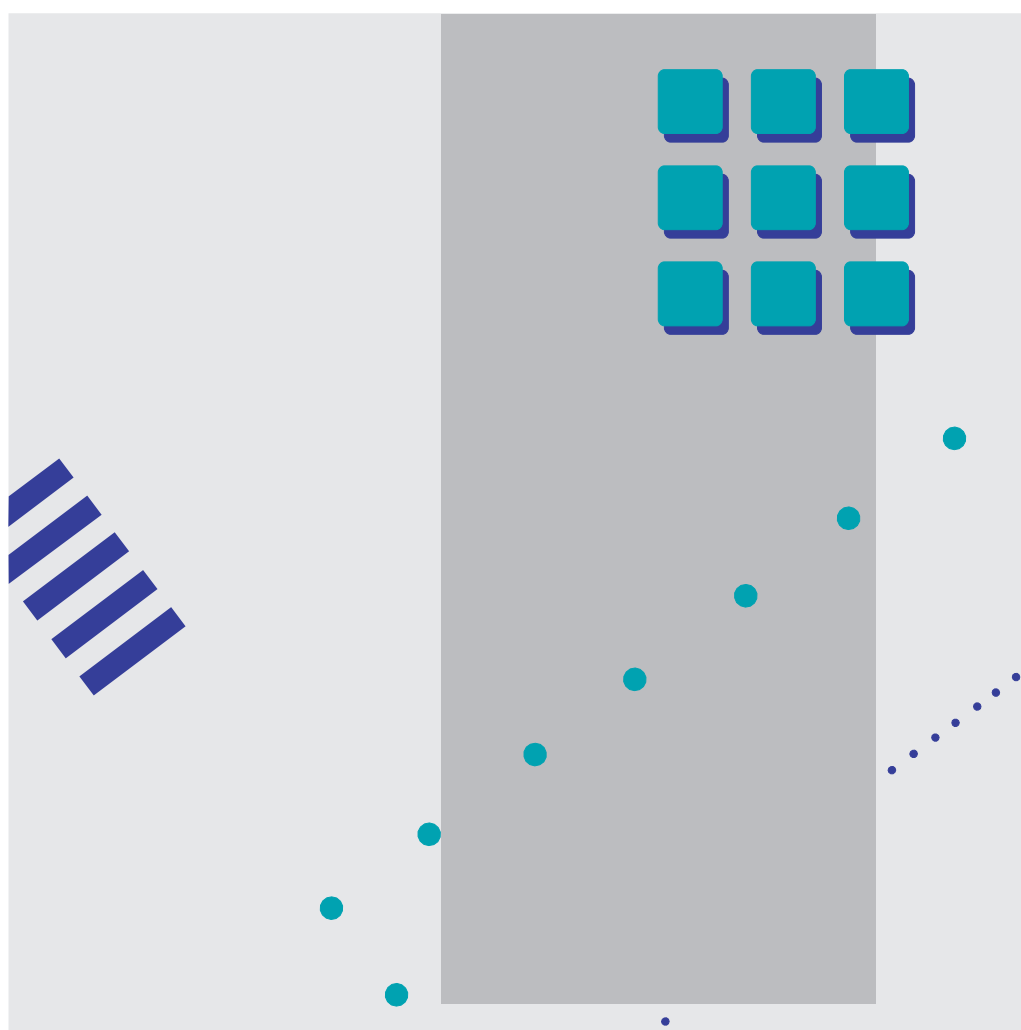


Master Index

Version 3 Release 5



TPNS
Teleprocessing Network Simulator



Master Index

Version 3 Release 5

Note!

Before using this information and the product it supports, be sure to read the general information under “Notices” on page v.

Fourth Edition (September 1996)

This document applies to the Teleprocessing Network Simulator Version 3 Release 5 (program number 5688-121), an IBM licensed program, which runs under the following operating systems:

MVS/370 (MVS/SP Version 1 or later)

MVS/Extended Architecture (MVS/SP Version 2 or later)

MVS/Enterprise System Architecture (MVS/SP Version 3 or later)

OS/390

Any VM system (with or without High Performance Option) that supports Group Control System (GCS).

Publications are not stocked at the address given below. If you want more IBM publications, ask your IBM representative or write to the IBM branch office serving your locality.

A form for readers comments has been provided at the back of this document. If this form has been removed, you may address comments to IBM Corporation, TPNS Development and Support, Dept. LWZA/B502, P. O. Box 12195, 4205 S. Miami Blvd., Research Triangle Park, North Carolina 27709, U.S.A. IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

© **Copyright International Business Machines Corporation 1991, 1996. All rights reserved.**

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Contents

Notices	v
Trademarks and Service Marks	v
 About This Book	vii
Who Should Use This Book	vii
How to Use This Book	vii
Where to Find More Information	vii
 Introduction	1
What is Teleprocessing Network Simulator (TPNS)?	1
How to Use This Index	1
 Master Bibliography	3
TPNS Library	3
General Information Manuals	3
Planning and Installation Manuals	3
Operation Manuals	3
Reference Manuals	4
User's Manuals	4
Samples Manuals	4
Messages and Codes	4
 Master Index	5

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make them available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of the intellectual property rights of IBM may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
500 Columbus Avenue
Thornwood, New York 10594
United States of America

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement.

This document is not intended for production use and is furnished as is without any warranty of any kind, and all warranties are hereby disclaimed including the warranties of merchantability and fitness for a particular purpose.

Trademarks and Service Marks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

ACF/VTAM	APPN	CICS
DFSORT	Enterprise System/9370	IBM
IMS	MVS	MVS/ESA
MVS/SP	MVS/XA	NetView
OS/390	S/370	SP
Systems Application Architecture	VM/ESA	VM/XA
VTAM		

Other company, product, and service names, which may be denoted by a double asterisk (**), may be trademarks or service marks of others.

About This Book

This book provides a master index and bibliography to the users of Teleprocessing Network Simulator (TPNS).

Who Should Use This Book

This book is intended for anyone who uses TPNS.

How to Use This Book

Use this book if you need to look up a particular item, but do not know what book to check.

Where to Find More Information

The following list shows the books in the TPNS library. For information on related publications, see the "Master Bibliography" on page 3.

Education

<i>TPNS General Information</i>	GH20-2487
<i>TPNS Primer</i>	SC31-6043

Planning and Installation

<i>TPNS Planning and Installation</i>	SH20-2488
---------------------------------------	-----------

Resource and Message Traffic Definition

<i>Defining TPNS Networks</i>	SC31-6008
<i>Creating TPNS Message Generation Decks</i>	SC31-6009
<i>TPNS Language Reference</i>	SH20-2489
<i>TPNS Script Generating Utilities</i>	SC30-3453
<i>TPNS Samples</i>	SC30-3454
<i>TPNS STL Reference Card</i>	SX75-0065
<i>Using the TPNS Structured Translator Language (STL) and the STL Translator</i>	SC31-6013

Operation

<i>TPNS Operation</i>	SC30-3289
<i>TPNS Messages and Codes</i>	SC30-3310
<i>TPNS General Utilities</i>	SC30-3290

Customization

<i>TPNS User Exits</i>	SC31-6071
------------------------	-----------

Miscellaneous

TPNS Master Index

GC31-6059

TPNS Library (all manuals except *TPNS General Information*)

SBOF-1426

IBM Online Library: IBM Networking Systems Softcopy Collection Kit
(CD-ROM containing softcopy of all manuals except for *TPNS Master Index*
and *TPNS STL Reference Card*)

SK2T-6012

Introduction

This book contains index entries for selected books in the TPNS library. Use this book whenever you need to find information about a particular topic, but do not know where to look. See “How to Use This Index” for more information about using this index.

This book also contains the master bibliography for TPNS, listed in “Master Bibliography” on page 3.

What is Teleprocessing Network Simulator (TPNS)?

TPNS is a terminal and network simulation tool. You can use TPNS to determine system performance and response time, to evaluate network design, to perform functional testing, and to automate regression testing. Used as a basic tool in a comprehensive test plan, TPNS increases the effectiveness of system testing by providing a structured and systematic approach to all phases of system testing.

TPNS runs on any IBM host processor that supports:

- MVS/370 (MVS/SP* Version 1 or later)
- MVS/XA* (MVS/SP Version 2 or later)
- MVS/ESA* (MVS/SP Version 3 or later)
- OS/390*
- Any VM system (with or without High Performance Option) that supports Group Control System (GCS).

In this book, MVS is any environment running MVS/370 or MVS/XA (unless explicitly stated otherwise). VM is any environment running a VM/SP, VM/XA*, or VM/ESA* system that supports GCS.

How to Use This Index

To locate a particular topic, do the following steps:

1. Find its entry in the index. Entries in this index are shown alphabetically, with page numbers prefaced by a book code.

The following list shows the book codes used in this index.

Code	Book and Order Number
DEF	<i>Defining TPNS Networks</i> , SC31-6008
GU	<i>TPNS General Utilities</i> , SC30-3290
LR	<i>TPNS Language Reference</i> , SH20-2489
MGD	<i>Creating TPNS Message Generation Decks</i> , SC31-6009
OP	<i>TPNS Operation</i> , SC30-3289
PI	<i>TPNS Planning and Installation</i> , SH20-2488
PRI	<i>TPNS Primer</i> , SC31-6043

SGU	<i>TPNS Script Generating Utilities</i> , SC30-3453
SMP	<i>TPNS Samples</i> , SC30-3454
STL	<i>Using the TPNS Structured Translator Language (STL) and the STL Translator</i> , SC31-6013
UE	<i>TPNS User Exits</i> , SC31-6071

2. Get the book indicated by the book code and turn to the specified page number in that book.

For example, to find information about the A subcommand of the F operator command, turn to the index in this book and locate the following entry:

A subcommand (F command) **OP**:143

Then, get a copy of *TPNS Operation* (as indicated by the book code, OP) and turn to page 137 of that book. You can find the needed information there.

Master Bibliography

The following manuals provide additional information about the definition and operation of networks simulated by TPNS:

TPNS Library

Creating TPNS Message Generation Decks, SC31-6009

Defining TPNS Networks, SC31-6008

TPNS General Information, GH20-2487

TPNS General Utilities, SC30-3290

TPNS Language Reference, SH20-2489

TPNS Master Index, GC31-6059

TPNS Messages and Codes, SC30-3310

TPNS Operation, SC30-3289

TPNS Planning and Installation, SH20-2488

TPNS Primer, SC31-6043

TPNS Samples, SC30-3454

TPNS Script Generating Utilities, SC30-3453

TPNS STL Reference Card, SX75-0065

TPNS User Exits, SC31-6071

Using the TPNS Structured Translator Language (STL) and the STL Translator, SC31-6013

General Information Manuals

IBM 3101 Display Terminal Description, GA18-2033

Planning and Installation Manuals

ACF/VTAM Installation and Resource Definition, SC27-0610

IBM 8775 Display Terminal Component Description, GA33-3044

I/O Installation—Physical Planning for S/360 and S/370, GC22-7064

MVS Installation and Tuning Guide, GC28-0681

Netview Installation and Administration Guide, SC30-3360

NCP Installation and Resource Definition Guide, SC30-3253

VM/SP Planning Guide and Reference, SC19-6201

VM/SP HPO Planning Guide and Reference, SC19-6223

Operation Manuals

ACF/VTAM Operation, SC27-0612

BTS Program Reference and Operation Manual, SH20-5523

DFSORT Application Programming: Guide, SC33-4035

IBM 3270 Information Display Unit, GV20-9707

IBM 3644 Automatic Data Unit Programming Guide, GC24-5178

IBM 3720/3725 Communication Controller Principle of Operation, GA33-0013

IBM 3725 Communication Original Equipment Manual, GA33-0017

IBM 3745 Communication Controller: Principle of Operation, SA33-0102

NetView Customization: Command Lists, SC30-3423

OS/VS2/MVS System Programming Library: Job Management, GC28-0267

OS/VS2/MVS System Programming Library: Service Aids, GC28-0629

OS/VS2/MVS System Programming Library: TSO/E Customization, SC28-1380

OS/VS2/MVS System Programming Library: Utilities, GC26-3902

VM/SP Group Control System Guide, SC24-5249

VTAM Network Implementation Guide, SC31-6404

Reference Manuals

ACF/NCP Reference Summary and Data Areas for the IBM 3725, LY30-3070

ACF/NCP Reference Summary and Data Areas for the IBM 3745, LY30-5570

ACF/NCP/VS Program Reference Summary, LY30-3058

ACF/VTAM Programmer's Guide, SC23-0115

VTAM Programming for LU 6.2, SC31-6437

IBM Token-Ring Network Architecture Reference, SC30-3374

IBM 3725 Communication Controller Configuration Guide, SA33-0012

IBM 3745 Communication Controller Line Interface Coupler and Cable Guide, SA33-0094

IBM 3745 Communication Controller Original Equipment Manufacturers Information Manual, SA33-0094

IBM 5250 Information Display System Functions Reference Manual, SA21-9247

NCP and EP Reference Summary and Data Areas (Version 5), LY30-5603

NCP, SSP, and EP Diagnosis Guide, LY30-5591

NetView Administration Reference, SC30-3360

SNA Formats and Protocol Reference Manual: Architecture Logic, SC30-3112

SNA Formats and Protocol Reference Manual: Architecture Logic for Type 2.1 Nodes, SC30-3422

System Network Architecture: Reference Summary, GA27-3136

Systems Application Architecture Common Programming Interface Communications Reference, SC26-4399-06. (TPNS does not support CPI-C functions that have been added in later releases of this document.)

User's Manuals

NetView Performance Monitor User's Guide and Reference, SH20-6360

Random Number Generation and Testing, GC20-8011

Samples Manuals

OS/VS2 JES2 Logic Manual, SY24-6000

Messages and Codes

ACF/VTAM Programming, SC27-0611

VSE/VSAM Messages and Codes, SC24-5146

Master Index

Special Characters

/ operator **STL:33**
// operator **STL:33**
- operator **STL:33**
/* (comment) **PRI:45**
@EJECT statement **STL:156**
@ENDGENERATE statement **STL:37, STL:157—158**
@ENDNETWORK statement **STL:163**
@ENDPROGRAM statement **STL:113, STL:165**
@GENERATE statement **STL:37, STL:157—158**
@IFNUM statement **STL:131, STL:159—160**
@INCLUDE statement **STL:38, STL:161**
@NETWORK statement **STL:163**
@PROGRAM statement **PRI:61, STL:113, STL:165**
\$ (Exit) operator command **OP:171**
\$ATTR\$ data field option **MGD:167**
\$BADGE\$ data field option **MGD:182**
\$CARD\$ data field option **MGD:182**
\$DIDO\$ data field option **MGD:182**
\$DUP\$ data field option **MGD:185**
\$FM\$ data field option **MGD:160**
\$MAG10\$ data field option **MGD:182**
\$MAG10S\$ data field option **MGD:182**
\$MAG63\$ data field option **MGD:182**
\$MAG63S\$ data field option **MGD:182**
\$NL\$ data field option **MGD:160**
\$RNUM\$ data field option
 using to specify a range of random
 numbers **MGD:32**
 using with RN network definition statement **MGD:32**
\$UTBL\$ data field option
 generating messages with **MGD:35**
 selecting entries with **MGD:35**
* (asterisk) **PRI:21**
* (comment) command
 Log Compare Utility **GU:160**
 Loglist Utility **GU:96**
 Response Time Utility **GU:224**
* (comment) command, ITPSGEN **SGU:93**
* (Comment) operator command **OP:170**
* operator **STL:33**
*/ (comment) **PRI:45**
*data subcommand (F command) **OP:143**
& relational operator **STL:52**
&= operator **STL:49**
%. control command **SGU:185**
+ operator **STL:33**
+data subcommand (F command) **OP:144**
< operator **STL:49**
<= operator **STL:49**

<> operator **STL:49**
= operator **STL:49**
=< operator **STL:49**
=> operator **STL:49**
> operator **STL:49**
>< operator **STL:49**
>= operator **STL:49**
| relational operator **STL:52**
|| operator **STL:34**

Numerics

2741
 CMS example **SMP:19**
 TSO example **SMP:45**
3101 displays, changing parameters for **UE:3**
3270
 CICS example **SMP:17**
 CMS example (BSC) **SMP:20**
 CMS example (SNA) **SMP:21**
 IMS example **SMP:26**
 TSO/SPF example (SNA) **SMP:50**
 TSO/VTAM example (SNA) **SMP:47**
3270 BSC terminal, simulating errors in **MGD:161**
3270 Data Analysis/APL Character Set
 lists of extended characters **MGD:223, MGD:227**
 simulating **MGD:162**
3270 display records **MGD:209**
3270 displays, changing parameters for **UE:3**
3270 extended functions, simulating **MGD:163**
3270 functions with ITPECHO **GU:228—232**
3270 key options **MGD:160**
3270 records **UE:55**
3270 screen attributes
 described **STL:65**
 using ATTR3270 function to identify **STL:65**
3270 SDLC terminal, simulating errors in **MGD:161**
3270 terminals
 comparing display records **PI:111**
 displaying images **PI:102**
3271
 TSO/VTAM example (SDLC) **SMP:47**
3274
 IMS example **SMP:27**
3274 Local Clear key **MGD:160**
3278 Magnetic Stripe Reader, simulating **MGD:161**
3289 Line Printer, Model 3 **MGD:182**
3289 Printer, Model 3
 See IBM 8100 Information System
3600
 CICS example **SMP:18**
 IMS example **SMP:28**

3641 Reporting Terminal **MGD:182**
See also IBM 8100 Information System
 3642 Encoder Printer
See IBM 8100 Information System
 3643 displays, changing parameters for **UE:3**
 3643 Keyboard Display
See IBM 8100 Information System
 3643 records **UE:55**
 3644 Automatic Data Unit **MGD:183**
See also IBM 8100 Information System
 3645 Printer **MGD:183**
See also IBM 8100 Information System
 3646 Scanner Control Unit **MGD:184**
See also IBM 8100 Information System
 3647 Time and Attendance Terminal **MGD:184**
See also IBM 8100 Information System
 3767
 CICS example **SMP:18**
 TSO/VTAM example **SMP:45, SMP:47**
 3770
 RJE example **SMP:33**
 RJE example with multiple logical units **SMP:36**
 RJE example with single logical unit **SMP:40**
 3780
 RJE example **SMP:34**
 5250 displays, changing parameters for **UE:3**
 5250 records **UE:56**
 5250 scripts **SMP:50**
 8100 processor devices
 TPNS support for **PI:138**
 8775 displays, changing parameters for **UE:3**
 8775 records **UE:55**

