusammen, die in der Quellenergebnisliste markiert sind. Ressourcentabellendatensätze können mit den Befehlen MARK und UNMARK markiert werden.

**NOTFILTER**(*CPSM-Token*)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option NOTFILTER fasst nur die Ressourcentabellendatensätze zusammen, die den angegebenen Filterkriterien nicht entsprechen.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

**NOTMARKED**

Fasst nur die Ressourcentabellendatensätze zusammen, die in der Quellenergebnisliste nicht markiert sind. Ressourcentabellendatensätze können mit den Befehlen MARK und UNMARK markiert werden.

**REASON**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

**RESPONSE**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**SUMOPT**(*Datenbereich*)

Gibt einen Puffer mit dem für diese Operation zu verwendenden Zusammenfassungsausdruck an. Mit dem Wert SUMOPT werden die Standardzusammenfassungsoptionen für die Ressourcentabellenattribute überschrieben.

Weitere Informationen zum Bilden eines Zusammenfassungsausdrucks finden Sie in Developing CICSplex SM applications. Eine Liste der Standardzusammenfassungsoptionen für eine angegebene Ressourcentabelle finden Sie in CICSplex SM-Ressourcentabellen in den Referenzinformationen.

**THREAD**(*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

**TO**(*CPSM-Token*)

Gibt die Zielergebnisliste für diese Operation an.

CICSplex SM erstellt eine neue zusammenfassende Ergebnisliste und gibt das entsprechende identifizierende Token im selben Feld zurück, wenn für dieses Feld Folgendes gilt:

- Es ist auf eine binäre Null festgelegt (in COBOL, C, PL/I oder Assembler)
- Es ist eine nicht initialisierte Variable (in REXX)

In anderen Fällen können Sie ein vorhandenes zusammenfassendes Ergebnis angeben, das durch einen vorherigen Befehl des Typs GROUP für die in der Option FROM angegebene Ergebnisliste erstellt wurde. Das heißt, Sie können eine zusammenfassende Ergebnisliste wiederverwenden, aber nur, um die Datensätze in derselben Ergebnisliste erneut zusammenzufassen.

**Anmerkung:** Wenn Sie das Token einer zuvor erstellten zusammenfassenden Ergebnisliste angeben, müssen Sie sicherstellen, dass die Ergebnisliste noch vorhanden ist. Wenn Sie eine Quellenergebnisliste löschen, werden alle zusammenfassenden Ergebnislisten, die aus ihr erstellt wurden, ebenfalls verworfen.

## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl GROUP zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

**NODATA**

Es wurden keine Datensätze gefunden, die mit den angegebenen Suchkriterien übereinstimmen.

**BUSY** Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Aktiv':

**FROM**

Die in der Option FROM angegebene Quellenergebnisliste wird von einem anderen Befehl verarbeitet.

**TO** Die in der Option TO angegebene Zielergebnisliste wird von einem anderen Befehl verarbeitet.

**ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

**NOSERVICE**

Das Anwendungsstubprogramm konnte das API-Servicemodul nicht laden.

**NOSTORAGE**

Das Anwendungsstubprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**INVALIDCMD**

Aus dem folgenden Grund ist der Befehl ungültig:

**LENGTH**

Die Gesamtlänge aller Optionen im Befehl überschreitet den maximalen Grenzwert.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- BY
- FILTER
- FROM
- LENGTH
- NOTFILTER
- SUMOPT
- THREAD
- TO.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

**NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

**APITASK**

Die API-Steuersubtask ist nicht aktiv.

**CPSMAPI**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

**SERVERGONE**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

**VERSIONINVL**

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

**NOTSUPPORTED**

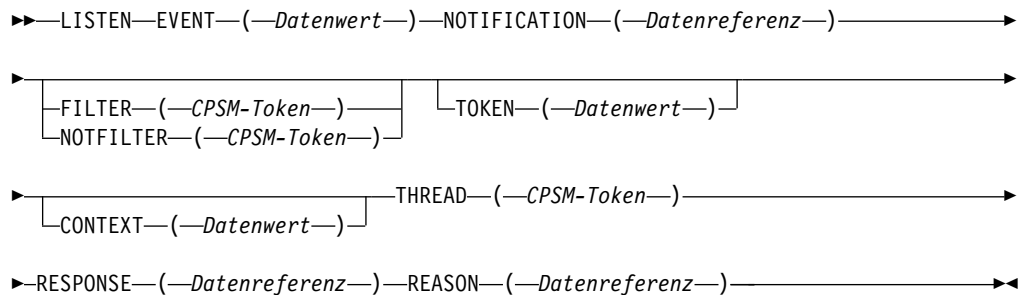
Die Version des Anwendungssubprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

## NOTVSNCONN

Die Version des Anwendungsstubsprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

## LISTEN

Anfordern einer Benachrichtigung, die an den Verarbeitungsthread gesendet werden soll.



### Beschreibung

Mit diesem Befehl wird das Senden einer Benachrichtigung an den Verarbeitungsthread angefordert, wenn es im CICSplex zu einem bestimmten Ereignis kommt.

- Ein Ereignis wird durch eine Ressourcentabelle mit einem CPSM-Benachrichtigungstyp dargestellt.
- Der Befehl LISTEN wird in Verbindung mit dem Befehl RECEIVE verwendet. Wenn Sie LISTEN verwenden, um eine Benachrichtigung über ein Ereignis anzufordern, müssen Sie einen nachfolgenden Befehl des Typs RECEIVE verwenden, um Informationen über das Ereignis abzurufen.
- Ein API-Verarbeitungsthread kann maximal 256 abgeschlossene asynchrone Anforderungen aufweisen, die gleichzeitig ausstehend sind. Wenn Sie den Befehl RECEIVE nicht in regelmäßigen Intervallen ausgeben und Ihr Verarbeitungsthread sein Maximum von 256 erreicht, werden asynchrone Anforderungen verworfen und nicht verarbeitet. Eine vollständige Beschreibung der asynchronen Verarbeitung finden Sie in *Developing CICSplex SM applications*.

### Zugehörige Befehle

ADDRESS, CANCEL, RECEIVE

### Optionen

#### CONTEXT (*Datenwert*)

Gibt den Kontext für diesen Befehl an. Bei dem Kontext muss es sich um einen aus 1 bis 8 Zeichen bestehenden Namen eines CMAS oder CICSplex handeln.

Wenn Sie die Option CONTEXT nicht angeben, wird der Standardkontext für den Thread angenommen.

#### EVENT (*Datenwert*)

Gibt die Ressourcentabelle an, die das Ereignis darstellt, auf das gewartet werden soll. Bei diesem Wert muss es sich um den aus 1 bis 8 Zeichen bestehen-

den Namen einer gültigen CICSplex SM Notification-Ressourcentabelle handeln. Eine Liste der CICSplex SM-Ressourcentabellen nach Typ finden Sie in *Developing CICSplex SM applications*.

**FILTER**(*CPSM-Token*)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option FILTER ist nur für die Ereignisse empfangsbereit, die die angegebenen Filterkriterien erfüllen.

Mit der Option FILTER können Sie die Benachrichtigungen begrenzen, die Sie zu Ereignissen erhalten, die einem bestimmten CMAS oder CICSplex zugeordnet sind. So können Sie beispielsweise den Filter

PLEXNAME=EYUPLX01.

erstellen und diesen Filter im Befehl LISTEN angeben, damit Sie nur über Ereignisse benachrichtigt werden, die mit der CICSplex-Nachricht EYUPLX01 generiert wurden.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

**NOTFILTER**(*CPSM-Token*)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option NOTFILTER ist nur für die Ereignisse empfangsbereit, die nicht die angegebenen Filterkriterien erfüllen.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

**NOTIFICATION**(*Datenreferenz*)

Benennt eine Variable, um das Vollwort-Token zu empfangen, das CICSplex SM dieser Benachrichtigungsanforderung zuordnet.

Dieses identifizierende Token muss im Befehl CANCEL angegeben werden, wenn die Benachrichtigungsanforderung abgebrochen werden soll.

**REASON**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

**RESPONSE**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**THREAD**(*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

**TOKEN**(*Datenwert*)

Definiert ein aus 1 bis 4 Zeichen bestehendes Token, das Sie für die Korrelation dieser Anforderung des Typs LISTEN mit dem Ergebnis eines nachfolgenden Befehls des Typs RECEIVE auswählen. Dieses Token dient zur Verwendung durch Ihr Programm; CICSplex SM verwendet den Wert nicht. Das Token wird durch den Befehl RECEIVE zurückgegeben, wenn es zu einem Ereignis des angegebenen Typs kommt.

## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl LISTEN zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

### ENVIRONERROR

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

#### NOSERVICE

Das Anwendungsstumpprogramm konnte das API-Servicemodul nicht laden.

#### NOSTORAGE

Das Anwendungsstumpprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

#### SOCRESOURCE

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

### FAILED

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

#### ABENDED

Die Befehlsverarbeitung wurde abnormal beendet.

#### EXCEPTION

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

### INCOMPATIBLE

Aus den folgenden Gründen kam es zu einer Inkompatibilitätsbedingung:

#### INVALIDEVT

Das angegebene Ereignis ist mit dem Filter, der in der Option FILTER oder NOTFILTER angegeben wurde, nicht kompatibel.

### INVALIDPARM

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- CONTEXT
- EVENT
- FILTER
- NOTFILTER
- NOTIFICATION
- THREAD
- TOKEN.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

### NOTAVAILABLE

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

#### APITASK

Die API-Steuersubtask ist nicht aktiv.



### CPSMAPI

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread versucht hat, eine Verbindung herzustellen, ist für die API-Verarbeitung nicht verfügbar.

### PLEXMGR

Der CMAS, mit dem der Verarbeitungsthread derzeit verbunden ist, ist nicht an der Verwaltung des angegebenen CICSplex beteiligt und es steht kein anderer CMAS zur Verfügung, der das CICSplex verwaltet.

### SERVERGONE

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

### VERSIONINVL

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

#### NOTSUPPORTED

Die Version des Anwendungsstubprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

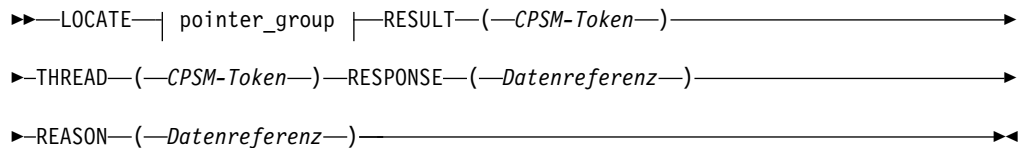
#### NOTVSNCONN

Die Version des Anwendungsstubprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

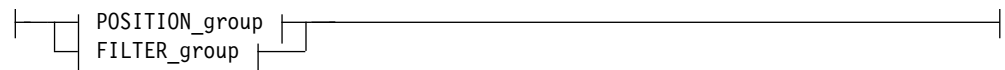
---

## LOCATE

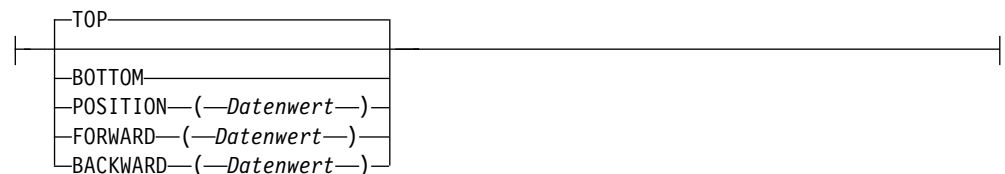
Positionieren des Datensatzzeigers in einer Ergebnisliste.



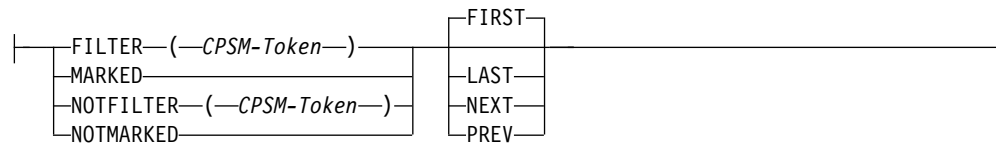
### pointer\_group



### POSITION\_group



## **FILTER\_group**



## **Beschreibung**

Mit diesem Befehl wird der Datensatzzeiger in einer Ergebnisliste positioniert.

- API-Befehle, mit denen Datensätze bearbeitet oder die Daten in einer Ergebnisliste aktualisieren, wirken sich auf die Position des Datensatzzeigers aus:
  - Nach einem Befehl des Typs GET wird der Zeiger auf den Anfang der Ergebnisliste positioniert.
  - Nach einem Befehl des Typs FETCH wird der Zeiger auf den nächsten Datensatz in der Ergebnisliste positioniert (d. h. der Datensatz nach dem letzten Datensatz, der in der jeweiligen Richtung, in der sich der Zeiger bewegte, abgerufen wurde; dies kann vorwärts oder rückwärts sein). Wenn keine Datensätze abgerufen wurden (da keine Datensätze mit den angegebenen Kriterien übereinstimmten), wird der Zeiger an den Anfang oder an das Ende der Ergebnisliste positioniert, in Abhängigkeit davon, in welche Richtung es sich bewegt hat.

Nachdem Sie einen anderen Befehl ausgegeben haben, der Datensätze bearbeitet oder Daten aktualisiert, hängt die Position des Datensatzzeigers von einer Kombination von Faktoren ab, einschließlich der Optionen, die Sie im Befehl angegeben haben. Um die Position des Zeigers sicherzustellen, müssen Sie den Befehl LOCATE verwenden, um ihn explizit in der Ergebnisliste zu positionieren.

- Mit dem Befehl LOCATE werden alle gelöschten Datensätze in der Ergebnisliste übersprungen. Wenn Sie versuchen, den Datensatzzeiger auf einen gelöschten Datensatz zu positionieren, empfangen Sie den RESPONSE-Wert NODATA.

## **Zugehörige Befehle**

COPY, DELETE, FETCH, GETDEF, GROUP, MARK, ORDER, PERFORM OBJECT, PERFORM SET, REFRESH, SET, SPECIFY FILTER, UNMARK

## **Optionen**

### **BACKWARD**(*Datenwert*)

Bewegt den Datensatzzeiger zurück, und zwar um die angegebene Anzahl an Ressourcentabellendatensätzen.

Wenn der Zeiger den Anfang der Ergebnisliste erreicht, bleibt er auf dem ersten Ressourcentabellendatensatz positioniert. Der Zeiger bewegt sich nicht weiter rückwärts bis zum Ende der Ergebnisliste.

### **BOTTOM**

Bewegt den Datensatzzeiger zum letzten Ressourcentabellendatensatz in der Ergebnisliste.

### **FILTER**(*CPSM-Token*)

Gibt einen Filter an, der für diese Operation verwendet werden soll.

Mit der Option FILTER wird der Datensatzzeiger auf einen Ressourcentabellendatensatz positioniert, der die angegebenen Filterkriterien erfüllt. Die Option

FIRST, LAST, NEXT oder PREV bestimmt, an welcher Stelle in der Ergebnisliste die Suche beginnt und in welcher Richtung sie weiter ausgeführt wird.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

#### **FIRST**

Startet eine Suche auf der Basis von Filter- oder Markierungskriterien mit dem ersten Ressourcentabellendatensatz in der Ergebnisliste. Die Suche wird in Vorwärtsrichtung durch die Ergebnisliste fortgesetzt, bis eine Übereinstimmung gefunden wird.

#### **FORWARD** (*Datenwert*)

Bewegt den Datensatzzeiger vorwärts, und zwar um die angegebene Anzahl an Ressourcentabellendatensätzen.

Wenn der Zeiger das Ende der Ergebnisliste erreicht, bleibt er auf dem letzten Ressourcentabellendatensatz positioniert. Der Zeiger bewegt sich nicht weiter vorwärts bis zum Anfang der Ergebnisliste.

#### **LAST**

Startet eine Suche auf der Basis von Filter- oder Markierungskriterien mit dem letzten Ressourcentabellendatensatz in der Ergebnisliste. Die Suche wird in Rückwärtsrichtung durch die Ergebnisliste fortgesetzt, bis eine Übereinstimmung gefunden wird.

#### **MARKED**

Positioniert den Datensatzzeiger auf einen markierten Ressourcentabellendatensatz. Die Option FIRST, LAST, NEXT oder PREV bestimmt, an welcher Stelle in der Ergebnisliste die Suche beginnt und in welcher Richtung sie weiter ausgeführt wird.

Ressourcentabellendatensätze können mit den Befehlen MARK und UNMARK markiert werden.

#### **NEXT**

Startet eine Suche auf der Basis von Filter- oder Markierungskriterien mit dem aktuellen Ressourcentabellendatensatz in der Ergebnisliste. Die Suche wird in Vorwärtsrichtung durch die Ergebnisliste fortgesetzt, bis eine Übereinstimmung gefunden wird.

#### **NOTFILTER** (*CPSM-Token*)

Gibt einen Filter an, der für diese Operation verwendet werden soll.

Mit der Option NOTFILTER wird der Datensatzzeiger auf einen Ressourcentabellendatensatz positioniert, der die angegebenen Filterkriterien nicht erfüllt. Die Option FIRST, LAST, NEXT oder PREV bestimmt, an welcher Stelle in der Ergebnisliste die Suche beginnt und in welcher Richtung sie weiter ausgeführt wird.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

#### **NOTMARKED**

Positioniert den Datensatzzeiger auf einen nicht markierten Ressourcentabellendatensatz. Die Option FIRST, LAST, NEXT oder PREV bestimmt, an welcher Stelle in der Ergebnisliste die Suche beginnt und in welcher Richtung sie weiter ausgeführt wird.

Ressourcentabellendatensätze können mit den Befehlen MARK und UNMARK markiert werden.

**POSITION**(*Datenwert*)

Bewegt den Datensatzzeiger zum n-ten Ressourcentabellendatensatz in der Ergebnisliste.

Dieser Wert muss eine Zahl sein, die die relative Position des Datensatzes in der Ergebnisliste angibt. Der erste Datensatz in einer Ergebnisliste wird durch die Zahl 1 angegeben.

Wenn Sie beispielsweise den Datensatzzeiger an den fünften Ressourcentabellendatensatz in einer Ergebnisliste verschieben möchten, geben Sie POSITION(5) an.

**PREV**

Startet eine Suche auf der Basis von Filter- oder Markierungskriterien mit dem vorherigen Ressourcentabellendatensatz in der Ergebnisliste. Die Suche wird in Rückwärtsrichtung durch die Ergebnisliste fortgesetzt, bis eine Übereinstimmung gefunden wird.

**REASON**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

**RESPONSE**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**RESULT**(*CPSM-Token*)

Gibt die API-Ergebnisliste an, die von dieser Operation verarbeitet werden soll. Die Ergebnisliste kann durch einen der folgenden Befehle erzeugt werden:

- COPY
- GET
- GETDEF
- GROUP
- PERFORM OBJECT.

**THREAD**(*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

**TOP**

Bewegt den Datensatzzeiger zum ersten Ressourcentabellendatensatz in der Ergebnisliste.

## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl LOCATE zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

**NODATA**

Es wurden keine Datensätze gefunden, die mit den angegebenen Suchkriterien übereinstimmen, und zwar aus einem der folgenden Gründe:

**BACKWARD**

Es sind keine weiteren Datensätze vorhanden, die die Suchkriterien in Rückwärtsrichtung erfüllen.

**FORWARD**

Es sind keine weiteren Datensätze vorhanden, die die Suchkriterien in Vorwärtsrichtung erfüllen.

**BUSY** Aus den folgenden Gründen kam es zu einer Bedingung des Typs 'Aktiv':

**RESULT**

Die in der Option RESULT angegebene Ergebnisliste wird von einem anderen Befehl verarbeitet.

**ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

**NOSERVICE**

Das Anwendungsstumpprogramm konnte das API-Servicemodul nicht laden.

**NOSTORAGE**

Das Anwendungsstumpprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

**SOLRESOURCE**

Eine erforderliche Ressource, deren Eigner ein lokaler Eigner ist (d. h. Eigner des Adressraums, in dem der Verarbeitungsthread ausgeführt wird), ist nicht verfügbar.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- BACKWARD
- FILTER
- FORWARD
- NOTFILTER
- POSITION
- RESULT
- THREAD.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

**NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

**APITASK**

Die API-Steuersubtask ist nicht aktiv.

### CPSMAPI

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread versucht hat, eine Verbindung herzustellen, ist für die API-Verarbeitung nicht verfügbar.

### SERVERGONE

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

### VERSIONINVL

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

#### NOTSUPPORTED

Die Version des Anwendungsstubsprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

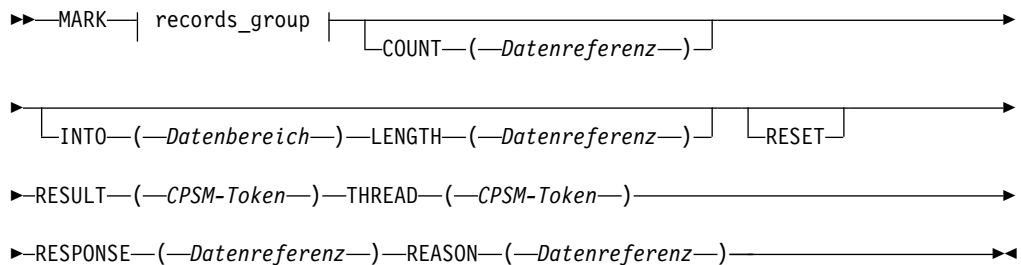
#### NOTVSNCONN

Die Version des Anwendungsstubsprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

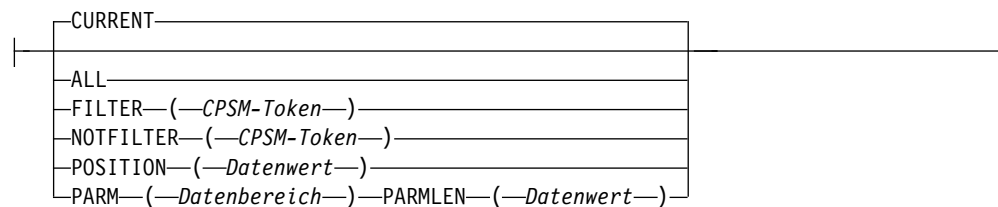
---

## MARK

Markieren ausgewählter Ressourcentabellendatensätze in einer Ergebnisliste.



### records\_group



## Beschreibung

Mit diesem Befehl werden ausgewählte Ressourcentabellendatensätze in einer Ergebnisliste markiert.

- Der Befehl MARK beginnt immer mit der Verarbeitung des letzten abgerufenen Datensatzes und nicht mit dem nächsten Datensatz in der Ergebnisliste.
- Alle Ressourcentabellendatensätze, die Sie zuvor in der Ergebnisliste markiert haben, bleiben markiert, es sei denn, Sie verwenden die Option RESET.

## Zugehörige Befehle

COPY, DELETE, EXPAND, FETCH, GROUP, LOCATE, PERFORM SET, REFRESH, SET, SPECIFY FILTER, UNMARK

## Optionen

### ALL

Markiert alle Ressourcentabellendatensätze in der Ergebnisliste. Wenn Sie ALL angeben, wird die Option RESET ignoriert.

### COUNT (Datenreferenz)

Benennt eine Variable, um die Anzahl der Ressourcentabellendatensätze zu empfangen, die nicht markiert werden konnten.

### CURRENT

Markiert nur den aktuellen Ressourcentabellendatensatz.

### FILTER (CPSM-Token)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option FILTER markiert nur die Ressourcentabellendatensätze, die den angegebenen Filterkriterien entsprechen.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

### INTO (Datenbereich)

Gibt einen Puffer an, um eine Liste der Ressourcentabellendatensätze zu empfangen, die nicht markiert werden konnten.

Dieser Puffer muss lang genug sein, um die maximale Anzahl von Datensatznummern aufzunehmen, die aus Ihrer Anforderung des Typs MARK resultieren können (wenn keine von ihnen markiert werden kann). Datensatznummern werden im INTO-Puffer einzeln aufgelistet (nicht nach Bereich) und werden durch Kommas voneinander getrennt.

**Anmerkung:** Wenn Sie den RESPONSE-Wert WARNING AREATOOSMALL empfangen (weil der Puffer nicht lang genug war), stellen die Daten, die in diesem Puffer zurückgegeben wurden, eine Teilliste der Datensätze dar, die nicht markiert werden konnten.

### LENGTH (Datenreferenz)

Ein Vollwortwert, der die Länge des INTO-Puffers angibt.

Der Wert, den CICSplex SM in diesem Feld zurückgibt, ist von dem RESPONSE-Wert für den Befehl MARK abhängig:

**OK** Die tatsächliche Länge der Daten, die im INTO-Puffer zurückgegeben werden.

### WARNING AREATOOSMALL

Die Pufferlänge, die erforderlich wäre, um eine vollständige Liste der Datensätze aufzunehmen, die nicht markiert werden konnten.

### NOTFILTER (CPSM-Token)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option NOTFILTER markiert nur die Ressourcentabellendatensätze, die den angegebenen Filterkriterien nicht entsprechen.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

**PARM**(*Datenbereich*)

Gibt einen Puffer mit dem Parameterausdruck an, mit dem die zu markierenden Ressourcentabellendatensätze aufgelistet werden.

Der Parameterausdruck für den Befehl MARK ist eine Zeichenfolge mit Datensatznummern. Beispiel:

```
PARM('1,3,6:9,24.')
```

Wenn Sie einzelne Datensätze angeben möchten, trennen Sie die Datensatznummern durch ein Komma voneinander. Um einen Bereich von Datensätzen anzugeben, trennen Sie die niedrigen und hohen Datensatznummern durch einen Doppelpunkt voneinander. Der gesamte Parameterausdruck muss mit einem Punkt enden.

Details zur Verwendung eines Parameterausdrucks mit dem Befehl MARK finden Sie in *Developing CICSplex SM applications*.

**PARMLEN**(*Datenwert*)

Ein Vollwortwert, der die Länge des PARM-Puffers angibt.

**POSITION**(*Datenwert*)

Markiert den n-ten Ressourcentabellendatensatz in der Ergebnisliste.

Dieser Wert muss eine Zahl sein, die die relative Position des Datensatzes in der Ergebnisliste angibt. Der erste Datensatz in einer Ergebnisliste wird durch die Zahl 1 angegeben.

Wenn Sie beispielsweise den fünften Ressourcentabellendatensatz in einer Ergebnisliste markieren möchten, geben Sie POSITION(5) an.

**REASON**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

**RESET**

Entfernt alle Markierungen, die zuvor in Ressourcentabellendatensätzen in der Ergebnisliste platziert wurden, und markiert nur die Datensätze, die Sie in der aktuellen MARK-Anforderung angeben.

Wenn Sie die Option RESET nicht verwenden, bleiben alle Datensätze markiert, die Sie zuvor markiert haben. Das heißt, die in der aktuellen MARK-Anforderung angegebenen Datensätze werden zusätzlich zu allen zuvor markierten Datensätzen markiert.

**RESPONSE**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**RESULT**(*CPSM-Token*)

Gibt die API-Ergebnisliste an, die von dieser Operation verarbeitet werden soll. Die Ergebnisliste kann durch einen der folgenden Befehle erzeugt werden:

- COPY
- GET
- GETDEF
- GROUP
- PERFORM OBJECT.

**THREAD**(*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.



## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl MARK zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

### **NODATA**

Es wurden keine Datensätze gefunden, die mit den angegebenen Suchkriterien übereinstimmen.

### **WARNING**

Der Befehl hat die Verarbeitung aus einem der folgenden Gründe mit einer Warnung beendet:

#### **AREATOOSMALL**

Sie haben die Optionen INTO und LENGTH angegeben, aber der Puffer war nicht lang genug, um die Zeichenfolge der Datensätze aufzunehmen, die nicht markiert werden konnten.

#### **DATAERROR**

Mindestens einer der im PARM-Puffer angegebenen Datensätze konnte nicht gefunden werden, um markiert zu werden. Wenn Sie die Option COUNT angegeben haben, wird die Anzahl der Datensätze, die nicht markiert werden konnten, zurückgegeben. Wenn Sie die Optionen INTO und LENGTH angegeben haben, wird eine Liste der Datensätze in dem Puffer zurückgegeben.

**BUSY** Aus den folgenden Gründen kam es zu einer Bedingung des Typs 'Aktiv':

#### **RESULT**

Die in der Option RESULT angegebene Ergebnisliste wird von einem anderen Befehl verarbeitet.