A

A (Alter) operator command **OP:126**
 altering user time intervals with **MGD:74**
 operands **MGD:74**
 referencing message generation decks
 with **MGD:192**
 signaling events with **MGD:135**
 A subcommand (F command) **OP:143**
 ABORT statement **STL:84, STL:168**
 ACKDLYCT operand (LLS statement) **LR:142, LR:152**
 ACKDLYTO operand (LLS statement) **LR:142, LR:152**
 Activate/Deactivate Cross-Domain Resources (V) oper-
 ator command **OP:163**
 Activate/Deactivate TPNS Control Program Line Trace
 (B) operator command **OP:139**
 Activate/Deactivate TPNS Control Program SIT Trace
 (N) operator command **OP:152**
 activating logic tests **MGD:100**
 Active Command List **GU:109**
 activities flowchart **PRI:10**
 adapter, token-ring
 See token-ring adapter

adaptive session pacing **DEF:182**
 ADDR operand
 DEV (terminal) statement **LR:177**
 LU (cross-domain) statement **LR:265**
 LU (type 2.1 node) statement **LR:223**
 PU (cross-domain) statement **LR:283**
 PU (type 2.1 node) statement **LR:239**
 TERM statement **LR:196**
 addressing mode considerations **UE:2**
 adjacent domain
 defined **PI:37**
 ADJNODE statement **DEF:185, LR:216**
 AID keys
 coding on a TRANSMIT statement **STL:76**
 list of AID keys that can be simulated **STL:75**
 specific devices, keys that can be simulated
 for **STL:385**
 transmitting message with **STL:75**
 AID operand (M command) **OP:148**
 Airline Line Control example
 ALCI definition **SMP:103**
 message generation decks **SMP:98**
 network definition statements **SMP:96**
 sample network **SMP:96**
 STL procedures **SMP:101**
 TPNS control program generation **SMP:103**
 Airlines Line Control (ALC) terminals **DEF:245, MGD:177**
 ALC example
 ALCI definition **SMP:103**
 message generation decks **SMP:98**
 network definition statements **SMP:96**
 sample network **SMP:96**
 STL procedures **SMP:101**
 TPNS control program generation **SMP:103**
 ALC support **STL:76, STL:292**
 ALCSEOM operand **MGD:177**
 MSGTXT statement **LR:386**
 TEXT statement **LR:423**
 alert simulation **MGD:156, PI:143**
 ALLOCATE statement **STL:31, STL:169—170**
 allocating data sets **DEF:25**
 allocating data sets, ITPLSGEN **SGU:47**
 allocating IDC data sets **SGU:4**
 allocating TPNS resources **STL:31, STL:131**
 ALTCSET operand
 DEV (TCP/IP) statement **LR:128**
 DEV (terminal) statement **LR:177**
 LU (cross-domain) statement **LR:266**
 LU (type 2.1 node) statement **LR:224**
 LU (VTAMAPPL) statement **LR:110**
 Alter Network Operands (A) operator
 command **OP:126**
 altering
 sequential processing **MGD:123**
 user time intervals with the A (Alter) operator
 command **MGD:74**

analyzing results

- comparing 3270 display records **MGD:209**
- determining response times **MGD:214**
- using operator reports **MGD:204**
- using other monitoring programs **MGD:203**
- using the Log Compare Utility **MGD:209**
- using the Loglist Utility **MGD:206**
- using the TPNS Display Monitor Function **MGD:219**
- using TPNS output **MGD:203**

analyzing the log data set **UE:42**

APL operand (CHARSET statement) **LR:304**

APL/Text Character Set, simulating **MGD:163**

APLCSID operand **DEF:228**

- DEV (TCP/IP) statement **LR:128**
- DEV (terminal) statement **LR:178**
- LU (cross-domain) statement **LR:266**
- LU (type 2.1 node) statement **LR:224**
- LU (VTAMAPPL) statement **LR:110**

APPCLU

- canceling **OP:140**
- counter allocation **MGD:127**
- JCL for **OP:11**
- query network for **OP:33**
- starting activity for **OP:160**
- stopping activity for **OP:155**

APPCLU command

- Loglist Utility **GU:60**
- Response Time Utility **GU:191**

APPCLU statement **DEF:89, LR:94, PI:60**

APPCLUID data field option **LR:437**

APPCLUID function **STL:304**

application program password, defining

- CPI-C **LR:94**
- VTAM **LR:123**

Application Test Processor (TPNS/ATP)

- CLIST for non-VSAM data sets, example of **SGU:110**
- CLIST for VSAM data sets, example of **SGU:107**
- creating JCL and control statements **SMP:142**
- data base, managing **SGU:177—182**
- data sets, specifying **SGU:120**
- defining transactions and testcases **SMP:132**
- description **SGU:103**
- displaying TPNS/ATP Option menu **SGU:118**
- documentation processors **SGU:104**
- editor **SGU:104, SGU:116—135**
- editor, option of ISPF/PDF Primary Option Menu **SGU:112**
- epilogue, example of **SGU:159**
- example of **SMP:131**
- execution parameter, specifying **SGU:121**
- facilities **SGU:104**
- formatted input **SMP:149**
- generating scripts **SMP:144**
- identifying MFS screen source **SMP:131**
- identifying screen sequences and data **SMP:131**

Application Test Processor (TPNS/ATP) (*continued*)

- identifying the scenario **SMP:131**
- ITPATPSG **SGU:153—173**
- ITPSDL **SGU:142—153**
- ITPTCL **SGU:136—142**
- JCL for non-VSAM data sets, example of **SGU:108**
- JCL for VSAM data sets, example of **SGU:105**
- message generation epilogues **SGU:157**
- message generation prologues **SGU:157**
- MFS source statements **SMP:147**
- MFS, support of **SGU:173—177**
- Program Function (PF) keys **SGU:135, SGU:136—142**
- prologue, example of **SGU:159**
- reviewing output **SMP:144**
- Screen Data Listing processor **SGU:142—153**
- script generator **SGU:104, SGU:153—173**
- selection options **SGU:119**
- setup required **SGU:104**
- support **SGU:104**
- tailoring for TPNS/ATP editor **SGU:112**
- Testcase Listing **SGU:136**
- unformatted input **SMP:152**
- VSAM data bases required **SGU:105**

applications

- prototyping with TPNS **PI:45**
- testing with TPNS **PI:45**

APPLID operand

- APPCLU statement **LR:94**
- VTAMAPPL statement **LR:123**

APPLID parameter that identifies ITPECHO **GU:233**

APPN Type 2.1 end node **DEF:187**

AREA operand

- DATASAVE statement **LR:330**
- IF (message generation) statement **LR:368**
- IF (network configuration) statement **LR:50**
- LOG statement **LR:383**
- on the DATASAVE statement **MGD:43**
- on the IF statement **MGD:88**
- on the LOG statement **MGD:122**

area operand (A command) **OP:131**

ASSIGN control command **SGU:160**

assignment statements, use of **STL:18, STL:32**

asterisk (Comment) operator command **OP:170**

asynchronous conditions

- expressions not allowed in, listed **STL:387**
- reserved variables, using in **STL:24**
- restrictions on expressions in **STL:90**
- restrictions on the SUBSTR function in **STL:91**
- setting up **STL:90**
- testing **STL:82**

asynchronous statements **PRI:83**

- defined **STL:82**
- listed **STL:82**

asynchronous statements exercise **PRI:83**

- asynchronous subset statements
 - ABORT statement **STL:84, STL:168**
 - CALL statement **STL:84, STL:174**
 - CANCEL statement **STL:87, STL:175**
 - defined **STL:84**
 - EXECUTE statement **STL:85, STL:234**
 - listed **STL:84**
 - NORESP statement **STL:87, STL:258**
- ATO operand (LINE (type 2.1 node) statement) **LR:218**
- ATP
 - See Application Test Processor (TPNS/ATP)*
- ATPAID menu **SGU:131**
- ATPDEF menu **SGU:120**
- ATPDFO menu **SGU:132**
- ATPPRIM menu **SGU:119**
- ATPSDEL menu **SGU:129**
- ATPSEL menu **SGU:122**
- ATPTC menu **SGU:125**
- ATPTCLST panel **SGU:126**
- ATPTCSEL menu **SGU:124**
- ATRABORT operand
 - DEV (TCP/IP) statement **LR:128**
 - DEV (terminal) statement **LR:178**
 - LU (cross-domain) statement **LR:266**
 - LU (type 2.1 node) statement **LR:224**
 - LU (VTAMAPPL) statement **LR:111**
 - TERM statement **LR:197**
- ATRABORT operand (A command) **OP:131**
- ATRABORT operand on network definition statements **MGD:193**
- ATRDECK network definition operand, defining programs used **STL:127**
- ATRDECK operand
 - DEV (TCP/IP) statement **LR:129**
 - DEV (terminal) statement **LR:178**
 - LU (cross-domain) statement **LR:266**
 - LU (type 2.1 node) statement **LR:224**
 - LU (VTAMAPPL) statement **LR:111**
 - TERM statement **LR:197**
- ATRDECK operand (A command) **OP:131**
- ATRDECK operand on network definition statements **MGD:193**
- attention key **STL:58**
- attention key simulation **MGD:151**
- ATTR data field option **LR:437**
- ATTR statement **LR:297, MGD:174, STL:171**
- ATTR3270 function **STL:65, STL:305—310**
- ATTRIBUTE command **GU:147**
- authorizing TPNS
 - on MVS **PI:14**
 - on VM **PI:17**
 - under TSO **PI:15**
- AUTOLF operand (TERM statement) **LR:197**
- automatic string function **GU:230, GU:231**
- automatic terminal recovery **DEF:273**

- automatic UTI adjustment **DEF:262**
- automating the test
 - OPCMND statement **STL:101, STL:262**
 - operator commands, including in STL programs **STL:101**
- automating TPNS operation **PI:100**
- AVMON (Availability Monitor)
 - configurations **SMP:156**
 - description **SMP:153**
 - modifying **SMP:155**
 - processing **SMP:153**
 - summary reports **SMP:160**
- AVMON scripts
 - ACHKNETV **SMP:177**
 - ACHKTSO **SMP:186**
 - ACTRLNET **SMP:168**
 - AFORTIME **SMP:171**
 - ALOGNETV **SMP:175**
 - ALOGTSO **SMP:185**
 - AMONNETV **SMP:172**
 - AMONTSO **SMP:182**
- AVMON STL procedures **SMP:190**
 - constant declarations **SMP:194**
 - procedure ACHKNETV **SMP:210**
 - procedure ACHKTSO **SMP:218**
 - procedure ACTRLNET **SMP:202**
 - procedure ALOGNETV **SMP:208**
 - procedure ALOGTSO **SMP:216**
 - procedure AMONNETV **SMP:204**
 - procedure AMONTSO **SMP:212**
 - table declarations **SMP:201**
 - utility procedures **SMP:219**
 - variable declarations **SMP:196**

B

- B (Activate/Deactivate TPNS Control Program Line Trace) operator command **OP:139**
- B operand (A command) **OP:131**
- back tab key statement **LR:299**
- back tab key, simulating **LR:299**
- BADGE data field option **LR:438**
- BADGE function **STL:59, STL:311**
- badge reader, simulating
 - See devices, simulating, badge reader*
- BASECSID operand **DEF:228**
 - DEV (TCP/IP) statement **LR:129**
 - DEV (terminal) statement **LR:178**
 - LU (cross-domain) statement **LR:267**
 - LU (type 2.1 node) statement **LR:225**
 - LU (VTAMAPPL) statement **LR:111**
- binary synchronous communication
 - line discipline **PI:136**
 - terminals **PI:135**
- BIND **DEF:143, DEF:144**

BIND image, ITPECHO **GU**:228
 BIND pacing **DEF**:181
 BIND response, modifying **SMP**:93
 BIND segmenting **DEF**:181
 BINDSEG operand (PU21 statement) **LR**:243
 bit
 condition, defined **STL**:49
 constant, defined **STL**:26
 expression, defined **STL**:35
 variable, defined **STL**:22
 BIT statement **STL**:28, **STL**:172
 bit variables **PRI**:43
 BLKDLY operand (TERM statement) **LR**:197
 BLKDLY operand on the TERM network definition statement **MGD**:76
 BLOCK operand
 LOG statement **LR**:384
 PU (PU type 4) statement **LR**:284
 blocked PIUs **DEF**:153
 boundary channel-attached Type 2.1 nodes
 defining **DEF**:184
 modifying NCP for **DEF**:63, **DEF**:64
 physical configuration for **DEF**:12
 TPNS simulation support **DEF**:183
 BOUNDARY operand (RTM statement) **LR**:84
 BRANCH statement **LR**:298, **MGD**:123
 break execution (QUIESCE) **PRI**:17
 BREAK operand (TEXT statement) **LR**:423
 BSC 3270
 logging off **STL**:70
 logging on **STL**:70
 BSC1 terminals **MGD**:175
 BSC2 switched lines **DEF**:245
 BSC2 terminals **MGD**:176
 BSC3 terminals **MGD**:176
 BSCx terminals
 BSC1 terminals **DEF**:244
 BSC2 terminals **DEF**:245
 BSC3 terminals **DEF**:245
 BTAB statement **LR**:299, **STL**:173
 BTRANS command **GU**:173, **GU**:192—194
 BTU trace, command **OP**:161
 BUFFER **STL**:24
 BUFFER reserved variable **PRI**:44
 BUFSIZE operand **PRI**:21
 APPCLU statement **LR**:95
 CNTLR statement **LR**:252
 LINE (type 2.1 node) statement **LR**:218
 LINK statement **LR**:260
 SSCP statement **LR**:287
 TCPIP statement **LR**:138
 TERM statement **LR**:198
 VTAMAPPL statement **LR**:123
 BUFSIZE parameter **GU**:233
 building user data table **LR**:91

C

C (Cancel Network Resources) operator
 command **OP**:140
 C2D function **STL**:320
 C2X function **STL**:321
 CALC statement **LR**:300, **MGD**:118
 calculating a value in save area or user area **LR**:300
 calculating response times **GU**:171—173
 CALL statement **LR**:302, **MGD**:124
 asynchronous subset statement **STL**:84, **STL**:174
 structured flow-of-control statement **STL**:42, **STL**:174
 call subroutine statement **LR**:302
 CALldata operand (A command) **OP**:132
 CALldata operand (LCH statement) **LR**:158
 CALldata statement **DEF**:199, **LR**:157
 CALldly operand (A command) **OP**:132
 CALldly operand (LCH statement) **LR**:158
 CALLIN operand (MCH statement) **LR**:164
 calling a subroutine **LR**:302
 calling other procedures exercise **PRI**:57
 CALLIO operand (MCH statement) **LR**:164
 CALLOUT operand (MCH statement) **LR**:165
 Cancel Event statement **LR**:303
 CANCEL statement **LR**:303, **MGD**:135
 asynchronous subset statement **STL**:87, **STL**:175—177
 used to cancel events **STL**:98, **STL**:175—177
 canceling
 delays **STL**:87
 events **STL**:98
 capacity planning tests
 conducting **PI**:27
 capturing data with IDC
 adding statements to the IDC log **SGU**:16
 capturing session initiation **SGU**:14
 capturing session termination **SGU**:21
 changing escape keys **SGU**:20
 changing log data sets **SGU**:18
 controlling data capture **SGU**:16
 IDC Add Statements Panel **SGU**:18
 IDC Change Escape Key Panel **SGU**:20
 IDC Change Log Data Sets Panel **SGU**:19
 IDC Escape Actions Panel **SGU**:14
 using escape actions **SGU**:14
 CARD data field option **LR**:438
 CARD function **STL**:59, **STL**:312
 card reader, simulating
 See devices, simulating, card reader
 CART
 See command and response token (CART)
 CCOL function **PRI**:73, **STL**:64, **STL**:313
 CCSIZE operand
 DEV (TCP/IP) statement **LR**:129
 DEV (terminal) statement **LR**:179

CCSIZE operand (*continued*)
 LU (cross-domain) statement **LR:267**
 LU (type 2.1 node) statement **LR:225**
 LU (VTAMAPPL) statement **LR:111**
 CDLOG (cross-domain logging) option
 example **GU:248**
 using **GU:247**
 CDLOG command **GU:61**
 CDLOG operand (A command) **OP:132**
 CDRM statement **DEF:123, LR:248**
 CDRM-CDRM session
 activating **DEF:146**
 deactivating **DEF:147**
 CDU trace, command **OP:161**
 CGRAPH command **GU:195**
 CHAIN operand (RH statement) **LR:401**
 chaining messages **MGD:149**
 CHAINING operand
 DEV (terminal) statement **LR:179**
 LU (cross-domain) statement **LR:267**
 LU (type 2.1 node) statement **LR:225**
 LU (VTAMAPPL) statement **LR:112**
 TERM statement **LR:199**
 CHANDD operand (CHANL statement) **LR:250**
 changing message generation paths **LR:298**
 CHANL command **GU:62**
 CHANL statement **DEF:124, LR:250**
 channel
 listing address of **OP:147**
 logical, calling **OP:166**
 query network for **OP:33**
 trace **OP:49**
 tracing **OP:161**
 CHAR function **PRI:73, STL:314**
 character set ID **DEF:228**
 character set select statement **LR:304**
 character set, selecting **LR:304**
 CHARATTR command **GU:148**
 CHARS operand (DELETE statement) **LR:342**
 CHARSET statement **LR:304, MGD:163, MGD:164, STL:178**
 checking the ISPF editor profile **PRI:14, PRI:94**
 checklist for developing message generation
 decks **MGD:12**
 CHECKONLY command **GU:149**
 CICS **DEF:71**
 CICS scripts
 with 3270 BSC and SNA **SMP:17**
 with 3600 **SMP:18**
 with 3767 SNA **SMP:18**
 class of service table begin statement **LR:257**
 class of service table end statement **LR:256**
 classes
 declaring shared variables **PRI:44**
 declaring unshared variables **PRI:44**
 shared variables **PRI:43**

classes (*continued*)
 unshared variables **PRI:43**
 classes, variables
See variables, classes
 CLEAR AID key **STL:75**
 clear function **GU:232**
 Clear key statement **LR:305**
 Clear key, simulating **LR:305**
 Clear Partition key statement **LR:306**
 Clear Partition key, simulating **LR:306**
 CLEAR statement **LR:305**
 CLEARPTN AID key **STL:75**
 CLEARPTN statement **LR:306, MGD:165**
 CLIST
 samples **OP:98**
 triggered by TPNS messages **OP:100**
 using **OP:98**
 using to run STL Translator **STL:117**
 CLISTs
 defining the TPNS/NetView operator interface **PI:22**
 using for operator interface between TPNS and the
 NetView program **PI:102**
 Closedown (Z) operator command **OP:169**
 CMACCP statement **LR:320, STL:179**
 CMALLC statement **LR:320, STL:180**
 CMCFCM statement **LR:320, STL:181**
 CMCFMD statement **LR:320, STL:182**
 CMD statement **LR:307**
 CMD1-24 AID keys **STL:75**
 CMDEAL statement **LR:320, STL:183**
 CMECS statement **LR:320, STL:184**
 CMECT statement **LR:320, STL:185**
 CMEMN statement **LR:320, STL:186**
 CMEPLN statement **LR:320, STL:187**
 CMESL statement **LR:320, STL:188**
 CMFLUS statement **LR:320, STL:189**
 CMINIT statement **LR:320, STL:190**
 CMND statement **DEF:122, LR:308—317, MGD:150, MGD:154**
 CMONTH data field option **LR:438**
 CMONTH function **STL:315**
 CMPTR statement **LR:320, STL:191**
 CMRCV statement **LR:320, STL:192**
 CMRTS statement **STL:194**
 CMS scripts
 with 2741 **SMP:19**
 with 3270 BSC **SMP:20**
 with 3270 SNA **SMP:21**
 CMSCT statement **LR:320, STL:195**
 CMSDT statement **LR:321, STL:196**
 CMSED statement **LR:321, STL:197**
 CMSEND statement **LR:320, STL:198**
 CMSERR statement **LR:320, STL:200**
 CMSF statement **LR:321, STL:201, STL:202**
 CMSFM5 statement **LR:321**

CMSLD statement **LR:321, STL:203**
 CMSMN statement **LR:321, STL:204**
 CMSPLN statement **LR:321, STL:205**
 CMSPTR statement **LR:321, STL:206**
 CMSRC statement **LR:321, STL:207**
 CMSRT statement **LR:320, LR:321, STL:208**
 CMSSL statement **LR:321, STL:209**
 CMSST statement **LR:321, STL:210**
 CMSTPN statement **LR:321, STL:211**
 CMTRTS statement **LR:321, STL:212**
 CMxxxx CPI-C simulation statements **LR:295, LR:318—320**
 CMxxxx simulation statements **MGD:117, MGD:119**
 CNOS operand (APPCLU statement) **LR:95**
 CNSL command **GU:63**
 CNTLR statement **DEF:124, DEF:172, LR:252—254**
 CNTLR subarea definition statement **LR:252—254**
 CNTLREX operand
 discussion **UE:29**
 parameter list **UE:30**
 return codes **UE:30**
 CNTLREX operand (NTWRK statement) **LR:71**
 CNTLRID data field option **LR:438**
 CNTLRID function **STL:316**
 CNTR data field option **LR:438**
 CNTRS operand (NTWRK statement) **LR:71**
 CNTRSEED operand (NTWRK statement) **LR:71**
 CNTRX data field option **LR:438**
 CODE operand (HELP statement) **LR:360**
 coding
 AREA operand on the DATASAVE statement **MGD:43**
 BLKDLY operand on the TERM network definition statement **MGD:76**
 comment field **MGD:20**
 DATASAVE statement **MGD:43**
 DBCS2SB function (DATASAVE statement) **MGD:47**
 DBCSADD function (DATASAVE statement) **MGD:47**
 DBCSADJ function (DATASAVE statement) **MGD:47**
 DBCSDEL function (DATASAVE statement) **MGD:47**
 DELAY operand on network definition statements **MGD:75**
 DELAY statement **MGD:77**
 DELETE function (DATASAVE statement) **MGD:46**
 ENDTXT statement **MGD:21**
 IF statement **MGD:24**
 IF statement operands
 AREA **MGD:88**
 CURSOR **MGD:84**
 ELSE **MGD:92**
 EVENT **MGD:84**
 LENG **MGD:87**
 list of **MGD:86**

coding (*continued*)
 IF statement operands (*continued*)
 LOC **MGD:84**
 LOCTEXT **MGD:91**
 SCANCNTR **MGD:98**
 STATUS=HOLD **MGD:102**
 TEXT **MGD:88**
 THEN **MGD:92**
 UTBL **MGD:97**
 UTBLCNTR **MGD:87, MGD:97**
 WHEN **MGD:88**
 INSERT function (DATASAVE statement) **MGD:46**
 IUTI operand **MGD:72**
 LEFT function (DATASAVE statement) **MGD:46**
 LENG operand on the DATASAVE statement **MGD:45**
 LOC operand on the DATASAVE statement **MGD:44**
 message generation decks **MGD:21**
 message generation decks with delimiters **MGD:67**
 MSGTXT statement **MGD:21**
 name field **MGD:18**
 operand field **MGD:18**
 RATE statement **MGD:75**
 RIGHT function (DATASAVE statement) **MGD:46**
 SB2DBCS function (DATASAVE statement) **MGD:47**
 SB2MDBCS function (DATASAVE statement) **MGD:47**
 scripts with intermessage delays **MGD:79**
 SETUTI statement **MGD:73**
 statement field **MGD:18**
 TEXT operand on the DATASAVE statement **MGD:44**
 TEXT statement **MGD:23**
 TRANSLATE function (DATASAVE statement) **MGD:46**
 WAIT statement **MGD:23**
 coding conventions
 comments **LR:3**
 continuing statements **LR:3**
 continuing text operands **LR:3**
 Name field **LR:1**
 operands **LR:2**
 overview **LR:1**
 with TSO **LR:2**
 COFF function **STL:64, STL:317**
 color for display, selecting **LR:325**
 COLOR operand
 DEV (TCP/IP) statement **LR:129**
 DEV (terminal) statement **LR:179**
 LU (cross-domain) statement **LR:267**
 LU (type 2.1 node) statement **LR:226**
 LU (VTAMAPPL) statement **LR:112**
 COLOR statement **LR:325, MGD:163, STL:213**

COLUMN operand
 CURSOR statement **LR:327**
 SELECT statement **LR:406**
 combining procedures from different STL programs
 declaring variables for **STL:130**
 example of **STL:132**
 numbering IF statements for **STL:131**
 reasons for **STL:130**
 combining statements **PRI:75**
 combining types of data **MGD:30**
 command and response token (CART) **OP:25, OP:59**
 COMMAND operand
 CMND statement **LR:308**
 FE statement **LR:43**
 commands
 \$—Exit **OP:171**
 *—Comment **OP:170**
 A—Alter Network Parameters **OP:55, OP:126—138**
 B—Activate/Deactivate TPNS Control Program Line
 Trace **OP:49, OP:139**
 C—Cancel Network Resources **OP:55, OP:140**
 coding conventions **OP:125**
 D—Dump Control Blocks **OP:50, OP:141**
 E—Restart Message Logging **OP:81, OP:142**
 entering from a TSO terminal **OP:25**
 entering from RESLIST **OP:92**
 entering under MVS **OP:24, OP:25**
 entering under VM **OP:26**
 execution order of **OP:22**
 F—Enter Console Recovery **OP:56, OP:143—144**
 G—Terminal Status Query **OP:145**
 HELP **OP:88**
 I—Initialize a Network **OP:27, OP:146**
 issuing from NetView program **OP:85—98**
 issuing in full-screen mode operation **OP:88—97**
 issuing in line mode operation **OP:85**
 issuing with MODIFY **OP:21**
 L—List Resource **OP:147**
 links—LAN interface, specifying **OP:24**
 links—X.25 interface, specifying **OP:23**
 M—Display Monitor Facility **OP:106—110, OP:148—151**
 N—Activate/Deactivate TPNS Control Program SIT
 Trace **OP:50, OP:152**
 NCP subarea links, specifying **OP:23**
 O—Output Data/NetView Control **OP:154**
 operator, specifying at console **OP:21**
 P—Stop Network Resources **OP:155**
 Q—Query Network Resources **OP:28—48, OP:156**
 query device **OP:36—41**
 query subarea node **OP:31**
 R—Reset a Network **OP:54, OP:159**
 RESLIST **OP:89—97**
 S—Start Network Resources **OP:28, OP:160**
 SHOW **OP:97**
 specifying resources **OP:22**
 commands (*continued*)
 specifying with the OPCMND Statement **OP:21**
 T—Dispatcher and Channel Traces **OP:49, OP:161**
 terms used, list of **OP:123**
 U—Unsolicited RU Transmission **OP:162**
 using **OP:27—48**
 V—Activate/Deactivate Cross-Domain
 Resources **OP:51—54, OP:163**
 W—RSTATS Query **OP:165**
 X—X.25 Command **OP:166**
 Y—Communication Controller Control **OP:168**
 Z—Closedown **OP:169**
 commands, control for loglist **PRI:65, PRI:68**
 commands, entering from the console
 (ITPLSGEN) **SGU:54**
 commands, generating debugging comments
 (ITPLSGEN) **SGU:54**
 comment (*) command
 Log Compare Utility **GU:160**
 Loglist Utility **GU:96**
 Response Time Utility **GU:224**
 comment (*) command, ITPSGEN **SGU:93**
 Comment (*) operator command **OP:170**
 comment field **MGD:20**
 comment statement, TPNS
 See TPNS comment command
 comments
 coding **STL:20**
 documenting STL programs using **STL:15**
 multiple line comments **PRI:45**
 nesting **STL:20**
 single line comments **PRI:45**
 comments, coding of **LR:3**
 Common Programming Interface Communications
 (CPI-C)
 See also CPI-C
 message logging **GU:241, GU:251**
 trace records **GU:36, GU:241**
 communication controller
 authorizing TPNS to **PI:14**
 dedicated to TPNS **DEF:27**
 for CPI-C transaction program configuration **PI:12, PI:46**
 for direct-attach configuration **PI:13, PI:54**
 for link-attach configuration **PI:12, PI:51**
 for TCP/IP configuration **PI:12**
 for VTAMAPPL configuration **PI:12, PI:44**
 generating TPNS control program for **DEF:27—61**
 in TPNS physical configurations **DEF:11**
 JCL with **OP:5—9**
 JCL without **OP:9—12**
 modifying NCP for **DEF:63—65**
 requirements **PI:12**
 resynchronizing clock in **OP:168**
 shared by TPNS **DEF:63**

- Communication Controller Control (Y) operator command **OP:168**
- communication network management
 - goals of **PI:143**
 - integrating TPNS with **PI:143**
 - monitor features **PI:145**
- Compare List **GU:110, PI:113**
- comparing data
 - IF (network configuration) statement **LR:46**
- comparing user table entries
 - See user tables, comparing entries using UTBLSCAN function
- comparisons
 - See conditions
- compatible exits **UE:38**
- Complete Records List **GU:109, PI:113**
- complex conditions
 - See conditions, complex
- COMPROT operand (MODEENT statement) **LR:63**
- concatenation, string expressions **STL:34**
- CONCHAR operand (MSGTXT statement) **LR:387**
- CONCHAR operand on the MSGTXT statement **MGD:22**
- COND operand
 - IF (message generation) statement **LR:368**
 - IF (network configuration) statement **LR:51**
- conditional delimiters
 - CMND **MGD:65**
 - ENDTXT **MGD:65**
 - sending messages with **MGD:65—67**
 - TEXT **MGD:65**
- conditional tab statement **LR:326**
- conditions
 - bit
 - comparisons by shorthand **STL:49**
 - defined **STL:49**
 - complex
 - defined **STL:48, STL:52**
 - examples of **STL:52**
 - defined **STL:48**
 - integer, defined **STL:50**
 - relational operators **STL:49**
 - simple
 - defined **STL:48**
 - syntax of **STL:49**
 - string, defined **STL:50**
 - test under mask **STL:51**
- conditions under which logic test is not evaluated **LR:461**
- conditions, asynchronous
 - See asynchronous conditions
- configuring the system
 - deciding what to simulate **PI:40**
 - defined **PI:37**
 - determining test objectives **PI:40**
 - example **PI:39**

- configuring the system (*continued*)
 - identifying logical configuration **PI:40**
 - identifying physical configuration **PI:41**
 - steps to follow **PI:39**
- CONNECT operand (LLS statement) **LR:142, LR:152**
- connecting TPNS to system under test
 - external modem and real line
 - method **DEF:281—282**
 - local attachment cables method **DEF:285—286**
 - modem elimination method **DEF:282—285**
- CONNOUT operand (LLS statement) **LR:143, LR:153**
- CONRATE operand (A command) **OP:132**
- CONRATE operand (NTWRK statement) **LR:71**
- console records **UE:42**
- console, entering commands from (ITPLSGEN) **SGU:54**
- CONSTANT statement **STL:28, STL:214**
- constants
 - defined **STL:22**
 - named
 - declaring **STL:28**
 - defined **STL:26**
 - naming conventions for **STL:22**
 - operators, string constants **STL:25**
 - types
 - bit, defined **STL:26**
 - hexadecimal string, defined **STL:26**
 - integer, defined **STL:24**
 - string, defined **STL:24**
- continuation of statements **STL:19**
- continuation statements **LR:3**
- continuing
 - information over multiple statement lines **MGD:19**
 - messages **MGD:20**
- control block paging data set definition
 - statement **LR:68**
- control blocks
 - DEV control block **UE:58**
 - discussion **UE:57**
 - LDS control block **UE:63**
 - LIN control block **UE:65**
 - LOG control block **UE:66**
 - NCB control block **UE:71**
 - PRT control block **UE:72**
 - PTN control block **UE:74**
 - PU control block **UE:76**
 - RSP control block **UE:77**
 - SAV control block **UE:78**
 - TRM control block **UE:79**
- control command categories
 - data type selection **GU:58**
 - general control **GU:58**
 - overall selection **GU:58**
 - primary resource selection **GU:58**
 - record selection **GU:58**
 - resource selection **GU:58**

control command categories (*continued*)

secondary resource selection **GU:58**

control command coding conventions **GU:57**

control commands, loglist **PRI:65, PRI:68**

control program generation macros

See TPNS control program generation macros

control program, generating

See TPNS control program, generating

control statements

BRANCH **MGD:123**

CALC **MGD:118**

CALL **MGD:124**

CANCEL **MGD:135**

CMxxxx **MGD:117, MGD:119**

DATASAVE **MGD:118**

DEACT **MGD:136**

DELAY **MGD:119**

described **MGD:27, MGD:117—141**

ERROR **MGD:120**

EVENT **MGD:134**

EXIT **MGD:119**

LABEL **MGD:124**

list of **MGD:117**

LOG **MGD:122**

MONITOR **MGD:122**

MSGTXT **MGD:117**

ON **MGD:136**

OPCMND **MGD:121**

RESET **MGD:120**

RESP **MGD:120**

RETURN **MGD:124**

RH **MGD:121**

SET **MGD:125**

SETSW **MGD:125**

SHOW **MGD:122**

TH **MGD:120**

using for specific devices **MGD:119**

WTO **MGD:122**

controller simulation **PI:141**

controller-level exit

discussion **UE:29**

parameter list **UE:30**

return codes **UE:30**

controlling events **MGD:133**

controlling switches and counters **MGD:125**

conversation

accepting **STL:179**

allocating **STL:180**

confirming **STL:181, STL:182**

deallocate type **STL:196**

deallocating **STL:183**

error direction **STL:197**

FM header 5 extension **STL:202**

initializing **STL:190**

log data **STL:203**

mode name **STL:204**

conversation (*continued*)

partner LU name **STL:187, STL:205**

prepare to receive **STL:191, STL:206**

receive type **STL:208**

request-to-send received **STL:192, STL:198,**

STL:200, STL:212

return control **STL:207**

send type **STL:210**

state **STL:184**

symbolic destination name **STL:190**

synchronization level **STL:188, STL:209**

TP name **STL:211**

type **STL:185, STL:186, STL:195**

CONVERT operand (DATASAVE statement) **LR:331**

converting DBCS data **MGD:47**

correcting preprocessor errors **PRI:22**

correcting syntax errors **PRI:17**

correlation records, creating

See statement correlation records, creating

COS operand

CMND statement **LR:310**

MODEENT statement **LR:63**

COS statement **DEF:122, LR:255**

COSEND statement **DEF:122, LR:256**

COSTAB statement **DEF:122, LR:257**

COUNT operand (MSGTXT statement) **LR:387**

COUNT operand on the MSGTXT statement **MGD:22**

counter allocation **LR:457**

counter tests **GU:40**

counters **DEF:257—260, STL:22**

See also sequence counters, index counters

altering values with the SET

statement **MGD:125—133**

generating messages dynamically with **MGD:38—42**

setting a counter

to a specific value or random number **MGD:128**

to EBCDIC data **MGD:132**

to hexadecimal data **MGD:130**

to the cursor's row, column, or offset **MGD:132**

to the index of the last item in a user

table **MGD:133**

to the length of user area or save area

data **MGD:131**

to the number of rows or columns in a

display **MGD:133**

with multiplication, division, and remainder

division **MGD:128**

setting values with the SET statement **MGD:125**

CPI-C **DEF:87, STL:39**

APPCLU as a VTAM application program

password **LR:94**

symbolic name, defining **LR:94**

control statements **MGD:117, MGD:119**

delimiters **MGD:55**

IF statement restrictions **MGD:84**

transaction program

defining **LR:99**

- CPI-C Architecture
 - application programming interface (API) **DEF:87**
 - conversation **DEF:90, DEF:92**
 - inbound **DEF:87**
 - outbound **DEF:87**
 - logical unit **DEF:89, DEF:91**
 - partner LU name **DEF:87, DEF:88, DEF:90**
 - type 6.2 (APPC) **DEF:92**
 - side information **DEF:89**
 - Systems Application Architecture CPI-C **DEF:87**
 - TPNS differences from CPI-C 1.1 **DEF:93**
 - CM_SYNC_POINT conversation synchronization level **DEF:93**
 - maximum send length on mapped conversations **DEF:93**
- CPI-C example
 - with multiple-instance TPs
 - message generation decks **SMP:127**
 - network definition statements **SMP:126**
 - sample network **SMP:125**
 - STL procedures **SMP:129**
 - with single-instance TPs
 - message generation decks **SMP:119**
 - network definition statements **SMP:118**
 - sample network **SMP:117**
 - STL procedures **SMP:122**
- CPI-C logical unit (APPCLU statement)
 - password (PASSWD operand) **DEF:89**
 - session limit (CNOS operand) **DEF:90**
 - side information (SIDEINFO operand) **DEF:90**
 - symbolic name (APPLID operand) **DEF:89, DEF:91**
- CPI-C model **SMP:396**
- CPI-C simulation **DEF:89**
- CPI-C trace records **UE:42**
- CPI-C transaction program
 - client **DEF:87, DEF:90, DEF:92**
 - CNOS **DEF:90, DEF:92**
 - instance **DEF:87, DEF:88, DEF:90**
 - server **DEF:87, DEF:90, DEF:92**
 - symbolic destination name **DEF:89**
 - transaction program name **DEF:92**
- CPI-C transaction program configuration
 - communication controller for **PI:12, PI:47**
 - network definition statements for **PI:59**
 - requirements **PI:11**
 - use of **PI:42**
 - using **PI:44, PI:46, PI:47**
- CPI-C transaction program network definition
 - message generation deck example **DEF:101—103**
 - network definition considerations
 - CYCLIC operand on PATH statement **DEF:92**
 - DIST statement **DEF:92**
 - sharing save areas and counters **DEF:93**
 - using counters **DEF:93**
 - using literal values **DEF:93**
 - using named constants **DEF:93**
 - using network save areas **DEF:93**
- CPI-C transaction program network definition (*continued*)
 - network definition statements
 - APPCLU **DEF:89, DEF:91, DEF:92**
 - SIDEEND **DEF:89**
 - SIDEENT **DEF:89**
 - SIDEINFO **DEF:89**
 - TP **DEF:88, DEF:90, DEF:92**
 - network generation STL example **DEF:95—100**
 - side information
 - on APPCLU statement **DEF:89, DEF:90**
 - statement group **DEF:89**
- CPI-C transaction program simulation
 - activating **DEF:87, DEF:92**
 - coding **DEF:91**
 - designing **DEF:91**
 - logging **DEF:90**
 - logical network configuration figure **DEF:88**
 - message deck **DEF:90**
 - path **DEF:87, DEF:88, DEF:90, DEF:92**
 - physical configuration for **PI:44, PI:46**
 - repeating **DEF:88, DEF:90, DEF:92**
 - stagger time **DEF:90**
 - statistics **DEF:90**
 - terminating **DEF:88, DEF:92**
 - trace information **DEF:88, DEF:90**
 - user control data (UCD) **DEF:90**
- CPI-C verbs
 - defining **STL:179—212**
 - using **STL:39**
- CPI-C VTAM considerations
 - coordinating CPI-C and VTAM **DEF:94**
 - VTAM ACB **DEF:89, DEF:94**
 - VTAM access method control block (ACB) **DEF:89**
- CPICCON CPI-C constants **STL:395**
- CPICVAR CPI-C variables **STL:395**
- CPINIT operand **DEF:187**
- CPINIT operand (PU21 statement) **LR:243**
- CPITRACE operand **GU:36**
- CPITRACE operand (A command) **OP:132**
- CPITRACE operand (TP statement) **LR:99**
- CPOS function **STL:65, STL:318—318**
- CPSESCAP operand **DEF:187**
- CPSESCAP operand (PU21 statement) **LR:244**
- CPSESS operand **DEF:187**
- CPSESS operand (ADJNODE statement) **LR:216**
- CR operand (TEXT statement) **LR:424**
- CRDATA LN operand
 - DEV (TCP/IP) statement **LR:130**
 - DEV (terminal) statement **LR:180**
 - LU (cross-domain) statement **LR:268**
 - LU (type 2.1 node) statement **LR:226**
 - LU (VTAMAPPL) statement **LR:112**
 - TERM statement **LR:199**
- creating self-checking scripts **MGD:111, MGD:112**

- cross-domain link
 - defined **PI:38**
- cross-domain logging (CDLOG) option
 - example **GU:248**
 - using **GU:247**
- cross-domain resource manager definition
 - statement **LR:248**
- cross-domain resources, controlling **OP:51—54**
- cross-domain simulation **LR:15**
- CROSSKEY operand (CDRM statement) **LR:248**
- CROW function **PRI:73, STL:64, STL:319**
- cryptography
 - See encrypting/decrypting messages
- CTAB statement **LR:326, STL:215**
- CTRC command **GU:64**
- CUADDR operand
 - LINE (type 2.1 node) statement **LR:218**
 - LPORT statement **LR:146**
- Cumulative Response Time Distribution **GU:169**
- CURSOR command **GU:151**
- cursor movement, simulating **MGD:159, STL:57**
- CURSOR operand
 - IF (message generation) statement **LR:364**
 - IF (network configuration) statement **LR:47**
- cursor position
 - functions used to identify **STL:64**
- cursor position tests **GU:39**
- cursor positioning keys, simulating **LR:327**
- Cursor Select key statement **LR:329**
- CURSOR statement **LR:327, MGD:159, STL:57, STL:216—217**
- CURSSEL AID key **STL:75**
- CURSSEL statement **LR:329, STL:56, STL:218**
- customizing Product Identifier Subvector
 - statement **LR:238**
- customizing response time monitor **LR:84**
- CWL operand (APPCLU statement) **LR:95**
- CWP operand (APPCLU statement) **LR:95**
- CYCLIC operand (PATH statement) **LR:81**
- CYCLIC operand on network definition
 - statements **MGD:191**
- cyclic path selection **MGD:191**

D

- D (Dump Control Block) operator command **OP:141**
- D operand (A command) **OP:133**
- data **STL:24**
 - combining types of **MGD:30**
 - entering into messages generated manually **MGD:29**
 - inserting into a message **MGD:48**
 - saving in an area **MGD:43**
- data base
 - for TPNS test **PI:72**