### **ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

#### **NOSERVICE**

Das Anwendungsstubprogramm konnte das API-Servicemodul nicht laden.

#### **NOSTORAGE**

Das Anwendungsstubprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

#### **SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

#### **SOLRESOURCE**

Eine erforderliche Ressource, deren Eigner ein lokaler Eigner ist (d. h. Eigner des Adressraums, in dem der Verarbeitungsthread ausgeführt wird), ist nicht verfügbar.

### **FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

#### **ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- COUNT
- FILTER
- INTO
- LENGTH
- NOTFILTER
- PARM
- PARMLEN
- RESULT
- THREAD.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

**NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

**APITASK**

Die API-Steuersubtask ist nicht aktiv.

**CPSMAPI**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

**SERVERGONE**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

**VERSIONINVL**

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

**NOTSUPPORTED**

Die Version des Anwendungsstubprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

**NOTVSNCONN**

Die Version des Anwendungsstubprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

---

## ORDER

Sortieren der Ressourcentabellendatensätze in einer Ergebnisliste.

- ORDER—BY—(—*Datenbereich*—)—LENGTH—(—*Datenwert*—)——————►
- RESULT—(—*CPSM-Token*—)—THREAD—(—*CPSM-Token*—)——————►
- RESPONSE—(—*Datenreferenz*—)—REASON—(—*Datenreferenz*—)——————►◄

## Beschreibung

Mit diesem Befehl werden die Ressourcentabellendatensätze in einer Ergebnisliste in eine benutzerdefinierte Reihenfolge sortiert.

- Standardmäßig werden die Datensätze nach den Schlüsselattributen für die Ressourcentabelle sortiert.
- Die von Ihnen angegebene Sortierreihenfolge für eine Ergebnisliste bleibt so lange wirksam, bis Sie einen anderen Befehl des Typs ORDER ausgeben.
- Wenn die Ergebnisliste gelöschte Datensätze enthält, werden diese Datensätze in den Sortierprozess eingeschlossen. Sie werden nach denselben Attributen sortiert wie andere Datensätze und ihre Position in der neu geordneten Ergebnisliste ist möglicherweise schwer zu ermitteln. Um dies zu verhindern, setzen Sie den Befehl REFRESH vor dem Befehl ORDER ab; durch REFRESH werden alle gelöschten Datensätze aus der Ergebnisliste entfernt.

## Zugehörige Befehle

COPY, GET, GETDEF, GROUP, LOCATE, PERFORM OBJECT

## Optionen

### BY(*Datenbereich*)

Gibt einen Puffer mit dem für diese Operation zu verwendenden Ausdruck an.

Bei einem Sortierausdruck handelt es sich um eine Liste von Attributen, die beim Sortieren der Ressourcentabellendatensätze verwendet werden sollen. Beispiel:

CICSSYS,TRANID.

Dabei werden die Attributnamen durch Kommas oder Leerzeichen voneinander getrennt und der gesamte Ausdruck wird mit einem Punkt beendet.

In diesem Beispiel werden die Ressourcentabellendatensätze mit dem CICS-Systemnamen als primärem Sortierschlüssel und der Transaktions-ID als sekundärem Schlüssel sortiert. Die Standardsortierreihenfolge ist aufsteigend. Wenn Sie Attributwerte in absteigender Reihenfolge sortieren möchten, fügen Sie /D am Ende des Attributnamens hinzu.

Weitere Informationen zur Verwendung von Sortierausdrücken mit dem Befehl ORDER finden Sie in Developing CICSplex SM applications.

### LENGTH(*Datenwert*)

Ein Vollwortwert, der die Länge des BY-Puffers angibt.

**Anmerkung:** Die von Ihnen angegebene Pufferlänge darf keine anderen Daten als einen Sortierausdruck enthalten.

### REASON(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

### RESPONSE(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

### RESULT(*CPSM-Token*)

Gibt die API-Ergebnisliste an, die von dieser Operation verarbeitet werden soll. Die Ergebnisliste kann durch einen der folgenden Befehle erzeugt werden:

- COPY

- GET
- GETDEF
- GROUP
- PERFORM OBJECT.

#### **THREAD** (*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

## **Bedingungen**

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl ORDER zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

**BUSY** Aus den folgenden Gründen kam es zu einer Bedingung des Typs 'Aktiv':

#### **RESULT**

Die in der Option RESULT angegebene Ergebnisliste wird von einem anderen Befehl verarbeitet.

#### **ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

#### **NOSERVICE**

Das Anwendungsstubprogramm konnte das API-Servicemodul nicht laden.

#### **NOSTORAGE**

Das Anwendungsstubprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

#### **SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

#### **FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

#### **ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

#### **EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

#### **INVALIDCMD**

Aus dem folgenden Grund ist der Befehl ungültig:

#### **LENGTH**

Die Gesamtlänge aller Optionen im Befehl überschreitet den maximalen Grenzwert.

#### **INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- BY

- LENGTH
- RESULT
- THREAD.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

#### NOTAVAILABLE

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

#### APITASK

Die API-Steuersubtask ist nicht aktiv.

#### CPSMAPI

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

#### SERVERGONE

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

#### VERSIONINVL

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

#### NOTSUPPORTED

Die Version des Anwendungsstubsprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

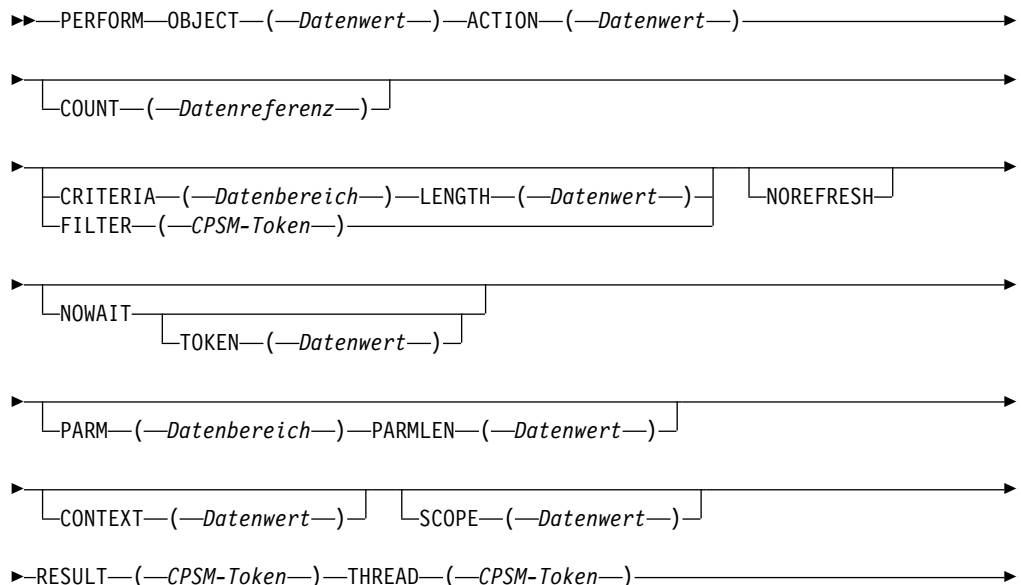
#### NOTVSNCONN

Die Version des Anwendungsstubsprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

---

## PERFORM OBJECT

Durchführen einer Aktion für eine oder mehrere Ressourcen.



## Beschreibung

Mit diesem Befehl wird für eine oder mehrere Ressourcen eine Aktion durchgeführt.

- Die Ressourcen, an denen mit **PERFORM OBJECT** gearbeitet werden soll, müssen nicht als Datensätze in einer Ergebnisliste vorhanden sein. Eine Ergebnisliste wird implizit durch diesen Prozess erstellt.
- Wenn der gültige Kontext und Bereich beim Absetzen des Befehls **PERFORM OBJECT** CICS-Systeme umfasst, die die angeforderte Aktion nicht unterstützen, wird die Anforderung für diese CICS-Systeme ignoriert.
- Der Befehl **PERFORM OBJECT** beinhaltet zwei Phasen: bei der ersten Phase wird die Ergebnisliste erstellt, bei der zweiten Phase wird die angeforderte Aktion für die Datensätze in der Ergebnisliste durchgeführt. Wenn während der Erstellung der Ergebnisliste ein Fehler auftritt, die Ergebnisliste jedoch nicht leer ist, wird die angeforderte Aktion für die Datensätze, die in der Ergebnisliste vorhanden sind, dennoch versucht.

## Zugehörige Befehle

DISCARD, GET, LOCATE, PERFORM SET, QUERY, SET, SPECIFY FILTER

## Optionen

### **ACTION**(Datenwert)

Gibt die durchzuführende Aktion an. Bei diesem Wert muss es sich um einen aus 1 bis 12 Zeichen bestehenden Namen einer gültigen Aktion für die Ressourcentabelle handeln.

Eine Beschreibung der Aktionen, die für eine angegebene Ressourcentabelle gültig sind, finden Sie im Abschnitt zu den CICSplex SM-Ressourcentabellen in den Referenzinformationen.

### **CONTEXT**(Datenwert)

Gibt den Kontext für diesen Befehl an. Bei dem Kontext muss es sich um einen aus 1 bis 8 Zeichen bestehenden Namen eines CMAS oder CICSplex handeln.

Wenn Sie die Option CONTEXT nicht angeben, wird der Standardkontext für den Thread angenommen.

### **COUNT**(Datenreferenz)

Benennt eine Variable, um die Anzahl der Ressourcentabellendatensätze in der Zielergebnisliste zu empfangen, wenn diese Operation abgeschlossen ist.

### **CRITERIA**(Datenbereich)

Gibt einen Puffer mit dem für diese Operation zu verwendenden Filterausdruck an. Die Option CRITERIA ruft nur die Ressourcentabellendatensätze ab, die den angegebenen Filterkriterien entsprechen.

Weitere Informationen zum Bilden eines Filterausdrucks finden Sie in Developing CICSplex SM applications.

### **FILTER**(CPSM-Token)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option FILTER ruft nur die Ressourcentabellendatensätze ab, die den angegebenen Filterkriterien entsprechen.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl *SPECIFY FILTER* zurückgegeben.

**LENGTH**(*Datenwert*)

Ein Vollwortwert, der die Länge des *CRITERIA*-Puffers angibt.

**Anmerkung:** Die von Ihnen angegebene Pufferlänge darf keine anderen Daten als einen Filterausdruck enthalten.

**NOREFRESH**

Gibt an, dass die Ressourcentabellendatensätze in der Ergebnisliste, die durch *PERFORM OBJECT* erstellt wurden, nicht aktualisiert werden dürfen. Die Datensätze spiegeln den Status der Ressourcen bei der Erstellung der Ergebnisliste wider.

Wenn Sie die Option *NOREFRESH* nicht angeben, werden die Ressourcentabellendatensätze aktualisiert, um den Ressourcenstatus nach Abschluss dieser Operation widerzuspiegeln.

**NOWAIT**

Gibt Ihrem Programm die Kontrolle zurück, sobald der Befehl *PERFORM OBJECT* akzeptiert wurde; dadurch kann der Befehl asynchron verarbeitet werden.

Wenn Sie die Option *NOWAIT* angeben, müssen Sie einen nachfolgenden Befehl des Typs *RECEIVE* verwenden, um zu testen, ob diese Anforderung ausgeführt wurde. Die Ergebnisse einer asynchronen Anforderung werden als Ressourcentabellendatensätze des Typs *ASYNCREQ* zurückgegeben. Eine vollständige Beschreibung der asynchronen Verarbeitung finden Sie in *Developing CICSplex SM applications*.

**Anmerkung:** Wenn Sie die Option *TOKEN* angeben, wird die Option *NOWAIT* standardmäßig angenommen.

**OBJECT**(*Datenwert*)

Gibt die Ressourcentabelle an, für die die Aktion ausgeführt werden soll. Bei diesem Wert muss es sich um einen aus 8 Zeichen bestehenden Namen einer gültigen Ressourcentabelle handeln.

**PARM**(*Datenbereich*)

Gibt einen Puffer mit dem Parameterausdruck an, der zum Durchführen der Aktion verwendet werden soll.

Details zur Verwendung eines Parameterausdrucks mit dem Befehl *PERFORM OBJECT* finden Sie in *Developing CICSplex SM applications*. Eine Beschreibung der Parameter, die für eine angegebene Ressourcentabellenaktion erforderlich sind, finden Sie im Abschnitt zu den *CICSplex SM-Ressourcentabellen* in den Referenzinformationen.

**PARMLEN**(*Datenwert*)

Ein Vollwortwert, der die Länge des *PARM*-Puffers angibt.

**REASON**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

**RESPONSE**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

### **RESULT** (*CPSM-Token*)

Gibt die API-Ergebnisliste an, die von dieser Operation verarbeitet werden soll. Die Ergebnisliste kann durch einen der folgenden Befehle erzeugt werden:

- COPY
- GET
- GROUP
- PERFORM OBJECT.

CICSplex SM ersetzt den Inhalt der vorhandenen Ergebnisliste durch die Ressourcentabellendatensätze, die durch diesen Befehl des Typs **PERFORM OBJECT** angefordert werden.

CICSplex SM erstellt eine neue Ergebnisliste und gibt das identifizierende Token im selben Feld zurück, wenn für dieses Feld Folgendes gilt:

- Es ist auf eine binäre Null gesetzt (in COBOL, C, PL/I oder Assembler)
- Es ist eine nicht initialisierte Variable (in REXX)

### **SCOPE** (*Datenwert*)

Gibt den Bereich für diesen Befehl an.

Um die Option SCOPE verwenden zu können, muss der aktuelle Kontext (der durch diesen Befehl oder einen vorherigen Befehl des Typs CONNECT oder QUALIFY definiert wurde) ein CICSplex sein. Für den Bereich kann Folgendes gelten:

- Es ist der aus 1 bis 8 Zeichen bestehende Name des CICSplex selbst
- Es ist ein CICS-System oder eine CICS-Systemgruppe innerhalb des CICSplex
- Es ist ein logischer Bereich wie er in einer CICSplex SM-Ressourcenbeschreibung (RESDESC) definiert wurde

Wenn OBJECT eine CICS-Definitionsressource angibt und die Option PARM den Parameter **CSDGROUP** beinhaltet, kann ein gültiger Bereich angegeben werden. Für den Bereich kann Folgendes gelten:

- Es ist ein CICS-System im CICSplex

Wenn es sich bei dem aktuellen Kontext um einen CMAS handelt oder die Option OBJECT einen anderen Typ von Ressourcentabelle angibt, ist diese Option bedeutungslos und wird ignoriert.

Wenn Sie die Option SCOPE nicht angeben, wird der Standardbereich für den Thread angenommen. Wenn der aktuelle Kontext ein CICSplex ist und kein Standardbereich für den Befehl CONNECT oder QUALIFY festgelegt wurde, erhalten Sie eine Antwort des Typs INVALIDPARM für die Option SCOPE.

### **THREAD** (*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

### **TOKEN** (*Datenwert*)

Definiert ein aus 1 bis 4 Zeichen bestehendes Token, das Sie für die Korrelation einer asynchronen Anforderung des Typs **PERFORM OBJECT** mit dem Ergebnis eines nachfolgenden Befehls des Typs RECEIVE auswählen. Dieses Token dient zur Verwendung durch Ihr Programm; CICSplex SM verwendet den Wert nicht. Das Token wird durch den Befehl RECEIVE zurückgegeben, wenn diese **PERFORM OBJECT**-Anforderung abgeschlossen ist.



## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl **PERFORM OBJECT** zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

### **SCHEDULED**

Der Befehl wurde für die Verarbeitung geplant.

### **NODATA**

Es wurden keine Datensätze gefunden, die mit den angegebenen Suchkriterien übereinstimmen.

**BUSY** Aus den folgenden Gründen kam es zu einer Bedingung des Typs 'Aktiv':

#### **RESULT**

Die in der Option RESULT angegebene Ergebnisliste wird von einem anderen Befehl verarbeitet.

### **ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

#### **NOSERVICE**

Das Anwendungsstubsprogramm konnte das API-Servicemodul nicht laden.

#### **NOSTORAGE**

Das Anwendungsstubsprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

#### **NOTPROCESSED**

Einer der MASSs, an den die Anforderung übertragen wurde, konnte die Anforderung nicht verarbeiten.

#### **REQTIMEOUT**

Einer der CMASs oder MASSs, an die die Anforderung übertragen wurde, antwortete nicht.

#### **SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

### **FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

#### **ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

#### **EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

### **INVALIDATA**

Es wurden ungültige Daten gefunden. Der Parameter, der ungültige Daten enthält, wird als Ursachenwert zurückgegeben:

**PARM** Ein Attributwert, der im PARM-Puffer aufgelistet ist, ist für das angegebene Attribut nicht gültig.

**CRITERIA**

Ein Attributwert, der im CRITERIA-Puffer aufgelistet ist, ist für das angegebene Attribut nicht gültig.

**INVALIDCMD**

Aus einem der folgenden Gründe ist der Befehl ungültig:

**FILTER**

Der für die Operation übergebene Filterausdruck ist zu umfangreich oder zu komplex.

**LENGTH**

Die Gesamtlänge aller Optionen im Befehl überschreitet den maximalen Grenzwert.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- ACTION
- CONTEXT
- CRITERIA
- FILTER
- LENGTH
- OBJECT
- PARM
- PARMLEN
- RESULT
- SCOPE
- THREAD
- TOKEN.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

**NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

**APITASK**

Die API-Steuersubtask ist nicht aktiv.

**CMAS**

Ein CMAS (CICSplex SM-Adressraum), an den die Anforderung übertragen wurde, ist nicht verfügbar.

**CPSMAPI**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

**MAINTPOINT**

Der Wartungspunkt für den aktuellen Kontext ist nicht verfügbar.

**PLEXMGR**

Der CMAS, mit dem der Verarbeitungsthread derzeit verbunden ist, ist nicht an der Verwaltung des angegebenen CICSplex beteiligt und es steht kein anderer CMAS zur Verfügung, der das CICSplex verwaltet.

**SCOPE**

Entweder ist keiner der MASs im angegebenen Bereich verfügbar, oder keiner von ihnen unterstützt die angeforderte Aktion.

**WORKLOAD**

Die in der API-Anforderung angegebene Workload ist in dem lokalen CMAS nicht verfügbar.

**NOTFOUND**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht gefunden':

**ACTION**

Die in der Option ACTION angegebene Aktion wurde für die angegebene Ressourcentabelle nicht gefunden.

**ATTRIBUTE**

Ein Attribut, das im CRITERIA- oder PARM-Puffer angegeben wurde, wurde für die angegebene Ressourcentabelle nicht gefunden.

**NOTPERMIT**

Aus den folgenden Gründen kam es zu einer Unzulässigkeitsbedingung:

**USRID**

Die Benutzer-ID, die dem Verarbeitungsthread zugeordnet ist, verfügt nicht über die erforderliche Sicherheitsberechtigung.

**TABLEERROR**

Ein Ressourcentabellendatensatz ist aus dem folgenden Grund ungültig:

**DATAERROR**

Der Wert, der einem oder mehreren Ressourcentabellenattributen zugeordnet ist, ist ungültig. Dieser Fehler kann auftreten, wenn der Ressourcentabelle erforderliche Felder fehlen, sie ein oder mehrere miteinander in Konflikt stehende Felder enthält oder sie doppelt vorhanden ist. Bei BAS kann dieser Fehler auch auftreten, wenn Sie nicht über die erforderliche Sicherheitsberechtigung verfügen. Verwenden Sie den Befehl FEEDBACK, um weitere Informationen zu diesem Fehler abzurufen.

**SERVERGONE**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

**VERSIONINVL**

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

**ACTION**

Die angegebene Aktion wird für die Version in Kombination mit dem Befehl CONNECT nicht unterstützt.

**NOTSUPPORTED**

Die Version des Anwendungsstubsprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

**NOTVSNCONN**

Die Version des Anwendungsstubsprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

## WARNING

Aus einem der folgenden Gründe wurde der Befehl PERFORM OBJECT möglicherweise nur teilweise ausgeführt:

## RESULT

Während der Erstellung der Ergebnisliste, die für den Befehl verwendet werden soll, wurde eine Antwort empfangen, die besagt, dass etwas nicht OK sei. Es waren jedoch einige Datensätze der Ergebnisliste verfügbar und die angeforderte Aktion wurde erfolgreich für sie ausgeführt. Verwenden Sie den Befehl FEEDBACK ohne die Option RESULT, um Informationen über die Antwort zu erhalten, die besagt, dass etwas nicht OK sei.

## ACTION

Während der Erstellung der Ergebnisliste, die für den Befehl verwendet werden soll, wurde eine Antwort empfangen, die besagt, dass etwas nicht OK sei. Es waren jedoch einige Datensätze der Ergebnisliste verfügbar und die Ausführung der angeforderten Aktion wurde versucht. Die angegebene Aktion wurde für mindestens einen Datensatz der Ergebnisliste nicht erfolgreich ausgeführt, da es zu einer CICSplex SM-Antwort bzw. -Ursache des Typs TABLEERROR oder DATAERROR kam.

Verwenden Sie den Befehl FEEDBACK ohne die Option RESULT, um Informationen über den Fehler zu erhalten, der während der Erstellung der Ergebnisliste aufgetreten ist. Verwenden Sie den Befehl FEEDBACK mit der Option RESULT, um Informationen zu den Datensätzen zu erhalten, die zu der Antwort bzw. der Ursache des Typs TABLEERROR oder DATAERROR geführt haben.

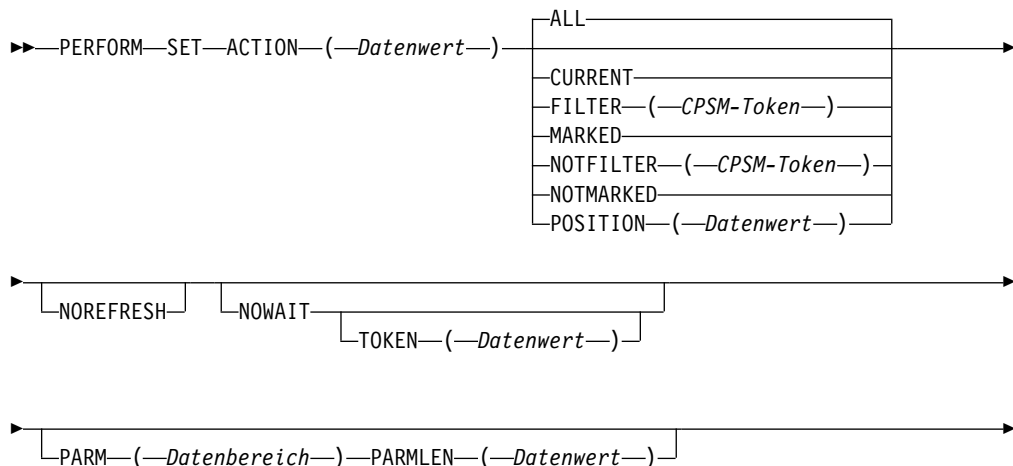
## WORKLOAD

Die in der API-Anforderung angegebene Workload ist in dem lokalen CMAS nicht verfügbar.

---

## PERFORM SET

Durchführen einer Aktion für eine oder mehrere Ressourcen.



►RESULT—(—CPSM-Token—)—THREAD—(—CPSM-Token—)—————►

►RESPONSE—(—Datenreferenz—)—REASON—(—Datenreferenz—)—————◄◄

## Beschreibung

Mit diesem Befehl wird eine Aktion für eine oder mehrere Ressourcen durchgeführt, die durch Ressourcentabellendatensätze in einer vorhandenen Ergebnisliste dargestellt werden. Wenn der gültige Kontext und Bereich beim Absetzen des Befehls PERFORM SET CICS-Systeme umfasst, die die angeforderte Aktion nicht unterstützen, wird die Anforderung für diese CICS-Systeme ignoriert.

## Zugehörige Befehle

LOCATE, MARK, PERFORM OBJECT, SET, SPECIFY FILTER

## Optionen

### **ACTION**(Datenwert)

Gibt die durchzuführende Aktion an. Bei diesem Wert muss es sich um einen aus 1 bis 12 Zeichen bestehenden Namen einer gültigen Aktion für die Ressourcentabelle handeln.

Eine Beschreibung der Aktionen, die für eine angegebene Ressourcentabelle gültig sind, finden Sie im Abschnitt zu den CICSplex SM-Ressourcentabellen in den Referenzinformationen.

### **ALL**

Führt die angegebene Aktion für alle Ressourcentabellendatensätze in der Ergebnisliste durch.

### **CURRENT**

Führt die angegebene Aktion nur für den aktuellen Ressourcentabellendatensatz durch.

### **FILTER**(CPSM-Token)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option FILTER führt die Aktion nur für die Ressourcentabellendatensätze durch, die den angegebenen Filterkriterien entsprechen.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

### **MARKED**

Führt die angegebene Aktion nur für die Ressourcentabellendatensätze durch, die in der Ergebnisliste markiert sind. Ressourcentabellendatensätze können mit den Befehlen MARK und UNMARK markiert werden.

### **NOREFRESH**

Gibt an, dass die Ressourcentabellendatensätze in der Quellenergebnisliste nicht aktualisiert werden dürfen. Die Datensätze spiegeln den Status der Ressourcen vor der Verarbeitung des Befehls PERFORM SET wider.

Wenn Sie die Option NOREFRESH nicht angeben, werden die Ressourcentabellendatensätze aktualisiert, um den Ressourcenstatus nach Abschluss dieser Operation widerzuspiegeln.

**NOTFILTER**(*CPSM-Token*)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option NOTFILTER führt die Aktion nur für die Ressourcentabellendatensätze durch, die nicht den angegebenen Filterkriterien entsprechen.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

**NOTMARKED**

Führt die angegebene Aktion nur für die Ressourcentabellendatensätze durch, die in der Ergebnisliste nicht markiert sind. Ressourcentabellendatensätze können mit den Befehlen MARK und UNMARK markiert werden.

**NOWAIT**

Gibt Ihrem Programm die Kontrolle zurück, sobald der Befehl PERFORM SET akzeptiert wurde; dadurch kann der Befehl asynchron verarbeitet werden.

Wenn Sie die Option NOWAIT angeben, müssen Sie einen nachfolgenden Befehl des Typs RECEIVE verwenden, um zu testen, ob diese Anforderung ausgeführt wurde. Die Ergebnisse einer asynchronen Anforderung werden als Ressourcentabellendatensätze des Typs ASYNCREQ zurückgegeben. Eine vollständige Beschreibung der asynchronen Verarbeitung finden Sie in *Developing CICSplex SM applications*.

**Anmerkung:** Wenn Sie die Option TOKEN angeben, wird die Option NOWAIT standardmäßig angenommen.

**PARM**(*Datenbereich*)

Gibt einen Puffer mit dem Parameterausdruck an, der zum Durchführen der Aktion verwendet werden soll.

Details zur Verwendung eines Parameterausdrucks mit dem Befehl PERFORM SET finden Sie in *Developing CICSplex SM applications*. Eine Beschreibung der Parameter, die für eine angegebene Ressourcentabellenaktion erforderlich sind, finden Sie im Abschnitt zu den CICSplex SM-Ressourcentabellen in den Referenzinformationen.

**PARMLEN**(*Datenwert*)

Ein Vollwortwert, der die Länge des PARM-Puffers angibt.

**POSITION**(*Datenwert*)

Führt die angegebene Aktion für den n-ten Ressourcentabellendatensatz in der Ergebnisliste durch.

Dieser Wert muss eine Zahl sein, die die relative Position des Datensatzes in der Ergebnisliste angibt. Der erste Datensatz in einer Ergebnisliste wird durch die Zahl 1 angegeben.

Wenn Sie beispielsweise die angegebene Aktion im fünften Ressourcentabellendatensatz in einer Ergebnisliste durchführen möchten, geben Sie POSITION(5) an.

**REASON**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

**RESPONSE**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**RESULT** (*CPSM-Token*)

Gibt die API-Ergebnisliste an, die von dieser Operation verarbeitet werden soll. Die Ergebnisliste kann durch einen der folgenden Befehle erzeugt werden:

- COPY
- GET
- GROUP
- PERFORM OBJECT.