- data base management, TPNS/ATP
 - archiving for recovery **SGU:177**
 - CLIST, example of **SGU:180**
 - data sets required **SGU:182**
 - free space, reallocating **SGU:177**
 - JCL, example of **SGU:178**
- data circuit-terminating equipment
 - See X.25 packet-switched network (PSN), DCE (data circuit-terminating equipment)
- DATA command **GU:65**
- data field options **LR:437—447**
 - \$ATTR\$ **MGD:167**
 - \$BADGE\$ **MGD:31, MGD:182**
 - \$CARD\$ **MGD:182**
 - \$DIDO\$ **MGD:182**
 - \$DSEQ\$ **MGD:39**
 - \$DUP\$ **MGD:185**
 - \$FN\$ **MGD:160**
 - \$LSEQ\$ **MGD:39**
 - \$MAG10\$ **MGD:182**
 - \$MAG10S\$ **MGD:182**
 - \$MAG63\$ **MGD:182**
 - \$MAG63S\$ **MGD:182**
 - \$MONTH\$ **MGD:31**
 - \$NL\$ **MGD:160**
 - \$NSEQ\$ **MGD:39**
 - \$RNUM\$ **MGD:32**
 - \$SEQ\$ **MGD:39**
 - \$TAB\$ **MGD:160**
 - \$TSEQ\$ **MGD:39**
 - \$UTBL\$ **MGD:35**
 - \$YEAR\$ **MGD:31**
 - syntax conventions **MGD:30**
 - understanding **MGD:31**
- data link control role **DEF:181**
- data messages
 - logging **GU:245**
- DATA operand
 - CMND statement **LR:310**
 - FILE statement **LR:44**
- DATA operand (U command) **OP:162**
- data set
 - SYSPRINT, dynamic allocation of **OP:12**
- data sets, allocating **DEF:25**
- data terminal equipment
 - See X.25 packet-switched network (PSN), DTE (data terminal equipment)
- data tests **GU:37**
- data type selection control commands **GU:58**
- DATAHOST operand (SSCP statement) **LR:288**
- DATASAVE operand
 - IF (message generation) statement **LR:369**
 - IF (network configuration) statement **LR:51**
- DATASAVE statement **LR:330—337**
 - converting DBCS data in a save area with **MGD:47**
 - manipulating data in a save area with **MGD:46**

DATASAVE statement (*continued*)
 placing data in a user or save area with **MGD:43**
 using **MGD:118**
 date and time
 functions for providing **STL:61**
 obtaining **STL:61**
 DATE data field option **LR:438**
 DATE function **STL:322**
 date functions
 listed **STL:61**
 using **STL:61**
 DAY data field option **LR:439**
 DAY function **STL:324**
 DBCS
 See also double-byte character set (DBCS)
 character set ID **DEF:228**
 logic testing **MGD:105**
 simulating **MGD:170**
 simulating devices **DEF:234**
 DBCS data, converting **MGD:47**
 DBCS data, simulating **STL:58**
 DBCS operand **DEF:234**
 DEV (TCP/IP) statement **LR:130**
 DEV (terminal) statement **LR:180**
 LU (cross-domain) statement **LR:268**
 LU (type 2.1 node) statement **LR:226**
 LU (VTAMAPPL) statement **LR:113**
 DBCS, logic testing **STL:59**
 DBCS2SB function **STL:328**
 DBCS2SB function (DATASAVE statement) **LR:332**
 DBCS2SB function on the DATASAVE
 statement **MGD:47**
 DBCSADD function **STL:325**
 DBCSADD function (DATASAVE statement) **LR:332**
 DBCSADD function on the DATASAVE
 statement **MGD:47**
 DBCSADJ function **STL:326**
 DBCSADJ function (DATASAVE statement) **LR:332**
 DBCSADJ function on the DATASAVE
 statement **MGD:47**
 DBCSCSID operand **DEF:228**
 DEV (TCP/IP) statement **LR:130**
 DEV (terminal) statement **LR:180**
 LU (cross-domain) statement **LR:268**
 LU (type 2.1 node) statement **LR:226**
 LU (VTAMAPPL) statement **LR:113**
 DBCSDEL function **STL:327**
 DBCSDEL function (DATASAVE statement) **LR:332**
 DBCSDEL function on the DATASAVE
 statement **MGD:47**
 DBITMOD operand (MCH statement) **LR:165**
 DBITSENT operand (MCH statement) **LR:165**
 DBITSUP operand (MCH statement) **LR:165**
 DC operand (A command) **OP:133**
 DCE
 See X.25 packet-switched network (PSN), DCE (data
 circuit-terminating equipment)

DDNAME operand
 MSGDISK statement **LR:68**
 NTRKLOG statement **LR:80**
 DEACT statement **LR:338**
 coding example **MGD:136**
 deactivating events with **MGD:136**
 deactivating logic tests with **MGD:101**
 deactivating ON SIGNED conditions **STL:97**
 deactivating ONIN and ONOUT conditions **STL:83**
 described **STL:219—220**
 deactivate logic test statement **LR:338**
 deactivating cross-domain resources **OP:52**
 deactivating logic tests **MGD:101**
 deactivating ONIN statements
 See ONIN statement, deactivating
 deactivating ONOUT statements
 See ONOUT statement, deactivating
 DEBUG command, ITPLSGEN **SGU:54**
 DEBUG operand (A command) **OP:133**
 DEBUG option
 in message logging **GU:245**
 debugging IDC problems
 3270 log display (DSPY) records **SGU:40**
 analyzing the IDC log **SGU:40**
 analyzing the IDC trace **SGU:41**
 IDC restrictions **SGU:41**
 receive (RECV) records **SGU:40**
 transmit (XMIT) records **SGU:40**
 declarative statements
 allocating TPNS resources using **STL:31**
 coding, placement of **STL:27**
 constants, declaring names **STL:28**
 use of **STL:18, STL:27**
 user tables, declaring **STL:29**
 variables, declaring explicitly **STL:28**
 decryption
 See encrypting/decrypting messages
 DEF operand (RTM statement) **LR:84**
 defining
 intermessage delays for specific resources **MGD:75**
 selection of message generation decks **MGD:190**
 user tables with the UTBL network definition state-
 ment **MGD:34**
 defining networks
 See networks, defining
 defining the message generation deck name,
 ITPLSGEN **SGU:51**
 defining the STL program name, ITPLSGEN **SGU:51**
 defining the TPNS/NetView operator interface
 command processors and CLISTs **PI:22**
 defining the operator interface to the NetView
 program **PI:21**
 subtasks **PI:22**
 TPNS load library **PI:21**
 defining the operator interface to the operating
 system **PI:20**

- defining the TPNS/NetView operator interface (*continued*)
 - defining the operator interface to TPNS **PI:23**
 - defining the operator interface to VTAM **PI:20**
 - JCL **PI:23**
- defining user delays, ITPLSGEN **SGU:52**
- defining user exits **STL:39**
- definition statements, network
 - See network definition statements
- definitions **PRI:4**
- DELAY command **SGU:84**
- DELAY command, ITPLSGEN **SGU:52**
- DELAY network definition operand, using to control intermessage delay **STL:79**
- DELAY operand **PRI:21**
 - DEV (TCP/IP) statement **LR:130**
 - DEV (terminal) statement **LR:180**
 - IF (message generation) statement **LR:369**
 - IF (network configuration) statement **LR:52**
 - LU (cross-domain) statement **LR:269**
 - LU (type 2.1 node) statement **LR:227**
 - LU (VTAMAPPL) statement **LR:113**
 - TERM statement **LR:199**
 - TP statement **LR:100**
- DELAY operand on network definition statements **MGD:75**
- DELAY statement **LR:340, STL:80, STL:221—222**
 - coding **MGD:77**
 - using **MGD:119**
- delay value
 - defined **MGD:70**
 - specifying for individual resources **MGD:75**
- delay, intermessage
 - defined **STL:78**
 - factors determining length
 - listed **STL:79**
 - load on TPNS system **STL:81**
 - network definition statements **STL:79**
 - STL statements **STL:80**
 - receiving messages during **STL:81**
- delaying execution **LR:340**
- delays, canceling **STL:87**
- delays, defining user (ITPLSGEN) **SGU:52**
- delays, message generation
 - See network options
- delays, overriding user (ITPLSGEN) **SGU:53**
- DELETE function (DATASAVE statement) **LR:332**
- DELETE function on the DATASAVE statement **MGD:46**
- delete key statement **LR:342**
- delete key, simulating **LR:342, MGD:159**
- DELETE statement **LR:342, MGD:159, STL:223**
- delimiters
 - See *also* conditional delimiters, unconditional delimiters
 - affecting the message generation process **MGD:53**

- delimiters (*continued*)
 - classifying **MGD:54**
 - coding message generation decks with **MGD:67**
 - conditional **MGD:65—67**
 - CPI-C simulation **MGD:55**
 - described **MGD:26**
 - interrupting message generation with **MGD:56—65**
 - unconditional **MGD:56—65**
 - use of **MGD:53—68**
- DELSTR function **STL:329**
- DELYSEED operand (NTWRK statement) **LR:71**
- dependent LUs **DEF:182**
- designing STL programs
 - documenting the STL program
 - comment block at beginning **STL:15**
 - comments in programs **STL:15**
 - test plans, retaining **STL:15**
 - planning
 - message content, determining **STL:14**
 - procedures, structuring **STL:14**
 - programs, structuring **STL:13**
 - steps for program development **STL:13**
 - special requirements
 - message generation statements, including **STL:13**
 - products, programs, and resources **STL:12**
 - running the test **STL:12**
 - testing STL programs
 - logic testing **STL:15**
 - syntax testing **STL:15**
- designing the test plan **PI:30**
- DESTNAME operand
 - SIDEENT statement **LR:87**
 - SIDEINFO operand (APPCLU statement) **LR:97**
- DESTSA operand (PATH statement) **LR:280**
- determining test objectives
 - example **PI:40**
- determining variable types **PRI:44**
- DEV command **GU:131**
- DEV control block **UE:58**
- DEV statement **DEF:22, DEF:23, DEF:172, LR:127, LR:176, PI:61, PI:63**
- developing a network definition **PRI:4, PRI:9**
 - multiple device simulation **PRI:34**
 - multiple procedures **PRI:30**
 - simple simulation **PRI:20**
- developing STL input **PRI:5, PRI:9**
 - calling other procedures **PRI:57**
 - DO groups **PRI:49**
 - functions **PRI:74**
 - IF statements **PRI:47**
 - multiple procedures **PRI:29**
 - ONIN statements **PRI:84**
 - SELECT groups **PRI:53**
 - simple simulation **PRI:15**
 - user tables **PRI:80**

- device control block **UE:58**
- device counters **LR:457**
- device definition statement **LR:127—137, LR:176—191**
- device ID functions
 - described **STL:61**
 - listed **STL:101**
- device key statements
 - for various devices **STL:385**
 - listed **STL:55**
 - types of **STL:55**
 - using **STL:54**
- device keys, simulating
 - editing keys **STL:55**
 - keys available for various devices **STL:385**
 - keys for moving around on screen **STL:55, STL:56**
 - screen attribute keys **STL:55**
- device parameters, changing **UE:2**
- device type codes **UE:53**
- device, specifying (IPTLSGEN) **SGU:52**
- devices, simulating
 - Airlines Line Control (ALC) terminals **DEF:245**
 - badge reader **STL:59**
 - BSCx terminals **DEF:244—245**
 - card reader **STL:59**
 - digital input source **STL:59**
 - IBM 1050 Data Communication System **DEF:237**
 - IBM 2740 Communication Terminal, Model 1 and 2 **DEF:238**
 - IBM 2741 Communication Terminal **DEF:239**
 - IBM 3101 Model 2 Display Terminal **DEF:242—244**
 - IBM 3174 Dynamic Definition of Dependent Logical Unit **DEF:236**
 - IBM 3174 Network Asset Management Function **DEF:236**
 - IBM 3270 Information Display System **DEF:108, DEF:223—228**
 - IBM 3274 Response Time Monitor **DEF:234—236**
 - IBM 3290 Information Panel **DEF:228—234**
 - IBM 5250 Display System **DEF:247**
 - IBM 8100 Information System **DEF:249—255**
 - IBM 8775 Display Terminal **DEF:246**
 - IBM Personal Computers (PCs) **DEF:255**
 - magnetic input **STL:59**
 - magnetic stripe reader **STL:59**
 - Model 20 Multileaving Workstation **DEF:248**
 - TPNS USER terminal **DEF:249**
 - Western Union Model 33/35 TWX Terminal **DEF:241—242**
- devices, simulating specific types of **PI:67**
- DEVID data field option **LR:439**
- DEVID function **STL:330**
- DIAG operand (MCH statement) **LR:165**
- DIAG98 option for virtual machine **PI:17**
- dial lines (SDLC) **DEF:117**
- dial lines (TWX) **DEF:241**
- DIAL operand
 - LINE (single-domain) statement **LR:193**
 - LINE (type 2.1 node) statement **LR:218**
- DIAL operand (A command) **OP:133**
- DIDO data field option **LR:439**
- DIDO function **STL:59, STL:331**
- Differences Report **GU:112, PI:114**
- digital input source, simulating
 - See* devices, simulating, digital input source
- direct-attach configuration
 - See also* physical configuration, types
 - communication controller for **PI:13, PI:54**
 - limitations of **PI:53**
 - use of **PI:42**
 - using **PI:53**
- direct-attached cross-domain model **SMP:316**
- Dispatcher and Channel Traces (T) operator command **OP:161**
- dispatcher trace
 - command **OP:161**
 - description of **OP:49**
- display color select statement **LR:325**
- display devices, changing parameters for **UE:3**
- display highlight select statement **LR:361**
- display image, logging for formatting by the Loglist Utility **MGD:169**
- Display Monitor **PRI:5, PRI:9**
 - entering data **PRI:26, PRI:31**
 - establishing multiple sessions **PRI:36**
 - example screen **PRI:26**
 - logging on **PRI:25, PRI:31**
 - stopping Display Monitor **PRI:26**
- Display Monitor Facility
 - See* TPNS Display Monitor Facility
- Display Monitor Facility (M) operator command **OP:148**
- DISPLAY operand **PRI:22**
 - DEV (TCP/IP) statement **LR:131**
 - DEV (terminal) statement **LR:181**
 - LOG statement **LR:384**
 - LU (cross-domain) statement **LR:269**
 - LU (type 2.1 node) statement **LR:227**
 - LU (VTAMAPPL) statement **LR:114**
 - TERM statement **LR:200**
- display partition control block **UE:74**
- display records, formatting **PRI:68**
- display terminal
 - following message generation for **MGD:146**
 - generating messages for **MGD:143**
 - logic testing for **MGD:108**
- displaying image at NetView terminal **LR:417**
- displaying the simulated 3270 display image **LR:385**
- DIST network definition statement **MGD:191**
- DIST statement **LR:42**
- DLCI operand (LLS (frame relay) statement) **LR:153**
- DLOGMOD operand
 - DEV (terminal) statement **LR:181**

- DLOGMOD operand (*continued*)
 - LU (cross-domain) statement **LR:270**
 - LU (type 2.1 node) statement **LR:228**
 - LU (VTAMAPPL) statement **LR:114**
 - TERM statement **LR:200**
- DLOGMOD operand (M command) **OP:149**
- DO FOREVER statement **STL:46, STL:225**
- DO FOREVER statement groups **PRI:49**
 - example **PRI:49**
 - explanation **PRI:51**
- DO statement groups exercise **PRI:49**
- DO statement, iterative **STL:47, STL:226**
- DO statement, simple **STL:45, STL:224**
- DO statements **STL:45, STL:224—226**
- DO WHILE statement **STL:46, STL:224**
- DO WHILE statement groups
 - general usage **PRI:49**
 - logon discussion **PRI:16**
 - logon example **PRI:16**
- documenting STL programs
 - comment block at beginning **STL:15**
 - comments in programs **STL:15**
 - test plans, retaining **STL:15**
- dollar sign (Exit) operator command **OP:171**
- domain
 - defined **PI:37**
- domain simulation
 - defined **PI:37**
- double-byte character set
 - See DBCS
- double-byte character set (DBCS)
 - \$ (Exit) operator command **OP:171**
 - A (Alter) operator command **OP:131, OP:134**
 - BTRANS command (Response Time Utility) **GU:194**
 - CHECKONLY command (Log Compare Utility) **GU:149**
 - ETRANS command (Response Time Utility) **GU:197**
 - EXCLUDE command (Log Compare Utility) **GU:133**
 - F (Enter Console Recovery) operator command **OP:144**
 - Log Compare Utility **GU:113**
 - Loglist Utility **GU:47**
 - M (Display Monitor Facility) operator command **OP:149**
 - MASK command (Log Compare Utility) **GU:155**
 - SELECT command (Log Compare Utility) **GU:140**
 - START command (Log Compare Utility) **GU:141**
 - SYNCPOINT command (Log Compare Utility) **GU:142**
- DOWN operand
 - CURSOR statement **LR:327**
 - SCROLL statement **LR:405**
- DSEQ data field option **LR:439**
- DSEQ operand (A command) **OP:133**
- DSPLY command **GU:67**

- DSPY records, understanding of **GU:99**
- DTE
 - See X.25 packet-switched network (PSN), DTE (data terminal equipment)
- DTEADDR operand (MCH statement) **LR:165**
- Dump Control Blocks (D) operator command **OP:141**
- DUP data field option **LR:439**
- DUP statement **LR:343, STL:227**
- duplicate key statement **LR:343**
- duplicate key, simulating **LR:343**

E

- E (execute) for THEN and ELSE on the IF statement **MGD:94**
- E (Restart Message Logging) operator command **OP:81, OP:142**
- E operand (A command) **OP:133**
- E2D function **STL:332—332**
- EBCDIC string
 - using to represent a random number **STL:61**
- echo function, non-3270 device **GU:232**
- ECHO script **SMP:85**
- editor profile, ISPF **PRI:14, PRI:94**
- EL data field option **LR:439**
- ELEMENT operand
 - CDRM statement **LR:248**
 - SSCP statement **LR:288**
- Eligible Terminal Report (SNA 3270 Reformatter Utility) **SGU:63**
- ELSE operand
 - IF (message generation) statement **LR:369**
 - IF (network configuration) statement **LR:52**
- ELSE operand on the IF statement
 - E (execute) on **MGD:94**
 - IGNORE action on **MGD:97**
 - on multiple IF statements **MGD:96**
 - use of **MGD:92**
 - VERIFY—(data) on **MGD:95**
- EMTRATE **DEF:263**
- EMTRATE network definition operand, using to control intermessage delay **STL:80**
- EMTRATE operand (NTWRK statement) **LR:72**
- ENA (extended network addressing)
 - See SNA (Systems Network Architecture), extended network addressing
- ENCR operand
 - DEV (terminal) statement **LR:182**
 - LU (cross-domain) statement **LR:270**
 - LU (type 2.1 node) statement **LR:228**
 - LU (VTAMAPPL) statement **LR:114**
 - MODEENT statement **LR:63**
 - TERM statement **LR:201**
- encrypted data
 - logging **GU:246**

- encrypting/decrypting messages
 - cryptographic keys, types **DEF:292**
 - general simulation support **DEF:292—293**
 - in TPNS simulations **DEF:291—305**
 - MVS Programmed Cryptographic Facility (PCF)
 - network definition statement operands **DEF:291**
 - subarea simulations
 - cryptographic session level **DEF:303—303**
 - network definition example **DEF:298**
 - simulating primary LU **DEF:303—305**
 - simulating secondary LU **DEF:302—303**
 - TPNS requirements **DEF:300—301**
 - VTAM requirements **DEF:299—300**
 - terminal simulations
 - command sequences **DEF:296—297**
 - cryptographic session level **DEF:297**
 - network definition example **DEF:293**
 - TPNS requirements **DEF:295**
 - VTAM requirements **DEF:294—295**
 - VTAM Encrypt/Decrypt program **DEF:291**
- encryption
 - See* encrypting/decrypting messages
- END Command
 - ITPECHO **GU:235**
 - Log Compare Utility **GU:152**
 - Loglist Utility **GU:68**
 - Response Time Utility **GU:196**
- END command, ITPLSGEN **SGU:55**
- end message generation deck statement **LR:344**
- end of run reports **MGD:205, OP:67**
 - using **PI:109**
- ending ITPLSGEN **SGU:55**
- ENDTXT statement **LR:344, PRI:17, STL:228**
- ENDTXT statement, coding **MGD:21**
- ENDUTBL statement **STL:229**
- ENTER AID key **STL:75**
- Enter Console Recovery (F) operator
 - command **OP:143**
- Enter key statement **LR:345**
- Enter key, simulating **LR:345, MGD:145**
- ENTER statement **LR:345, MGD:144, MGD:145**
- enter-echo function **GU:229**
- entering
 - data for messages generated manually **MGD:29**
 - message text **MGD:19**
 - operand names and values **MGD:20**
- entering commands from the console,
 - ITPLSGEN **SGU:54**
- entering data on a simulated terminal (TYPE statement) **PRI:16**
- entering Display Monitor data **PRI:26, PRI:31**
- entries in user table, selecting **STL:63**
- entry criteria **PI:35**
- EOB operand (TEXT statement) **LR:424**
- ER operand (PATH statement) **LR:280**
- Erase Input key statement **LR:348**
- Erase Input key, simulating **LR:348**
- Erase to End of Screen statement **LR:347**
- Erase to the End of Field key statement **LR:346**
- Erase to the End of Field key, simulating **LR:346**
- EREOF statement **LR:346, STL:230**
- EREOS key, simulating **LR:347**
- EREOS statement **LR:347, STL:231**
- ERIN statement **LR:348, STL:232**
- ERLIMIT operand
 - CNTLR statement **LR:253**
 - SSCP statement **LR:288**
- ERRCOUNT command **GU:132**
- error messages, MFS **SGU:114**
- error simulation statement **LR:349**
- ERROR statement **LR:349, MGD:120, MGD:161**
- error, logical simulating **LR:399**
- error, simulating **LR:349**
- errors from undeclared variables
 - discussion **PRI:44**
 - error messages **PRI:44**
- errors, correcting
 - error messages, reading **STL:135**
 - event dictionary, using **STL:138**
 - logic **STL:140**
 - proper method for **STL:139**
 - syntax **STL:133**
 - variable dictionary, using **STL:136**
- errors, line **OP:119**
- ERRORT operand (FPORT statement) **LR:150**
- ESC key, simulating **LR:350**
- ESC statement **LR:350, STL:233**
- establishing a session **PRI:15**
- establishing a session using IDC
 - IDC Main Panel **SGU:11**
 - IDC Start Session Panel **SGU:12**
 - logging on to an application **SGU:12**
 - logging on to IDC **SGU:10**
- establishing Display Monitor sessions **PRI:5, PRI:9**
 - entering data **PRI:26, PRI:31**
 - establishing multiple sessions **PRI:36**
 - example screen **PRI:26**
 - logging on **PRI:25, PRI:31**
 - stopping Display Monitor **PRI:26**
- establishing multiple Display Monitor sessions **PRI:36**
- Ethernet model **SMP:323**
- ETHTYPE operand (LLS statement) **LR:143**
- ETRANS command **GU:173, GU:197—199**
- evaluating logic tests
 - using WHEN=IMMED **MGD:103**
 - using WHEN=IN **MGD:103**
 - using WHEN=OUT **MGD:103**
- event dictionary
 - example **STL:139**
 - reading **STL:139**
 - using to find errors **STL:138**

EVENT operand
 FE statement **LR:43**
 IF (message generation) statement **LR:364**
 IF (network configuration) statement **LR:47**
 ON statement **LR:391**
 WAIT statement **LR:430**
 EVENT statement **LR:351—352, MGD:134**
 event tests **GU:38**
 EVENT WAIT indicator **MGD:63**
 events
 canceling **STL:98**
 coding examples **MGD:140**
 coding variable event names **MGD:137**
 controlling communication with **MGD:138**
 deactivating with the DEACT statement **MGD:133**
 names **STL:92, STL:95**
 ON/SIGNAL facility **MGD:139**
 posting **STL:92**
 signaling **STL:95**
 TAG **STL:92, STL:95**
 using to synchronize multiple devices **MGD:140**
 WAIT/POST facility **MGD:138**
 EVENTTAG operand
 CANCEL Statement **LR:303**
 EVENT statement **LR:352**
 example Display Monitor screen **PRI:26**
 example Loglist Utility output
 display records **PRI:69**
 ONIN and ONOUT statements **PRI:85**
 STL functions **PRI:78**
 STL trace records **PRI:66**
 user tables **PRI:81**
 example of SELECT statement **PRI:53**
 example of variable declarations **PRI:45**
 example preprocessor error listing **PRI:23**
 example STL error listing **PRI:18**
 example STL program listing **PRI:67**
 EXCLUDE command **GU:133**
 EXDEV command
 Loglist Utility **GU:69**
 EXEC
 using to run STL Translator **STL:118**
 execute for THEN or ELSE on the IF
 statement **MGD:94**
 execute procedure
 defined **STL:85**
 functions not allowed in **STL:86**
 statements not allowed in **STL:85**
 EXECUTE statement **STL:85—86, STL:234**
 executing the Loglist Utility **PRI:65**
 execution parameters **OP:17**
 ITPATPSG **SGU:168**
 ITPECHO **GU:233**
 ITPIDC **SGU:9**
 ITPLSGEN **SGU:49**
 ITPLU2RF **SGU:61**
 execution parameters (*continued*)
 ITPSDL **SGU:148**
 ITPSGEN **SGU:81**
 ITPTCL **SGU:140**
 Log Compare Utility **GU:113**
 Loglist Utility **GU:47**
 Response Time Utility **GU:177**
 TPNS Preprocessor **GU:22**
 TPNS/ATP **SGU:121**
 execution parameters, STL Translator
 NOPDSOUT **STL:114**
 NOSEQOUT **STL:114**
 NOSOURCE **STL:114**
 NOTPNS **STL:114**
 PROGRAM=name **STL:115**
 PRTLNCNT=nnn **STL:115**
 using **STL:114**
 exercises
 calling other procedures **PRI:57**
 DO statement groups **PRI:49**
 functions **PRI:73**
 IF statements **PRI:47**
 multiple devices **PRI:33**
 multiple procedures **PRI:29**
 ONIN statements **PRI:83**
 overview **PRI:9**
 SELECT statements **PRI:53**
 simple simulation **PRI:13**
 trace records **PRI:61**
 user tables **PRI:79**
 variable declarations and comments **PRI:43**
 Exit (\$) operator command **OP:171**
 EXIT command
 Loglist Utility **GU:70**
 Response Time Utility **GU:200**
 exit criteria **PI:35**
 exit interface routine
 description of **UE:31**
 registers **UE:32**
 return codes **UE:36**
 EXIT operand
 parameter list **UE:51**
 sample routine **UE:52**
 EXIT operand (NTWRK statement) **LR:72**
 exit routines **DEF:276**
 CNTLREX operand **UE:29**
 EXIT operand **UE:51**
 EXIT statement **UE:22**
 INEXIT operand **UE:3**
 INFOEXIT operand **UE:16**
 LNEREXIT operand **UE:19**
 NCTLEXIT operand **UE:10**
 NETEXIT operand **UE:14**
 OUTEXIT operand **UE:7**
 PUEXIT operand **UE:30**
 UCMDEXIT operand **UE:11**

- exit routines (*continued*)
 - UXOCEXIT operand **UE:21**
 - VREXIT operand **UE:28**
- EXIT statement **LR:353, MGD:119**
 - discussion **UE:22**
 - parameter list **UE:23**
 - return codes **UE:23**
 - sample routine **UE:24**
 - sample statements **UE:27**
- exiting from message generation processing **LR:353**
- expressions
 - bit, defined **STL:35**
 - integer
 - constant, defined **STL:33**
 - defined **STL:33**
 - operators for **STL:33**
 - string
 - defined **STL:34**
 - operators for **STL:34**
- expressions not allowed in asynchronous
 - conditions **STL:387**
- extended BIND **DEF:182**
- extended color **MGD:163**
- extended highlighting **MGD:164**
- extended network addressing (ENA)
 - See* SNA (Systems Network Architecture), extended network addressing
- extended subarea addressing
 - See* SNA (Systems Network Architecture), extended subarea addressing
- extended UNBIND **DEF:183**
- EXTERM command
 - Loglist Utility **GU:71**
 - Response Time Utility **GU:201**
- external modem and real line connection method
 - See* connecting TPNS to system under test
- EXTFUN operand
 - DEV (TCP/IP) statement **LR:131**
 - DEV (terminal) statement **LR:182**
 - LU (cross-domain) statement **LR:270**
 - LU (type 2.1 node) statement **LR:228**
 - LU (VTAMAPPL) statement **LR:115**
- EXTP command **GU:72**

F

- F (Enter Console Recovery) operator
 - command **OP:143**
- facilitating program development **STL:16**
- FE statement **DEF:260, LR:43**
- Field Advance key, simulating **LR:355**
- Field Backspace key, simulating **LR:356**
- Field Exit key, simulating **LR:358**
- Field Mark key, simulating **LR:359**
- Field Minus (F-) key statement **LR:357**

- Field Minus (F-) key, simulating **LR:357**
- field outlining **DEF:234**
- Field Plus (F+) key, simulating **LR:358**
- file characteristics, defining
 - See* FTP
- FILE statement **DEF:111, LR:44**
 - See also* FTP
- File Transfer Protocol (FTP) **DEF:105, DEF:111**
 - See also* FTP
- File Transfer Protocol (FTP) client simulation
 - time stamping messages **GU:244**
- File Transfer Protocol (FTP) configuration
 - communication controller for **PI:12**
 - network definition statements for **PI:61**
 - requirements **PI:11**
 - use of **PI:42**
 - using **PI:49**
- File Transfer Protocol (FTP) example
 - message generation decks **SMP:112**
 - network definition statements **SMP:111**
 - sample network **SMP:111**
 - STL procedures **SMP:113**
- File Transfer Protocol (FTP) model **SMP:383**
- finding strings **STL:36, STL:337**
- fixed selection of table entries **MGD:36**
- FLAGS operand (RTM statement) **LR:85**
- FLDADV statement **LR:355, STL:235**
- FLDBKSP statement **LR:356, STL:236**
- FLDMINUS statement **LR:357, STL:237**
- FLDOUTLN operand **DEF:234**
 - DEV (TCP/IP) statement **LR:132**
 - DEV (terminal) statement **LR:183**
 - LU (cross-domain) statement **LR:271**
 - LU (type 2.1 node) statement **LR:229**
 - LU (VTAMAPPL) statement **LR:115**
- FLDPLUS statement **LR:358, STL:238**
- FLDVALID operand
 - DEV (TCP/IP) statement **LR:132**
 - DEV (terminal) statement **LR:183**
 - LU (cross-domain) statement **LR:271**
 - LU (type 2.1 node) statement **LR:229**
 - LU (VTAMAPPL) statement **LR:115**
- flow-of-control statements
 - See* structured flow-of-control statements
- flowchart of primer activities **PRI:10**
- FLOWNEG operand (MCH statement) **LR:166**
- FM data field option **LR:440**
- FM function **STL:333**
- FM statement **LR:359, STL:56, STL:239**
- FMPROF operand (MODEENT statement) **LR:64**
- FMTSNA command **GU:73**
- format control statements **LR:459**
- formatting
 - log data set **PI:111**
- formatting display records **PRI:68**

FPORT

See frame relay port (FPORT)

FPORT statement **DEF:170, LR:150**

FPORTID data field option **LR:440**

FPORTID function **STL:334**

frame relay **PI:65, PI:140, STL:334**

frame relay example

message generation decks **SMP:106**

network definition statements **SMP:105**

sample network **SMP:104**

STL procedures **SMP:107**

frame relay model **SMP:389**

frame relay port (FPORT)

activating scanner internal trace for **OP:152**

canceled **OP:140**

link interface, specifying **OP:24**

listing addresses of **OP:147**

query network for **OP:41, OP:156**

starting activity for **OP:160**

stopping activity for **OP:155**

frame-relay network

activating resources **DEF:176**

coordinating statements **DEF:178**

deactivating resources **DEF:176**

defining resources **DEF:169**

configuration **DEF:169**

lines and terminals **DEF:171**

example network **DEF:175**

LLS Resource Interval Report **DEF:176**

logical configuration **DEF:174**

network definition example **DEF:173**

network definition statements

CNTRLR **DEF:172**

DEV **DEF:172**

FPORT **DEF:170**

LINE **DEF:171**

LINK **DEF:172**

LLS **DEF:170**

PU **DEF:172**

PU21 **DEF:172**

TERM **DEF:171**

physical configuration **DEF:173**

simulating devices **DEF:169**

TPNS simulation support **DEF:169**

utilities **DEF:176**

FRMLGTH operand (MCH statement) **LR:166**

FRSTTXT network definition operand, defining programs used **STL:127**

FRSTTXT operand **PRI:33**

DEV (TCP/IP) statement **LR:132**

DEV (terminal) statement **LR:183**

LU (cross-domain) statement **LR:271**

LU (type 2.1 node) statement **LR:229**

LU (VTAMAPPL) statement **LR:115**

TERM statement **LR:201**

TP statement **LR:101**

FRSTTXT operand on network definition statements **MGD:192**

FTP

See also TCP/IP

defining file characteristics **DEF:111**

generating FTP messages **DEF:112**

foreign files or directories **DEF:112**

supported FTP commands **DEF:112, DEF:113**

using TPNS logic testing statements **DEF:112**

using TYPE or TEXT statements **DEF:112**

network definition statements

FILE **DEF:111**

FTPSPORT operand (TCPIP statement) **LR:139**

full-duplex model **SMP:357**

full-screen mode operation

help panels (HELP command) **OP:88**

issuing commands from **OP:88—97**

resource list function (RESLIST command) **OP:89**

simulated device display function (SHOW command) **OP:97**

FUNCTION operand (DATASAVE statement) **LR:331**

function testing **MGD:11**

function tests

conducting **PI:25**

functions **PRI:73**

CCOL function **PRI:73**

CHAR function **PRI:73**

CROW function **PRI:73**

INDEX function **PRI:73**

LENGTH function **PRI:73**

LUID function **PRI:73**

SUBSTR function **PRI:73**

TOD function **PRI:73**

functions, defined **STL:18, STL:35**

functions, non-3270 **GU:232**

future event statement **LR:43**

future events **DEF:260**

G

G (Terminal Status Query) operator command **OP:145**

GENEND

See TPNS control program generation macros

general control commands **GU:58**

general definition statements, list of **LR:291**

Generalized Trace Facility (GTF)

description **SGU:74**

reformatting data from **SGU:76**

GENERATE command, ITPLSGEN **SGU:53**

generate escape sequence statement **LR:350**

generate message generation deck

statement **LR:422—427**

generating a message generation deck,

ITPLSGEN **SGU:52**

generating an STL program, ITPLSGEN **SGU:51**

- generating changed data fields, ITPLSGEN **SGU:53**
- generating debugging comments, ITPLSGEN **SGU:54**
- generating messages
 - dynamically **MGD:30—50**
 - with \$UTBL\$ **MGD:35**
 - with index counters **MGD:40**
 - with random numbers **MGD:32**
 - with save and user areas **MGD:42**
 - with sequence counters **MGD:39**
 - with user tables **MGD:33**
 - for display terminals **MGD:143**
 - for simulated 3270 terminals **MGD:157**
 - for SNA terminals **MGD:149**
 - manually **MGD:29—30**
 - with the TEXT statement **MGD:29—51**
- generating NCP
 - See NCP
- generating rate tables
 - See rate tables, generating
- generating scripts for LU 6.2 simulations **SGU:100**
- generating scripts with IDC
 - adding panel verification logic **SGU:29**
 - choosing the type of script **SGU:21**
 - generating a message generation deck **SGU:26**
 - generating an STL program **SGU:22**
 - generating user delays **SGU:35**
 - IDC Generate Message Generation Deck Panel **SGU:26**
 - IDC Generate STL Program Panel **SGU:22**
 - introduction **SGU:21**
 - sample message generation deck **SGU:28**
 - sample message generation deck with user delays **SGU:38**
 - sample message generation deck with verification logic **SGU:33**
 - sample STL program **SGU:25**
 - sample STL program with user delays **SGU:36**
 - sample STL program with verification logic **SGU:30**
- generating scripts, ITPLSGEN **SGU:53**
- generating SENSE data **MGD:150, MGD:151**
- generating TPNS control program
 - See TPNS control program, generating
- generating unsolicited data from SNA physical units **MGD:156**
- generation macros
 - See TPNS control program generation macros
- GRAPH command **GU:202**
- GTF
 - See Generalized Trace Facility (GTF)

H

- H operand (A command) **OP:134**
- half-duplex model **SMP:357**
- halt execution (QUIESCE) **PRI:17**

- HEAD operand (NTWRK statement) **LR:72**
- HEADER command
 - Log Compare Utility **GU:153**
 - Loglist Utility **GU:74**
 - Response Time Utility **GU:203**
- header records, log data set **UE:42**
- HELP AID key **STL:75**
- HELP key, simulating **LR:360**
- help panels **OP:88, PI:101**
- HELP statement **LR:360, MGD:179**
- HEX function **STL:335**
- hexadecimal data, representing **STL:26**
- hierarchy of network definition statements
 - See network definition statements, hierarchy
- HIGH operand (RN statement) **LR:83**
- highlighting option key, simulating **LR:361**
- HIGHLIGHT operand
 - DEV (TCP/IP) statement **LR:132**
 - DEV (terminal) statement **LR:183**
 - LU (cross-domain) statement **LR:271**
 - LU (type 2.1 node) statement **LR:229**
 - LU (VTAMAPPL) statement **LR:116**
- HIGHLIGHT statement **LR:361, MGD:164, STL:240**
- HOME key, simulating **LR:362**
- HOME statement **LR:362, STL:241**
- host
 - defining logical unit in **LR:264**
- HOST operand (CNTLR statement) **LR:253**
- host processor
 - TPNS requirements **PI:1**
- host processor virtual storage
 - estimating **PI:89**
- host subarea, defining
 - See SNA (Systems Network Architecture), defining
 - host subarea resources
- HWSUBFLD operand **DEF:236**
 - DEV (terminal) statement **LR:183**
 - LU (cross-domain) statement **LR:271**
 - PSUBVECT statement **LR:238**
 - PU (cross-domain) statement **LR:284**
 - TERM statement **LR:201**

I

- I (Initialize) operator command **OP:146**
- I operand (A command) **OP:134**
- IBM 1050 Data Communication System **MGD:171**
 - logic testing **DEF:238**
 - network definition example **DEF:237**
 - simulating **DEF:237**
- IBM 2740 Communication Terminal Model 1
 - logic testing **DEF:238**
 - network definition example **DEF:238**
 - simulating **DEF:238**
- IBM 2740 Communication Terminal Model 2
 - logic testing **DEF:239**