**THREAD** (*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

**TOKEN** (*Datenwert*)

Definiert ein aus 1 bis 4 Zeichen bestehendes Token, das Sie für die Korrelation einer asynchronen Anforderung des Typs PERFORM SET mit dem Ergebnis eines nachfolgenden Befehls des Typs RECEIVE auswählen. Dieses Token dient zur Verwendung durch Ihr Programm; CICSplex SM verwendet den Wert nicht. Das Token wird durch den Befehl RECEIVE zurückgegeben, wenn diese PERFORM SET-Anforderung abgeschlossen ist.

## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl PERFORM SET zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

**SCHEDULED**

Der Befehl wurde für die Verarbeitung geplant.

**NODATA**

Es wurden keine Datensätze gefunden, die mit den angegebenen Suchkriterien übereinstimmen. Bei Angabe der Option ALL wird möglicherweise der folgende Grund zurückgegeben:

**FORWARD**

Es sind keine weiteren Datensätze vorhanden, die die Suchkriterien in Vorwärtsrichtung erfüllen.

**BUSY** Aus den folgenden Gründen kam es zu einer Bedingung des Typs 'Aktiv':

**RESULT**

Die in der Option RESULT angegebene Ergebnisliste wird von einem anderen Befehl verarbeitet.

**ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

**NOSERVICE**

Das Anwendungsstubprogramm konnte das API-Servicemodul nicht laden.

**NOSTORAGE**

Das Anwendungsstubprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**NOTPROCESSED**

Einer der MASs, an den die Anforderung übertragen wurde, konnte die Anforderung nicht verarbeiten.

**REQTIMEOUT**

Einer der CMASs oder MASs, an die die Anforderung übertragen wurde, antwortete nicht.

**SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**INVALIDDATA**

Es wurden ungültige Daten gefunden. Der Parameter, der ungültige Daten enthält, wird als Ursachenwert zurückgegeben:

**PARM** Ein Attributwert, der im PARM-Puffer aufgelistet ist, ist für das angegebene Attribut nicht gültig.

**INVALIDCMD**

Aus dem folgenden Grund ist der Befehl ungültig:

**LENGTH**

Die Gesamtlänge aller Optionen im Befehl überschreitet den maximalen Grenzwert.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- ACTION
- FILTER
- NOTFILTER
- PARM
- PARMLEN
- POSITION
- RESULT
- THREAD
- TOKEN.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

**NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

**APITASK**

Die API-Steuersubtask ist nicht aktiv.



**CMAS**

Ein CMAS (CICSplex SM-Adressraum), an den die Anforderung übertragen wurde, ist nicht verfügbar.

**CPSMAPI**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

**MAINTPOINT**

Der Wartungspunkt für den aktuellen Kontext ist nicht verfügbar.

**PLEXMGR**

Der CMAS, mit dem der Verarbeitungsthread derzeit verbunden ist, ist nicht an der Verwaltung des angegebenen CICSplex beteiligt und es steht kein anderer CMAS zur Verfügung, der das CICSplex verwaltet.

**SCOPE**

Mindestens einer der in dem Bereich angegebenen MASs ist nicht verfügbar.

**WORKLOAD**

Die in der API-Anforderung angegebene Workload ist in dem lokalen CMAS nicht verfügbar.

**NOTFOUND**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht gefunden':

**ACTION**

Die in der Option ACTION angegebene Aktion wurde für die angegebene Ressourcentabelle nicht gefunden.

**ATTRIBUTE**

Ein Attribut, das im CRITERIA- oder PARM-Puffer angegeben wurde, wurde für die angegebene Ressourcentabelle nicht gefunden.

**NOTPERMIT**

Aus den folgenden Gründen kam es zu einer Unzulässigkeitsbedingung:

**USRID**

Die Benutzer-ID, die dem Verarbeitungsthread zugeordnet ist, verfügt nicht über die erforderliche Sicherheitsberechtigung.

**SERVERGONE**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

**TABLEERROR**

Ein Ressourcentabellendatensatz ist aus dem folgenden Grund ungültig:

**DATAERROR**

Der Wert, der einem oder mehreren Ressourcentabellenattributen zugeordnet ist, ist ungültig. Dieser Fehler kann auftreten, wenn der Ressourcentabelle erforderliche Felder fehlen, sie ein oder mehrere miteinander in Konflikt stehende Felder enthält oder sie doppelt vorhanden ist. Bei BAS kann dieser Fehler auch auftreten, wenn Sie nicht über die erforderliche Sicherheitsberechtigung verfügen. Verwenden Sie den Befehl FEEDBACK, um weitere Informationen zu diesem Fehler abzurufen.

## VERSIONINVL

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

### ACTION

Die angegebene Aktion wird für die Version in Kombination mit dem Befehl CONNECT nicht unterstützt.

### NOTSUPPORTED

Die Version des Anwendungsstubprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

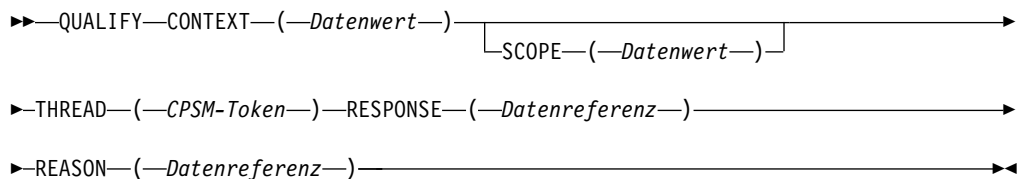
### NOTVSNCONN

Die Version des Anwendungsstubprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

---

## QUALIFY

Definieren des CICSplex SM-Kontexts und -Bereichs.



## Zugehörige Befehle

### CONNECT

## Beschreibung

Mit diesem Befehl werden der CICSplex SM-Kontext und -Bereich für nachfolgende Befehle definiert, die von einem API-Verarbeitungsthread ausgegeben werden.

## Optionen

### CONTEXT(*Datenwert*)

Gibt den Kontext für nachfolgende Befehle an, die für diesen Thread ausgegeben werden. Bei dem Kontext muss es sich um einen aus 1 bis 8 Zeichen bestehenden Namen eines CMAS oder CICSplex handeln.

Der angegebene Kontext bleibt für den Thread wirksam, bis Sie ihn überschreiben oder ihn in einem nachfolgenden Befehl ändern.

### REASON(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

### RESPONSE(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

### SCOPE(*Datenwert*)

Gibt den Bereich für nachfolgende Befehle an, die für diesen Thread ausgegeben werden.

Mit der Option SCOPE wird die Option CONTEXT qualifiziert. Wenn es sich bei dem Kontext um ein CICSplex handelt, kann für den Bereich Folgendes gelten:

- Es ist der aus 1 bis 8 Zeichen bestehende Name des CICSplex selbst
- Es ist ein CICS-System oder eine CICS-Systemgruppe innerhalb des CICSplex
- Es ist ein logischer Bereich wie er in einer CICSplex SM-Ressourcenbeschreibung (RESDESC) definiert wurde

Wenn es sich bei dem Kontext um einen CMAS handelt, ist diese Option bedeutungslos und wird ignoriert.

Der angegebene Bereich bleibt für den Thread wirksam, es sei denn, Sie überschreiben ihn für einen bestimmten Befehl oder ändern ihn durch Absetzen eines weiteren Befehls vom Typ QUALIFY. Wenn Sie die Option SCOPE nicht angeben, wird kein Bereichswert angenommen (d. h. der durch den Befehl CONNECT für den Thread eingerichtete Standardbereich wird nicht beibehalten).

**Anmerkung:** Für bestimmte API-Befehle ist ein gültiger Bereich erforderlich, wenn der Kontext ein CICSplex ist. Wenn Sie keinen Bereich im Befehl QUALIFY angeben, müssen Sie die Option SCOPE beim Ausgeben eines der folgenden Befehle für eine Ressourcentabelle angeben, die eine CICS-Ressource darstellt:

- GET
- PERFORM OBJECT
- PERFORM SET
- REFRESH
- SET.

#### **THREAD** (*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

## **Bedingungen**

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl QUALIFY zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

#### **ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

##### **NOSERVICE**

Das Anwendungsstubprogramm konnte das API-Servicemodul nicht laden.

##### **NOSTORAGE**

Das Anwendungsstubprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- CONTEXT
- SCOPE
- THREAD.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

**NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

**APITASK**

Die API-Steuersubtask ist nicht aktiv.

**CPSMAPI**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

**PLEXMGR**

Der CMAS, mit dem der Verarbeitungsthread derzeit verbunden ist, ist nicht an der Verwaltung des angegebenen CICSplex beteiligt und es steht kein anderer CMAS zur Verfügung, der das CICSplex verwaltet.

**SERVERGONE**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

**VERSIONINVL**

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

**NOTSUPPORTED**

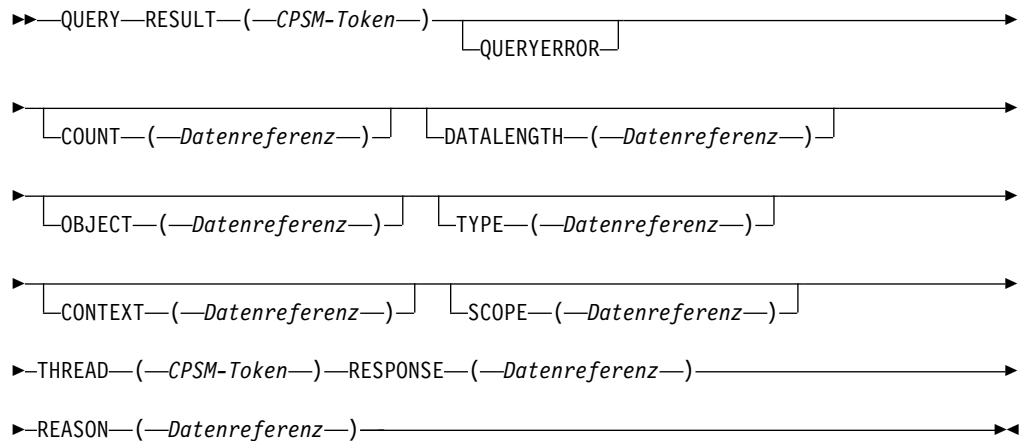
Die Version des Anwendungsstubprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

**NOTVSNCONN**

Die Version des Anwendungsstubprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

## QUERY

Abrufen von Informationen über eine Ergebnisliste und den darin enthaltenen Ressourcentabellendatensätzen.



### Beschreibung

Mit diesem Befehl werden Informationen über eine Ergebnisliste und die darin enthaltenen Ressourcentabellendatensätze abgerufen.

- Mit dem Befehl QUERY kann Folgendes festgestellt werden:
  - Kontext und Umfang der Ergebnisliste
  - Typ der Ressourcentabellendatensätze, die die Ergebnisliste enthält
  - Ob die Datensätze aus der CICSplex SM-Ressourcentabelle stammen oder aus einer benutzerdefinierten Ansicht dieser Tabelle
  - Anzahl der Ressourcentabellendatensätze in der Ergebnisliste
  - Länge der Ressourcentabellendatensätze
- Für Programme, die in REXX geschrieben werden, ist die Ausgabe des Befehls QUERY die einzige Möglichkeit, die Länge eines bestimmten Ressourcentabellendatensatzes zu bestimmen.

### Zugehörige Befehle

COPY, GET, GETDEF, GROUP, PERFORM OBJECT

### Optionen

#### CONTEXT(*Datenreferenz*)

Benennt eine Variable, um den Kontext zu empfangen, der der Ergebnisliste zugeordnet ist.

#### COUNT(*Datenreferenz*)

Benennt eine Variable, um die Anzahl der Ressourcentabellendatensätze in der Ergebnisliste zu empfangen.

#### DATALENGTH(*Datenreferenz*)

Benennt eine Variable, um die Länge der Ressourcentabellendatensätze in der Ergebnisliste zu empfangen.

**OBJECT** *(Datenreferenz)*

Benennt eine Variable, um den Namen der Ressourcentabelle zu empfangen, die der Ergebnisliste derzeit zugeordnet ist.

**Anmerkung:** Wenn QUERYERROR angegeben wird, ist das zurückgegebene Objekt (OBJECT) MASQRYER, nicht das Objekt oder die Ansicht in der Ergebnisliste.

**QUERYERROR**

Gibt an, dass diese Anforderung Informationen zu MASQRYER-Ressourcen zurückgeben soll, die durch den letzten Befehl des Typs GET, PERFORM oder SET generiert wurden, um mit der Ergebnisliste zu arbeiten.

**REASON** *(Datenreferenz)*

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

**RESPONSE** *(Datenreferenz)*

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**RESULT** *(CPSM-Token)*

Gibt die API-Ergebnisliste an, die von dieser Operation verarbeitet werden soll. Die Ergebnisliste kann durch einen der folgenden Befehle erzeugt werden:

- COPY
- GET
- GETDEF
- GROUP
- PERFORM OBJECT.

**SCOPE** *(Datenreferenz)*

Benennt eine Variable, um den Bereich zu empfangen, der der Ergebnisliste zugeordnet ist. Dieser Wert ist für Ergebnislisten, die CMAS-Typressourcen enthalten, möglicherweise leer.

**THREAD** *(CPSM-Token)*

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

**TYPE** *(Datenreferenz)*

Benennt eine Variable, um einen aus einem Zeichen bestehenden Wert zu erhalten, der angibt, welcher Typ von Datensätzen in der Ergebnisliste enthalten ist:

- |          |   |
|----------|---|
| <b>T</b> | Von CICSplex SM bereitgestellte Ressourcentabellen.   |
| <b>V</b> | Ansichten einer Ressourcentabelle, die durch einen Befehl des Typs SPECIFY VIEW erstellt wurden, der zuvor in diesem Verarbeitungsthread abgesetzt wurde. |

## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl QUERY zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

**BUSY** Aus den folgenden Gründen kam es zu einer Bedingung des Typs 'Aktiv':

**RESULT**

Die in der Option RESULT angegebene Ergebnisliste wird von einem anderen Befehl verarbeitet.

**ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

**NOSERVICE**

Das Anwendungsstubsprogramm konnte das API-Servicemodul nicht laden.

**NOSTORAGE**

Das Anwendungsstubsprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- CONTEXT
- DATALENGTH
- OBJECT
- RESULT
- THREAD
- TYPE.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

**NODATA**

Der Befehl hat Informationen zu den MASQRYER-Ressourcen angefordert, die vom letzten Befehl zum Verarbeiten der Ergebnisliste generiert wurden; der letzte Befehl wurde jedoch erfolgreich ausgeführt und es gibt keine MASQRYER-Ressourcen. Wenn COUNT angefordert wurde, wird ein Wert von null zurückgegeben.

**NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

**APITASK**

Die API-Steuersubtask ist nicht aktiv.

**CPSMAPI**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

## SERVERGONE

Der CMA5 (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

## VERSIONINVL

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

### NOTSUPPORTED

Die Version des Anwendungsstubprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

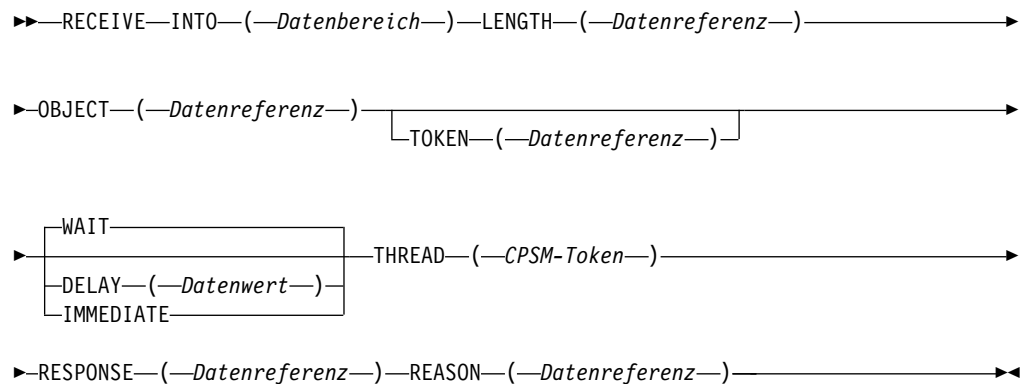
### NOTVSNCONN

Die Version des Anwendungsstubprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

---

## RECEIVE

Empfangen der Ausgabe aus abgeschlossenen asynchronen Anforderungen.



## Beschreibung

Durch diesen Befehl wird die Ausgabe aus abgeschlossenen asynchronen Anforderungen empfangen, die dem Verarbeitungsthread zugeordnet sind.

- Zu einer asynchronen Ausgabe kann es kommen, wenn Sie zuvor entweder einen Befehl des Typs LISTEN oder einen der folgenden Befehle mit der Option NOWAIT ausgegeben haben:
  - GET
  - PERFORM OBJECT
  - PERFORM SET
  - REFRESH
  - SET.
- Um festzustellen, ob eine asynchrone Ausgabe für den Empfang bereitsteht, setzen Sie den Befehl ADDRESS ab und prüfen Sie den Wert SENTINEL, bevor Sie den Befehl RECEIVE absetzen.
- Ein API-Verarbeitungsthread kann maximal 256 abgeschlossene asynchrone Anforderungen aufweisen, die gleichzeitig ausstehend sind. Wenn Sie den Befehl RECEIVE nicht in regelmäßigen Intervallen ausgeben und Ihr Verarbeitungsthread sein Maximum von 256 erreicht, werden asynchrone Anforderungen ver-



worfen und nicht verarbeitet. Eine vollständige Beschreibung der asynchronen Verarbeitung finden Sie in Developing CICSplex SM applications.

## Zugehörige Befehle

ADDRESS, GET, LISTEN, PERFORM OBJECT, PERFORM SET, REFRESH, SET

## Optionen

### **DELAY**(*Datenwert*)

Gibt die Anzahl der Sekunden an, die die Verarbeitung wartet, wenn beim Absetzen des Befehls RECEIVE keine Ausgabe verfügbar ist. Ist die angegebene Anzahl von Sekunden abgelaufen, wird die Steuerung an den Verarbeitungsthread zurückgegeben, unabhängig davon, ob eine Ausgabe verfügbar wird oder nicht. Wenn die Ausgabe während der Verzögerungszeit verfügbar wird, wird die Steuerung an den Verarbeitungsthread zurückgegeben. Wenn die Ausgabe sofort verfügbar ist, gibt es keine Verzögerung; die Steuerung wird sofort an den Verarbeitungsthread zurückgegeben.

DELAY muss einen Wert ungleich null angeben. Wenn Sie sicherstellen wollen, dass Ihr Programm niemals in einen Wartestatus gerät, verwenden Sie die Option IMMEDIATE anstelle von DELAY.

### **IMMEDIATE**

Gibt die Steuerung sofort an den Verarbeitungsthread zurück, unabhängig davon, ob eine Ausgabe verfügbar ist oder nicht.

### **INTO**(*Datenbereich*)

Gibt einen Puffer zum Empfang von asynchroner Ausgabe an, sofern für diesen Thread vorhanden. Dieser Puffer muss lang genug sein, um die gesamte empfangene Ausgabe aufnehmen zu können.

Bei der zurückgegebenen Ausgabe kann es sich um Folgendes handeln:

- Einen Ressourcentabellendatensatz, der ein Ereignis darstellt, das in einem früheren Befehl des Typs LISTEN benannt wurde
- Einen ASYNCREQ-Ressourcentabellendatensatz, der die Beendigung einer asynchronen GET-, PERFORM-, REFRESH- oder SET-Anforderung darstellt

### **LENGTH**(*Datenreferenz*)

Ein Vollwortwert, der die Länge des INTO-Puffers angibt.

Nachdem die Operation abgeschlossen ist, wird dieses Feld auf die tatsächliche Länge der Daten gesetzt, die im INTO-Puffer zurückgegeben werden. Kann die Operation nicht abgeschlossen werden, weil der Puffer nicht lang genug ist, wird dieses Feld auf die erforderliche Länge gesetzt.

### **OBJECT**(*Datenreferenz*)

Benennt eine Variable, um einen Ressourcentabellennamen zu empfangen, wenn die Ausgabe für diesen Thread verfügbar ist.

### **REASON**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

### **RESPONSE**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**THREAD**(*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

**WAIT**

Wartet, bis die asynchrone Ausgabe verfügbar wird, bevor die Steuerung an den Verarbeitungsthread zurückgegeben wird.

**Anmerkung:** Die Option WAIT wartet unbegrenzt auf eine asynchrone Ausgabe. Stellen Sie sicher, dass abgeschlossene asynchrone Anforderungen ausstehen; setzen Sie hierfür den Befehl ADDRESS ab, bevor Sie den Befehl RECEIVE absetzen.

**TOKEN**(*Datenreferenz*)

Benennt eine Variable, um das benutzerdefiniert Token zu empfangen, das der asynchronen Ausgabe zugeordnet ist. Dieser Wert ist das Token, das Sie für den Befehl GET, LISTEN, PERFORM, REFRESH oder SET definiert haben, durch den die Ausgabe erzeugt wurde.

**Bedingungen**

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl RECEIVE zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

**NODATA**

Es waren keine Daten zum Empfangen vorhanden.

**WARNING**

Der Befehl hat die Verarbeitung aus folgendem Grund mit einer Warnung beendet:

**AREATOOSMALL**

Der INTO-Puffer reicht nicht mehr aus, um die Anzahl der angeforderten und verfügbaren Datensätze aufzunehmen.

**ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

**NOSERVICE**

Das Anwendungsstubsprogramm konnte das API-Servicemodul nicht laden.

**NOSTORAGE**

Das Anwendungsstubsprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

### EXCEPTION

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

### INVALIDPARM

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- DELAY
- INTO
- LENGTH
- OBJECT
- THREAD
- TOKEN.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

### NOTAVAILABLE

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

#### APITASK

Die API-Steuersubtask ist nicht aktiv.

#### CPSMAPI

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

### SERVERGONE

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

### VERSIONINVL

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

#### NOTSUPPORTED

Die Version des Anwendungssubprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

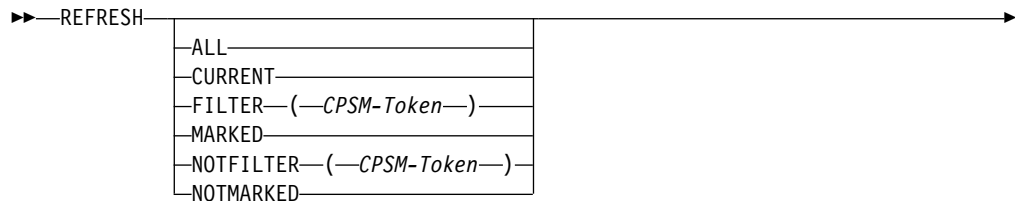
#### NOTVSNNCONN

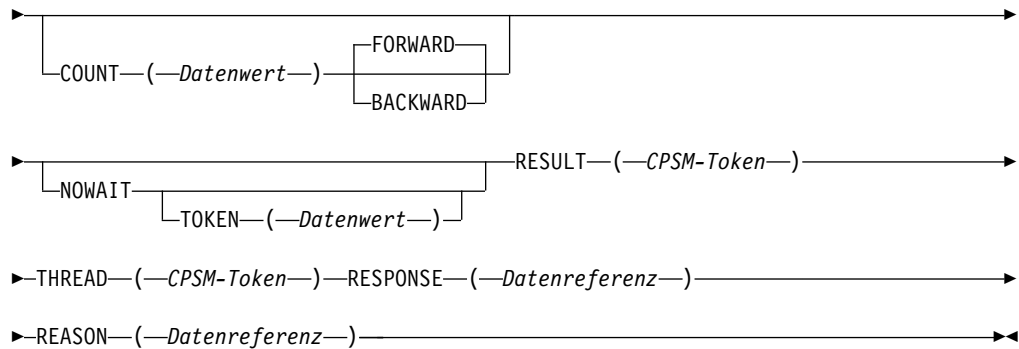
Die Version des Anwendungssubprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

---

## REFRESH

Aktualisieren von Daten für Ressourcentabellendatensätze.





## Beschreibung

- Durch diesen Befehl werden die Daten für einige oder für alle Ressourcentabellendatensätze in einer Ergebnisliste aktualisiert.
- Für die MAS-Ressourcentabelle stellt REFRESH nur dann Daten bereit, wenn MAS bei der letzten Erstellung der Ergebnisliste aktiv war.

## Zugehörige Befehle

COPY, GET, LOCATE, MARK, PERFORM OBJECT, SPECIFY FILTER

## Optionen

### ALL

Aktualisiert alle Ressourcentabellendatensätze in der Ergebnisliste. Bei der Angabe der Option ALL gilt Folgendes:

- Die Option COUNT wird ignoriert.
- Alle Datensätze, die gelöscht wurden, werden aus der Ergebnisliste entfernt. Alle Positionen, die zuvor von gelöschten Datensätzen belegt waren, werden gefüllt und die verbleibenden Datensätze werden neu nummeriert. Aus diesem Grund kann die relative Position eines bestimmten Datensatzes in einer Ergebnisliste nach einer Aktualisierung unterschiedlich sein.

### BACKWARD

Aktualisiert den vorherigen Ressourcentabellendatensatz und setzt die Verarbeitung durch die Ergebnisliste in Rückwärtsrichtung fort und es werden so viele Datensätze aktualisiert wie durch die Option COUNT angegeben wird.

**Anmerkung:** Erreicht der Datensatzzeiger das Ende der Ergebnisliste, wird der aktuelle Datensatz (der der letzte Datensatz ist) mit BACKWARD aktualisiert; anschließend wird mit den vorherigen Datensätzen fortgefahren.

### COUNT (Datenwert)

Gibt die Anzahl der zu aktualisierenden Ressourcentabellendatensätze an. Wenn Sie die Option COUNT nicht angeben, wird nur ein einziger Datensatz aktualisiert.

Wenn Sie die Option FORWARD oder BACKWARD nicht angeben, wird der Aktualisierungsprozess in Vorwärtsrichtung durch die Ergebnisliste durchgeführt.

### CURRENT

Aktualisiert nur den aktuellen Ressourcentabellendatensatz. Wenn Sie CURRENT angeben, wird die Option COUNT ignoriert.

**FILTER**(*CPSM-Token*)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option FILTER gibt an, dass nur die Ressourcentabellendatensätze, die den angegebenen Filterkriterien entsprechen, bei der Aktualisierung berücksichtigt werden sollen.

Die Anzahl der aktualisierten Datensätze wird durch die Option COUNT bestimmt. Wenn Sie die Option COUNT nicht angeben, wird nur der erste Datensatz aktualisiert, der die Filterkriterien erfüllt.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

**FORWARD**

Aktualisiert den aktuellen Ressourcentabellendatensatz und setzt die Verarbeitung durch die Ergebnisliste in Vorwärtsrichtung fort und es werden so viele Datensätze aktualisiert wie durch die Option COUNT angegeben wird.

**MARKED**

Gibt an, dass nur die Ressourcentabellendatensätze, die in der Ergebnisliste markiert sind, bei der Aktualisierung berücksichtigt werden sollen.

Die Anzahl der aktualisierten Datensätze wird durch die Option COUNT bestimmt. Wenn Sie die Option COUNT nicht angeben, wird nur der erste Datensatz aktualisiert, der markiert ist.

Ressourcentabellendatensätze können mit den Befehlen MARK und UNMARK markiert werden.

**NOTFILTER**(*CPSM-Token*)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option NOTFILTER gibt an, dass nur die Ressourcentabellendatensätze, die den angegebenen Filterkriterien nicht entsprechen, bei der Aktualisierung berücksichtigt werden sollen.

Die Anzahl der aktualisierten Datensätze wird durch die Option COUNT bestimmt. Wenn Sie die Option COUNT nicht angeben, wird nur der erste Datensatz aktualisiert, der die Filterkriterien nicht erfüllt.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

**NOTMARKED**

Gibt an, dass nur die Ressourcentabellendatensätze, die in der Ergebnisliste nicht markiert sind, bei der Aktualisierung berücksichtigt werden sollen.

Die Anzahl der aktualisierten Datensätze wird durch die Option COUNT bestimmt. Wenn Sie die Option COUNT nicht angeben, wird nur der erste Datensatz aktualisiert, der nicht markiert ist.

Ressourcentabellendatensätze können mit den Befehlen MARK und UNMARK markiert werden.

**NOWAIT**

Gibt die Steuerung an Ihr Programm zurück, sobald der Befehl REFRESH akzeptiert wurde; dadurch kann der Befehl asynchron verarbeitet werden.

Wenn Sie die Option NOWAIT angeben, müssen Sie einen nachfolgenden Befehl des Typs RECEIVE verwenden, um zu testen, ob diese Anforderung ausgeführt wurde. Die Ergebnisse einer asynchronen Anforderung werden als Ressourcentabellendatensätze des Typs ASYNCREQ zurückgegeben. Eine vollständige Beschreibung der asynchronen Verarbeitung finden Sie in Developing CICSplex SM applications.

**Anmerkung:** Wenn Sie die Option TOKEN angeben, wird die Option NO-WAIT standardmäßig angenommen.

**REASON**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

**RESPONSE**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**RESULT**(*CPSM-Token*)

Gibt die API-Ergebnisliste an, die von dieser Operation verarbeitet werden soll. Die Ergebnisliste kann durch einen der folgenden Befehle erzeugt werden:

- COPY
- GET
- PERFORM OBJECT.

**THREAD**(*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

**TOKEN**(*Datenwert*)

Definiert ein aus 1 bis 4 Zeichen bestehendes Token, das Sie für die Korrelation einer asynchronen Anforderung des Typs REFRESH mit dem Ergebnis eines nachfolgenden Befehls des Typs RECEIVE auswählen. Dieses Token dient zur Verwendung durch Ihr Programm; CICSplex SM verwendet den Wert nicht. Das Token wird durch den Befehl RECEIVE zurückgegeben, wenn diese REFRESH-Anforderung abgeschlossen ist.

## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl REFRESH zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

**SCHEDULED**

Der Befehl wurde für die Verarbeitung geplant.

**BUSY** Aus den folgenden Gründen kam es zu einer Bedingung des Typs 'Aktiv':

**RESULT**

Die in der Option RESULT angegebene Ergebnisliste wird von einem anderen Befehl verarbeitet.

**ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

**NOSERVICE**

Das Anwendungsstubprogramm konnte das API-Servicemodul nicht laden.

**NOSTORAGE**

Das Anwendungsstubprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**NOTPROCESSED**

Einer der MASs, an den die Anforderung übertragen wurde, konnte die Anforderung nicht verarbeiten.

**REQTIMEOUT**

Einer der CMASs oder MASs, an die die Anforderung übertragen wurde, antwortete nicht.

**SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- COUNT
- FILTER
- NOTFILTER
- RESULT
- THREAD
- TOKEN.

Überprüfen Sie die Befehlsbeschreibung auf eine gültige Parametersyntax hin.

**NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

**APITASK**

Die API-Steuersubtask ist nicht aktiv.

**CMAS**

Ein CMAS (CICSplex SM-Adressraum), an den die Anforderung übertragen wurde, ist nicht verfügbar.

**CPSMAPI**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

**MAINTPOINT**

Der Wartungspunkt für den aktuellen Kontext ist nicht verfügbar.

**RESOURCE**

Der Ressourcentabellentyp für die Ergebnisliste, die aktualisiert wird, wird von diesem Befehl nicht unterstützt. Um die Ergebnisliste zu aktualisieren, führen Sie den Befehl GET erneut aus, mit dem sie ursprünglich angefordert wurde. Derzeit sind Ergebnislis-

ten für die Ressourcentabellen EVCSPEC, HTASK, OSGIBUND, OSGISERV, SYSPARM und RULE nicht aktualisierbar.

## SCOPE

Mindestens einer der in dem Bereich angegebenen MASs ist nicht verfügbar. Es wird ein Ressourcentabellendatensatz des Typs MASQRYER erstellt, um alle MAS zu identifizieren, die nicht auf die Anforderung reagiert haben.

**NOTPERMIT**

Aus den folgenden Gründen kam es zu einer Unzulässigkeitsbedingung:

## USRID

Die Benutzer-ID, die dem Verarbeitungsthread zugeordnet ist, verfügt nicht über die erforderliche Sicherheitsberechtigung.

## SERVERGONE

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

## TABLEERROR

Ein Ressourcentabellendatensatz ist aus dem folgenden Grund ungültig:

## DATAERROR

Der Wert, der einem oder mehreren Ressourcentabellenattributen zugeordnet ist, ist ungültig. Dieser Fehler kann auftreten, wenn der Ressourcentabelle erforderliche Attribute fehlen, sie ein oder mehrere miteinander in Konflikt stehende Attribute enthält oder sie nicht vorhanden ist. Verwenden Sie den Befehl `FEEDBACK`, um weitere Informationen zu diesem Fehler abzurufen.

## VERSIONINVL

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

## NOTSUPPORTED

Die Version des Anwendungstreibers, das für diesen Befehl verwendet wird, wird nicht unterstützt.

## NOTVSNCONN

Die Version des Anwendungsstubsprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

**REMOVE**

### Entfernen einer CICSplex SM- oder CICS-Definition aus dem Datenrepository.

```

▶▶ REMOVE OBJECT (—Datenwert—) FROM (—Datenbereich—)
▶ LENGTH (—Datenwert—)
▶ [ PARM (—Datenbereich—) PARMLen (—Datenwert—) ]
▶ [ CONTEXT (—Datenwert—) ] [ SCOPE (—Datenwert—) ]
▶▶ THREAD (—CPSM-Token—) RESPONSE (—Datenreferenz—)

```



## Beschreibung

Durch diesen Befehl wird eine CICSplex SM- oder CICS-Definition aus dem Datenrepository entfernt. Für Definitionen mit einem CICSplex als Kontext (z. B. Workload-Management- oder Echtzeitanalysedefinitionen) wird die Definition auch aus den Datenrepositorys aller an der Verwaltung des CICSplex beteiligten CMASs entfernt.

## Zugehörige Befehle

CREATE, UPDATE

## Optionen

### CONTEXT (Datenwert)

Gibt den Kontext für diesen Befehl an. Bei dem Kontext muss es sich um einen aus 1 bis 8 Zeichen bestehenden Namen eines CMAS oder CICSplex handeln.

Wenn Sie die Option CONTEXT nicht angeben, wird der Standardkontext für den Thread angenommen.

### FROM (Datenbereich)

Gibt einen Puffer an, der einen Ressourcentabellendatensatz enthält, der die zu entfernenden Definitionen darstellt. Der Datensatz muss alle Attribute für die Ressourcentabelle enthalten, die in der Option OBJECT angegeben ist.

### LENGTH (Datenwert)

Ein Vollwortwert, der die Länge des FROM-Puffers angibt.

### OBJECT (Datenwert)

Gibt die Ressourcentabelle an, die die zu entfernende Definition darstellt. Bei diesem Wert muss es sich um einen aus 1 bis 8 Zeichen bestehenden Namen einer gültigen Ressourcentabelle einer CICSplex SM-Definition oder CICS-Definition handeln. Eine Liste der CICSplex SM-Ressourcentabellen nach Typ finden Sie in *Developing CICSplex SM applications*.

### PARM (Datenbereich)

Gibt einen Puffer mit dem Parameterausdruck an, der zum Entfernen der Definition verwendet werden soll.

Details zur Verwendung eines Parameterausdrucks mit dem Befehl REMOVE finden Sie in *Developing CICSplex SM applications*. Eine Beschreibung der Parameter, die für eine angegebene Ressourcentabelle gültig sind, finden Sie im Abschnitt zu den CICSplex SM-Ressourcentabellen in den Referenzinformationen.

### PARMLN (Datenwert)

Ein Vollwortwert, der die Länge des PARM-Puffers angibt.

### REASON (Datenreferenz)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

### RESPONSE (Datenreferenz)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**SCOPE** (*Datenwert*)

Gibt den Bereich für diesen Befehl an.

Wenn der aktuelle Kontext (definiert durch diesen Befehl oder einen früheren Befehl des Typs CONNECT oder QUALIFY) ein CICSplex ist und die Option OBJECT eine CICS-Definitionsressource angibt und die Option PARM den Parameter CSD beinhaltet, kann ein gültiger Bereich angegeben werden.

Der Bereich kann ein CICS-System innerhalb des CICSplex sein. Wenn es sich bei dem aktuellen Kontext um einen CMAS handelt oder die Option OBJECT einen anderen Typ von Ressourcentabelle angibt oder der Parameter CSD in keiner CICS-Definitionsressource angegeben ist, ist diese Option bedeutungslos und wird ignoriert.

Wenn SCOPE für den Befehl gültig ist und Sie die Option SCOPE nicht angeben, wird der Standardbereich für den Thread angenommen. Wenn der aktuelle Kontext ein CICSplex ist und kein Standardbereich für den Befehl CONNECT oder QUALIFY festgelegt wurde, erhalten Sie eine Antwort des Typs INVALIDPARM für die Option SCOPE.

**THREAD** (*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

**Bedingungen**

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl REMOVE zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

**ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

**NOSERVICE**

Das Anwendungsstubprogramm konnte das API-Servicemodul nicht laden.

**NOSTORAGE**

Das Anwendungsstubprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**NOTPROCESSED**

Einer der MASSs, an den die Anforderung übertragen wurde, konnte die Anforderung nicht verarbeiten (nur bei CSD-Anforderungen).

**REQTIMEOUT**

Einer der CMASs, an die die Anforderung übertragen wurde, antwortete nicht.

**SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**INVALIDCMD**

Aus dem folgenden Grund ist der Befehl ungültig:

**LENGTH**

Die Gesamtlänge aller Optionen im Befehl überschreitet den maximalen Grenzwert.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- CONTEXT
- FROM
- LENGTH
- OBJECT
- PARM
- PARMLen
- THREAD.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

**NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

**APITASK**

Die API-Steuersubtask ist nicht aktiv.

**CMAS**

Ein CMAS (CICSplex SM-Adressraum), an den die Anforderung übertragen wurde, ist nicht verfügbar.

**CPSMAPI**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

**CSDAPI**

Es ist keine Unterstützung für die CICSplex SM-API für den Zugriff auf CICS CSD verfügbar.

**MAINTPOINT**

Der Wartungspunkt für den aktuellen Kontext ist nicht verfügbar.

**NOTPERMIT**

Aus den folgenden Gründen kam es zu einer Unzulässigkeitsbedingung:

**USRID**

Die Benutzer-ID, die dem Verarbeitungsthread zugeordnet ist, verfügt nicht über die erforderliche Sicherheitsberechtigung.

**SERVERGONE**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

## TABLEERROR

Ein Ressourcentabellendatensatz ist aus einem der folgenden Gründe ungültig:

### DATAERROR

Der Wert, der einem oder mehreren Ressourcentabellenattributen zugeordnet ist, ist ungültig. Dieser Fehler kann auftreten, wenn der Ressourcentabelle erforderliche Attribute fehlen, sie ein oder mehrere miteinander in Konflikt stehende Attribute enthält oder sie nicht vorhanden ist. Verwenden Sie den Befehl FEEDBACK, um weitere Informationen zu diesem Fehler abzurufen.

### INVALIDATTR

Eines der Ressourcentabellenattribute ist ungültig.

### INVALIDVER

Die angegebene Version der Ressourcentabelle wird von CICSplex SM nicht unterstützt.

## VERSIONINVL

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

### NOTSUPPORTED

Die Version des Anwendungsstubsprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

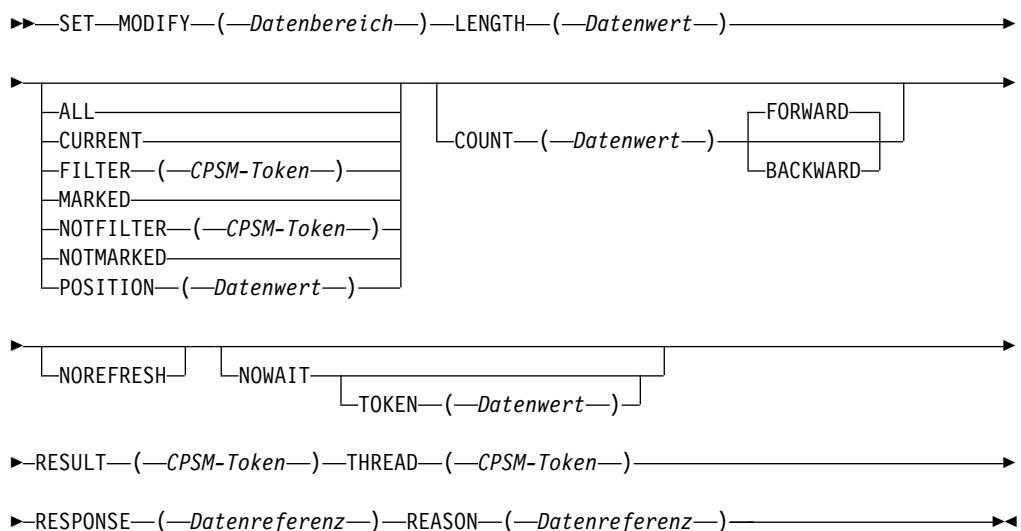
### NOTVSNCONN

Die Version des Anwendungsstubsprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

---

## SET

Ändern der Attribute einer oder mehrerer Ressourcen.



## Beschreibung

Mit diesem Befehl werden die Attribute einer oder mehrerer Ressourcen modifiziert, die durch Ressourcentabellendatensätze in einer vorhandenen Ergebnisliste dargestellt werden.

- Der Befehl SET ist nur für CICS- und einige CICSplex SM-Ressourcentabellen gültig.
- Wenn der gültige Kontext und Bereich beim Absetzen des Befehls SET CICS-Systeme umfasst, die die angeforderte Modifikation nicht unterstützen, wird die Anforderung für diese CICS-Systeme ignoriert.

## Zugehörige Befehle

COPY, GET, GROUP, LOCATE, MARK, PERFORM OBJECT, PERFORM SET, SPECIFY FILTER

## Optionen

### ALL

Modifiziert alle Ressourcentabellendatensätze in der Ergebnisliste. Wenn Sie ALL angeben, wird die Option COUNT ignoriert.

### BACKWARD

Modifiziert den vorherigen Ressourcentabellendatensatz und setzt die Verarbeitung durch die Ergebnisliste in Rückwärtsrichtung fort und es werden so viele Datensätze modifiziert wie durch die Option COUNT angegeben wird.

**Anmerkung:** Erreicht der Datensatzzeiger das Ende der Ergebnisliste, wird der aktuelle Datensatz (der der letzte Datensatz ist) mit BACKWARD modifiziert; anschließend wird mit den vorherigen Datensätzen fortgefahren.

### COUNT (*Datenwert*)

Gibt die Anzahl der zu modifizierenden Ressourcentabellendatensätze an. Wenn Sie die Option COUNT nicht angeben, wird nur ein einziger Datensatz aktualisiert.

Wenn Sie die Option FORWARD oder BACKWARD nicht angeben, wird der Modifikationsprozess in Vorwärtsrichtung durch die Ergebnisliste durchgeführt.

### CURRENT

Modifiziert nur den aktuellen Ressourcentabellendatensatz. Wenn Sie CURRENT angeben, wird die Option COUNT ignoriert.

### FILTER (*CPSM-Token*)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option FILTER gibt an, dass nur die Ressourcentabellendatensätze, die den angegebenen Filterkriterien entsprechen, beim Modifizieren berücksichtigt werden sollen.

Die Anzahl der modifizierten Datensätze wird durch die Option COUNT bestimmt. Wenn Sie die Option COUNT nicht angeben, wird nur der erste Datensatz modifiziert, der die Filterkriterien erfüllt.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

### FORWARD

Modifiziert den aktuellen Ressourcentabellendatensatz und setzt die Verarbeitung durch die Ergebnisliste in Vorwärtsrichtung fort und es werden so viele Datensätze modifiziert wie durch die Option COUNT angegeben wird.

### LENGTH (*Datenwert*)

Ein Vollwortwert, der die Länge des MODIFY-Puffers angibt.

**Anmerkung:** Die von Ihnen angegebene Pufferlänge darf keine anderen Daten als einen Modifikationsausdruck enthalten.

#### **MARKED**

Gibt an, dass nur die Ressourcentabellendatensätze, die in der Ergebnisliste markiert sind, beim Modifizieren berücksichtigt werden sollen.

Die Anzahl der modifizierten Datensätze wird durch die Option COUNT bestimmt. Wenn Sie die Option COUNT nicht angeben, wird nur der erste Datensatz modifiziert, der markiert ist.

Ressourcentabellendatensätze können mit den Befehlen MARK und UNMARK markiert werden.

#### **MODIFY**(*Datenbereich*)

Gibt einen Puffer mit dem Modifikationsausdruck an, der für die Modifikation der Ressourcentabellendatensätze verwendet werden soll.

Weitere Informationen zum Bilden eines Modifikationsausdrucks finden Sie in Developing CICSplex SM applications.

#### **NOREFRESH**

Gibt an, dass die Ressourcentabellendatensätze in der Quellenergebnisliste nicht aktualisiert werden dürfen. Die Datensätze spiegeln den Status der Ressourcen vor der Verarbeitung des Befehls SET wider.

Wenn Sie die Option NOREFRESH nicht angeben, werden die Ressourcentabellendatensätze aktualisiert, um den Ressourcenstatus nach Abschluss dieser Operation widerzuspiegeln.

#### **NOTFILTER**(*CPSM-Token*)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option NOTFILTER gibt an, dass nur die Ressourcentabellendatensätze, die den angegebenen Filterkriterien nicht entsprechen, beim Modifizieren berücksichtigt werden sollen.

Die Anzahl der modifizierten Datensätze wird durch die Option COUNT bestimmt. Wenn Sie die Option COUNT nicht angeben, wird nur der erste Datensatz modifiziert, der die Filterkriterien nicht erfüllt.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

#### **NOTMARKED**

Gibt an, dass nur die Ressourcentabellendatensätze, die in der Ergebnisliste nicht markiert sind, beim Modifizieren berücksichtigt werden sollen.

Die Anzahl der modifizierten Datensätze wird durch die Option COUNT bestimmt. Wenn Sie die Option COUNT nicht angeben, wird nur der erste Datensatz modifiziert, der nicht markiert ist.

Ressourcentabellendatensätze können mit den Befehlen MARK und UNMARK markiert werden.

#### **NOWAIT**

Gibt die Steuerung an Ihr Programm zurück, sobald der Befehl SET akzeptiert wurde; dadurch kann der Befehl asynchron verarbeitet werden.

Wenn Sie die Option NOWAIT angeben, müssen Sie einen nachfolgenden Befehl des Typs RECEIVE verwenden, um zu testen, ob diese Anforderung ausgeführt wurde. Die Ergebnisse einer asynchronen Anforderung werden als Res-

sourcentabellendatensätze des Typs ASYNCREQ zurückgegeben. Eine vollständige Beschreibung der asynchronen Verarbeitung finden Sie in Developing CICSplex SM applications.

**Anmerkung:** Wenn Sie die Option TOKEN angeben, wird die Option NO-WAIT standardmäßig angenommen.

**POSITION**(*Datenwert*)

Ändert den n-ten Ressourcentabellendatensatz in der Ergebnisliste. Wenn Sie POSITION angeben, wird die Option COUNT ignoriert.

Dieser Wert muss eine Zahl sein, die die relative Position des Datensatzes in der Ergebnisliste angibt. Der erste Datensatz in einer Ergebnisliste wird durch die Zahl 1 angegeben.

Wenn Sie beispielsweise den fünften Ressourcentabellendatensatz in einer Ergebnisliste modifizieren möchten, geben Sie POSITION(5) an.

**REASON**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

**RESPONSE**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**RESULT**(*CPSM-Token*)

Gibt die API-Ergebnisliste an, die von dieser Operation verarbeitet werden soll. Die Ergebnisliste kann durch einen der folgenden Befehle erzeugt werden:

- COPY
- GET
- GROUP
- PERFORM OBJECT.

**THREAD**(*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

**TOKEN**(*Datenwert*)

Definiert ein aus 1 bis 4 Zeichen bestehendes Token, das Sie für die Korrelation einer asynchronen Anforderung des Typs SET mit dem Ergebnis eines nachfolgenden Befehls des Typs RECEIVE auswählen. Dieses Token dient zur Verwendung durch Ihr Programm; CICSplex SM verwendet den Wert nicht. Das Token wird durch den Befehl RECEIVE zurückgegeben, wenn diese SET-Anforderung abgeschlossen ist.

## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl SET zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

**SCHEDULED**

Der Befehl wurde für die Verarbeitung geplant.

**BUSY** Aus den folgenden Gründen kam es zu einer Bedingung des Typs 'Aktiv':

**RESULT**

Die in der Option RESULT angegebene Ergebnisliste wird von einem anderen Befehl verarbeitet.

**ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

**NOSERVICE**

Das Anwendungsstubsprogramm konnte das API-Servicemodul nicht laden.

**NOSTORAGE**

Das Anwendungsstubsprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**NOTPROCESSED**

Einer der MASs, an den die Anforderung übertragen wurde, konnte die Anforderung nicht verarbeiten.

**REQTIMEOUT**

Einer der CMASs oder MASs, an die die Anforderung übertragen wurde, antwortete nicht.

**SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**INVALIDDATA**

Aus einem der folgenden Gründe kam es zu einem Fehler über ungültige Daten:

**MODIFY**

Ein Attributwert, der im MODIFY-Puffer aufgelistet ist, ist für das angegebene Attribut nicht gültig.

**NOTSUPPORTED**

Ein Attribut, das im MODIFY-Puffer aufgeführt ist, kann nicht geändert werden.

**INVALIDCMD**

Aus dem folgenden Grund ist der Befehl ungültig:

**LENGTH**

Die Gesamtlänge aller Optionen im Befehl überschreitet den maximalen Grenzwert.

**INVALIDPARM**

In der Befehlsfolge oder im MODIFY-Puffer wurde ein ungültiger Parameter festgestellt. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- ATTRIBUTE



- COUNT
- FILTER
- LENGTH
- MODIFY
- NOTFILTER
- POSITION
- RESULT
- THREAD
- TOKEN.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

#### **NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

##### **APITASK**

Die API-Steuersubtask ist nicht aktiv.

##### **CMAS**

Ein CMAS (CICSplex SM-Adressraum), an den die Anforderung übertragen wurde, ist nicht verfügbar.

##### **CPSMAPI**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

##### **MAINTPOINT**

Der Wartungspunkt für den aktuellen Kontext ist nicht verfügbar.

##### **SCOPE**

Entweder ist keiner der MASs im angegebenen Bereich verfügbar, oder keiner von ihnen unterstützt die angeforderte Modifikation.

#### **NOTFOUND**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht gefunden':

##### **ACTION**

Eine Aktion, die im MODIFY-Puffer angefordert wurde, wurde für die angegebene Ressourcentabelle nicht gefunden.

##### **ATTRIBUTE**

Ein Attribut, das im MODIFY-Puffer angegeben wurde, wurde für die angegebene Ressourcentabelle nicht gefunden.

#### **NOTPERMIT**

Aus den folgenden Gründen kam es zu einer Unzulässigkeitsbedingung:

##### **USRID**

Die Benutzer-ID, die dem Verarbeitungsthread zugeordnet ist, verfügt nicht über die erforderliche Sicherheitsberechtigung.

#### **SERVERGONE**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

#### **TABLEERROR**

Ein Ressourcentabellendatensatz ist aus einem der folgenden Gründe ungültig:

#### DATAERROR

Der Wert, der einem oder mehreren Ressourcentabellenattributen zugeordnet ist, ist ungültig. Dieser Fehler kann auftreten, wenn der Ressourcentabelle erforderliche Attribute fehlen, sie ein oder mehrere miteinander in Konflikt stehende Attribute enthält oder sie doppelt vorhanden ist. Verwenden Sie den Befehl FEEDBACK, um weitere Informationen zu diesem Fehler abzurufen.

#### INVALIDVER

Die angegebene Version der Ressourcentabelle wird von CICSplex SM nicht unterstützt.

#### VERSIONINVL

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

##### NOTSUPPORTED

Die Version des Anwendungsstubsprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

##### NOTVSNNCONN

Die Version des Anwendungsstubsprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

---

## SPECIFY FILTER

Definieren eines Attribut- oder Wertfilters und Zuweisen eines identifizierenden Tokens zu diesem Filter.

```
►►—SPECIFY—FILTER—(—Datenreferenz—)—CRITERIA—(—Datenbereich—)—►►
►►—LENGTH—(—Datenwert—)—OBJECT—(—Datenwert—)—THREAD—(—CPSM-Token—)—►►
►►—RESPONSE—(—Datenreferenz—)—REASON—(—Datenreferenz—)—►►
```

### Beschreibung

Durch diesen Befehl wird ein Attribut- oder Wertfilter definiert und ihm ein identifizierendes Token zugeordnet.

- Filter werden dem jeweiligen Verarbeitungsthread zugeordnet, bei dem sie definiert sind; eine gemeinsame Nutzung durch andere Verarbeitungsthreads ist nicht möglich.
- Sie können mehrere Filter zur Verwendung durch einen Verarbeitungsthread definieren; CICSplex SM weist jedem ein eindeutiges identifizierendes Token zu.
- Wenn ein Verarbeitungsthread beendet wird, werden alle von ihm definierten Filter gelöscht.

### Zugehörige Befehle

COPY, DELETE, DISCARD, FETCH, GET, GROUP, LISTEN, LOCATE, MARK, PERFORM OBJECT, PERFORM SET, REFRESH, SET, UNMARK

### Optionen

#### CRITERIA(*Datenbereich*)

Gibt einen Puffer mit dem für diese Operation zu verwendenden Filteraus-

druck an. Die Option CRITERIA filtert nur die Ressourcentabellendatensätze, die den angegebenen Kriterien entsprechen.

Weitere Informationen zum Bilden eines Filterausdrucks finden Sie in Developing CICSplex SM applications.

**Anmerkung:** Das Attribut EYU\_CICSNAME oder EYU\_CICSREL kann in einem Filterausdruck nicht angegeben werden.

**FILTER**(*Datenreferenz*)

Benennt eine Variable, um das Token zu empfangen, das CICSplex SM diesem Filter zuordnet.

Dieses identifizierende Token muss in allen nachfolgenden Befehlen angegeben werden, die diesen Filter verwenden.

**LENGTH**(*Datenwert*)

Ein Vollwortwert, der die Länge des CRITERIA-Puffers angibt.

**Anmerkung:** Die von Ihnen angegebene Pufferlänge darf keine anderen Daten als einen Filterausdruck enthalten.

**OBJECT**(*Datenwert*)

Gibt die Ressourcentabelle an, für die ein Filter erstellt wird. Bei diesem Wert muss es sich um einen aus 8 Zeichen bestehenden Namen einer gültigen Ressourcentabelle handeln.

**REASON**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

**RESPONSE**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**THREAD**(*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl SPECIFY FILTER zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

**ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

**NOSERVICE**

Das Anwendungsstubprogramm konnte das API-Servicemodul nicht laden.

**NOSTORAGE**

Das Anwendungsstubprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**INVALIDDATA**

Es wurden ungültige Daten gefunden. Der Parameter, der ungültige Daten enthält, wird als Ursachenwert zurückgegeben:

**CRITERIA**

Ein Attributwert, der im CRITERIA-Puffer aufgelistet ist, ist für das angegebene Attribut nicht gültig.

**INVALIDCMD**

Aus einem der folgenden Gründe ist der Befehl ungültig:

**FILTER**

Der für die Operation übergebene Filterausdruck ist zu umfangreich oder zu komplex.

**LENGTH**

Die Gesamtlänge aller Optionen im Befehl überschreitet den maximalen Grenzwert.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- CRITERIA
- FILTER
- LENGTH
- OBJECT
- THREAD.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

**NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

**APITASK**

Die API-Steuersubtask ist nicht aktiv.

**CPSMAPI**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

**NOTFOUND**

Aus den folgenden Gründen kam es zu einer Bedingung des Typs 'Nicht gefunden':

### ATTRIBUTE

Ein Attribut, das im CRITERIA-Puffer angegeben wurde, wurde für die angegebene Ressourcentabelle nicht gefunden.

### SERVERGONE

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

### VERSIONINVL

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

#### NOTSUPPORTED

Die Version des Anwendungsstubsprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

#### NOTVSNCONN

Die Version des Anwendungsstubsprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

---

## SPECIFY VIEW

Erstellen einer angepassten Ansicht einer angegebenen Ressourcentabelle.

```
►—SPECIFY—VIEW—(—Datenwert—)—FIELDS—(—Datenbereich—)——————►
►—LENGTH—(—Datenwert—)—OBJECT—(—Datenwert—)—THREAD—(—CPSM-Token—)—————►
►—RESPONSE—(—Datenreferenz—)—REASON—(—Datenreferenz—)——————►◀
```

### Beschreibung

Mit diesem Befehl wird eine angepasste Ansicht einer angegebenen Ressourcentabelle erstellt.