- IBM 2740 Communication Terminal Model 2 (*continued*)
 - network definition example **DEF:239**
 - simulating **DEF:239**
- IBM 2740 Communication Terminal, Model 1 **MGD:171**
- IBM 2740 Communication Terminal, Model 2 **MGD:172**
- IBM 2741 Communication Terminal **MGD:172**
 - logic testing **DEF:240**
 - network definition example **DEF:240**
 - simulating **DEF:239**
- IBM 3101 Model 2 Display Terminal
 - generating messages for **MGD:174**
 - logic testing **DEF:244**, **MGD:175**
 - network definition example **DEF:243**
 - simulating **DEF:242—244**
- IBM 3174 Dynamic Definition of Dependent Logical Unit (DDDLU) **DEF:236**
- IBM 3174 Network Asset Management Function
 - simulating **DEF:236**
 - specifying product set identification information **DEF:236**
- IBM 3270 character set ID **DEF:228**
- IBM 3270 Information Display System **MGD:157**
 - 3271 Control Unit (BSC) definition example **DEF:225**, **DEF:226**
 - 3271 Control Unit (SDLC) definition example **DEF:226**
 - 3274 Control Unit (BSC) definition example **DEF:226**
 - 3274 Control Unit (SNA) definition example **DEF:227**
 - 3275 display station (BSC) definition example **DEF:225**
 - 3275 display station (SDLC) definition example **DEF:226**
 - 3276 Control Unit (BSC) definition example **DEF:226**
 - 3276 Control Unit (SNA) definition example **DEF:227**
 - addressing **DEF:227**
 - compatible devices **DEF:225**
 - components simulated using BSC **DEF:224**
 - components simulated using SDLC **DEF:223**
 - components, network definition examples **DEF:225—227**
 - display monitor facility **DEF:228**
 - network definition examples for components **DEF:225—227**
 - polling **DEF:227**
 - simulating **DEF:223—228**
- IBM 3270 Terminal **MGD:160**
- IBM 3274 Response Time Monitor (RTM)
 - customizing **DEF:235**
 - data flow **DEF:236**
 - response time calculation **DEF:235**
 - simulating **DEF:234—236**
- IBM 3290 Information Panel
 - operands used to define **DEF:229**
 - screen definition
 - character cell size (CCSIZE) **DEF:231**
 - display size (DISPLAY) **DEF:231**
 - screen size (UASIZE) **DEF:230**
 - screen split **DEF:230**
 - screen definition examples **DEF:231—234**
 - simulating **DEF:228—234**
- IBM 5250 Display System
 - format table, formatting **DEF:248**
 - generating messages for **MGD:179**
 - logic testing **DEF:248**, **MGD:180**
 - screen image buffer, formatting **DEF:248**
 - simulating **DEF:247—248**
 - simulating errors in **MGD:180**
 - using **MGD:178**
- IBM 8100 Information System **MGD:182**
- 3289 Line Printer, Model 3 **DEF:250**
- 3641 Reporting Terminal **DEF:251**
- 3642 Encoder Printer **DEF:251**
- 3643 Keyboard Display **DEF:251**
- 3644 Automatic Data Unit **DEF:251**
- 3645 Printer **DEF:252**
- 3646 Scanner Control Unit **DEF:252**
- 3647 Time and Attendance Terminal **DEF:252**
- network definition examples **DEF:252—255**
- simulating **DEF:249—255**
- IBM 8775 Display Terminal
 - field validation **DEF:247**
 - field validation with **MGD:178**
 - multiple partitions **DEF:247**
 - scrolling **DEF:247**
 - simulating **DEF:246**
 - using **MGD:178**
- IBM Enterprise System/9370 equipped with a token-ring adapter
 - using **PI:52**
- IBM Personal Computers (PCs) **DEF:255**
- IBM Token-Ring model **SMP:363**
- IBM Token-Ring Network script, with 3270 SNA **SMP:77**
- ID data field option **LR:440**
- ID function **STL:336**
- IDBLKNO operand (PU (type 2.1 node) statement) **LR:239**
- IDC
 - See Interactive Data Capture (IDC)
- IDC Add STL Statements Panel **SGU:18**
- IDC Change Escape Key Panel **SGU:20**
- IDC Change Log Data Set Panel **SGU:19**
- IDC data sets, allocating **SGU:4**
- IDC Escape Actions Panel **SGU:14**
- IDC examples
 - sample message generation deck **SGU:28**
 - sample message generation deck with user delays **SGU:38**

IDC examples (*continued*)

- sample message generation deck with verification logic **SGU:33**
- sample network definition **SGU:44**
- sample STL program **SGU:25**
- sample STL program with user delays **SGU:36**
- sample STL program with verification logic **SGU:30**

IDC Generate Message Generation Deck

- Panel **SGU:26**

IDC Generate STL Program Panel **SGU:22**

IDC Main Panel **SGU:11**

IDC restrictions **SGU:41**

IDC Start Session Panel **SGU:12**

identifying network resources in use

- list of functions for **STL:101**

IF (message generation) statement **LR:363—378**

IF (network configuration) statement **LR:46—60**

IF message generation statement

- activating logic tests **MGD:100**
- coding **MGD:24, MGD:85**
- deactivating logic tests **MGD:101**
- defining logic tests **MGD:83**
- examples of coding **MGD:85**
- message-level logic tests **MGD:84**

operands

- AREA **MGD:88**
- CURSOR **MGD:84**
- ELSE **MGD:92**
- EVENT **MGD:84**
- LENG **MGD:87**
- list of **MGD:86**
- LOC **MGD:84, MGD:90**
- RESP **MGD:153**
- SCANCNTR **MGD:98**
- STATUS=HOLD **MGD:102**
- TEXT **MGD:88**
- THEN **MGD:92**
- UTBL **MGD:97**
- UTBLCNTR **MGD:97**
- WHEN=IMMED **MGD:88**
- WHEN=IN **MGD:88**
- WHEN=OUT **MGD:88**

IF network definition statement

- coding **MGD:84**
- network-level logic tests **MGD:84**

IF number **STL:159**

IF statement **DEF:268, STL:43, STL:242**

IF statement exercise **PRI:47**

IF statement, categories of **GU:37**

IF statement, logic test **LR:449**

IF statements, network-level **STL:88**

IFS operand (DEACT statement) **LR:338**

IGNORE action for logic tests **MGD:97**

IMS **DEF:71**

IMS application

- scripts **SMP:26**

IMS application (*continued*)

- with 3270 BSC and SNA **SMP:26**
- with 3274 SNA **SMP:27**
- with 3600 **SMP:28**
- with TWX **SMP:28**

Inactivity Report **MGD:205**

- using **PI:110**

inactivity reports **OP:73**

INACTTO operand (LLS statement) **LR:143, LR:153**

INBFRS operand (LINE (type 2.1 node) statement) **LR:219**

INCLUDE network definition statement **MGD:192**

INCLUDE network definition statement, defining programs used **STL:127**

INCLUDE statement **LR:61**

- including data from other data sets **STL:38**
- including network definitions in STL **STL:123**
- example **STL:123**

incorrect or missing message traffic **OP:117**

INDEPEND operand (LU (type 2.1 node) statement) **LR:229**

independent LU 6.2 **DEF:182**

index counters

- See also* network options, counters
- generating messages with **MGD:40—42**
- incrementing the value of **MGD:40**

INDEX function **PRI:73, STL:36, STL:337**

indicators

- INPUT INHIBITED **MGD:144**
- preserving over asynchronous IF statements **MGD:62**
- QUIESCE **MGD:65**
- WAIT **MGD:61**
- WAIT EVENT **MGD:63**

Ineligible Terminal Report (SNA 3270 Reformatter Utility) **SGU:65**

INEXIT operand

- discussion **UE:3**
- parameter list **UE:4**
- sample routine **UE:5**

INEXIT operand (NTWRK statement) **LR:72**

INFO command **GU:75**

INFOEXIT operand

- discussion **UE:16**
- parameter list **UE:17**
- return codes **UE:17**
- sample routine **UE:17**

INFOEXIT operand (NTWRK statement) **LR:72**

Information List **PRI:95**

Information Management System (IMS) scripts

information message exit

- discussion **UE:16**
- parameter list **UE:17**
- return codes **UE:17**
- sample routine **UE:17**

informational records **GU:35, UE:42**
 INHBTMSG operand (NTWRK statement) **LR:73**
 INIT operand **PRI:22**
 DEV (terminal) statement **LR:184**
 LU (cross-domain) statement **LR:271**
 LU (type 2.1 node) statement **LR:230**
 LU (VTAMAPPL) statement **LR:116**
 TERM statement **LR:201**
 initialize a network **OP:27**
 Initialize a Network (I) operator command **OP:146**
 initializing the network **PRI:4, PRI:9**
 flow of activities **PRI:10**
 how to **PRI:25**
 initiating sessions for SNA terminals **MGD:154**
 INITSELF statement **PRI:15, STL:67—69, STL:243**
 INN operand (SSCP statement) **LR:288**
 INPUT command, ITPLSGEN **SGU:51**
 input for Loglist Utility **PRI:65**
 INPUT INHIBITED indicator **MGD:144**
 input to STL Translator
 defined **STL:106**
 input, specifying (ITPLSGEN) **SGU:51**
 input/output (I/O)
 real **PI:10**
 INSERT function **STL:338**
 INSERT function (DATASAVE statement) **LR:332**
 INSERT function on the DATASAVE
 statement **MGD:46**
 Insert key, simulating **MGD:159**
 INSERT operand (DATASAVE statement) **LR:333**
 INSERT statement **LR:379, MGD:159, STL:245**
 INSTALL1 **PI:24, SMP:8**
 INSTALL2 **PI:24, SMP:16**
 installing ITPECHO **GU:232**
 installing the TPNS/ISPF Interface
 allocating data sets **PI:19**
 ISPF/PDF Primary Option Menu changes **PI:18**
 library setup **PI:19**
 installing TPNS
 checklist for **PI:24**
 communication controller requirements **PI:12**
 defining the NetView interface **PI:20**
 general requirements **PI:10**
 host processor requirements **PI:7**
 installing the TPNS/ISPF Interface **PI:17**
 migrating from TPNS Version 3 Release 4 **PI:121**
 MVS requirements **PI:7**
 printing the TPNS Program Directory **PI:13**
 using ITPECHO **PI:7**
 VM requirements **PI:10**
 INSTANCE operand (TP statement) **LR:101**
 instance, TP
 See transaction program, instance
 integer
 condition, defined **STL:50**
 constant, defined **STL:24**
 integer (*continued*)
 expression, defined **STL:33**
 operators **STL:33**
 variable, defined **STL:22**
 INTEGER statement **STL:28, STL:246**
 integer variables **PRI:43**
 integrating decks with network
 definitions **MGD:189—202**
 inter-user communication vehicle
 See IUCV cross-memory facility
 Inter-User Communication Vehicle (IUCV) **PI:10**
 Interactive Data Capture (IDC) **MGD:4, PI:11**
 capturing data
 adding statements to the IDC log **SGU:16**
 capturing session initiation **SGU:14**
 capturing session termination **SGU:21**
 changing escape keys **SGU:20**
 changing log data sets **SGU:18**
 controlling data capture **SGU:16**
 IDC Add Statements Panel **SGU:18**
 IDC Change Escape Key Panel **SGU:20**
 IDC Change Log Data Sets Panel **SGU:19**
 IDC Escape Actions Panel **SGU:14**
 using escape actions **SGU:14**
 creating network definitions
 controlling script execution flow **SGU:44**
 synchronizing multiple scripts **SGU:45**
 debugging problems
 3270 log display (DSPY) records **SGU:40**
 analyzing the IDC log **SGU:40**
 analyzing the IDC trace **SGU:41**
 IDC restrictions **SGU:41**
 overview **SGU:40**
 receive (RECV) records **SGU:40**
 transmit (XMIT) records **SGU:40**
 description **SGU:3**
 establishing a session
 IDC Main Panel **SGU:11**
 IDC Start Session Panel **SGU:12**
 introduction **SGU:40**
 logging on to an application **SGU:12**
 logging on to IDC **SGU:10**
 generating scripts
 adding panel verification logic **SGU:29**
 choosing the type of script **SGU:21**
 generating a message generation deck **SGU:26**
 generating an STL program **SGU:22**
 generating user delays **SGU:35**
 IDC Generate Message Generation Deck
 Panel **SGU:26**
 IDC Generate STL Program Panel **SGU:22**
 sample message generation deck **SGU:28**
 sample message generation deck with user
 delays **SGU:38**
 sample message generation deck with verification
 logic **SGU:33**
 sample STL program **SGU:25**

Interactive Data Capture (IDC) (*continued*)

- generating scripts (*continued*)
 - sample STL program with user delays **SGU:36**
 - sample STL program with verification logic **SGU:30**
- modifying scripts **SGU:39**
- running TPNS **SGU:46**
- setting up IDC
 - allocating data sets **SGU:4**
 - defining IDC to VTAM **SGU:4**
- starting IDC
 - defining the job stream **SGU:6**
 - running IDC as a batch job **SGU:6**
 - running IDC as a started procedure **SGU:7**
 - running IDC from a CLIST **SGU:7**
 - running IDC from GCS **SGU:8**
 - running IDC from the TPNS/ISPF interface **SGU:6**
 - specifying execution parameters **SGU:9**
- stopping ITPIDC **SGU:21**
- understanding return codes **SGU:45**
- using help **SGU:40**
- interface between the NetView program and TPNS
 - See operator interface between the NetView program and TPNS
- intermessage delay
 - defined **STL:78**
 - factors determining length
 - listed **STL:79**
 - load on TPNS system **STL:81**
 - network definition statements **STL:79**
 - STL statements **STL:80**
 - receiving messages during **STL:81**
- intermessage delays
 - altering user time intervals with A (Alter) operator command **MGD:74**
 - coding a script with **MGD:79**
 - for individual resources **MGD:77**
 - referencing multiple user time intervals **MGD:72**
 - specifying **MGD:69**
 - specifying delay values for individual resources **MGD:75**
 - specifying multiple user time intervals **MGD:72**
 - understanding **MGD:69**
 - use of **MGD:69—81**
 - using THKTIME to determine the start of **MGD:71**
- interpreting Loglist Utility output **PRI:65**
 - display records **PRI:68**
 - ONIN and ONOUT statements **PRI:86**
 - STL functions **PRI:77**
 - STL trace records **PRI:68**
 - user tables **PRI:82**
- interval reports **MGD:204, OP:61—63**
 - information included in **PI:108**
 - using **PI:108**
- INXEXPND operand (NTWRK statement) **LR:73**
- ISPF editor profile **PRI:14, PRI:94**
- ISPF, operating TPNS under **OP:14**
- ISPF/PDF Primary Option Menu **SGU:113, SGU:118**
- ISPF/PDF scrolling facility **SGU:135**
- ISTATUS operand
 - CDRM statement **LR:249**
 - CHANL statement **LR:250**
 - CNTLR statement **LR:253**
 - LINE (peripheral link) statement **LR:258**
 - LINK statement **LR:261**
 - LU (cross-domain) statement **LR:272**
 - MCH statement **LR:166**
 - PADPARMS statement **LR:169**
 - PU (cross-domain) statement **LR:284**
 - SSCP statement **LR:289**
- ITERATE statement **STL:48, STL:247**
- iterative DO statement **STL:47, STL:226**
- iterative DO statement groups **PRI:49**
 - example **PRI:49**
 - explanation **PRI:51**
- ITIME operand (NTWRK statement) **LR:74**
- ITPATPSG
 - ASSIGN control command **SGU:160**
 - CLIST, example of **SGU:169**
 - control commands **SGU:159—167**
 - data sets required **SGU:170**
 - description **SGU:153**
 - execution parameters **SGU:168**
 - input **SGU:153**
 - JCL, example of **SGU:169**
 - message generation epilogues **SGU:157**
 - message generation prologues **SGU:157**
 - model network **SGU:154**
 - model network, example of **SGU:155**
 - output **SGU:171**
 - running **SGU:168**
 - SCRIPT control command **SGU:163**
 - signing off **SGU:157**
 - signing on **SGU:157**
 - SIGNOFF control command **SGU:166**
 - SIGNON control command **SGU:161**
- ITPBTS **MGD:5, SMP:149**
 - %. control command **SGU:185**
 - CLIST, example of **SGU:187**
 - control commands, specifying **SGU:183**
 - conversion **SGU:189**
 - data set requirements **SGU:186**
 - description **SGU:183**
 - EXEC, example of **SGU:187**
 - input, specifying **SGU:183**
 - JCL, example of **SGU:187**
 - output **PI:82, SGU:190**
 - output, example of **SGU:191**
 - processing **SGU:188**
 - restrictions **SGU:190**

ITPBTS (*continued*)

- return codes **SGU:189**
- running **SGU:186**
- running application in response and nonresponse modes **PI:82**
- running from the TPNS/ISPF interface **SGU:188**
- running in nonresponse mode **SGU:183**
- running in response mode **SGU:183**
- special handling **SGU:190**
- TPNS members, updated **SGU:193**
- using **PI:82**

ITPECHO **PRI:5, SMP:3**

- 3270 functions, list of **GU:228**
- APPLID parameter **GU:233**
- automatic string function **GU:230, GU:231**
- BIND **GU:228**
- BUFSIZE parameter **GU:233**
- changing parameters **GU:234**
- clear function **GU:232**
- description of **GU:227**
- discussion **PRI:7**
- END command **GU:235**
- enter-echo function **GU:229**
- example of screen **PRI:73**
- EXEC, example of **GU:234**
- execution parameters **GU:233**
- functional specifications of **GU:227—232**
- initial screen **GU:229**
- initiating sessions **GU:235**
- installing **GU:232**
- JCL **PRI:7**
- JCL, example of **GU:234**
- logging on **GU:235**
- logging on and off **PRI:8**
- logoff function **GU:232**
- modifications **PRI:7**
- non-3270 functions **GU:232**
- NOSMSG command **GU:235**
- NOTRACE command **GU:235**
- operator commands **GU:235**
- PASSWD parameter **GU:233**
- programming requirements **GU:227**
- repeat function **GU:231**
- repetition function **GU:230, GU:231**
- restarting **PRI:8**
- return codes **GU:236**
- running **GU:233**
- SMSG command **GU:235**
- SNA protocols **GU:228**
- storage requirements **GU:227**
- TRACE command **GU:235**
- TSO CLIST, example of **GU:234**
- VTAMAPPL statement, defining **GU:232**
- WRITE command **GU:230**

ITPECHO Utility **PI:7**

ITPIDC

See Interactive Data Capture (IDC)

ITPLGWTR

- description of **OP:77**
- preparing to use **OP:77**
- return codes **OP:80**
- running **OP:78**
- specifying use of **OP:79**
- using an EXEC, example of **OP:79**

ITPLSGEN

- description **SGU:47**
- generating output **SGU:55**
 - generating scripts from BSC or SDLC 3270 logs **SGU:55**
- running ITPLSGEN
 - introduction **SGU:48**
 - specifying execution parameters **SGU:49**
 - using CLISTs **SGU:48**
 - using EXECs **SGU:49**
 - using JCL **SGU:48**
 - using the TPNS/ISPF interface **SGU:49**
- setting up ITPLSGEN **SGU:47**
- using control commands
 - DEBUG command **SGU:54**
 - DELAY command **SGU:52**
 - END command **SGU:55**
 - entering commands **SGU:50**
 - GENERATE command **SGU:53**
 - INPUT command **SGU:51**
 - introduction **SGU:50**
 - LU command **SGU:52**
 - MSGTXT command **SGU:51**
 - NODELAY command **SGU:53**
 - NOVERIFY command **SGU:54**
 - OUTPUT command **SGU:51**
 - P command **SGU:54**
 - RUN command **SGU:55**
 - STL command **SGU:51**
 - TPNS command **SGU:52**
 - VERIFY command **SGU:53**

ITPLSGEN control commands

- DEBUG command **SGU:54**
- DELAY command **SGU:52**
- END command **SGU:55**
- entering commands **SGU:50**
- GENERATE command **SGU:53**
- INPUT command **SGU:51**
- introduction **SGU:50**
- LU command **SGU:52**
- MSGTXT command **SGU:51**
- NODELAY command **SGU:53**
- NOVERIFY command **SGU:54**
- OUTPUT command **SGU:51**
- P command **SGU:54**
- RUN command **SGU:55**
- STL command **SGU:51**

ITPLSGEN control commands *(continued)*

- TPNS command **SGU:52**
- VERIFY command **SGU:53**

ITPLSGEN examples

- generating scripts from BSC or SDLC 3270 logs **SGU:55**

ITPLU2RF

- allocating data sets **SGU:60**
- description **SGU:59**
- planning to use **SGU:59**
- running
 - specifying execution parameters **SGU:61**
 - using CLISTs **SGU:61**
 - using EXECs **SGU:61**
 - using JCL **SGU:60**
 - using the TPNS/ISPF interface **SGU:61**
- understanding output
 - Eligible Terminal Report **SGU:63**
 - Ineligible Terminal Report **SGU:65**
 - Summary Report **SGU:62**
- understanding restrictions **SGU:67**
- understanding return codes **SGU:68**
- using the TPNS log data set
 - calculating response times **SGU:66**
 - comparing log data sets **SGU:67**
 - formatting **SGU:66**
 - generating a TPNS script **SGU:66**
 - introduction **SGU:65**

ITPS99TU

- formatting **OP:12**

ITPSDL

- CLIST, example of **SGU:149**
- control commands **SGU:144—148**
- data sets required **SGU:150**
- description **SGU:142**
- execution parameter **SGU:148**
- JCL, example of **SGU:148**
- output, example of **SGU:153**
- REPORT control command **SGU:145**
- SEARCH control command **SGU:146**
- SELECT control command **SGU:147**

ITPSGEN **PI:76**

- * (comment) command **SGU:93**
- CLIST, example of **SGU:98**
- control commands **SGU:82—92**
- control input, example of **SGU:97**
- data set requirements **SGU:81**
- DELAY command **SGU:84**
- description **SGU:80**
- EXEC, example of **SGU:98**
- execution parameters **SGU:81**
- input network definitions **SGU:78**
- input tape, format of **SGU:74**
- JCL, example of **SGU:97**
- LIMIT command **SGU:85**
- LIST command **SGU:86**