- Ansichten können nur für Ressourcentabellen mit einem Typ von CICS-Ressource erstellt werden.
- Ansichten werden dem jeweiligen Verarbeitungsthread zugeordnet, bei dem sie erstellt werden; eine gemeinsame Nutzung durch andere Verarbeitungsthreads ist nicht möglich.
- Wenn ein Verarbeitungsthread beendet wird, werden alle von ihm erstellten Ansichten gelöscht.
- Der Name, den Sie einer Ansicht zuordnen, hat Vorrang vor allen vorhandenen Ressourcentabellennamen. Sie können einen vorhandenen Ressourcentabellennamen erneut definieren, um eine angepasste Ansicht der Ressourcentabelle darzustellen.
- Es ist empfehlenswert, Namen für angepasste Ansichten zu verwenden, die nicht bereits anderen angepassten Ansichten oder von CICSplex SM bereitgestellten Ressourcentabellen zugeordnet sind. Wenn Sie einen Namen verwenden, der bereits zugeordnet ist, sollten Sie sich darüber im Klaren sein, dass Ihre Verarbeitung negativ beeinflusst werden könnte. Weitere Details hierzu finden Sie in *Developing CICSplex SM applications*.
- Wenn Sie ein Upgrade auf eine spätere Version von CICSplex SM durchführen, müssen Sie überprüfen, ob neue Ressourcentabellen nicht dieselben Namen wie angepasste Ansichten haben. Weitere Details hierzu finden Sie in *Developing CICSplex SM applications*.

## Zugehörige Befehle

DISCARD, GET

## Optionen

### **FIELDS**(*Datenbereich*)

Gibt einen Puffer mit dem für diese Operation zu verwendenden Ausdruck an.

Details zur Verwendung eines Sortierausdrucks mit dem Befehl SPECIFY VIEW finden Sie in *Developing CICSplex SM applications*.

**Anmerkung:** Das Attribut EYU\_CICSNAME oder EYU\_CICSREL kann in einem Sortierausdruck nicht angegeben werden.

### **LENGTH**(*Datenwert*)

Ein Vollwortwert, der die Länge des FIELDS-Puffers angibt.

**Anmerkung:** Die von Ihnen angegebene Pufferlänge darf keine anderen Daten als einen Sortierausdruck enthalten.

### **OBJECT**(*Datenwert*)

Gibt die Ressourcentabelle an, für die eine Ansicht erstellt wird. Bei diesem Wert muss es sich um einen aus 1 bis 8 Zeichen bestehenden Namen einer gültigen CICS-Ressourcentabelle handeln. Eine Liste der CICSplex SM-Ressourcentabellen nach Typ finden Sie in *Developing CICSplex SM applications*.

### **REASON**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

### **RESPONSE**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

### **THREAD**(*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

### **VIEW**(*Datenwert*)

Definiert einen aus 1 bis 8 Zeichen bestehenden Namen für die zu erstellende Ansicht.

## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl SPECIFY VIEW zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

**DUPE** Aus den folgenden Gründen kam es zu einer Duplikatsbedingung:

**VIEW** Die angegebene Ansicht ist bereits vorhanden und kann nicht erstellt werden.

### **ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

**NOSERVICE**

Das Anwendungsstubsprogramm konnte das API-Servicemodul nicht laden.

**NOSTORAGE**

Das Anwendungsstubsprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**INVALIDCMD**

Aus dem folgenden Grund ist der Befehl ungültig:

**LENGTH**

Die Gesamtlänge aller Optionen im Befehl überschreitet den maximalen Grenzwert.

**INVALIDPARM**

In der Befehlsfolge oder im FIELDS-Puffer wurde ein ungültiger Parameter festgestellt. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- ATTRIBUTE
- FIELDS
- LENGTH
- OBJECT
- THREAD
- VIEW.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

**NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

**APITASK**

Die API-Steuersubtask ist nicht aktiv.

**CPSMAPI**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

**NOTFOUND**

Aus den folgenden Gründen kam es zu einer Bedingung des Typs 'Nicht gefunden':

**ATTRIBUTE**

Ein Attribut, das im FIELDS-Puffer angegeben wurde, wurde für die angegebene Ressourcentabelle nicht gefunden.

**SERVERGONE**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

**TABLEERROR**

Ein Ressourcentabellendatensatz ist aus einem der folgenden Gründe ungültig:

**DATAERROR**

Der Wert, der einem oder mehreren Ressourcentabellenattributen zugeordnet ist, ist ungültig. Dieser Fehler kann auftreten, wenn der Ressourcentabelle erforderliche Attribute fehlen, sie ein oder mehrere miteinander in Konflikt stehende Attribute enthält oder sie nicht vorhanden ist. Verwenden Sie den Befehl FEEDBACK, um weitere Informationen zu diesem Fehler abzurufen.

**INVALIDVER**

Die angegebene Version der Ressourcentabelle wird von CICSplex SM nicht unterstützt.

**VERSIONINVL**

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

**NOTSUPPORTED**

Die Version des Anwendungsstubprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

**NOTVSNCONN**

Die Version des Anwendungsstubprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

---

## TERMINATE

Beenden der gesamten API-Verarbeitung für alle aktiven Threads.

►►—TERMINATE—RESPONSE—(—Datenreferenz—)—REASON—(—Datenreferenz—)—►►

**Beschreibung**

Durch diesen Befehl wird die gesamte API-Verarbeitung für alle aktiven Threads beendet, die durch die CICS- oder MVS-Task erstellt wurden, die den Befehl absetzt.

- Das Absetzen des Befehls TERMINATE entspricht dem Absetzen des Befehls DISCONNECT für jeden aktiven Thread einzeln.
- Alle Ressourcen, die dem Thread zugeordnet sind, werden freigegeben, einschließlich Ergebnislisten, Filter, Ansichten, Diagnosedaten und ausstehenden asynchronen Anforderungen.

**Zugehörige Befehle**

CONNECT, DISCONNECT



## Optionen

### REASON(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

### RESPONSE(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl TERMINATE zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

### ENVIRONERROR

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

#### NOSERVICE

Das Anwendungsstubprogramm konnte das API-Servicemodul nicht laden.

#### NOSTORAGE

Das Anwendungsstubprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

### FAILED

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

#### ABENDED

Die Befehlsverarbeitung wurde abnormal beendet.

#### EXCEPTION

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

---

## TRANSLATE

Umsetzen von Attributwerten von Ressourcentabellen.

### Befehl

```
►►—TRANSLATE—OBJECT—(—Datenwert—)—ATTRIBUTE—(—Datenwert—)—►►
►└─FROMCV—(—Datenwert—)—TOCHAR—(—Datenreferenz—)—►
  └─FROMCHAR—(—Datenwert—)—TOCV—(—Datenreferenz—)—►
►—THREAD—(—CPSM-Token—)—RESPONSE—(—Datenreferenz—)—►
►—REASON—(—Datenreferenz—)—►►
```

### Beschreibung

Mit diesem Befehl werden Attributwerte von Ressourcentabellen, die in codierter Form (z. B. als EYUDA- und CVDA-Werte) verwaltet werden, zwischen ihrem internen codierten Format und einem externen Anzeigeformat umgesetzt.

- Wenn Ihr Programm in REXX geschrieben ist, können Sie mit dem Befehl TPARSE auf einen Ressourcentabellendatensatz zugreifen und dessen Attributwerte umsetzen. Wenn Sie jedoch die Option ASIS mit TPARSE verwenden, werden die Attributwerte nicht in ihr externes Format umgesetzt. In diesem Fall müssen Sie TRANSLATE verwenden, nachdem Sie TPARSE verwendet haben, um die formatierten Anzeigewerte zu empfangen. Eine Beschreibung des Befehls TPARSE finden Sie in Kapitel 2, „REXX-Funktionen und -Befehle“, auf Seite 13.
- In einer CICS-Umgebung können durch die Funktion DFHVALUE inkompatible CVDA-Werte für die folgenden Ressourcentabellenattribute zurückgegeben werden:

| Ressourcentabelle | Attributwert         |
|-------------------|----------------------|
| CONNECT           | RECOVSTATUS(NRS)     |
| IPCONN            | RECOVSTATUS(NRS)     |
| LIBRARY           | CRITSTATUS(CRITICAL) |
| LOCTRAN           | RESSEC(RESSECEXT)    |
| PROGDEF           | API(CICSAPI)         |
| PROGRAM           | APIST(CICSAPI)       |
| PROGRAM           | LPASTAT(NOTSVA)      |
| PROGRAM           | LPASTAT(SVA)         |

Da diese CVDA-Werte in Konflikt mit Werten stehen, die in CICS verwendet werden, muss CICSplex SM sie modifizieren, damit sie ihre Eindeutigkeit beibehalten. Für alle CICS-CVDA-Attribute fügt CICSplex SM entweder 8000 oder 9000 zum Wert hinzu, der von DFHVALUE zurückgegeben wird.

Wenn Sie eines dieser Attribute umsetzen möchten, müssen Sie entweder 8000 oder 9000 zu dem Wert hinzufügen, den Sie über DFHVALUE erhalten haben, bevor Sie das Attribut bei CICSplex SM vorweisen.

## Optionen

### ATTRIBUTE(*Datenwert*)

Gibt das Ressourcentabellenattribut an, das umgesetzt werden soll. Bei diesem Wert muss es sich um einen aus 1 bis 12 Zeichen bestehenden Namen eines gültigen Attributs für die Ressourcentabelle handeln.

### FROMCHAR(*Datenwert*)

Gibt den aus 1 - 12 Zeichen bestehenden Wert für das angegebene Attribut an.

### FROMCV(*Datenwert*)

Gibt den 4 Byte großen internen codierten Wert für das angegebene Attribut an.

### OBJECT(*Datenwert*)

Gibt die Ressourcentabelle an, zu der das umgesetzte Attribut gehört. Bei diesem Wert muss es sich um einen aus 8 Zeichen bestehenden Namen einer gültigen Ressourcentabelle handeln.

### REASON(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

### RESPONSE(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**THREAD**(*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

**TOCHAR**(*Datenreferenz*)

Benennt eine Variable, um das Ergebnis der Umsetzung eines internen codierten Werts in den aus 1 bis 12 Zeichen bestehenden Wert für das angegebene Attribut zu erhalten.

**TOCV**(*Datenreferenz*)

Benennt eine Variable, um das Ergebnis der Umsetzung eines Zeichenwerts in den 4 Byte großen internen Wert für das angegebene Attribut zu erhalten.

## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl TRANSLATE zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

**ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

**NOSERVICE**

Das Anwendungsstubsprogramm konnte das API-Servicemodul nicht laden.

**NOSTORAGE**

Das Anwendungsstubsprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

**SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

**FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

**ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- ATTRIBUTE
- FROMCHAR
- FROMCV
- OBJECT
- THREAD
- TOCHAR
- TOCV.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

#### NOTAVAILABLE

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

#### APITASK

Die API-Steuersubtask ist nicht aktiv.

#### CPSMAPI

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

#### SERVERGONE

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

#### TABLEERROR

Ein Ressourcentabellendatensatz ist aus dem folgenden Grund ungültig:

#### INVALIDVER

Die angegebene Version der Ressourcentabelle wird von CICSplex SM nicht unterstützt.

#### VERSIONINVL

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

#### NOTSUPPORTED

Die Version des Anwendungsstubprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

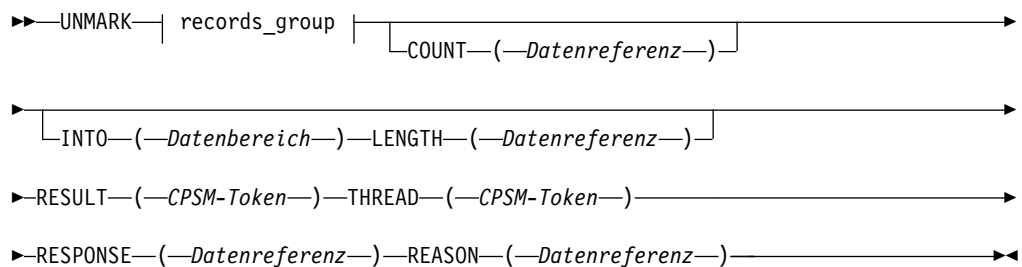
#### NOTVSNCONN

Die Version des Anwendungsstubprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.

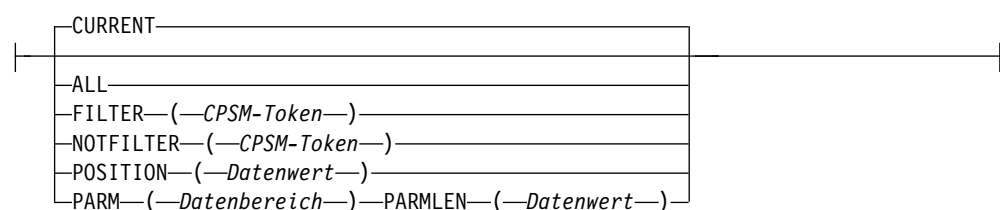
---

## UNMARK

Entfernen von Markierungen aus Ressourcentabellendatensätzen.



#### records\_group



## Beschreibung

Mit diesem Befehl werden die Markierungen entfernt, die durch einen vorherigen Befehl MARK in Ressourcentabellendatensätzen platziert wurden. Der Befehl UNMARK beginnt immer mit der Verarbeitung des letzten abgerufenen Datensatzes und nicht mit dem nächsten Datensatz in der Ergebnisliste.

## Zugehörige Befehle

EXPAND, LOCATE, MARK

## Optionen

### ALL

Entfernt die Markierungen aus allen Ressourcentabellendatensätzen in der Ergebnisliste.

### COUNT (*Datenreferenz*)

Benennt eine Variable, um die Anzahl der Ressourcentabellendatensätze zu empfangen, deren Markierung nicht aufgehoben werden konnte.

### CURRENT

Entfernt nur die Markierung des aktuellen Ressourcentabellendatensatzes.

### FILTER (*CPSM-Token*)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option FILTER entfernt die Markierungen nur von den Ressourcentabellendatensätzen, die den angegebenen Filterkriterien entsprechen.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

### INTO (*Datenbereich*)

Gibt einen Puffer an, um eine Liste der Ressourcentabellendatensätze zu empfangen, deren Markierung nicht aufgehoben werden konnte.

Dieser Puffer muss lang genug sein, um die maximale Anzahl von Datensatznummern aufzunehmen, die aus Ihrer Anforderung des Typs UNMARK resultieren können (wenn die Markierung keines Datensatzes aufgehoben werden kann). Datensatznummern werden im INTO-Puffer einzeln aufgelistet (nicht nach Bereich) und werden durch Kommas voneinander getrennt.

**Anmerkung:** Wenn Sie den RESPONSE-Wert WARNING AREATOOSMALL empfangen (weil der Puffer nicht lang genug war), stellen die Daten, die in diesem Puffer zurückgegeben wurden, eine Teilliste der Datensätze dar, deren Markierung nicht aufgehoben werden konnte.

### LENGTH (*Datenreferenz*)

Ein Vollwortwert, der die Länge des INTO-Puffers angibt.

Der Wert, den CICSplex SM in diesem Feld zurückgibt, ist von dem RESPONSE-Wert für den Befehl UNMARK abhängig:

**OK** Die tatsächliche Länge der Daten, die im INTO-Puffer zurückgegeben werden.

### WARNING AREATOOSMALL

Die Pufferlänge, die erforderlich wäre, um eine vollständige Liste der Datensätze aufzunehmen, deren Markierung nicht aufgehoben werden konnte.

**NOTFILTER**(*CPSM-Token*)

Gibt einen Filter an, der für diese Operation verwendet werden soll. Die Option NOTFILTER entfernt die Markierungen nur von den Ressourcentabellendatensätzen, die nicht den angegebenen Filterkriterien entsprechen.

Der Wert für *CPSM-Token*, der einen Filter angibt, wird durch den Befehl SPECIFY FILTER zurückgegeben.

**PARM**(*Datenbereich*)

Gibt einen Puffer mit dem Parameterausdruck an, mit dem die Ressourcentabellendatensätze aufgelistet werden, deren Markierung aufgehoben werden soll.

Der Parameterausdruck für den Befehl UNMARK ist eine Zeichenfolge aus Datensatznummern. Beispiel:

```
PARM('1,3,6:9,24.')
```

Wenn Sie einzelne Datensätze angeben möchten, trennen Sie die Datensatznummern durch ein Komma voneinander. Um einen Bereich von Datensätzen anzugeben, trennen Sie die niedrigen und hohen Datensatznummern durch einen Doppelpunkt voneinander. Der gesamte Parameterausdruck muss mit einem Punkt enden.

Details zur Verwendung eines Parameterausdrucks mit dem Befehl UNMARK finden Sie in Developing CICSplex SM applications.

**PARMLEN**(*Datenwert*)

Ein Vollwortwert, der die Länge des PARM-Puffers angibt.

**POSITION**(*Datenwert*)

Entfernt die Markierung aus dem n-ten Ressourcentabellendatensatz in der Ergebnisliste.

Dieser Wert muss eine Zahl sein, die die relative Position des Datensatzes in der Ergebnisliste angibt. Der erste Datensatz in einer Ergebnisliste wird durch die Zahl 1 angegeben.

Wenn Sie beispielsweise die Markierung des fünften Ressourcentabellendatensatzes in einer Ergebnisliste aufheben möchten, geben Sie POSITION(5) an.

**REASON**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

**RESPONSE**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**RESULT**(*CPSM-Token*)

Gibt die API-Ergebnisliste an, die von dieser Operation verarbeitet werden soll. Die Ergebnisliste kann durch einen der folgenden Befehle erzeugt werden:

- COPY
- GET
- GETDEF
- GROUP
- PERFORM OBJECT.

**THREAD**(*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

## Bedingungen

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl UN-MARK zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

### NODATA

Es wurden keine Datensätze gefunden, die mit den angegebenen Suchkriterien übereinstimmen.

### WARNING

Der Befehl hat die Verarbeitung aus einem der folgenden Gründe mit einer Warnung beendet:

#### AREATOOSMALL

Sie haben die Optionen INTO und LENGTH angegeben, aber der Puffer war nicht lang genug, um die Zeichenfolge der Datensätze aufzunehmen, deren Markierung nicht aufgehoben werden konnte.

#### DATAERROR

Mindestens einer der im PARM-Puffer angegebenen Datensätze konnte nicht gefunden werden, um seine Markierung aufzuheben. Wenn Sie die Option COUNT angegeben haben, wird die Anzahl der Datensätze, deren Markierung nicht aufgehoben werden konnte, zurückgegeben. Wenn Sie die Optionen INTO und LENGTH angegeben haben, wird eine Liste der Datensätze in dem Puffer zurückgegeben.

**BUSY** Aus den folgenden Gründen kam es zu einer Bedingung des Typs 'Aktiv':

#### RESULT

Die in der Option RESULT angegebene Ergebnisliste wird von einem anderen Befehl verarbeitet.

### ENVIRONERROR

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

#### NOSERVICE

Das Anwendungsstubprogramm konnte das API-Servicemodul nicht laden.

#### NOSTORAGE

Das Anwendungsstubprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

#### SOCRESOURCE

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

#### SOLRESOURCE

Eine erforderliche Ressource, deren Eigner ein lokaler Eigner ist (d. h. Eigner des Adressraums, in dem der Verarbeitungsthread ausgeführt wird), ist nicht verfügbar.

### FAILED

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

#### ABENDED

Die Befehlsverarbeitung wurde abnormal beendet.

**EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

**INVALIDCMD**

Aus dem folgenden Grund ist der Befehl ungültig:

**LENGTH**

Die Gesamtlänge aller Optionen im Befehl überschreitet den maximalen Grenzwert.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- COUNT
- FILTER
- INTO
- LENGTH
- NOTFILTER
- PARM
- PARMLen
- RESULT
- THREAD.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

**NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

**APITASK**

Die API-Steuersubtask ist nicht aktiv.

**CPSMAPI**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

**SERVERGONE**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

**VERSIONINVL**

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

**NOTSUPPORTED**

Die Version des Anwendungsstubprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

**NOTVSNNCONN**

Die Version des Anwendungsstubprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.



---

## UPDATE

Aktualisieren einer vorhandenen CICSplex SM- oder CICS-Definition.

```
►—UPDATE—OBJECT—(—Datenwert—)—————►
|
| FROM—(—Datenbereich—)—————►
| └─RESULT—(—CPSM-Token—)—MODIFY—(—Datenbereich—)─┘
|
| ►—LENGTH—(—Datenwert—)—————►
|
| └─PARM—(—Datenbereich—)—PARMLEN—(—Datenwert—)─┘
|
| └─CONTEXT—(—Datenwert—)─┘ └─SCOPE—(—Datenwert—)─┘
|
| ►—THREAD—(—CPSM-Token—)—RESPONSE—(—Datenreferenz—)—————►
|
| ►—REASON—(—Datenreferenz—)—————►◀
```

### Beschreibung

Mit diesem Befehl wird eine vorhandene CICSplex SM- oder CICS-Definition aktualisiert, und zwar entsprechend den von Ihnen angegebenen Attributwerten.

- Die aktualisierte Definition ersetzt die vorhandene Definition im CICSplex SM-Datenrepository.
- Für Definitionen mit einem CICSplex als Kontext (z. B. Workload-Management- oder Echtzeitanalysedefinitionen) wird die Definition auch in den Datenrepositorys aller an der Verwaltung des CICSplex beteiligten CMASs aktualisiert.

### Zugehörige Befehle

CREATE, REMOVE

### Optionen

#### CONTEXT (Datenwert)

Gibt den Kontext für diesen Befehl an. Bei dem Kontext muss es sich um einen aus 1 bis 8 Zeichen bestehenden Namen eines CMAS oder CICSplex handeln.

Wenn Sie die Option CONTEXT nicht angeben, wird der Standardkontext für den Thread angenommen.

#### FROM (Datenbereich)

Gibt einen Puffer an, der einen Ressourcentabellendatensatz enthält, der die zu aktualisierende Definition darstellt.

Der Datensatz muss alle Attribute für die Ressourcentabelle enthalten, die in der Option OBJECT angegeben ist. Für optionale Attribute, die Sie nicht angeben möchten, setzen Sie das Feld auf Nullwerte (d. h. 0).

#### LENGTH (Datenwert)

Ein Vollwortwert, der die Länge des FROM- oder MODIFY-Puffers angibt.

**Anmerkung:** Die von Ihnen angegebene Pufferlänge darf keine anderen Daten als einen Ressourcentabellendatensatz oder einen Modifikationsausdruck enthalten.

**MODIFY**(*Datenbereich*)

Gibt einen Puffer mit dem Modifikationsausdruck an, der für die Modifikation der Ressourcentabellendatensätze der CICS-Definition verwendet werden soll.

Weitere Informationen zum Bilden eines Modifikationsausdrucks finden Sie in *Developing CICSplex SM applications*.

**OBJECT**(*Datenwert*)

Gibt die Ressourcentabelle an, die die zu aktualisierende Definition darstellt. Bei diesem Wert muss es sich um einen aus 8 Zeichen bestehenden Namen einer gültigen Ressourcentabelle einer CICSplex SM-Definition oder CICS-Definition handeln. Eine Liste der CICSplex SM-Ressourcentabellen nach Typ finden Sie in *Developing CICSplex SM applications*.

**PARM**(*Datenbereich*)

Gibt einen Puffer mit dem Parameterausdruck an, der zum Aktualisieren der Definition verwendet werden soll.

Details zur Verwendung eines Parameterausdrucks mit dem Befehl UPDATE finden Sie in *Developing CICSplex SM applications*. Eine Beschreibung der Parameter, die für eine angegebene Ressourcentabelle gültig sind, finden Sie in CICSplex SM-Ressourcentabellen in den Referenzinformationen.

**PARMLEN**(*Datenwert*)

Ein Vollwortwert, der die Länge des PARM-Puffers angibt.

**REASON**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortursache zu empfangen.

**RESPONSE**(*Datenreferenz*)

Benennt eine Variable, um den von diesem Befehl zurückgegebenen Wert für die Vollwortantwort zu empfangen.

**RESULT**(*CPSM-Token*)

Gibt die API-Ergebnisliste an, die von dieser Operation verarbeitet werden soll. Die Ergebnisliste muss Ressourcentabellendatensätze der CICS-Definition enthalten. Die Datensätze werden dem Modifikationsausdruck entsprechend aktualisiert, den Sie im MODIFY-Puffer angeben.

Die Ergebnisliste kann durch einen der folgenden Befehle erzeugt werden:

- COPY
- GET
- GROUP
- PERFORM OBJECT.

**SCOPE**(*Datenwert*)

Gibt den Bereich für diesen Befehl an.

Wenn der aktuelle Kontext (definiert durch diesen Befehl oder einen früheren Befehl des Typs CONNECT oder QUALIFY) ein CICSplex ist, die Option OBJECT eine CICS-Definitionsressource angibt und die Option PARM den Parameter CSD beinhaltet, kann ein gültiger Bereich angegeben werden.

Der Bereich kann ein CICS-System innerhalb des CICSplex sein. Wenn es sich bei dem aktuellen Kontext um einen CMAS handelt oder die Option OBJECT einen anderen Typ von Ressourcentabelle angibt oder der Parameter CSD in keiner CICS-Definitionsressource angegeben ist, ist diese Option bedeutungslos und wird ignoriert.

Wenn SCOPE für den Befehl gültig ist und Sie die Option SCOPE nicht angeben, wird der Standardbereich für den Thread angenommen. Wenn der aktuelle Kontext ein CICSplex ist und kein Standardbereich für den Befehl CONNECT oder QUALIFY festgelegt wurde, erhalten Sie eine Antwort des Typs INVALIDPARAM für die Option SCOPE.

#### **THREAD** (*CPSM-Token*)

Gibt den API-Thread an, der für diese Operation verwendet werden soll. Der Wert für *CPSM-Token*, der einen Thread angibt, wird durch den Befehl CONNECT zurückgegeben.

## **Bedingungen**

Im Folgenden finden Sie eine Liste der RESPONSE-Werte, die vom Befehl UPDATE zurückgegeben werden können. Die Beschreibung der einzelnen RESPONSE-Werte enthält eine Liste der zugeordneten REASON-Werte, sofern dies zutrifft.

**OK** Die Verarbeitung des Befehls wurde erfolgreich ausgeführt.

#### **ENVIRONERROR**

Aus einem der folgenden Gründe kam es zu einem Umgebungsfehler:

##### **NOSERVICE**

Das Anwendungsstubprogramm konnte das API-Servicemodul nicht laden.

##### **NOSTORAGE**

Das Anwendungsstubprogramm konnte den erforderlichen Speicher in dem Adressraum, in dem der Verarbeitungsthread ausgeführt wird, nicht abrufen.

##### **NOTPROCESSED**

Einer der MASSs, an den die Anforderung übertragen wurde, konnte die Anforderung nicht verarbeiten (nur bei CSD-Anforderungen).

##### **REQTIMEOUT**

Einer der CMASs oder MASSs, an die die Anforderung übertragen wurde, antwortete nicht.

##### **SOCRESOURCE**

Eine erforderliche Ressource, deren Eigner der CMAS (CICSplex SM address space = CICSplex SM-Adressraum) ist, ist nicht verfügbar.

#### **FAILED**

Aus einem der folgenden Gründe ist der Befehl fehlgeschlagen:

##### **ABENDED**

Die Befehlsverarbeitung wurde abnormal beendet.

##### **EXCEPTION**

Bei der Befehlsverarbeitung wurde eine Ausnahmebedingung festgestellt.

#### **INVALIDDATA**

Aus einem der folgenden Gründe kam es zu einem Fehler über ungültige Daten:

##### **MODIFY**

Ein Attributwert, der im MODIFY-Puffer aufgelistet ist, ist für das angegebene Attribut nicht gültig.

**NOTSUPPORTED**

Ein Attribut, das im MODIFY-Puffer aufgeführt ist, kann nicht geändert werden.

**INVALIDCMD**

Aus dem folgenden Grund ist der Befehl ungültig:

**LENGTH**

Die Gesamtlänge aller Optionen im Befehl überschreitet den maximalen Grenzwert.

**INVALIDPARM**

Es wurde ein ungültiger Parameter gefunden. Der Parameter, der ungültig ist, wird als Ursachenwert zurückgegeben:

- CONTEXT
- FROM
- LENGTH
- MODIFY
- OBJECT
- PARM
- PARMLEN
- RESULT
- THREAD.

Ziehen Sie die Befehlsbeschreibung für die gültige Parametersyntax zurate.

**NOTAVAILABLE**

Aus einem der folgenden Gründe kam es zu einer Bedingung des Typs 'Nicht verfügbar':

**APITASK**

Die API-Steuersubtask ist nicht aktiv.

**CMAS**

Ein CMAS (CICSplex SM-Adressraum), an den die Anforderung übertragen wurde, ist nicht verfügbar.

**CPSMAPI**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden ist, ist für die API-Verarbeitung nicht verfügbar.

**CSDAPI**

Es ist keine Unterstützung für die CICSplex SM-API für den Zugriff auf CICS CSD verfügbar.

**MAINTPOINT**

Der Wartungspunkt für den aktuellen Kontext ist nicht verfügbar.

**NOTPERMIT**

Aus den folgenden Gründen kam es zu einer Unzulässigkeitsbedingung:

**USRID**

Die Benutzer-ID, die dem Verarbeitungsthread zugeordnet ist, verfügt nicht über die erforderliche Sicherheitsberechtigung.

**SERVERGONE**

Der CMAS (CICSplex SM-Adressraum), mit dem der Verarbeitungsthread verbunden war, ist nicht mehr aktiv.

**TABLEERROR**

Ein Ressourcentabellendatensatz ist aus einem der folgenden Gründe ungültig:

**DATAERROR**

Der Wert, der einem oder mehreren Ressourcentabellenattributen zugeordnet ist, ist ungültig. Dieser Fehler kann in folgenden Situationen auftreten:

- Wenn der Ressourcentabelle erforderliche Attribute fehlen, sie ein oder mehrere miteinander in Konflikt stehende Attribute enthält oder sie nicht vorhanden ist.
- Wenn eine CICS-Ressourcendefinition Attribute enthält, die dazu führen würden, dass der Befehl EXEC CICS CREATE Warnungen ausgeben würde.

Verwenden Sie den Befehl FEEDBACK, um weitere Informationen zu diesem Fehler abzurufen.

**INVALIDATTR**

Eines der Ressourcentabellenattribute ist ungültig.

**INVALIDVER**

Die angegebene Version der Ressourcentabelle wird von CICSplex SM nicht unterstützt.

**VERSIONINVL**

Aus einem der folgenden Gründe kam es zu einem Versionskonflikt:

**NOTSUPPORTED**

Die Version des Anwendungsstubsprogramms, das für diesen Befehl verwendet wird, wird nicht unterstützt.

**NOTVSNCONN**

Die Version des Anwendungsstubsprogramms, die für diesen Befehl verwendet wird, entspricht nicht der Version, die mit dem Befehl CONNECT verwendet wurde.



---

## Kapitel 4. Liste der CICSplex SM-API-Befehlsargumente

Es werden die Argumente der CICSplex SM-API aufgeführt.

Für jeden Befehl des Typs EXEC CPSM fügt der CICS-Umsetzer einen Aufruf in das EXEC-Schnittstellenstubprogramm von CICSplex SM ein. Siehe *Link editing your program*.

Parameter des Befehls EXEC CPSM werden mit der Standardaufrufparameterliste übergeben; im Folgenden sehen Sie die Liste der EXEC CPSM-Befehlsargumente, ARG0 - ARG*n*:

### **Argument 0**

ARG0 enthält befehlsspezifische Daten. Die ersten beiden Byte von ARG0 enthalten den Funktionscode des CPSM-Befehls. Eine Liste der Funktionscodewerte finden Sie in Kapitel 5, „Funktionscodewerte für CICSplex SM-API-Befehle“, auf Seite 153.

### **Argument 1**

ARG1 enthält den THREAD-Wert. Für den Befehl TERMINATE enthält ARG1 Nullen.

### **Argument 2**

ARG2 enthält den OBJECT-Wert, falls dies möglich sein sollte. Eine Liste der Befehle, die einen gültigen OBJECT-Wert enthalten können, finden Sie in Kapitel 5, „Funktionscodewerte für CICSplex SM-API-Befehle“, auf Seite 153.

### **Argument 3**

ARG3 enthält befehlsspezifische Daten.

### **Argument 4**

ARG4 enthält befehlsspezifische Daten.

### **Argument 5**

ARG5 gibt den Antwortcode zurück. Siehe Kapitel 6, „RESPONSE- und REASON-Werte“, auf Seite 155.

### **Argument 6**

ARG6 gibt den Ursachencode zurück. Siehe Kapitel 6, „RESPONSE- und REASON-Werte“, auf Seite 155.

### **Argument 7 und höher**

ARG7 und höher enthalten befehlsspezifische Daten.





---

## Kapitel 5. Funktionscodewerte für CICSplex SM-API-Befehle

Es werden die CICSplex SM-API-Befehle und ihre numerischen Funktionscodewerte aufgelistet.

Tabelle 1 ist nach Befehlsname sortiert und Tabelle 2 auf Seite 154 ist nach Funktionscodewert sortiert.

Informationen zu den Argumenten finden Sie in Kapitel 4, „Liste der CICSplex SM-API-Befehlsargumente“, auf Seite 151.

*Tabelle 1. Nach Befehlsname sortierte Funktionscodewerte*

| CICSplex SM-Befehl | Anzahl der Argumente | Funktionscode | OBJECT-Wert in ARG2 |
|--------------------|----------------------|---------------|---------------------|
| ADDRESS            | 7                    | F016          | Nein                |
| CANCEL             | 10                   | F002          | Nein                |
| CONNECT            | 13                   | F003          | Nein                |
| COPY               | 11                   | F004          | Nein                |
| CREATE             | 10                   | F005          | Ja                  |
| DELETE             | 7                    | F006          | Nein                |
| DISCARD            | 7                    | F007          | Nein                |
| DISCONNECT         | 7                    | F008          | Nein                |
| EXPAND             | 16                   | F021          | Nein                |
| FEEDBACK           | 10                   | F020          | Nein                |
| FETCH              | 10                   | F009          | Nein                |
| GET                | 7                    | F00A          | Ja                  |
| GETDEF             | 7                    | F017          | Ja                  |
| GROUP              | 12                   | F01E          | Ja                  |
| LISTEN             | 14                   | F018          | Nein                |
| LOCATE             | 7                    | F00B          | Nein                |
| MARK               | 7                    | F00C          | Nein                |
| ORDER              | 10                   | F00D          | Nein                |
| PERFORM OBJECT     | 10                   | F00F          | Nein                |
| PERFORM SET        | 10                   | F00E          | Ja                  |
| QUALIFY            | 7                    | F010          | Nein                |
| QUERY              | 7                    | F011          | Nein                |
| RECEIVE            | 7                    | F012          | Ja                  |
| REFRESH            | 7                    | F019          | Nein                |
| REMOVE             | 10                   | F013          | Ja                  |
| SET                | 10                   | F01A          | Nein                |
| SPECIFY FILTER     | 11                   | F014          | Ja                  |
| SPECIFY VIEW       | 11                   | F01B          | Ja                  |
| TERMINATE          | 7                    | F01C          | Nein                |

*Tabelle 1. Nach Befehlsname sortierte Funktionscodewerte (Forts.)*

| <b>CICSplex SM-Befehl</b> | <b>Anzahl der Argumente</b> | <b>Funktionscode</b> | <b>OBJECT-Wert in ARG2</b> |
|---------------------------|-----------------------------|----------------------|----------------------------|
| TRANSLATE                 | 13                          | F01D                 | Ja                         |
| UNMARK                    | 7                           | F015                 | Nein                       |
| UPDATE                    | 10                          | F01F                 | Ja                         |

*Tabelle 2. Nach Funktionscode sortierte Funktionscodewerte*

| <b>Funktionscode</b> | <b>Anzahl der Argumente</b> | <b>CICSplex SM-Befehl</b> | <b>OBJECT-Wert in ARG2</b> |
|----------------------|-----------------------------|---------------------------|----------------------------|
| F002                 | 10                          | CANCEL                    | Nein                       |
| F003                 | 13                          | CONNECT                   | Nein                       |
| F004                 | 11                          | COPY                      | Nein                       |
| F005                 | 10                          | CREATE                    | Ja                         |
| F006                 | 7                           | DELETE                    | Nein                       |
| F007                 | 7                           | DISCARD                   | Nein                       |
| F008                 | 7                           | DISCONNECT                | Nein                       |
| F009                 | 10                          | FETCH                     | Nein                       |
| F00A                 | 7                           | GET                       | Ja                         |
| F00B                 | 7                           | LOCATE                    | Nein                       |
| F00C                 | 7                           | MARK                      | Nein                       |
| F00D                 | 10                          | ORDER                     | Nein                       |
| F00E                 | 10                          | PERFORM SET               | Ja                         |
| F00F                 | 10                          | PERFORM OBJECT            | Nein                       |
| F010                 | 7                           | QUALIFY                   | Nein                       |
| F011                 | 7                           | QUERY                     | Nein                       |
| F012                 | 7                           | RECEIVE                   | Ja                         |
| F013                 | 10                          | REMOVE                    | Ja                         |
| F014                 | 11                          | SPECIFY FILTER            | Ja                         |
| F015                 | 7                           | UNMARK                    | Nein                       |
| F016                 | 7                           | ADDRESS                   | Nein                       |
| F017                 | 7                           | GETDEF                    | Ja                         |
| F018                 | 14                          | LISTEN                    | Nein                       |
| F019                 | 7                           | REFRESH                   | Nein                       |
| F01A                 | 10                          | SET                       | Nein                       |
| F01B                 | 11                          | SPECIFY VIEW              | Ja                         |
| F01C                 | 7                           | TERMINATE                 | Nein                       |
| F01D                 | 13                          | TRANSLATE                 | Ja                         |
| F01E                 | 12                          | GROUP                     | Ja                         |
| F01F                 | 10                          | UPDATE                    | Ja                         |
| F020                 | 10                          | FEEDBACK                  | Nein                       |
| F021                 | 16                          | EXPAND                    | Nein                       |

## Kapitel 6. RESPONSE- und REASON-Werte

In diesem Abschnitt finden Sie eine Übersicht der RESPONSE- und REASON-Werte, die durch die einzelnen API-Befehle zurückgegeben werden.

Beschreibungen dieser Werte finden Sie in der Beschreibung des Befehls, durch den sie zurückgegeben werden. Eine Liste der RESPONSE- und REASON-Zeichenwerte und ihrer numerischen Entsprechungen finden Sie in Kapitel 7, „EYUDA-Werte“, auf Seite 161. Eine Beschreibung der RESPONSE- und REASON-Optionen finden Sie in Developing CICSplex SM applications.

| Befehl  | RESPONSE     | REASONS   |
|---------|--------------|---|
| ADDRESS | OK           |   |
|         | ENVIRONERROR | NOSERVICE, NOSTORAGE  |
|         | FAILED       | ABENDED, EXCEPTION  |
|         | INVALIDPARM  | ECB, SENTINEL, THREAD   |
|         | NOTAVAILABLE | APITASK, CPSMAPI  |
|         | SERVERGONE   |   |
|         | VERSIONINVL  | NOTSUPPORTED, NOTVSNCONN  |
| CANCEL  | OK           |   |
|         | ENVIRONERROR | NOSERVICE, NOSTORAGE, SOCRESOURCE   |
|         | FAILED       | ABENDED, EXCEPTION  |
|         | INVALIDPARM  | NOTIFICATION, THREAD  |
|         | NOTAVAILABLE | APITASK, CPSMAPI  |
|         | SERVERGONE   |   |
|         | VERSIONINVL  | NOTSUPPORTED, NOTVSNCONN  |
| CONNECT | OK           |   |
|         | ENVIRONERROR | APITASKERR, INVALIDTCB, NOSERVICE, NOSTORAGE, SOCRESOURCE, SOERESOURCE, SOLRESOURCE |
|         | FAILED       | ABENDED, EXCEPTION  |
|         | INVALIDPARM  | CONTEXT, SCOPE, SIGNONPARM, USRID, VERSION  |
|         | NOTAVAILABLE | APITASK, CPSMAPI, CPSMSERVER, CPSMSYSTEM, CPSMVERSION                               |
|         | NOTPERMIT    | USRID   |
|         | VERSIONINVL  | NOTSUPPORTED  |
| COPY    | OK           |   |
|         | NODATA       |   |
|         | BUSY         | FROM, TO  |
|         | ENVIRONERROR | NOSERVICE, NOSTORAGE, SOCRESOURCE   |
|         | FAILED       | ABENDED, EXCEPTION  |
|         | INCOMPATIBLE | INVALIDOBJ  |
|         | INVALIDPARM  | FILTER, FROM, NOTFILTER, THREAD, TO   |
|         | NOTAVAILABLE | APITASK, CPSMAPI  |
|         | SERVERGONE   |   |
|         | TABLEERROR   | INVALIDVER  |
|         | VERSIONINVL  | NOTSUPPORTED, NOTVSNCONN  |
| CREATE  | OK           |   |
|         | ENVIRONERROR | NOSERVICE, NOSTORAGE, NOTPROCESSED, REQTIMEOUT, SOCRESOURCE                         |
|         | FAILED       | ABENDED, EXCEPTION  |
|         | INVALIDCMD   | LENGTH  |
|         | INVALIDPARM  | CONTEXT, FROM, LENGTH, OBJECT, PARM, PARMLN, THREAD                                 |
|         | NOTAVAILABLE | APITASK, CMAS, CPSMAPI, CSDAPI, MAINTPOINT  |
|         | NOTPERMIT    | USRID   |
|         |              |   |

| Befehl     | RESPONSE     | REASONS   |
|------------|--------------|---|
| DELETE     | SERVERGONE   | DATAERROR, INVALIDATTR, INVALIDVER<br>NOTSUPPORTED, NOTVSNCONN  |
|            | TABLEERROR   |   |
|            | VERSIONINVL  |   |
|            | OK           | RESULT<br>NOSERVICE, NOSTORAGE, SOCRESOURCE, SOLRESOURCE<br>ABENDED, EXCEPTION<br>FILTER, NOTFILTER, RESULT, THREAD<br>APITASK, CPSMAPI |
|            | NODATA       |   |
|            | BUSY         |   |
|            | ENVIRONERROR |   |
|            | FAILED       |   |
|            | INVALIDPARM  |   |
|            | NOTAVAILABLE |   |
|            | SERVERGONE   | NOTSUPPORTED, NOTVSNCONN  |
|            | VERSIONINVL  |   |
| DISCARD    | OK           | RESULT<br>NOSERVICE, NOSTORAGE, SOCRESOURCE<br>ABENDED, EXCEPTION<br>FILTER, VIEW<br>FILTER, RESULT, THREAD, VIEW<br>APITASK, CPSMAPI   |
|            | BUSY         |   |
|            | ENVIRONERROR |   |
|            | FAILED       |   |
|            | INUSE        |   |
|            | INVALIDPARM  |   |
|            | NOTAVAILABLE |   |
|            | SERVERGONE   | NOTSUPPORTED, NOTVSNCONN  |
|            | VERSIONINVL  |   |
| DISCONNECT | OK           | NOSERVICE, NOSTORAGE, SOCRESOURCE<br>ABENDED, EXCEPTION<br>THREAD<br>APITASK, CPSMAPI   |
|            | ENVIRONERROR |   |
|            | FAILED       |   |
|            | INVALIDPARM  |   |
|            | NOTAVAILABLE |   |
|            | SERVERGONE   | NOTSUPPORTED, NOTVSNCONN  |
|            | VERSIONINVL  |   |
|            | OK           |   |
|            | ENVIRONERROR |   |
|            | FAILED       |   |
| FEEDBACK   | WARNING      | AREATOOSMALL  |
|            | BUSY         | RESULT  |
|            | ENVIRONERROR | NOSERVICE, NOSTORAGE  |
|            | FAILED       | ABENDED, EXCEPTION  |
|            | INVALIDPARM  | COUNT, INTO, LENGTH, RESULT, THREAD   |
|            | NOTAVAILABLE | APITASK, CPSMAPI  |
|            | SERVERGONE   | NOTSUPPORTED, NOTVSNCONN  |
|            | VERSIONINVL  |   |
|            | OK           |   |
|            | NODATA       |   |
| FETCH      | WARNING      | AREATOOSMALL  |
|            | BUSY         | RESULT  |
|            | ENVIRONERROR | NOSERVICE, NOSTORAGE, SOCRESOURCE, SOLRESOURCE  |
|            | FAILED       | ABENDED, EXCEPTION  |
|            | INVALIDPARM  | COUNT, FILTER, INTO, LENGTH, NOTFILTER, POSITION, RESULT, THREAD  |
|            | NOTAVAILABLE | APITASK, CPSMAPI  |
|            | SERVERGONE   | NOTSUPPORTED, NOTVSNCONN  |
|            | VERSIONINVL  |   |
|            | OK           |   |
|            | NODATA       |   |
| GET        | WARNING      | AREATOOSMALL  |
|            | BUSY         | RESULT  |
|            | ENVIRONERROR | NOSERVICE, NOSTORAGE, NOTPROCESSED, REQTIMEOUT, SOCRESOURCE   |
|            | FAILED       | ABENDED, EXCEPTION  |
|            | INVALIDDATA  | CRITERIA  |

| Befehl | RESPONSE     | REASONS  |
|--------|--------------|--|
|        | INVALIDCMD   | FILTER,LENGTH  |
|        | INVALIDPARM  | CONTEXT, CRITERIA, FILTER, LENGTH, OBJECT, PARM, PARMLen, RESULT, SCOPE, THREAD, TOKEN |
|        | NOTAVAILABLE | APITASK, CMAS, CPSMAPI, CSDAPI, MAINTPOINT, SCOPE, WORKLOAD                            |
|        | NOTFOUND     | ATTRIBUTE  |
|        | NOTPERMIT    | USRID  |
|        | SERVERGONE   |  |
|        | TABLEERROR   | DATAERROR, INVALIDVER  |
|        | VERSIONINVL  | NOTSUPPORTED, NOTVSNCONN   |
| GETDEF | OK           |  |
|        | NODATA       |  |
|        | BUSY         | RESULT   |
|        | ENVIRONERROR | NOSERVICE, NOSTORAGE, SOCRESOURCE  |
|        | FAILED       | ABENDED, EXCEPTION   |
|        | INCOMPATIBLE | INVALIDOBJ   |
|        | INVALIDPARM  | ATTRIBUTE, OBJECT, RESOURCE, RESULT, THREAD  |
|        | NOTAVAILABLE | APITASK, CPSMAPI   |
|        | SERVERGONE   |  |
|        | TABLEERROR   | DATAERROR, INVALIDVER  |
|        | VERSIONINVL  | NOTSUPPORTED, NOTVSNCONN   |
| GROUP  | OK           |  |
|        | NODATA       |  |
|        | BUSY         | FROM, TO   |
|        | ENVIRONERROR | NOSERVICE, NOSTORAGE, SOCRESOURCE  |
|        | FAILED       | ABENDED, EXCEPTION   |
|        | INVALIDCMD   | LENGTH   |
|        | INVALIDPARM  | BY, FILTER, FROM, LENGTH, NOTFILTER, SUMOPT, THREAD, TO                                |
|        | NOTAVAILABLE | APITASK, CPSMAPI   |
|        | SERVERGONE   |  |
|        | TABLEERROR   | INVALIDVER   |
|        | VERSIONINVL  | NOTSUPPORTED, NOTVSNCONN   |
| LISTEN | OK           |  |
|        | ENVIRONERROR | NOSERVICE, NOSTORAGE, SOCRESOURCE  |
|        | FAILED       | ABENDED, EXCEPTION   |
|        | INCOMPATIBLE | INVALIDEVT   |
|        | INVALIDPARM  | CONTEXT, EVENT, FILTER, NOTFILTER, NOTIFICATION, THREAD, TOKEN                         |
|        | NOTAVAILABLE | APITASK, CPSMAPI, PLEXMGR  |
|        | SERVERGONE   |  |
|        | VERSIONINVL  | NOTSUPPORTED, NOTVSNCONN   |
| LOCATE | OK           |  |
|        | NODATA       | BACKWARD, FORWARD  |
|        | BUSY         | RESULT   |
|        | ENVIRONERROR | NOSERVICE, NOSTORAGE, SOCRESOURCE, SOLRESOURCE   |
|        | FAILED       | ABENDED, EXCEPTION   |
|        | INVALIDPARM  | BACKWARD, FILTER, FORWARD, NOTFILTER, POSITION, RESULT, THREAD                         |
|        | NOTAVAILABLE | APITASK, CPSMAPI   |
|        | SERVERGONE   |  |
|        | VERSIONINVL  | NOTSUPPORTED, NOTVSNCONN   |
| MARK   | OK           |  |
|        | NODATA       |  |
|        | WARNING      | AREATOOSMALL, DATAERROR  |
|        | BUSY         | RESULT   |
|        | ENVIRONERROR | NOSERVICE, NOSTORAGE, SOCRESOURCE, SOLRESOURCE   |
|        | FAILED       | ABENDED, EXCEPTION   |
|        | INVALIDPARM  | COUNT, FILTER, INTO, LENGTH, NOTFILTER, PARM, PARMLen, RESULT, THREAD                  |
|        | NOTAVAILABLE | APITASK, CPSMAPI   |

| Befehl            | RESPONSE     | REASONS  |
|-------------------|--------------|--|
| ORDER             | SERVERGONE   |  |
|                   | VERSIONINVL  | NOTSUPPORTED, NOTVSNCONN   |
|                   | OK           |  |
|                   | BUSY         | RESULT   |
|                   | ENVIRONERROR | NOSERVICE, NOSTORAGE SOCRESOURCE   |
|                   | FAILED       | ABENDED, EXCEPTION   |
|                   | INVALIDCMD   | LENGTH   |
|                   | INVALIDPARM  | BY, LENGTH, RESULT, THREAD   |
|                   | NOTAVAILABLE | APITASK, CPSMAPI   |
|                   | SERVERGONE   |  |
| PERFORM<br>OBJECT | VERSIONINVL  | NOTSUPPORTED, NOTVSNCONN   |
|                   | OK           |  |
|                   | SCHEDULED    |  |
|                   | NODATA       | BACKWARD, FORWARD  |
|                   | BUSY         | RESULT   |
|                   | ENVIRONERROR | NOSERVICE, NOSTORAGE, NOTPROCESSED, REQTIMEOUT, SOCRESOURCE                                    |
|                   | FAILED       | ABENDED, EXCEPTION   |
|                   | INVALIDDATA  | PARM, CRITERIA   |
|                   | INVALIDCMD   | FILTER,LENGTH  |
|                   | INVALIDPARM  | ACTION, CONTEXT, CRITERIA, FILTER, LENGTH, OBJECT, PARM, PARMLEN, RESULT, SCOPE, THREAD, TOKEN |
| PERFORM SET       | NOTAVAILABLE | APITASK, CMAS, CPSMAPI, CSDAPI, MAINTPOINT, PLEXMGR, SCOPE, WORKLOAD                           |
|                   | NOTFOUND     | ACTION, ATTRIBUTE  |
|                   | NOTPERMIT    | USRID  |
|                   | SERVERGONE   |  |
|                   | TABLEERROR   | DATAERROR  |
|                   | VERSIONINVL  | NOTSUPPORTED, NOTVSNCONN   |
|                   | WARNING      | RESULT, ACTION   |
|                   | OK           |  |
|                   | SCHEDULED    |  |
|                   | NODATA       | BACKWARD, FORWARD  |
| QUALIFY           | BUSY         | RESULT   |
|                   | ENVIRONERROR | NOSERVICE, NOSTORAGE, NOTPROCESSED, REQTIMEOUT SOCRESOURCE                                     |
|                   | FAILED       | ABENDED, EXCEPTION   |
|                   | INVALIDDATA  | PARM, CRITERIA   |
|                   | INVALIDCMD   | LENGTH   |
|                   | INVALIDPARM  | ACTION, FILTER, NOTFILTER, PARM, PARMLEN, POSITION, RESULT, THREAD, TO-<br>KEN                 |
|                   | NOTAVAILABLE | APITASK, CMAS, CPSMAPI, MAINTPOINT, PLEXMGR, SCOPE, WORKLOAD                                   |
|                   | NOTFOUND     | ACTION, ATTRIBUTE  |
|                   | NOTPERMIT    | USRID  |
|                   | SERVERGONE   |  |
| QUERY             | TABLEERROR   | DATAERROR  |
|                   | VERSIONINVL  | NOTSUPPORTED, NOTVSNCONN   |
|                   | OK           |  |
|                   | ENVIRONERROR | NOSERVICE, NOSTORAGE, SOCRESOURCE  |
|                   | FAILED       | ABENDED, EXCEPTION   |
|                   | INVALIDPARM  | CONTEXT, SCOPE, THREAD   |
|                   | NOTAVAILABLE | APITASK, CPSMAPI, PLEXMGR  |
|                   | SERVERGONE   |  |
|                   | VERSIONINVL  | NOTSUPPORTED, NOTVSNCONN   |
|                   | OK           |  |
| QUERY             | BUSY         | RESULT   |
|                   | ENVIRONERROR | NOSERVICE, NOSTORAGE, SOCRESOURCE  |
|                   | FAILED       | ABENDED, EXCEPTION   |
|                   |              |  |

| Befehl         | RESPONSE     | REASONS  |
|----------------|--------------|--|
| RECEIVE        | INVALIDPARM  | CONTEXT, DATALENGTH, OBJECT, RESULT, THREAD, TYPE                                    |
|                | NOTAVAILABLE | APITASK, CPSMAPI   |
|                | SERVERGONE   |  |
|                | VERSIONINVL  | NOTSUPPORTED, NOTVSNCONN   |
|                | OK           |  |
|                | NODATA       |  |
|                | WARNING      | AREATOOSMALL   |
|                | ENVIRONERROR | NOSERVICE, NOSTORAGE, SOCRESOURCE  |
|                | FAILED       | ABENDED, EXCEPTION   |
|                | INVALIDPARM  | DELAY, INTO, LENGTH, OBJECT, THREAD, TOKEN   |
| REFRESH        | NOTAVAILABLE | APITASK, CPSMAPI   |
|                | SERVERGONE   |  |
|                | VERSIONINVL  | NOTSUPPORTED, NOTVSNCONN   |
|                | OK           |  |
|                | SCHEDULED    |  |
|                | BUSY         | RESULT   |
|                | ENVIRONERROR | NOSERVICE, NOSTORAGE, NOTPROCESSED, REQTIMEOUT, SOCRESOURCE                          |
|                | FAILED       | ABENDED, EXCEPTION   |
|                | INVALIDPARM  | COUNT, FILTER, NOTFILTER, RESULT, THREAD, TOKEN                                      |
|                | NOTAVAILABLE | APITASK, CMAS, CPSMAPI, MAINTPOINT, SCOPE  |
| REMOVE         | NOTPERMIT    | USRID  |
|                | SERVERGONE   |  |
|                | TABLEERROR   | DATAERROR  |
|                | VERSIONINVL  | NOTSUPPORTED, NOTVSNCONN   |
|                | OK           |  |
|                | ENVIRONERROR | NOSERVICE, NOSTORAGE, NOTPROCESSED, REQTIMEOUT, SOCRESOURCE                          |
|                | FAILED       | ABENDED, EXCEPTION   |
|                | INVALIDCMD   | LENGTH   |
|                | INVALIDPARM  | CONTEXT, FROM, LENGTH, OBJECT, PARM, PARMLen, THREAD                                 |
|                | NOTAVAILABLE | APITASK, CMAS, CPSMAPI, CSDAPI, MAINTPOINT   |
| SET            | NOTPERMIT    | USRID  |
|                | SERVERGONE   |  |
|                | TABLEERROR   | DATAERROR, INVALIDATTR, INVALIDVER   |
|                | VERSIONINVL  | NOTSUPPORTED, NOTVSNCONN   |
|                | OK           |  |
|                | SCHEDULED    |  |
|                | BUSY         | RESULT   |
|                | ENVIRONERROR | NOSERVICE, NOSTORAGE, NOTPROCESSED, REQTIMEOUT, SOCRESOURCE                          |
|                | FAILED       | ABENDED, EXCEPTION   |
|                | INVALIDDATA  | MODIFY, NOTSUPPORTED   |
| SPECIFY FILTER | INVALIDCMD   | LENGTH   |
|                | INVALIDPARM  | ATTRIBUTE, COUNT, FILTER, LENGTH, MODIFY, NOTFILTER, POSITION, RESULT, THREAD, TOKEN |
|                | NOTAVAILABLE | APITASK, CMAS, CPSMAPI, MAINTPOINT, SCOPE  |
|                | NOTFOUND     | ACTION, ATTRIBUTE  |
|                | NOTPERMIT    | USRID  |
|                | SERVERGONE   |  |
|                | TABLEERROR   | DATAERROR, INVALIDVER  |
|                | VERSIONINVL  | NOTSUPPORTED, NOTVSNCONN   |
|                | OK           |  |
|                | ENVIRONERROR | NOSERVICE, NOSTORAGE, SOCRESOURCE  |
|                | FAILED       | ABENDED, EXCEPTION   |
|                | INVALIDCMD   | FILTER, LENGTH   |
|                | INVALIDPARM  | CRITERIA, FILTER, LENGTH, OBJECT, THREAD   |
|                | NOTAVAILABLE | APITASK, CPSMAPI   |
|                | NOTFOUND     | ATTRIBUTE  |
|                |              |  |