ITPSGEN *(continued)*

- LU6.2 simulations, generating scripts for **SGU:100**
- message generation decks **SGU:94**
- networks, updated **SGU:95**
- NODELAY command **SGU:84**
- NOLIMIT command **SGU:85**
- NOLIST command **SGU:86**
- NONTWRK command **SGU:87**
- NORESP command **SGU:89**
- NOSEQOUT command **SGU:90**
- NOSSCP command **SGU:91**
- NTWRK command **SGU:87**
- output, example of **SGU:97**
- output, printed **SGU:95**
- output, sequential **SGU:95**
- REPORT command **SGU:88**
- RESP command **SGU:89**
- return codes **SGU:96**
- running **SGU:80**
- SEQOUT command **SGU:90**
- SSCP command **SGU:91**
- terminal types supported **SGU:80**
- TIME command **SGU:92**
- TPNS/ISPF interface, running from **SGU:99**

ITPSYSIN **DEF:26, MGD:202**

- compared to the preprocessor **PI:89**
- description of **GU:26**
- EXEC, example of **GU:28**
- JCL, example of **GU:27**
- return codes **GU:28**
- TPNS/ISPF Interface, invoking **GU:27**
- TSO CLIST, example of **GU:27**
- using **PI:89**

ITPTCL

- CLIST, example of **SGU:141**
- control commands **SGU:136—140**
- data sets required **SGU:140**
- description **SGU:136**
- execution parameter **SGU:140**
- JCL, example of **SGU:140**
- output, example of **SGU:142**
- REPORT control command **SGU:137**
- TESTCASE control command **SGU:138**
- TRANSACT control command **SGU:139**

ITPVTBRF

- description **SGU:76**
- EXEC, example of **SGU:77**
- JCL, example of **SGU:77**
- return codes **SGU:77**
- TPNS/ISPF interface, running from **SGU:76**

ITPXREF

- CLIST, example of **SGU:149**
- JCL, example of **SGU:148**
- TRANSACT control command **SGU:143**

IUCV (Inter-User Communication Vehicle) **PI:10**

IUCV cross-memory facility **DEF:105**
 IUTI network definition operand, using to control inter-message delay **STL:79**
 IUTI operand
 DEV (TCP/IP) statement **LR:132**
 DEV (terminal) statement **LR:184**
 LU (cross-domain) statement **LR:272**
 LU (type 2.1 node) statement **LR:230**
 LU (VTAMAPPL) statement **LR:116**
 TERM statement **LR:202**
 TP statement **LR:101**
 IUTI operand (A command) **OP:134**
 IUTI operand on network definition statements **MGD:72**
 IWDWAKCT operand (LLS statement) **LR:144, LR:153**

J

JCL
 for CPI-C, VTAM, or TCP/IP simulations **OP:11**
 operating TPNS with **OP:5—12**
 using to run STL Translator **STL:116**
 with communication controllers **OP:5—9**
 without communication controllers **OP:9—12**
 JCL and EXEC statements **UE:37**
 JCL for ITPECHO **PRI:7**
 jump key statement **LR:380**
 JUMP key, simulating **LR:380**
 JUMP PARTITION key, simulating **LR:380**
 JUMP statement **LR:380, MGD:165, STL:248**

K

K subcommand (F command) **OP:144**
 KEY operand
 ATTR statement **LR:297**
 ESC statement **LR:350**
 keyboard actions, simulating
 cursor movement **STL:57**
 device keys **STL:54**
 text entry **STL:53**
 KEYLABEL operand
 DEV (terminal) statement **LR:184**
 LU (cross-domain) statement **LR:272**
 LU (type 2.1 node) statement **LR:230**
 TERM statement **LR:202**
 keys that can be simulated for various devices **STL:385**
 keyword statements, use of **STL:18, STL:32**

L

L (List Resource Addresses) operator
 command **OP:147**
 L data field option **LR:440**
 LABEL operand
 BRANCH statement **LR:298**

LABEL operand (*continued*)
 CALL statement **LR:302**
 LABEL statement **LR:381, MGD:124**
 label, putting in message generation deck **LR:381**
 labels
 defined **STL:19**
 naming conventions for **STL:19**
 syntax for **STL:20**
 LAN
 See local area network
 LAN port (LPORT)
 canceling **OP:140**
 link interface, specifying **OP:24**
 listing address of **OP:147**
 query network for **OP:41**
 starting activity for **OP:160**
 stopping activity for **OP:155**
 LAN resource records **GU:34**
 LAN simulation, defining physical port on **LR:146**
 LANTYPE operand (LPORT statement) **LR:147**
 LAP operand (MCH statement) **LR:166**
 LASTVERB data field option **LR:440**
 LASTVERB function **STL:339**
 LC operand (A command) **OP:134**
 LCGN operand (LCH statement) **LR:159**
 LCH
 See logical channel name (LCH)
 LCH operand
 LINE (single-domain) statement **LR:193**
 LINE (type 2.1 node) statement **LR:219**
 LINK statement **LR:261**
 VC statement **LR:171**
 LCH statement **DEF:200, LR:158—162**
 LCHID data field option **LR:440**
 LCHID function **STL:340**
 LCLADDR operand (LPORT statement) **LR:147**
 LCLEAR statement **LR:382, MGD:160, STL:249**
 LCLPARMS operand (PADPARMS statement) **LR:169**
 LCLSAP operand (LLS statement) **LR:144, LR:154**
 LCN operand (LCH statement) **LR:159**
 LDS control block **UE:63**
 LEAVE statement **STL:48, STL:250**
 LEFT function **STL:341**
 LEFT function (DATASAVE statement) **LR:332**
 LEFT function on the DATASAVE statement **MGD:46**
 LEFT operand (CURSOR statement) **LR:327**
 LENG operand
 CALC statement **LR:301**
 DATASAVE statement **LR:333**
 IF (message generation) statement **LR:370**
 IF (network configuration) statement **LR:52**
 LOG statement **LR:384**
 on the DATASAVE statement **MGD:45**
 on the IF statement **MGD:87**
 TEXT statement **LR:424**

LENGTH command **GU**:204
 LENGTH function **PRI**:73, **STL**:342
 LEVEL operand
 CNTLR statement **LR**:253
 SSCP statement **LR**:289
 light pen **SGU**:80
 LIGHTPEN statement **STL**:56, **STL**:251
 LIMIT command **SGU**:85
 LIN control block **UE**:65
 line
 activating scanner internal trace for **OP**:152
 canceling **OP**:140
 errors, caused by **OP**:119
 listing address of **OP**:147
 query network for **OP**:33
 starting activity for **OP**:160
 stopping activity for **OP**:155
 LINE (Peripheral Link) statement **LR**:258—259
 LINE (Single Domain) statement **LR**:192—194
 LINE (type 2.1 node) statement **LR**:217—221
 LINE command
 Log Compare Utility **GU**:134
 Loglist Utility **GU**:76
 Response Time Utility **GU**:205
 line control block **UE**:65
 line counters **LR**:457
 line disciplines
 controller simulation **PI**:141
 supported by TPNS **PI**:135
 unsupported by TPNS **PI**:140
 line error exit
 discussion **UE**:19
 parameter list **UE**:20
 sample routine **UE**:20
 line mode operation **OP**:85
 LINE statement **DEF**:22, **DEF**:23, **DEF**:127, **DEF**:171
 terminal simulation **PI**:62
 line trace
 activating **OP**:139
 description of **OP**:49
 report **OP**:69
 line trace report **PI**:109
 line types simulated by TPNS **DEF**:33
 LINEID operand
 FPORT (frame relay) statement **LR**:150
 LINE (single-domain) statement **LR**:193
 LINE (type 2.1 node) statement **LR**:219
 LINK statement **LR**:261
 LPORT statement **LR**:146
 MCH statement **LR**:163
 LINES operand (M command) **OP**:149
 LINID data field option **LR**:440
 LINID function **STL**:343
 link
 activating scanner internal trace for **OP**:152
 canceling **OP**:140
 link (*continued*)
 LAN interface, specifying **OP**:24
 listing address of **OP**:147
 multichannel, activating/deactivating **OP**:166
 NCP subarea, specifying **OP**:23
 query network for **OP**:33
 starting activity for **OP**:160
 stopping activity for **OP**:155
 tracing characters on **OP**:139
 X.25 interface, specifying **OP**:23
 LINK statement **DEF**:126, **DEF**:172, **LR**:260—263
 link-attach configuration
 See also physical configuration, types
 communication controller for **PI**:12, **PI**:51
 defined **PI**:50
 example **PI**:42
 using **PI**:42, **PI**:52
 link-attached cross-domain model **SMP**:328
 LINNAME data field option **LR**:440
 LINNAME function **STL**:344
 LIST command **SGU**:86
 List Resource Addresses (L) operator
 command **OP**:147
 LISTX execution parameter **GU**:168
 LITIMER operand (FPORT statement) **LR**:151
 LL data field option **LR**:440
 LLC operand (LCH statement) **LR**:159
 LLS
 See logical link station name (LLS)
 LLS operand
 LINE (single-domain) statement **LR**:194
 LINE (type 2.1 node) statement **LR**:220
 LINK statement **LR**:261
 LLS Resource End of Run Report **OP**:68
 LLS Resource Interval Report **OP**:66
 LLS statement **DEF**:160, **DEF**:170, **LR**:142—145,
 LR:152—154
 LLSID data field option **LR**:440
 LLSID function **STL**:345
 LMI operand (FPORT statement) **LR**:151
 LNEREXIT operand
 discussion **UE**:19
 parameter list **UE**:20
 sample routine **UE**:20
 LNEREXIT operand (NTWRK statement) **LR**:74
 load module
 in MVS system authorization **PI**:14
 load on TPNS system, affecting intermessage
 delay **STL**:81
 LOAD operand (CHANL statement) **LR**:251
 LOC operand
 CALC statement **LR**:300
 DATASAVE statement **LR**:333
 IF (message generation) statement **LR**:365
 IF (network configuration) statement **LR**:48
 on the DATASAVE statement **MGD**:44

LOC operand (*continued*)

- on the IF statement **MGD:90**
- on the IF statement, coding **MGD:90, MGD:91**

local area network

- activating resources **DEF:165**
- deactivating resources **DEF:165**
- defining resources
 - lines and terminals on token-ring **DEF:161—163**
 - token-ring configuration **DEF:159—160**
- LLS Resource Interval Report **DEF:166**
- Loglist Utility **DEF:166**
- message logging **GU:249**
- network definition example **DEF:163—165**
- network definition statements
 - DEV **DEF:162**
 - LINE **DEF:161**
 - LLS **DEF:160**
 - LPORT **DEF:159**
 - PU **DEF:162**
 - TERM **DEF:161**
- Response Time Utility **DEF:166**
- simulating Ethernet resources **DEF:158**
- simulating IBM Token-Ring
 - resources **DEF:157—166**
- statements for simulating **PI:65**
- terminal types simulated **DEF:161**
- time stamping messages **GU:245**
- TPNS simulation support **DEF:157—158**
- TPNS support for **PI:139**

local attachment cables connection method

- See connecting TPNS to system under test

local clear statement **LR:382**

local clear, simulating **LR:382**

LOCLENG operand

- IF (message generation) statement **LR:370**
- IF (network configuration) statement **LR:53**

LOCTEXT operand **MGD:91**

- IF (message generation) statement **LR:367**
- IF (network configuration) statement **LR:49**

log byte

- with CPI-C statements **STL:179—212**
- with INITSELF **STL:69, STL:243**
- with TERMSELF **STL:69, STL:291**
- with TRANSMIT **STL:75, STL:292**

LOG command **GU:77**

Log Compare Utility **MGD:209**

- * (Comment) command **GU:160**
- Active Command List **PI:113**
- ATTRIBUTE command **GU:147**
- CHARATTR command **GU:148**
- CHECKONLY command **GU:149**
- command syntax **GU:129**
- Compare List **PI:113**
- Complete Records List **PI:113**
- CURSOR command **GU:151**
- description of **GU:99**

Log Compare Utility (*continued*)

- DEV command **GU:131**
- Differences Report **PI:114**
- END command **GU:152**
- ERRCOUNT command **GU:132**
- EXCLUDE command **GU:133**
- EXEC, example of **GU:115**
- execution parameters **GU:47, GU:113**
- HEADER command **GU:153**
- JCL, example of **GU:114**
- LINE command **GU:134**
- LU command **GU:135**
- MASK command **GU:154**
- MSGTXT command **GU:137**
- NODE command **GU:138, GU:139**
- output **PI:113**
- output, examples of **GU:116—128**
- output, list of **GU:108**
- P command **GU:156**
- REPORT command **GU:157**
- return codes **GU:128**
- RUN command **GU:158**
- running **GU:112, PI:112**
- SELECT command **GU:140**
- START command **GU:141**
- Summary Report **PI:114**
- synchronizing **GU:106—113**
- synchronizing log data sets **PI:112**
- SYNCPPOINT command **GU:142**
- TCPIP command **GU:143**
- TERM command **GU:144**
- TPNS/ISPF Interface, running under **GU:113**
- TSO CLIST, example of **GU:115**
- UPPERCASE command **GU:159**
- VTAMAPPL command **GU:145**

LOG control block **UE:66**

log data set **PRI:5**

- comparing 3270 display records **MGD:209**
- comparing display records from **PI:111**
- formatting **MGD:206, PI:111**
- Loglist Utility **GU:31**
- record types **GU:46**
- records **GU:31**
- Response Time Utility **GU:163**
- role in tracing STL **PRI:61**
- types of records in **GU:241**
- used for message logging **PI:110**
- using **MGD:206**

log data set header records **UE:42**

log data set records

- console records **UE:42**
- CPI-C trace records **UE:42**
- header record **UE:42**
- informational records **UE:42**
- log display records **UE:42**
- log records **UE:42**

- log data set records (*continued*)
 - marker records **UE:42**
 - message data records **UE:43**
 - message trace records **UE:43**
 - miscellaneous data recorder (MDR) records **UE:43**
 - STL trace records **UE:43**
 - trace data record **UE:43**
 - verify data record **UE:43**
- log data set, analyzing **UE:42**
- log data set, defining for direct access storage device (DASD) **OP:76**
- log data set, writing data to **LR:383**
- log display attribute table header **GU:45**
- log display record formats
 - 3270/3643/8775 records **UE:55**
 - 5250 records **UE:56**
 - discussion **UE:55**
- log display record header format **UE:63**
- log display records **UE:42**
- LOG operand
 - CMND statement **LR:311**
 - CMxxxx statement **LR:319**
 - IF (message generation) statement **LR:371**
 - IF (network configuration) statement **LR:53**
 - TEXT statement **LR:425**
- log record header **GU:42**
- log record header format **UE:66**
- log records **GU:35, UE:42**
- LOG statement **LR:383—384, MGD:122, STL:100, STL:252**
- LOGDSPLY operand **PRI:61**
 - DEV (TCP/IP) statement **LR:132**
 - DEV (terminal) statement **LR:184**
 - LU (cross-domain) statement **LR:272**
 - LU (type 2.1 node) statement **LR:230**
 - LU (VTAMAPPL) statement **LR:116**
 - TERM statement **LR:202**
- LOGDSPLY operand (A command) **OP:134**
- logging messages
 - See* message logging
 - See* network options
- logging off an application
 - BSC 3270s **STL:70**
 - SNA LUs **STL:69**
 - Telnet 3270 **STL:70**
- logging on an application
 - BSC 3270s **STL:70**
 - determining logon code **STL:66**
 - placing logon code **STL:66**
 - SNA LUs **STL:66**
 - Telnet 3270 **STL:70**
- logging on and off ITPECHO **PRI:8**
- logging on to applications asynchronously **PRI:87**
- logging on to applications synchronously **PRI:16**
- logging on to Display Monitor **PRI:25, PRI:31**

- logging test data
 - LOG statement **STL:100**
 - methods of **STL:100**
- logic errors, correcting
 - See* errors, correcting, logic
- logic test locations **LR:449**
- logic test, conditions not evaluated **LR:461**
- logic tests **DEF:268—271**
 - See also* message-level logic tests, network-level logic tests
 - activating **MGD:100**
 - deactivating **MGD:101**
 - defining **MGD:83**
 - evaluating
 - with WHEN=IMMED **MGD:103**
 - with WHEN=IN **MGD:103**
 - with WHEN=OUT **MGD:103**
 - examples **MGD:105—110**
 - for display terminal using WHEN=IMMED **MGD:108**
 - for IBM 3270 terminal **MGD:160**
 - IF statements as **MGD:83**
 - list of operands for coding **MGD:86**
 - preventing the deactivation of **MGD:102**
 - processing **MGD:100—104**
 - understanding **MGD:26, MGD:83**
 - using to create self-checking scripts **MGD:111**
- logic tests, deactivating **LR:338**
- logical channel **DEF:200**
- logical channel name (LCH)
 - canceling **OP:140**
 - query network for **OP:45**
 - starting activity for **OP:160**
 - stopping activity for **OP:155**
- logical configuration
 - CPI-C transaction program simulation **PI:46**
 - defined **PI:38**
 - description of **DEF:11**
 - example **PI:40**
 - related to physical configuration **PI:38**
 - subarea simulation **PI:52**
 - TCP/IP simulation **PI:48**
 - terminal simulation **PI:49**
 - types
 - subareas **DEF:12, DEF:17—18**
 - TCP/IP **DEF:12, DEF:14—15**
 - terminals, Type 2.1 nodes, link-attached
 - subareas **DEF:12, DEF:15—15**
 - VTAM applications and LUs **DEF:12, DEF:13**
 - types of **PI:43**
 - VTAM application simulation **PI:44**
- logical link station definition statement **LR:142—145, LR:152—154**
- logical link station name (LLS)
 - canceling **OP:140**
 - query network for **OP:42**
 - starting activity for **OP:160**

logical link station name (LLS) (*continued*)

stopping activity for **OP**:155

logical unit

defined **PI**:37

defining in NCP node subarea **LR**:264

defining in Type 2.1 node **LR**:222

types simulated by TPNS **PI**:45, **PI**:147

logical unit definition statement **LR**:264—279

logical units using VTAM Application Program Interface (VTAM API)

See VTAM applications

loglist **PRI**:5, **PRI**:64—71

loglist control commands **PRI**:65, **PRI**:68

Loglist Utility **MGD**:206

* (comment) command **GU**:96

APPCLU command **GU**:60

CDLOG command **GU**:61

CHANL command **GU**:62

CNSL command **GU**:63

command syntax **GU**:57

control commands **PRI**:64

CTRC command **GU**:64

DATA command **GU**:65

description **PRI**:5

description of **GU**:31

DEV command **GU**:66

DSPLY command **GU**:67

END command **GU**:68

example output

display records **PRI**:69

ONIN and ONOUT statements **PRI**:85

STL functions **PRI**:78

STL trace records **PRI**:66

user tables **PRI**:81

EXDEV command **GU**:69

EXEC, example of **GU**:50

execution parameters **GU**:47

EXIT command **GU**:70

EXTERM command **GU**:71

EXTP command **GU**:72

FMTSNA command **GU**:73

formatting display records **PRI**:68

HEADER command **GU**:74

INFO command **GU**:75

interpreting output

display records **PRI**:68

ONIN and ONOUT statements **PRI**:86

STL functions **PRI**:77

STL trace records **PRI**:68

user tables **PRI**:82

JCL, example of **GU**:48

LINE command **GU**:76

LOG command **GU**:77

log record types, format of **GU**:31

MDR command **GU**:78

MSGTXT command **GU**:79

Loglist Utility (*continued*)