| Befehl       | RESPONSE     | REASONS   |
|--------------|--------------|---|
| SPECIFY VIEW | SERVERGONE   | NOTSUPPORTED, NOTVSNCONN  |
|              | VERSIONINVL  |   |
|              | OK           | VIEW<br>NOSERVICE, NOSTORAGE, SOCRESOURCE<br>ABENDED, EXCEPTION<br>CRITERIA<br>LENGTH<br>ATTRIBUTE, FIELDS, LENGTH, OBJECT, THREAD, VIEW<br>APITASK, CPSMAPI<br>ATTRIBUTE<br><br>DATAERROR, INVALIDVER<br>NOTSUPPORTED, NOTVSNCONN  |
|              | DUPE         |   |
|              | ENVIRONERROR |   |
|              | FAILED       |   |
|              | INVALIDDATA  |   |
|              | INVALIDCMD   |   |
|              | INVALIDPARM  |   |
|              | NOTAVAILABLE |   |
|              | NOTFOUND     |   |
|              | SERVERGONE   |   |
| TERMINATE    | TABLEERROR   |   |
|              | VERSIONINVL  |   |
|              | OK           | NOSERVICE, NOSTORAGE<br>ABENDED, EXCEPTION  |
| TRANSLATE    | ENVIRONERROR |   |
|              | FAILED       | NOSERVICE, NOSTORAGE, SOCRESOURCE<br>ABENDED, EXCEPTION<br>ATTRIBUTE, FROMCHAR, FROMCV, OBJECT, THREAD, TOCHAR, TOCV<br>APITASK, CPSMAPI<br><br>INVALIDVER, INVALIDVER<br>NOTSUPPORTED, NOTVSNCONN  |
|              | OK           |   |
|              | ENVIRONERROR |   |
|              | FAILED       |   |
|              | INVALIDPARM  |   |
|              | NOTAVAILABLE |   |
|              | SERVERGONE   |   |
|              | TABLEERROR   |   |
| UNMARK       | VERSIONINVL  |   |
|              | OK           | AREATOOSMALL, DATAERROR<br>RESULT<br>NOSERVICE, NOSTORAGE, SOCRESOURCE, SOLRESOURCE<br>ABENDED, EXCEPTION<br>LENGTH<br>COUNT, FILTER, INTO, LENGTH, NOTFILTER, PARM, PARMLLEN, RESULT, THREAD<br>APITASK, CPSMAPI<br><br>NOTSUPPORTED, NOTVSNCONN   |
|              | NODATA       |   |
|              | WARNING      |   |
|              | BUSY         |   |
|              | ENVIRONERROR |   |
|              | FAILED       |   |
|              | INVALIDCMD   |   |
|              | INVALIDPARM  |   |
|              | NOTAVAILABLE |   |
|              | SERVERGONE   |   |
| UPDATE       | VERSIONINVL  |   |
|              | OK           | NOSERVICE, NOSTORAGE, NOTPROCESSED, REQTIMEOUT, SOCRESOURCE<br>ABENDED, EXCEPTION<br>LENGTH<br>CONTEXT, FROM, LENGTH, MODIFY, OBJECT, PARM, PARMLLEN, RESULT, THREAD<br>APITASK, CMAS, CPSMAPI, CSDAPI, MAINTPOINT<br>USRID<br><br>DATAERROR, INVALIDATTR, INVALIDVER<br>NOTSUPPORTED, NOTVSNCONN |
|              | ENVIRONERROR |   |
|              | FAILED       |   |
|              | INVALIDCMD   |   |
|              | INVALIDPARM  |   |
|              | NOTAVAILABLE |   |
|              | NOTPERMIT    |   |
|              | SERVERGONE   |   |
|              | TABLEERROR   |   |
|              | VERSIONINVL  |   |



---

## Kapitel 7. EYUDA-Werte

In diesem Abschnitt werden die CICSplex SM-API-EYUDA-Werte und ihre numerischen Entsprechungen aufgeführt.

Es gibt drei Arten von EYUDA-Werten:

### Allgemeine Werte

Werte, die von CICSplex SM zur Beschreibung oder Definition einer Resource verwendet werden. Diesen EYUDA-Werten entsprechen numerische Werte im Bereich zwischen 1 und 776. Siehe „Allgemeine EYUDA-Werte in numerischer Reihenfolge“.

### RESPONSE-Werte

Durch die Option RESPONSE eines API-Befehls zurückgegebene Werte. Diesen EYUDA-Werten entsprechen numerische Werte im Bereich zwischen 1024 und 1042. Siehe „RESPONSE-Werte vom Typ EYUDA in numerischer Reihenfolge“ auf Seite 187.

### REASON-Werte

Durch die Option REASON eines API-Befehls zurückgegebene Werte. Diesen EYUDA-Werten entsprechen numerische Werte im Bereich zwischen 1280 und 1378. Siehe „REASON-Werte vom Typ EYUDA in numerischer Reihenfolge“ auf Seite 189.

**Anmerkung:** Die EYUDA-Werte und ihre für SUMMUNLIKE aufgeführten numerischen Entsprechungen werden nur für ein EYUDA-Attribut in einer API-Ergebnisliste für den Befehl GROUP oder eine tabellarische WUI-Ansicht zurückgegeben. In früheren Releases wurde dieser Wert auf N\_a gesetzt.

**Anmerkung:** Die EYUDA-Werte und ihre numerischen Entsprechungen, die für HOTPOOL aufgeführt sind, sind nur in CICS Transaction Server 2.3 gültig.

---

## Allgemeine EYUDA-Werte in numerischer Reihenfolge

In diesem Abschnitt werden die allgemeinen EYUDA-Werte in numerischer Reihenfolge aufgelistet und der Zeichenwert für die einzelnen Werte aufgeführt.

| Wert | EYUDA   |
|------|---------|
| 0    | N_A     |
| 0    | NA      |
| 1    | YES     |
| 2    | NO      |
| 3    | ON      |
| 4    | OFF     |
| 5    | VALID   |
| 6    | INVALID |
| 7    | TRUE    |
| 8    | FALSE   |
| 9    | VLS     |

| Wert | EYUDA     |
|------|-----------|
| 10   | LS        |
| 11   | LW        |
| 12   | NM        |
| 13   | HW        |
| 14   | HS        |
| 15   | VHS       |
| 16   | EQ        |
| 17   | NE        |
| 18   | GT        |
| 19   | LT        |
| 20   | LOW       |
| 21   | HIGH      |
| 22   | NORMAL    |
| 23   | IMMEDIATE |
| 24   | TAKEOVER  |
| 25   | SHUT      |
| 26   | NOSHUT    |
| 27   | GLOBAL    |
| 28   | SYSTEM    |
| 30   | SUSPEND   |
| 33   | VALUE     |
| 34   | THRESHOLD |
| 35   | SAM       |
| 36   | APM       |
| 37   | MRM       |
| 38   | TRANID    |
| 39   | TERMID    |
| 40   | SIGNID    |
| 41   | RACFGID   |
| 42   | USERID    |
| 43   | NULL      |
| 44   | CHARSTR   |
| 45   | ACTIVE    |
| 46   | INACTIVE  |
| 47   | WAITING   |
| 48   | QUIESCING |
| 49   | POOL      |
| 50   | LTRAN     |
| 51   | RTRAN     |
| 53   | GOAL      |
| 54   | QUEUE     |

| Wert | EYUDA     |
|------|-----------|
| 55   | LUNAME    |
| 58   | DELIMIT   |
| 59   | PCONV     |
| 60   | LOGON     |
| 61   | SIGNON    |
| 63   | PERMANENT |
| 64   | MCICS     |
| 65   | MGLBL     |
| 66   | MDBX      |
| 67   | MCONN     |
| 68   | MFILE     |
| 69   | MJRNL     |
| 70   | MPROG     |
| 71   | MTERM     |
| 72   | MTDQS     |
| 73   | MTRAN     |
| 74   | MAPPL     |
| 75   | ABOVE     |
| 76   | BELOW     |
| 77   | NOCOPY    |
| 78   | DSA       |
| 79   | CDSA      |
| 80   | UDSA      |
| 81   | LPA       |
| 82   | EDSA      |
| 83   | ECDSA     |
| 84   | EUDSA     |
| 85   | ERDSA     |
| 86   | ELPA      |
| 87   | CICS      |
| 88   | USER      |
| 89   | READONLY  |
| 90   | LU61      |
| 91   | LU62      |
| 92   | INDIRECT  |
| 93   | MRO       |
| 94   | NOTAPPLIC |
| 95   | LFILE     |
| 96   | RFILE     |
| 97   | CTABL     |
| 98   | UTABL     |

| Wert | EYUDA      |
|------|------------|
| 99   | INSTALLED  |
| 100  | PENDING    |
| 101  | INHERIT    |
| 102  | EXPLICIT   |
| 103  | CICSSYS    |
| 104  | SYSGROUP   |
| 105  | KEEP       |
| 106  | NAME       |
| 107  | FORCE      |
| 108  | NONE       |
| 109  | UNASSIGNED |
| 110  | DROP       |
| 111  | LOCAL      |
| 112  | REMOTE     |
| 113  | DEFAULT    |
| 114  | REMOVE     |
| 115  | DORMANT    |
| 116  | START      |
| 117  | END        |
| 118  | ADJACENT   |
| 119  | LOSTCON    |
| 120  | CREATING   |
| 121  | REMOVING   |
| 122  | QUIESCED   |
| 123  | LINKACTIVE |
| 124  | LINKDOWN   |
| 125  | ESSS       |
| 126  | CONACT     |
| 127  | RESET      |
| 128  | SYSDUMP    |
| 129  | TRANDUMP   |
| 130  | MAXTASK    |
| 131  | STALLED    |
| 132  | SOSUDSA    |
| 133  | SOSCDSA    |
| 134  | SOSEUDSA   |
| 135  | SOSECDSA   |
| 136  | SOSERDSA   |
| 137  | SOSSDSA    |
| 138  | SOSESDSA   |
| 139  | SOSRDSA    |

| Wert | EYUDA       |
|------|-------------|
| 140  | QUIESCE     |
| 141  | PRIMARY     |
| 142  | SECONDARY   |
| 143  | DUPLICATE   |
| 144  | FROZEN      |
| 145  | ALL         |
| 146  | ANY         |
| 147  | SUM         |
| 148  | MIN         |
| 149  | MAX         |
| 150  | AVG         |
| 151  | CNT         |
| 152  | LE          |
| 153  | GE          |
| 154  | SDSA        |
| 155  | ESDSA       |
| 156  | RDSA        |
| 157  | SOSMVS      |
| 158  | SOSBELOW    |
| 159  | SOSABOVE    |
| 161  | EXECUTE     |
| 162  | CHECK       |
| 163  | LOSTCMAS    |
| 164  | LOSTMAS     |
| 165  | AASTERISK   |
| 166  | BLANK       |
| 167  | INDEX       |
| 168  | DATA        |
| 169  | BOTH        |
| 170  | NETBIOS     |
| 171  | TCPIP       |
| 172  | AFTER       |
| 173  | ALLREQS     |
| 174  | ASA         |
| 177  | ASIS        |
| 178  | ASSEMBLER   |
| 179  | BACKOUTONLY |
| 180  | BEFORE      |
| 181  | BLUE        |
| 182  | C           |
| 183  | CLEARCONV   |

| Wert | EYUDA      |
|------|------------|
| 184  | CLOSE      |
| 185  | COBOL      |
| 186  | COLD       |
| 187  | CYCLIC     |
| 188  | DEFERRED   |
| 189  | DIP        |
| 190  | DISK       |
| 191  | EODS       |
| 192  | EXTA       |
| 193  | FILE       |
| 194  | FIRSTREF   |
| 195  | GREEN      |
| 196  | IDENTIFY   |
| 197  | IGNORE     |
| 198  | INITIAL    |
| 199  | INOUT      |
| 200  | INPUT      |
| 201  | LEAVE      |
| 202  | LE370      |
| 203  | LINEAR     |
| 204  | LINK       |
| 205  | LMS        |
| 206  | LOGICAL    |
| 207  | LOGOFF     |
| 208  | LRU        |
| 209  | MACHINE    |
| 210  | MESSAGE    |
| 211  | MIXIDPE    |
| 212  | MOD        |
| 213  | MODIFYREQS |
| 214  | MSRE       |
| 215  | NEUTRAL    |
| 216  | NEW        |
| 217  | NOFORCE    |
| 218  | NONVTAM    |
| 219  | OLD        |
| 220  | ONLY       |
| 221  | OPEN       |
| 222  | OPID       |
| 223  | OUTPUT     |
| 224  | PERSISTENT |

| Wert | EYUDA       |
|------|-------------|
| 225  | PHYSICAL    |
| 226  | PINK        |
| 227  | PLI         |
| 228  | PRINTER     |
| 229  | RED         |
| 230  | RECOVERY    |
| 231  | REJECT      |
| 232  | RELEASESESS |
| 233  | REREAD      |
| 234  | RPG         |
| 235  | SCS         |
| 236  | SECURITY    |
| 237  | SHR         |
| 238  | SKIP        |
| 239  | SPECIFIC    |
| 240  | STARTIO     |
| 241  | STARTUP     |
| 242  | STRFIELD    |
| 243  | SYSDEFAULT  |
| 244  | TAPE        |
| 245  | TERMINAL    |
| 246  | TRANSACTION |
| 247  | TRANSIENT   |
| 248  | TURQUOISE   |
| 249  | U           |
| 250  | UNCONDREL   |
| 251  | UPDATEONLY  |
| 252  | VB          |
| 253  | VERIFY      |
| 254  | VTAM        |
| 255  | YELLOW      |
| 256  | 3270        |
| 257  | AUTO        |
| 258  | DYNAM       |
| 259  | EXTRA       |
| 260  | INTRA       |
| 261  | IND         |
| 262  | STAT        |
| 263  | RELATED     |
| 264  | TARGET      |
| 265  | NEVER       |

| Wert | EYUDA        |
|------|--------------|
| 266  | ALWAYS       |
| 267  | COLDONLY     |
| 268  | WARMONLY     |
| 269  | PROMPT       |
| 270  | CONTINUE     |
| 271  | TERMINATE    |
| 272  | SHUTDOWN     |
| 273  | RTADEF       |
| 274  | STATDEF      |
| 275  | CONNDEF      |
| 276  | FILEDEF      |
| 277  | JRNLDEF      |
| 278  | JRNMDEF      |
| 279  | LSRDEF       |
| 280  | MAPDEF       |
| 281  | PARTDEF      |
| 282  | PRTNDEF      |
| 283  | PROFDEF      |
| 284  | PROGDEF      |
| 285  | SESSDEF      |
| 286  | TDQDEF       |
| 287  | TERMDEF      |
| 288  | TRANDEF      |
| 289  | TRNCLDEF     |
| 290  | TSQDEF       |
| 291  | TYPTMDEF     |
| 292  | MPSYNCCR     |
| 293  | ASSOCIATIONS |
| 294  | MEMBERS      |
| 295  | DB2CDEF      |
| 296  | DB2EDEF      |
| 297  | DB2TDEF      |
| 298  | FSEGDEF      |
| 299  | TSMDEF       |
| 300  | ENQMDEF      |
| 301  | TCPDEF       |
| 302  | DOCDEF       |
| 303  | FULL         |
| 304  | RELEASE      |
| 305  | PA1          |
| 306  | PA2          |



| Wert | EYUDA    |
|------|----------|
| 307  | PA3      |
| 308  | PF1      |
| 309  | PF2      |
| 310  | PF3      |
| 311  | PF4      |
| 312  | PF5      |
| 313  | PF6      |
| 314  | PF7      |
| 315  | PF8      |
| 316  | PF9      |
| 317  | PF10     |
| 318  | PF11     |
| 319  | PF12     |
| 320  | PF13     |
| 321  | PF14     |
| 322  | PF15     |
| 323  | PF16     |
| 324  | PF17     |
| 325  | PF18     |
| 326  | PF19     |
| 327  | PF20     |
| 328  | PF21     |
| 329  | PF22     |
| 330  | PF23     |
| 331  | PF24     |
| 332  | STANDARD |
| 333  | APPC     |
| 334  | BATCHDI  |
| 335  | BCHLU    |
| 336  | CONSOLE  |
| 337  | CONTLU   |
| 338  | INTLU    |
| 339  | LUTYPE2  |
| 340  | LUTYPE3  |
| 341  | LUTYPE4  |
| 342  | L3277    |
| 343  | L3284    |
| 344  | L3286    |
| 345  | PIPELINE |
| 346  | SCSPRINT |
| 347  | TLX      |

| Wert | EYUDA      |
|------|------------|
| 348  | TWX        |
| 349  | USERPROG   |
| 350  | 3270P      |
| 351  | 3275       |
| 352  | 3277       |
| 353  | 3277CM     |
| 354  | 3284       |
| 355  | 3284CM     |
| 356  | 3286       |
| 357  | 3286CM     |
| 358  | 3600       |
| 359  | 3614       |
| 360  | 3650       |
| 361  | 3653       |
| 362  | 3767       |
| 363  | 3767C      |
| 364  | 3767I      |
| 365  | 3770       |
| 366  | 3770B      |
| 367  | 3770C      |
| 368  | 3770I      |
| 369  | 3790       |
| 370  | BLINK      |
| 371  | REVERSE    |
| 372  | UNDERLINE  |
| 373  | INSTALL    |
| 374  | RESDEF     |
| 375  | RASINDSC   |
| 376  | RESTYPE    |
| 377  | SCOPETYP   |
| 378  | REBUILD    |
| 379  | RECONNECT  |
| 380  | CONNECTING |
| 381  | TWAIT      |
| 382  | NOTWAIT    |
| 383  | DISCONNING |
| 384  | CFTBL      |
| 385  | CF         |
| 386  | QUASIRENT  |
| 387  | THREADSAFE |
| 388  | PROCDEF    |

| Wert | EYUDA        |
|------|--------------|
| 389  | BAPPL        |
| 390  | ACTIVITY     |
| 391  | PROCESS      |
| 393  | PMWINDOW     |
| 394  | FULLSCREEN   |
| 395  | COM1         |
| 396  | COM2         |
| 397  | COM3         |
| 398  | COM4         |
| 399  | COM5         |
| 400  | COM6         |
| 401  | COM7         |
| 402  | COM8         |
| 403  | CLIENTAUTH   |
| 404  | RQMDEF       |
| 405  | 3270TERM     |
| 406  | 3270PRNT     |
| 407  | 3270DBTM     |
| 408  | 3270DBPR     |
| 409  | 3151TERM     |
| 410  | SEQTERM      |
| 411  | EOF          |
| 412  | EOT          |
| 413  | FEPOODEF     |
| 414  | FETRGDEF     |
| 415  | FENODDEF     |
| 416  | FEPRODEF     |
| 417  | DEBUG        |
| 418  | LEVSE        |
| 419  | KEY          |
| 420  | RBA          |
| 423  | EJCODEF      |
| 424  | EJDJDEF      |
| 425  | BASIC        |
| 426  | CERTIFICATE  |
| 427  | AUTOREGISTER |
| 428  | AUTOMATIC    |
| 429  | CLIENTCERT   |
| 430  | LINK3270     |
| 431  | FACILITY     |
| 432  | NOTOPEN      |

| Wert | EYUDA        |
|------|--------------|
| 433  | UNKNOWN      |
| 434  | REENTPROT    |
| 435  | NOREENTPROT  |
| 436  | ASSERTED     |
| 437  | MIRROR       |
| 438  | DPL          |
| 439  | ONCRPC       |
| 440  | WEB          |
| 441  | BRIDGE       |
| 442  | CICSBTS      |
| 443  | TDQUEUE      |
| 444  | TERMSTART    |
| 445  | XMRUN        |
| 446  | SOCKET       |
| 447  | RRS          |
| 448  | IIRQRECVR    |
| 449  | RZSTTRPT     |
| 450  | IIOP         |
| 451  | ECI          |
| 452  | HOTPOOL      |
| 453  | JVM          |
| 454  | HTASK        |
| 455  | IOERROR      |
| 456  | INVALIDFILE  |
| 457  | SUSPENDED    |
| 458  | SUSPENDING   |
| 459  | HISTORY      |
| 460  | REALTIME     |
| 461  | VELOCITY     |
| 462  | DISCRETIONRY |
| 463  | CONNECTED    |
| 464  | NOTCONNECTED |
| 465  | RESUMING     |
| 466  | URIMPDEF     |
| 467  | PIPEDEF      |
| 468  | WEBSVDEF     |
| 469  | STARTED      |
| 470  | STOPPED      |
| 471  | XPLINK       |
| 472  | SSL          |
| 473  | CICSAPI      |

| Wert | EYUDA       |
|------|-------------|
| 474  | OPENAPI     |
| 475  | EXCI        |
| 476  | RECREATED   |
| 477  | IPCONDEF    |
| 478  | SAME        |
| 479  | SUPPORTED   |
| 480  | SOSGCDSA    |
| 481  | JCL         |
| 482  | REGION      |
| 483  | IEFUSI      |
| 484  | SMF         |
| 485  | ABOVEBAR    |
| 486  | LIBDEF      |
| 487  | ENABLED     |
| 488  | DISABLED    |
| 489  | SNA         |
| 490  | IPIC        |
| 491  | PARTIAL     |
| 492  | ATOMDEF     |
| 493  | BUNDDEF     |
| 494  | MQCONDEF    |
| 495  | JVMSVDEF    |
| 496  | THREADED    |
| 497  | CSDLIST     |
| 498  | CSDGROUP    |
| 499  | CSDINLST    |
| 500  | CSDINGRP    |
| 501  | LOCKED      |
| 502  | UOW         |
| 503  | SIT         |
| 504  | EYU         |
| 505  | TABLE       |
| 506  | SYSIN       |
| 507  | WUI         |
| 508  | REQUIRED    |
| 509  | COMBINED    |
| 510  | LNQUEUE     |
| 511  | LNGOAL      |
| 512  | GROUPRESYNC |
| 513  | INSTALLING  |
| 514  | DISCARDING  |

| Wert | EYUDA            |
|------|------------------|
| 566  | SUMMUNLIKE       |
| 567  | ATTLSAWARE       |
| 568  | MQMON            |
| 640  | MODBNOTACTIVE    |
| 641  | PRIVATEMETHOD    |
| 642  | METHODLOCKED     |
| 643  | ENVIRONTIMEOUT   |
| 644  | ENVIRONCANCEL    |
| 645  | STARTFAILED      |
| 646  | STACKTOOLARGE    |
| 647  | OVERFLOWOSSBFAIL |
| 648  | BADMALTRANSFER   |
| 736  | TRANSMITTIMEOUT  |
| 737  | EXECUTETIMEOUT   |
| 738  | TRANSMITFAILED   |
| 739  | TARGETUNAVAIL    |
| 740  | SECVIOLATION     |
| 741  | MALRETURNFAILED  |
| 742  | MAXHISTORYRECS   |
| 743  | REQNOTPROCED     |
| 752  | NOTAUTHORIZED    |
| 753  | SECINTERFACEFAIL |
| 754  | TRUSTED          |
| 755  | ETDSA            |
| 756  | EMPTY            |
| 757  | INCOMPLETE       |
| 758  | INVALIDSCOPE     |
| 759  | DISABLING        |
| 760  | ENABLING         |
| 761  | SOMEDISABLED     |
| 762  | BYTE             |
| 763  | KILOBYTE         |
| 764  | MEGABYTE         |
| 765  | GIGABYTE         |
| 766  | THOUSAND         |
| 767  | SECOND           |
| 768  | MILLISECOND      |
| 769  | MICROSECOND      |
| 770  | ABEND            |
| 771  | PLATFORM         |
| 772  | APPLICATION      |

| Wert | EYUDA       |
|------|-------------|
| 773  | APPLMAJVER  |
| 774  | APPLMINVER  |
| 775  | APPLMICVER  |
| 776  | OPERATION   |
| 777  | IMPORTONLY  |
| 778  | AVAILABLE   |
| 779  | UNAVAILABLE |
| 780  | SOMEAVAIL   |
| 781  | 520         |
| 782  | MQMONDEF    |
| 783  | TASK        |

## Allgemeine Werte vom Typ EYUDA in alphabetischer Reihenfolge

In diesem Abschnitt sind die allgemeinen Werte vom Typ EYUDA in alphabetischer Reihenfolge nach Zeichenwert aufgeführt.

| EYUDA        | Wert |
|--------------|------|
| AASTERISK    | 165  |
| ABEND        | 770  |
| ABOVE        | 75   |
| ABOVEBAR     | 485  |
| ACTIVE       | 45   |
| ACTIVITY     | 390  |
| ADJACENT     | 118  |
| AFTER        | 172  |
| ALL          | 145  |
| ALLREQS      | 173  |
| ALWAYS       | 266  |
| ANY          | 146  |
| APM          | 36   |
| APPC         | 333  |
| APPLICATION  | 772  |
| APPLMAJVER   | 773  |
| APPLMICVER   | 775  |
| APPLMINVER   | 774  |
| ASA          | 174  |
| ASIS         | 177  |
| ASSEMBLER    | 178  |
| ASSERTED     | 436  |
| ASSOCIATIONS | 293  |
| ATOMDEF      | 492  |

| EYUDA        | Wert |
|--------------|------|
| AUTO         | 257  |
| AUTOMATIC    | 428  |
| AUTOREGISTER | 427  |
| AVAILABLE    | 778  |
| AVG          | 150  |
| BACKOUTONLY  | 179  |
| BAPPL        | 389  |
| BASIC        | 425  |
| BATCHDI      | 334  |
| BCHLU        | 335  |
| BEFORE       | 180  |
| BELOW        | 76   |
| BLANK        | 166  |
| BLINK        | 370  |
| BLUE         | 181  |
| BOTH         | 169  |
| BRIDGE       | 441  |
| BUNDDDEF     | 493  |
| BYTE         | 762  |
| C            | 182  |
| CDSA         | 79   |
| CERTIFICATE  | 426  |
| CF           | 385  |
| CFTBL        | 384  |
| CHARSTR      | 44   |
| CHECK        | 162  |
| CICS         | 87   |
| CICSAPI      | 473  |
| CICSBTS      | 442  |
| CICSSYS      | 103  |
| CLEARCONV    | 183  |
| CLIENTAUTH   | 403  |
| CLIENTCERT   | 429  |
| CLOSE        | 184  |
| CNT          | 151  |
| COBOL        | 185  |
| COLD         | 186  |
| COLDONLY     | 267  |
| COM1         | 395  |
| COM2         | 396  |
| COM3         | 397  |



| EYUDA         | Wert |
|---------------|------|
| COM4          | 398  |
| COM5          | 399  |
| COM6          | 400  |
| COM7          | 401  |
| COM8          | 402  |
| CONACT        | 126  |
| CONNDEF       | 275  |
| CONNECTED     | 463  |
| CONNECTING    | 380  |
| CONSOLE       | 336  |
| CONTINUE      | 270  |
| CONTLU        | 337  |
| CREATING      | 120  |
| CSDLIST       | 497  |
| CSDGROUP      | 498  |
| CSDINLST      | 500  |
| CTABL         | 97   |
| CYCLIC        | 187  |
| DATA          | 168  |
| DB2CDEF       | 295  |
| DB2EDEF       | 296  |
| DB2TDEF       | 297  |
| DEBUG         | 417  |
| DEFAULT       | 113  |
| DEFERRED      | 188  |
| DELIMIT       | 58   |
| DIP           | 189  |
| DISABLED      | 488  |
| DISCONNING    | 383  |
| DISCRETIONARY | 462  |
| DISK          | 190  |
| DOCDEF        | 302  |
| DORMANT       | 115  |
| DPL           | 438  |
| DROP          | 110  |
| DSA           | 78   |
| DUPLICATE     | 143  |
| DYNAM         | 258  |
| ECDSA         | 83   |
| ECI           | 451  |
| EDSA          | 82   |

| <b>EYUDA</b> | <b>Wert</b> |
|--------------|-------------|
| EJCODEF      | 423         |
| EJDJDEF      | 424         |
| ELPA         | 86          |
| ENABLED      | 487         |
| END          | 117         |
| ENQMDEF      | 300         |
| EOS          | 191         |
| EOF          | 411         |
| EOT          | 412         |
| EQ           | 16          |
| ERDSA        | 85          |
| ESDSA        | 155         |
| ESSS         | 125         |
| EUDSA        | 84          |
| EXECUTE      | 161         |
| EXPLICIT     | 102         |
| EXTA         | 192         |
| EXTRA        | 259         |
| FACILITY     | 431         |
| FALSE        | 8           |
| FENODDEF     | 415         |
| FEPODEF      | 413         |
| FEPRODEF     | 416         |
| FETRGDEF     | 414         |
| FILE         | 193         |
| FILEDEF      | 276         |
| FIRSTREF     | 194         |
| FORCE        | 107         |
| FROZEN       | 144         |
| FSEGDEF      | 298         |
| FULL         | 303         |
| FULLSCREEN   | 394         |
| GE           | 153         |
| GIGABYTE     | 765         |
| GLOBAL       | 27          |
| GOAL         | 53          |
| GREEN        | 195         |
| GT           | 18          |
| HIGH         | 21          |
| HISTORY      | 459         |
| HOTPOOL      | 452         |

| EYUDA       | Wert |
|-------------|------|
| HS          | 14   |
| HTASK       | 454  |
| HW          | 13   |
| IDENTIFY    | 196  |
| IEFUSI      | 483  |
| IGNORE      | 197  |
| IIOP        | 450  |
| IIRQRECVR   | 448  |
| IMMEDIATE   | 23   |
| INACTIVE    | 46   |
| IND         | 261  |
| INDEX       | 167  |
| INDIRECT    | 92   |
| INHERIT     | 101  |
| INITIAL     | 198  |
| INOUT       | 199  |
| INPUT       | 200  |
| INSTALL     | 373  |
| INSTALLED   | 99   |
| INTLU       | 338  |
| INTRA       | 260  |
| INVALID     | 6    |
| INVALIDFILE | 456  |
| IOERROR     | 455  |
| IPCONDEF    | 475  |
| JCL         | 481  |
| JRNLDEF     | 277  |
| JRNMDEF     | 278  |
| JVM         | 453  |
| JVMVDEF     | 495  |
| KEEP        | 105  |
| KEY         | 419  |
| KILOBYTE    | 763  |
| LE          | 152  |
| LEAVE       | 201  |
| LEVSE       | 418  |
| LE370       | 202  |
| LFILE       | 95   |
| LIBDEF      | 486  |
| LINEAR      | 203  |
| LINK        | 204  |

| EYUDA      | Wert |
|------------|------|
| LINKACTIVE | 123  |
| LINKDOWN   | 124  |
| LINK3270   | 430  |
| LMS        | 205  |
| LNGOAL     | 511  |
| LNQUEUE    | 510  |
| LOCAL      | 111  |
| LOCKED     | 501  |
| LOGICAL    | 206  |
| LOGOFF     | 207  |
| LOGON      | 60   |
| LOSTCMAS   | 163  |
| LOSTCON    | 119  |
| LOSTMAS    | 164  |
| LOW        | 20   |
| LPA        | 81   |
| LRU        | 208  |
| LS         | 10   |
| LSRDEF     | 279  |
| LT         | 19   |
| LTRAN      | 50   |
| LUNAME     | 55   |
| LUTYPE2    | 339  |
| LUTYPE3    | 340  |
| LUTYPE4    | 341  |
| LU61       | 90   |
| LU62       | 91   |
| LW         | 11   |
| L3277      | 342  |
| L3284      | 343  |
| L3286      | 344  |
| MACHINE    | 209  |
| MAPDEF     | 280  |
| MAPPL      | 74   |
| MAX        | 149  |
| MAXTASK    | 130  |
| MCICS      | 64   |
| MCONN      | 67   |
| MDBX       | 66   |
| MEGABYTE   | 764  |
| MEMBERS    | 294  |

| EYUDA        | Wert |
|--------------|------|
| MESSAGE      | 210  |
| MFILE        | 68   |
| MGLBL        | 65   |
| MICROSECOND  | 769  |
| MILLISECOND  | 768  |
| MIN          | 148  |
| MIRROR       | 437  |
| MIXIDPE      | 211  |
| MJRNL        | 69   |
| MOD          | 212  |
| MODIFYREQS   | 213  |
| MPROG        | 70   |
| MPSYNCCR     | 292  |
| MQCONDEF     | 494  |
| MRM          | 37   |
| MRO          | 93   |
| MSRE         | 214  |
| MTDQS        | 72   |
| MTERM        | 71   |
| MTRAN        | 73   |
| N_A          | 0    |
| NA           | 0    |
| NAME         | 106  |
| NE           | 17   |
| NETBIOS      | 170  |
| NEUTRAL      | 215  |
| NEVER        | 265  |
| NEW          | 216  |
| NM           | 12   |
| NO           | 2    |
| NOCOPY       | 77   |
| NOFORCE      | 217  |
| NONE         | 108  |
| NONVTAM      | 218  |
| NOREENTPORT  | 435  |
| NORMAL       | 22   |
| NOSHUT       | 26   |
| NOTAPPLIC    | 94   |
| NOTCONNECTED | 464  |
| NOTOPEN      | 432  |
| NOTWAIT      | 382  |

| EYUDA      | Wert |
|------------|------|
| NULL       | 43   |
| OFF        | 4    |
| OLD        | 219  |
| ON         | 3    |
| ONCRPC     | 439  |
| ONLY       | 220  |
| OPEN       | 221  |
| OPENAPI    | 474  |
| OPERATION  | 776  |
| OPID       | 222  |
| OUTPUT     | 223  |
| PARTDEF    | 281  |
| PA1        | 305  |
| PA2        | 306  |
| PA3        | 307  |
| PCONV      | 59   |
| PENDING    | 100  |
| PERMANENT  | 63   |
| PERSISTENT | 224  |
| PF1        | 308  |
| PF10       | 317  |
| PF11       | 318  |
| PF12       | 319  |
| PF13       | 320  |
| PF14       | 321  |
| PF15       | 322  |
| PF16       | 323  |
| PF17       | 324  |
| PF18       | 325  |
| PF19       | 326  |
| PF2        | 309  |
| PF20       | 327  |
| PF21       | 328  |
| PF22       | 329  |
| PF23       | 330  |
| PF24       | 331  |
| PF3        | 310  |
| PF4        | 311  |
| PF5        | 213  |
| PF6        | 313  |
| PF7        | 314  |

| EYUDA       | Wert |
|-------------|------|
| PF8         | 315  |
| PF9         | 316  |
| PHYSICAL    | 225  |
| PINK        | 226  |
| PIPEDEF     | 467  |
| PIPELINE    | 435  |
| PLATFORM    | 771  |
| PLI         | 227  |
| PMWINDOW    | 393  |
| POOL        | 49   |
| PRIMARY     | 141  |
| PRINTER     | 228  |
| PROCDEF     | 388  |
| PROCESS     | 391  |
| PROFDEF     | 283  |
| PROGDEF     | 284  |
| PROMPT      | 269  |
| PRTNDEF     | 282  |
| QUASIRENT   | 386  |
| QUEUE       | 54   |
| QUIESCE     | 140  |
| QUIESCED    | 122  |
| QUIESCING   | 48   |
| RACFGID     | 41   |
| RASINDSC    | 375  |
| RBA         | 420  |
| RDSA        | 156  |
| READONLY    | 89   |
| REALTIME    | 460  |
| REBUILD     | 378  |
| RECONNECT   | 379  |
| RECOVERY    | 230  |
| RED         | 229  |
| REENTPROT   | 434  |
| REGION      | 482  |
| REJECT      | 231  |
| RELATED     | 263  |
| RELEASE     | 304  |
| RELEASESESS | 232  |
| REMOTE      | 112  |
| REMOVE      | 114  |

| <b>EYUDA</b> | <b>Wert</b> |
|--------------|-------------|
| REMOVING     | 121         |
| REREAD       | 233         |
| RESDEF       | 374         |
| RESET        | 127         |
| RESTYPE      | 376         |
| RESUMING     | 465         |
| REVERSE      | 371         |
| RFILE        | 96          |
| RPG          | 234         |
| RQMDEF       | 404         |
| RRS          | 447         |
| RTADEF       | 273         |
| RTRAN        | 51          |
| RZSTTRPT     | 449         |
| SAM          | 35          |
| SCOPETYP     | 337         |
| SCS          | 235         |
| SCSPRINT     | 346         |
| SDSA         | 154         |
| SECOND       | 767         |
| SECONDARY    | 142         |
| SECURITY     | 236         |
| SEQTERM      | 410         |
| SESSDEF      | 285         |
| SHR          | 237         |
| SHUT         | 25          |
| SHUTDOWN     | 272         |
| SIGNID       | 40          |
| SIGNON       | 61          |
| SKIP         | 238         |
| SMF          | 484         |
| SOCKET       | 446         |
| SOMEAVAIL    | 780         |
| SOSABOVE     | 159         |
| SOSBELOW     | 158         |
| SOSCDSA      | 133         |
| SOSECDSA     | 135         |
| SOSERDSA     | 136         |
| SOSESDSA     | 138         |
| SOSEUDSA     | 134         |
| SOSGCDSA     | 480         |



| EYUDA      | Wert |
|------------|------|
| SOSMVS     | 157  |
| SOSRDSA    | 139  |
| SOSSDSA    | 137  |
| SOSUDSA    | 132  |
| SPECIFIC   | 239  |
| SSL        | 472  |
| STALLED    | 131  |
| STANDARD   | 332  |
| START      | 116  |
| STARTED    | 332  |
| STARTIO    | 240  |
| STARTUP    | 241  |
| STAT       | 262  |
| STATDEF    | 274  |
| STOPPED    | 470  |
| STRFIELD   | 242  |
| SUM        | 147  |
| SUMMUNLIKE | 566  |
| SUSPEND    | 30   |
| SUSPENDED  | 457  |
| SUSPENDING | 458  |
| SYSDEFAULT | 243  |
| SYSDUMP    | 128  |
| SYSGROUP   | 104  |
| SYSTEM     | 28   |
| TAKEOVER   | 24   |
| TAPE       | 244  |
| TARGET     | 264  |
| TCPDEF     | 301  |
| TCPIP      | 171  |
| TDQDEF     | 286  |
| TDQUEUE    | 443  |
| TERMDEF    | 287  |
| TERMIN     | 39   |
| TERMINAL   | 245  |
| TERMINATE  | 271  |
| TERMSTART  | 444  |
| THOUSAND   | 766  |
| THREADED   | 496  |
| THREADSAFE | 387  |
| THRESHOLD  | 34   |

| EYUDA       | Wert |
|-------------|------|
| TLX         | 347  |
| TRANDEF     | 288  |
| TRANDUMP    | 129  |
| TRANID      | 38   |
| TRANSACTION | 246  |
| TRANSIENT   | 247  |
| TRNCLDEF    | 289  |
| TRUE        | 7    |
| TSMDEF      | 299  |
| TSQDEF      | 290  |
| TURQUOISE   | 248  |
| TWAIT       | 381  |
| TWX         | 348  |
| TYPTMDEF    | 291  |
| U           | 249  |
| UDSA        | 80   |
| UNASSIGNED  | 109  |
| UNAVAILABLE | 779  |
| UNCONDREL   | 250  |
| UNDERLINE   | 372  |
| UNKNOWN     | 433  |
| UOW         | 502  |
| UPDATEONLY  | 251  |
| URIMPDEF    | 466  |
| USER        | 88   |
| USERID      | 42   |
| USERPROG    | 349  |
| UTABL       | 98   |
| VALID       | 5    |
| VALUE       | 33   |
| VB          | 252  |
| VELOCITY    | 461  |
| VERIFY      | 352  |
| VHS         | 15   |
| VLS         | 9    |
| VTAM        | 254  |
| WAITING     | 47   |
| WARMONLY    | 268  |
| WEB         | 440  |
| WEBSVDEF    | 468  |
| XMRUN       | 445  |

| EYUDA    | Wert |
|----------|------|
| XPLINK   | 471  |
| YELLOW   | 255  |
| YES      | 1    |
| 3151TERM | 409  |
| 3270     | 256  |
| 3270DBPR | 408  |
| 3270DBTM | 407  |
| 3270P    | 350  |
| 3270PRNT | 406  |
| 3270TERM | 405  |
| 3275     | 351  |
| 3277     | 352  |
| 3277CM   | 353  |
| 3284     | 354  |
| 3284CM   | 355  |
| 3286     | 356  |
| 3286CM   | 357  |
| 3600     | 358  |
| 3614     | 359  |
| 3650     | 360  |
| 3653     | 361  |
| 3767     | 362  |
| 3767C    | 363  |
| 3767I    | 364  |
| 3770     | 365  |
| 3770B    | 366  |
| 3770C    | 367  |
| 3770I    | 368  |
| 3790     | 369  |

---

## RESPONSE-Werte vom Typ EYUDA in numerischer Reihenfolge

In diesem Abschnitt sind die RESPONSE-Werte vom Typ EYUDA in numerischer Reihenfolge aufgeführt.

| Wert | EYUDA       |
|------|-------------|
| 1024 | OK          |
| 1025 | SCHEDULED   |
| 1026 | NOTFOUND    |
| 1027 | NODATA      |
| 1028 | INVALIDPARM |
| 1029 | FAILED      |

| Wert | EYUDA        |
|------|--------------|
| 1030 | ENVIRONERROR |
| 1031 | NOTPERMIT    |
| 1032 | BUSY         |
| 1033 | SERVERGONE   |
| 1034 | NOTAVAILABLE |
| 1035 | VERSIONINVL  |
| 1036 | INVALIDCMD   |
| 1037 | WARNING      |
| 1038 | TABLEERROR   |
| 1039 | INCOMPATIBLE |
| 1040 | INUSE        |
| 1041 | INVALIDATA   |
| 1042 | DUPE         |

---

## RESPONSE-Werte vom Typ EYUDA in alphabetischer Reihenfolge

In diesem Abschnitt sind die RESPONSE-Werte vom Typ EYUDA in alphabetischer Reihenfolge aufgeführt.

| Wert         | EYUDA |
|--------------|-------|
| BUSY         | 1032  |
| DUPE         | 1042  |
| ENVIRONERROR | 1030  |
| FAILED       | 1029  |
| INCOMPATIBLE | 1039  |
| INUSE        | 1040  |
| INVALIDATA   | 1041  |
| INVALIDCMD   | 1036  |
| INVALIDPARM  | 1028  |
| NODATA       | 1027  |
| NOTAVAILABLE | 1034  |
| NOTFOUND     | 1026  |
| NOTPERMIT    | 1031  |
| OK           | 1024  |
| SCHEDULED    | 1025  |
| SERVERGONE   | 1033  |
| TABLEERROR   | 1038  |
| VERSIONINVL  | 1035  |
| WARNING      | 1037  |

---

## REASON-Werte vom Typ EYUDA in numerischer Reihenfolge

In diesem Abschnitt sind die REASON-Werte vom Typ EYUDA in numerischer Reihenfolge aufgeführt.

| Wert | EYUDA        |
|------|--------------|
| 1280 | THREAD       |
| 1281 | OBJECT       |
| 1282 | CONTEXT      |
| 1283 | RESULT       |
| 1284 | COUNT        |
| 1285 | LENGTH       |
| 1286 | FILTER       |
| 1287 | NOTFILTER    |
| 1288 | FORWARD      |
| 1289 | BACKWARD     |
| 1290 | POSITION     |
| 1291 | DELAY        |
| 1292 | NOTIFICATION |
| 1293 | SIGNONPARM   |
| 1294 | SCOPE        |
| 1295 | RESOURCE     |
| 1296 | FROM         |
| 1297 | TO           |
| 1298 | INTO         |
| 1299 | CRITERIA     |
| 1300 | BY           |
| 1301 | ACTION       |
| 1302 | ECB          |
| 1303 | SENTINEL     |
| 1304 | FEEDBACK     |
| 1305 | EVENT        |
| 1306 | TOKEN        |
| 1307 | MODIFY       |
| 1308 | VIEW         |
| 1309 | FIELDS       |
| 1310 | ATTRIBUTE    |
| 1311 | FROMCV       |
| 1312 | TOCHAR       |
| 1313 | FROMCHAR     |
| 1314 | TOCV         |
| 1315 | PARM         |
| 1316 | PARMLN       |

| Wert | EYUDA        |
|------|--------------|
| 1317 | SUMOPT       |
| 1318 | TYPE         |
| 1319 | DATALength   |
| 1320 | SOLRESOURCE  |
| 1321 | SOCRESOURCE  |
| 1322 | SOERESOURCE  |
| 1323 | MAINTPOINT   |
| 1324 | SYSNOTACT    |
| 1325 | SYSVLBAD     |
| 1326 | SYSNOTLCL    |
| 1327 | CICSRELBAD   |
| 1328 | ARMNOTREG    |
| 1329 | ARMNOTACT    |
| 1330 | ARMPOLCHK    |
| 1331 | ABENDED      |
| 1332 | CPSMSYSTEM   |
| 1333 | CPSMVERSION  |
| 1334 | CPSMAPI      |
| 1335 | NOTSUPPORTED |
| 1336 | NOTVSNCONN   |
| 1337 | INVALIDATTR  |
| 1338 | APITASKERR   |
| 1339 | CPSMSERVER   |
| 1340 | APITASK      |
| 1341 | PLEXMGR      |
| 1342 | REQTIMEOUT   |
| 1344 | AREATOOSMALL |
| 1345 | USRID        |
| 1348 | VERSION      |
| 1352 | FILTERMATCH  |
| 1353 | INVALIDOBJ   |
| 1354 | INVALIDVER   |
| 1355 | TASKDATAKEY  |
| 1356 | INVALIDVERB  |
| 1357 | NOSTORAGE    |
| 1358 | NOSERVICE    |
| 1359 | EXCEPTION    |
| 1360 | INVALIDEVT   |
| 1361 | DATAERROR    |
| 1362 | CMAS         |
| 1363 | FIRST        |

| Wert | EYUDA        |
|------|--------------|
| 1364 | NEXT         |
| 1365 | EXPIRED      |
| 1366 | WORKLOAD     |
| 1367 | ACTIONPARM   |
| 1368 | CICSNAME     |
| 1369 | MAXRECORDS   |
| 1370 | QUERY        |
| 1371 | EXPAND       |
| 1372 | XINTO        |
| 1373 | XLENGTH      |
| 1374 | APICMD       |
| 1375 | CSDAPI       |
| 1376 | TEST         |
| 1377 | QUERYERROR   |
| 1378 | NOTPROCESSED |
| 1379 | INVALIDTCB   |

## REASON-Werte vom Typ EYUDA in alphabetischer Reihenfolge

In diesem Abschnitt sind die REASON-Werte vom Typ EYUDA in alphabetischer Reihenfolge aufgeführt.

| Wert         | EYUDA |
|--------------|-------|
| ABENDED      | 1331  |
| ACTION       | 1301  |
| ACTIONPARM   | 1367  |
| APITASK      | 1340  |
| APITASKERR   | 1338  |
| AREATOOSMALL | 1344  |
| ARMNOTACT    | 1329  |
| ARMNOTREG    | 1328  |
| ARMPOLCHK    | 1330  |
| ATTRIBUTE    | 1310  |
| BACKWARD     | 1289  |
| BY           | 1300  |
| CICSNAME     | 1368  |
| CICSRELBAD   | 1327  |
| CMAS         | 1362  |
| CONTEXT      | 1282  |
| COUNT        | 1284  |
| CPSMAPI      | 1334  |
| CPSMSERVER   | 1339  |

| Wert         | EYUDA |
|--------------|-------|
| CPSMSYSTEM   | 1332  |
| CPSMVERSION  | 1333  |
| CRITERIA     | 1299  |
| CSDAPI       | 1375  |
| DATAERROR    | 1361  |
| DATALength   | 1319  |
| DELAY        | 1291  |
| ECB          | 1302  |
| EVENT        | 1305  |
| EXCEPTION    | 1359  |
| EXPIRED      | 1365  |
| FEEDBACK     | 1394  |
| FIELDS       | 1309  |
| FILTER       | 1286  |
| FITLERMATCH  | 1352  |
| FIRST        | 1363  |
| FORWARD      | 1288  |
| FROM         | 1296  |
| FROMCHAR     | 1313  |
| FROMCV       | 1311  |
| INTO         | 1298  |
| INVALIDATTR  | 1337  |
| INVALIDEVT   | 1360  |
| INVALIDOBJ   | 1353  |
| INVALIDTCB   | 1379  |
| INVALIDVER   | 1354  |
| INVALIDVERB  | 1356  |
| LENGTH       | 1285  |
| MAINTPOINT   | 1323  |
| MAXRECORDS   | 1369  |
| MODIFY       | 1307  |
| NEXT         | 1364  |
| NOSERVICE    | 1358  |
| NOSTORAGE    | 1357  |
| NOTFILTER    | 1287  |
| NOTIFICATION | 1292  |
| NOTPROCESSED | 1378  |
| NOTSUPPORTED | 1335  |
| NOTVSNCONN   | 1336  |
| OBJECT       | 1281  |
| PARM         | 1315  |



| Wert        | EYUDA |
|-------------|-------|
| PARMLEN     | 1316  |
| PLEXMGR     | 1341  |
| POSITION    | 1290  |
| QUERY       | 1370  |
| REQTIMEOUT  | 1342  |
| RESOURCE    | 1295  |
| RESULT      | 1283  |
| SCOPE       | 1294  |
| SENTINEL    | 1303  |
| SIGNONPARM  | 1293  |
| SOCRESOURCE | 1321  |
| SOERESOURCE | 1322  |
| SOLRESOURCE | 1320  |
| SUMOPT      | 1317  |
| SYSVLBAD    | 1325  |
| SYSNOTACT   | 1324  |
| SYSNOTLCL   | 1326  |
| TASKDATAKEY | 1355  |
| THREAD      | 1280  |
| TO          | 1297  |
| TOCHAR      | 1312  |
| TOCV        | 1314  |
| TOKEN       | 1306  |
| TYPE        | 1318  |
| USRID       | 1345  |
| VERSION     | 1348  |
| VIEW        | 1308  |
| WORKLOAD    | 1366  |
| XINTO       | 1372  |
| XLENGTH     | 1373  |



---

## 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 Neuausgabe 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 Beispielpprogramme 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 Beispielpprogramme 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 Beispielpprogramme 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 Beispielpprogramme 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 Verwendungsfähigkeit 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

## A

ADDRESS, Befehl  
  Übersicht der RESPONSE-Werte 155  
Argumentwerte  
  für die Befehlsschnittstelle  
    mit Assembler 6  
    mit C 4  
    mit COBOL 3  
    mit PL/I 5  
    Typen 1  
  für die Laufzeitschnittstelle 7  
Assemblerprogramme  
  Argumentwerte für 6  
Attribute, Ressourcentabelle  
  umsetzen  
    mit EYUVALUE 9

## B

Befehle angeben  
  über die Befehlsschnittstelle 1  
  über die Laufzeitschnittstelle 6  
Befehlsantworten  
  Übersicht 155  
Befehlsschnittstelle  
  API-Befehle angeben 1  
  Argumentwerte angeben 1

## C

C-Programme  
  Argumentwerte für 4  
CANCEL, Befehl  
  Übersicht der RESPONSE-Werte 155  
COBOL-Programme  
  Argumentwerte für 3  
CONNECT, Befehl  
  Übersicht der RESPONSE-Werte 155  
COPY, Befehl  
  Übersicht der RESPONSE-Werte 155  
CREATE, Befehl  
  Übersicht der RESPONSE-Werte 155  
CVDA-Werte umsetzen 9

## D

DELETE, Befehl  
  Übersicht der RESPONSE-Werte 156  
DISCARD, Befehl  
  Übersicht der RESPONSE-Werte 156  
DISCONNECT, Befehl  
  Übersicht der RESPONSE-Werte 156

## E

ECB-Feld  
  anfordern 21  
Ereignissteuerblock (ECB)  
  anfordern 21

EXPAND, Befehl 46  
EYU9XESV, Sicherheitsroutine  
  Optionen bei CONNECT 26, 27  
EYUDA-Werte  
  umsetzen 9  
  Zusammenfassung 161  
EYUVALUE, Funktion  
  Beschreibung 9

## F

FEEDBACK, Befehl  
  Übersicht der RESPONSE-Werte 156  
FETCH, Befehl  
  Übersicht der RESPONSE-Werte 156  
Filterausdruck  
  angeben  
    bei GET 62  
    bei SPECIFY FILTER 130  
    mit PERFORM OBJECT 94  
Format von Befehlen  
  über die Befehlsschnittstelle 1  
  über die Laufzeitschnittstelle 6

## G

GET, Befehl  
  Übersicht der RESPONSE-Werte 156  
GETDEF, Befehl  
  Übersicht der RESPONSE-Werte 157  
GROUP, Befehl  
  Übersicht der RESPONSE-Werte 157

## L

Längenoptionen angeben 10  
LISTEN, Befehl  
  Übersicht der RESPONSE-Werte 157  
LOCATE, Befehl  
  Übersicht der RESPONSE-Werte 157

## M

MARK, Befehl  
  Übersicht der RESPONSE-Werte 157  
Modifikationsausdruck  
  angeben  
    bei SET 126  
    bei UPDATE 146  
MVS-Einschränkungen 8

## O

ORDER, Befehl  
  Übersicht der RESPONSE-Werte 158

## P

Parameterausdruck  
  angeben  
    bei CREATE 35  
    bei GET 63  
    bei MARK 88  
    bei PERFORM OBJECT 95  
    bei PERFORM SET 102  
    bei REMOVE 121  
    bei UNMARK 142  
    bei UPDATE 146  
PERFORM OBJECT, Befehl  
  Übersicht der RESPONSE-Werte 158  
PERFORM SET, Befehl  
  Übersicht der RESPONSE-Werte 158  
PL/I-Programme  
  Argumentwerte für 5

## Q

QUALIFY, Befehl  
  Übersicht der RESPONSE-Werte 158  
QUERY, Befehl  
  Übersicht der RESPONSE-Werte 158

## R

REASON, Option  
  Beschreibung 10  
  Übersicht der Werte 155  
RECEIVE, Befehl  
  Übersicht der RESPONSE-Werte 159  
REFRESH, Befehl  
  Übersicht der RESPONSE-Werte 159  
REMOVE, Befehl  
  Übersicht der RESPONSE-Werte 159  
RESPONSE, Befehl  
  Übersicht 155  
RESPONSE, Option  
  Beschreibung 10  
  Übersicht der Werte 155  
Ressourcentabelle  
  Umsetzen von Attributen  
    mit EYUVALUE 9  
REXX-Laufzeitschnittstelle  
  API-Befehle angeben 6  
  Argumentwerte angeben 7  
  Befehle 16

## S

Sentinel-Feld  
  anfordern 22  
SET, Befehl  
  Übersicht der RESPONSE-Werte 159  
Sicherheit  
  Optionen bei CONNECT 26, 27

- Sortierausdruck
  - angeben
    - bei ORDER 91
    - bei SPECIFY VIEW 134
- SPECIFY FILTER, Befehl
  - Übersicht der RESPONSE-Werte 159
- SPECIFY VIEW, Befehl
  - Übersicht der RESPONSE-Werte 160
- Sprachaspekte
  - allgemein 10

## T

- TERMINATE, Befehl
  - Übersicht der RESPONSE-Werte 160
- TRANSLATE, Befehl
  - Übersicht der RESPONSE-Werte 160

## U

- Umsetzen
  - Ressourcentabellenattribute
    - mit EYUVALUE 9
- UNMARK, Befehl
  - Übersicht der RESPONSE-Werte 160
- UPDATE, Befehl
  - Übersicht der RESPONSE-Werte 160

## Z

- Zusammenfassungsausdruck
  - angeben 75
- Zusammenfassungsoptionen
  - angeben 75