MTRC command **GU**:80

NOCDLOG command **GU**:61

NOCNSL command **GU**:63

NOCTRC command **GU**:64

NODATA command **GU**:65

NODSPLY command **GU**:67

NOFMT command **GU**:73

NOHDR command **GU**:81

NOINFO command **GU**:75

NOLOG command **GU**:77

NOMDR command **GU**:78

NOMTRC command **GU**:80

NORR command **GU**:82

NOSTRC command **GU**:87

NOTRACE command **GU**:92

NOVERIFY command **GU**:94

NOVRPAC command **GU**:83

NTWRK command **GU**:84

output **PI**:111

output, examples of **GU**:51—56

overview **PRI**:64

P command **GU**:85

return code **GU**:56

RUN command **GU**:86

running **GU**:45, **PI**:111, **PRI**:65

sample output **STL**:142

sample output, explanation of **STL**:146

STRC command **GU**:87

TCPIP command **GU**:88

TERM command **GU**:89

TIME command **GU**:90

TP command **GU**:91

TPNS/ISPF Interface, running under **GU**:46

TRACE command **GU**:92

tracing simulation activity **STL**:140

TSO CLIST, example of **GU**:50

UPCASE command **GU**:93

using with STL programs **STL**:140

VERIFY command **GU**:94

VERIFY command with **GU**:36, **GU**:41

VTAMAPPL command **GU**:95

Loglist Utility exits, coding

analyzing the log data set **UE**:42

discussion **UE**:41

log data set records **UE**:42

parameter list **UE**:41

reading the log data set **UE**:43

register linkage **UE**:41

sample exit routine **UE**:43

Loglist Utility input **PRI**:65, **PRI**:68

Loglist Utility user exit routines **UE**:41

loglists

CMS with 3270 BSC **SMP**:261

INSTALL1 network **SMP**:233

INSTALL2 network **SMP**:245

- loglists (*continued*)
 - real host ownership **SMP**:275
 - TPNS as an application **SMP**:256
 - type 2.1 node with independent LUs **SMP**:293
- LOGMODE operand (MODEENT statement) **LR**:64
- logoff function, ITPECHO **GU**:232
- logoff function, non-3270 device **GU**:232
- logoff procedure **PRI**:29
- logon mode table
 - assigning a statement to **LR**:63
 - defining beginning of **LR**:67
 - indicating end of **LR**:62
- logon procedure **PRI**:29, **PRI**:87
- LOGON, starting TPNS Display Monitor Facility
 - through **OP**:107
- loops **OP**:117
- LOW operand (RN statement) **LR**:83
- LPORT
 - See LAN port (LPORT)
- LPORT statement **DEF**:159, **LR**:146
- LPORTID data field option **LR**:440
- LPORTID function **STL**:346
- LSEQ data field option **LR**:441
- LSEQ operand (A command) **OP**:134
- LTACHAR operand (TERM statement) **LR**:203
- LU
 - See logical unit
- LU (cross-domain) statement **LR**:264—279
- LU (type 2.1 node) statement **LR**:222—237
- LU (VTAMAPPL) statement 105, **LR**:109—122
- LU 6.2 simulations, generating scripts for **SGU**:100
- LU command **GU**:135
- LU command, ITPLSGEN **SGU**:52
- LU statement **PRI**:22
 - NCP subareas, simulation **DEF**:127
 - SSCP simulation **DEF**:123
 - Type 2.1 node simulation **DEF**:185
 - VTAM simulation **DEF**:69
 - VTAMAPPL configuration **PI**:59
- LU-LU session
 - activating (subarea simulation) **DEF**:147—147
 - activating (terminal simulation) **DEF**:143—144
 - deactivating (subarea simulation) **DEF**:148
 - deactivating (terminal simulation) **DEF**:145
 - INIT-OTHER command **DEF**:148—148
 - request/response units (subarea simulation) **DEF**:153—155
 - request/response units (terminal simulation) **DEF**:150
- LU2 terminals, simulating errors in **MGD**:161
- LUCAP operand
 - DEV statement **LR**:185
 - LU (cross-domain) statement **LR**:273
- LUID data field option **LR**:441
- LUID function **PRI**:73, **STL**:347

- LUNAME operand
 - CNOS operand (APPCLU statement) **LR**:96
 - SIDEENT statement **LR**:87
 - SIDEINFO operand (APPCLU statement) **LR**:97
- LUNAMES operand
 - CDRM statement **LR**:249
 - PU (type 2.1 node) statement **LR**:239
- LUS operand (CMND statement) **LR**:312
- LUs using VTAM Application Program Interface
 - See VTAM applications
- LUSTAT **MGD**:179
- LUTYPE operand **PRI**:22
 - LU (cross-domain) statement **LR**:273
 - LU (type 2.1 node) statement **LR**:231
 - LU (VTAMAPPL) statement **LR**:117

M

- M (Display Monitor Facility) operator command **OP**:148
- M operand (A command) **OP**:135
- MAG10 data field option **LR**:441
- MAG10, MAG10S, MAG63, MAG63S
 - functions **STL**:59, **STL**:348—348
- MAG10S data field option **LR**:441
- MAG63 data field option **LR**:441
- MAG63S data field option **LR**:441
- magnetic input, simulating
 - See devices, simulating, magnetic input
- magnetic stripe reader data, defining **LR**:419
- magnetic stripe reader input statement **LR**:419
- magnetic stripe reader, simulating
 - See devices, simulating, magnetic stripe reader
- marker records **UE**:42
- MASK command **GU**:154
- MAXBFRU operand
 - CHANL statement **LR**:251
 - LINE (type 2.1 node) statement **LR**:220
- MAXBLU operand
 - PU (cross-domain) statement **LR**:284
 - PU (type 2.1 node) statement **LR**:240
 - TERM statement **LR**:203
- MAXCALL operand
 - DEV (TCP/IP) statement **LR**:133
 - DEV (terminal) statement **LR**:185
 - LU (cross-domain) statement **LR**:273
 - LU (type 2.1 node) statement **LR**:231
 - LU (VTAMAPPL) statement **LR**:117
 - TERM statement **LR**:204
 - TP statement **LR**:101
- MAXNOPTN operand
 - DEV (TCP/IP) statement **LR**:133
 - DEV (terminal) statement **LR**:185
 - LU (cross-domain) statement **LR**:273
 - LU (type 2.1 node) statement **LR**:231
 - LU (VTAMAPPL) statement **LR**:117

MAXOUT operand
 LLS statement **LR:144, LR:154**
 PU (cross-domain) statement **LR:285**
 PU (type 2.1 node) statement **LR:240**
 TERM statement **LR:204**
 MAXPKTL operand (LCH statement) **LR:160**
 MAXPTNSZ operand
 DEV (TCP/IP) statement **LR:133**
 DEV (terminal) statement **LR:185**
 LU (cross-domain) statement **LR:274**
 LU (type 2.1 node) statement **LR:231**
 LU (VTAMAPPL) statement **LR:117**
 MAXSAP operand (LPORT statement) **LR:147**
 MAXSESS operand
 LU (cross-domain) statement **LR:274**
 LU (type 2.1 node) statement **LR:231**
 LU (VTAMAPPL) statement **LR:117**
 MAXSTN operand (LPORT statement) **LR:148**
 MAXSUBA operand (NTWRK statement) **LR:74**
 MCH
 See multichannel name/address (MCH)
 MCH statement **DEF:199, LR:163—168**
 MCHID data field option **LR:441**
 MCHID function **STL:349**
 MCHTYPE operand (MCH statement) **LR:166**
 MDR command **GU:78**
 MDR records **UE:43**
 menus
 ATPAID **SGU:131**
 ATPDEF **SGU:120**
 ATPDFO **SGU:132**
 ATPPRIM **SGU:119**
 ATPSDSEL **SGU:129**
 ATPSEL **SGU:122**
 ATPTC **SGU:125**
 ATPTCSEL **SGU:124**
 IDC Add STL Statements Panel **SGU:18**
 IDC Change Escape Key Panel **SGU:20**
 IDC Change Log Data Sets Panel **SGU:19**
 IDC Escape Actions Panel **SGU:14**
 IDC Generate Message Generation Deck
 Panel **SGU:26**
 IDC Generate STL Program Panel **SGU:22**
 IDC Main Panel **SGU:11**
 IDC Start Session Panel **SGU:12**
 ISPF/PDF Primary Option **SGU:113, SGU:118**
 TPNS/ATP Attention Identifier Definition **SGU:131**
 TPNS/ATP Data Field Options **SGU:132**
 TPNS/ATP Datasets and Parameter
 Definition **SGU:120**
 TPNS/ATP Editor Selection **SGU:122**
 TPNS/ATP Primary Option **SGU:119**
 TPNS/ATP Screen Selection **SGU:129**
 TPNS/ATP Testcase Definition **SGU:125**
 TPNS/ATP Testcase Selection **SGU:124**
 message data records **UE:43**
 message data, generating **LR:422**
 Message Format Service source data sets
 See MFS source data sets
 message generation
 conditions for continuing **STL:78**
 entering **STL:77**
 exiting **STL:77**
 for a display terminal **MGD:146**
 for a TWX terminal **MGD:173**
 interrupting **STL:77**
 interrupting with delimiters **MGD:56**
 monitoring and automating with control
 statements **MGD:121**
 planning considerations **MGD:7—12**
 with IBM 5250 display system **MGD:179**
 message generation deck **DEF:1, PRI:4, PRI:17**
 3101 simulation statements, list of **LR:292**
 3270 simulation statements, list of **LR:292**
 3290 simulation statements, list of **LR:293**
 3641 simulation statements, list of **LR:294**
 3643 simulation statements, list of **LR:294**
 5250 simulation statements, list of **LR:294**
 8775 simulation statements, list of **LR:295**
 defining start of **LR:386**
 ending **LR:344**
 establishing label in **LR:381**
 general definition statements, list of **LR:291**
 generating **LR:422**
 logic testing **LR:363**
 process, stopping **LR:418**
 returning from subroutine **LR:400**
 SNA simulation statements, list of **LR:292, LR:295**
 specifying **LR:61**
 specifying running sequence **LR:81**
 stopping generation of **LR:397**
 message generation deck logic test
 statement **LR:363—378**
 message generation deck name, defining
 (ITPLSGEN) **SGU:51**
 message generation deck start
 statement **LR:386—388**
 message generation deck, generating
 (ITPLSGEN) **SGU:52**
 message generation decks
 checklist for developing **MGD:12**
 coding with delimiters **MGD:67**
 coding with message generation
 statements **MGD:21**
 creating with ITPBTS **MGD:5**
 creating with message generation
 statements **MGD:4**
 creating with the Structured Translator Language
 (STL) **MGD:4**
 deciding how to create **PI:73**
 defined **PI:69**

message generation decks (*continued*)

- defining selection of **MGD:190**
- described **MGD:5**
- designing **MGD:9**
- determining message content **MGD:9**
- developing TPNS scripts **MGD:11**
- documenting **MGD:10**
- examples for 8100 processor **MGD:185**
- examples for VTAM application **MGD:197**
- function testing **MGD:11**
- how to create **MGD:4**
- integrating with network definitions **MGD:10, MGD:189**
- introduced **MGD:3**
- methods for creating **PI:73**
- not referenced by terminals, including **MGD:192**
- related to network definition statements **PI:72**
- related to network definitions **MGD:3**
- specifying for error recovery **MGD:193**
- specifying the first **MGD:192**
- steps for creating **PI:69—72**
- structuring **MGD:9**
- syntax testing **MGD:10**
- testing **MGD:10**
- use of **MGD:17—27**
- using ITPBTS to create **PI:82**
- using ITPLU2RF to create **PI:82**
- using message generation statements to create **PI:73**
- using STL to create **PI:74**
- using the Script Generator Utility to create **PI:76**
- using TPNS/ATP to create **PI:80**
- message generation delays
 - See network options, delays, message generation
- message generation exit
 - discussion **UE:22**
 - parameter list **UE:23**
 - return codes **UE:23**
 - sample routine **UE:24**
 - sample statements **UE:27**
- message generation sequence statement **LR:81—81**
- message generation statements
 - ATTR MGD:174**
 - BRANCH MGD:123**
 - CANCEL MGD:135**
 - CHARSET MGD:163, MGD:164**
 - classifying **MGD:26**
 - CLEARPTN MGD:165**
 - CMND MGD:150, MGD:154**
 - CMxxxx CPI-C simulation statements MGD:117, MGD:119**
 - COLOR MGD:163**
 - CURSOR MGD:159**
 - DEACT MGD:136**
 - DELETE MGD:159**
 - ENDTXT MGD:21**

message generation statements (*continued*)

- ENTER MGD:144, MGD:145**
- ERROR MGD:120, MGD:161**
- EVENT MGD:124, MGD:134**
- HELP MGD:179**
- HIGHLITE MGD:164**
- IF MGD:24**
- INSERT MGD:159**
- JUMP MGD:165**
- LABEL MGD:124**
- LCLEAR MGD:160**
- LOG MGD:122**
- MONITOR MGD:122**
- MSGTXT MGD:21**
- MSGUTBL MGD:34**
- ON MGD:136**
- OPCMND MGD:121**
- RESET MGD:120, MGD:158**
- RESP MGD:120, MGD:152**
- RETURN MGD:124**
- RH MGD:121, MGD:149**
- SCROLL MGD:165**
- SEND MGD:144**
- SET MGD:125**
- SETSW MGD:125**
- SETUTI MGD:73**
- SHOW MGD:122**
- STOP MGD:64**
- STRIPE MGD:161**
- syntax conventions for **MGD:17**
- TAB MGD:145**
- TEXT MGD:29**
- TH MGD:120, MGD:149**
- using **MGD:4, PI:73**
- WAIT MGD:23, MGD:56**
- when to use **PI:74**
- WTO MGD:122**
- message generation statements in STL
 - programs **STL:37**
- message input exit
 - discussion **UE:3**
 - parameter list **UE:4**
 - sample routine **UE:5**
- message logging **MGD:206, PRI:4, PRI:61**
 - See also network options, logging messages
 - blocked PIUs **GU:246**
 - data messages **GU:245**
 - DEBUG option **GU:245**
 - description of **GU:241**
 - encrypted data **GU:246**
 - in multiple-domain environment **GU:247**
 - local area network **GU:249**
 - partial segments **GU:245**
 - passthrough virtual circuits **GU:250**
 - requirements **PI:10**
 - SNA terminals **GU:249**

- message logging (*continued*)
 - time stamping **GU:242**
 - types of records **GU:241**
 - UHDLC terminals **GU:251**
 - USER terminals **GU:250**
 - using a separate virtual machine **PI:98**
 - X.25 packets **GU:249**
- message output exit
 - discussion **UE:7**
 - parameter list **UE:7**
 - sample routine **UE:8**
- message scripting
 - 5250 example **SMP:50**
 - ALC network **SMP:96**
 - CICS examples **SMP:17**
 - CMS examples **SMP:19**
 - CPI-C transaction program (TP) example
 - with multiple-instance TPs **SMP:127**
 - with single-instance TPs **SMP:119**
 - File Transfer Protocol (FTP) example **SMP:111**
 - frame relay example **SMP:106**
 - IBM Token-Ring Network example **SMP:77**
 - IMS examples **SMP:26**
 - LU6 example **SMP:83**
 - modifying the TPNS BIND response **SMP:93**
 - real host ownership of a simulated NCP subarea **SMP:80**
 - RJE examples **SMP:30**
 - Simple TCP Client example **SMP:114**
 - Telnet 3270 example **SMP:109**
 - TPNS as an application **SMP:85**
 - TSO examples **SMP:45**
 - Type 2.1 node examples (PU21) **SMP:50**
- message text definition
 - statement **LR:61**
- message text, entering **MGD:19**
- message trace records **GU:35, UE:43**
 - using to test TPNS scripts **PI:83**
- message tracing **DEF:274**
- message-level logic tests
 - terminology used to describe **MGD:85**
 - understanding **MGD:84**
- messages
 - CLISTS triggered by **OP:100**
 - continuing **MGD:20**
 - description of logging **OP:75**
 - generating dynamically
 - with \$UTBL\$ **MGD:35**
 - with random numbers **MGD:32**
 - with user tables **MGD:33**
 - generating for display terminals **MGD:143**
 - generating for SNA terminals **MGD:149**
 - generating manually
 - combining types of data **MGD:30**
 - entering data **MGD:29**
 - generating with the TEXT statement **MGD:29—51**

- messages (*continued*)
 - in MVS environment **OP:75**
 - in VM environment **OP:77**
 - inhibiting logging, console **OP:80**
 - logging **OP:75—81**
 - logging, restarting **OP:142**
 - no traffic, causes of **OP:117**
 - RESLIST **OP:96**
 - restarting **OP:81**
 - routing under the operator interface between the NetView program and TPNS **OP:84**
 - sending with delimiters **MGD:65**
 - separate virtual machine for logging **OP:77—80**
 - with sequence counters **MGD:39**
 - with index counters **MGD:40**
 - with save and user areas **MGD:42**
- messages, receiving
 - during intermessage delay **STL:81**
 - updating terminal's display buffer **STL:81**
- messages, transmitting
 - statements used to transmit messages, listed **STL:77**
 - using AID keys **STL:75**
 - using TRANSMIT statement **STL:75**
- MFS source data sets
 - error messages **SGU:114**
 - example of **SGU:113**
- MFS source statements
 - INQUIRY screen **SMP:147**
 - LOGOFF screen **SMP:148**
 - LOGON screen **SMP:147**
 - PRIMARY screen **SMP:147**
- MFS, TPNS/ATP support of **SGU:173—177**
- migrating from TPNS Version 3 Release 4
 - installation considerations **PI:130**
 - new features
 - user considerations **PI:125**
- migrating statements **PRI:34**
- migration considerations **STL:393**
- MINLEN operand (FILE statement) **LR:45**
- miscellaneous data recorder records **UE:43**
- MLen operand
 - APPCLU statement **LR:96**
 - FPORT statement **LR:151**
 - LINE (peripheral link) statement **LR:258**
 - LINE (single-domain) statement **LR:194**
 - LINK statement **LR:262**
 - LPORT statement **LR:148**
 - MCH statement **LR:166**
 - PU21 statement **LR:244**
 - TCPIP statement **LR:139**
 - VTAMAPPL statement **LR:124**
- MLOG operand **PRI:61**
 - APPCLU statement **LR:96**
 - FPORT statement **LR:151**
 - LINE (peripheral link) statement **LR:259**

MLOG operand (*continued*)
 LINE (single-domain) statement **LR:194**
 LINK statement **LR:262**
 LPORT statement **LR:148**
 MCH statement **LR:167**
 PU21 statement **LR:244**
 TCPIP statement **LR:139**
 VTAMAPPL statement **LR:124**

MODE operand
 CMND statement **LR:313**
 LINE (type 2.1 node) statement **LR:220**
 LINK statement **LR:262**

mode table begin statement **LR:67**
 mode table end statement **LR:62**
 mode table entry, statement **LR:63—66**
 MODEEND statement **DEF:122, DEF:184, LR:62**
 MODEENT statement **DEF:122, DEF:184, LR:63—66**
 Model 20 Multileaving Workstation **DEF:248, MGD:181**

model networks
 CPI-C **SMP:396**
 direct-attached cross-domain **SMP:315**
 Ethernet **SMP:323**
 frame relay **SMP:389**
 FTP **SMP:383**
 full-duplex and half-duplex **SMP:356**
 IBM Token-Ring **SMP:363**
 link-attached cross-domain **SMP:328**
 real host ownership of NCP subarea **SMP:348**
 Simple TCP Client **SMP:386**
 Telnet 3270 **SMP:379**
 Type 2.1 node **SMP:336**
 VTAM application **SMP:371**
 X.25 **SMP:374**

MODEL operand
 DEV (terminal) statement **LR:185**
 TERM statement **LR:204**

modem elimination connection method
 See connecting TPNS to system under test

MODENAME operand
 CNOS operand (APPCLU statement) **LR:96**
 SIDEENT statement **LR:87**
 SIDEINFO operand (APPCLU statement) **LR:97**

MODETAB operand
 LU (cross-domain) statement **LR:274**
 LU (type 2.1 node) statement **LR:232**

MODETAB statement **DEF:122, DEF:184, LR:67**

modifications
 to ITPECHO job stream **PRI:7**

MODIFY
 description of **OP:21**
 routing messages **OP:59**

modifying NCP
 See NCP

modifying the BIND response **SMP:93**

MODULE operand
 CHANL statement **LR:250**

MODULE operand (*continued*)
 CNTLR statement **LR:254**
 EXIT statement **LR:353**

MODULO operand
 LINE (single-domain) statement **LR:194**
 LINE (type 2.1 node) statement **LR:220**
 LINK statement **LR:262**
 MCH statement **LR:167**

MONCMND operand (A command) **OP:135**

MONITOR statement **LR:385, MGD:122, STL:99, STL:253**

monitoring
 availability **PI:144**
 hardware **PI:143**
 response times **PI:145**

monitoring the test
 methods of **STL:99**
 MONITOR statement **STL:99**
 SAY statement **STL:99**
 SHOW statement **STL:99**
 using operator interface between TPNS and the NetView program **STL:99**
 using TPNS Display Monitor Facility **STL:99**

MONLINK operand (LINK statement) **LR:263**

MONTH data field option **LR:441**

MONTH function **STL:350**

MORE operand (TEXT statement) **LR:425**

MSG operand (A command) **OP:135**

MSGDD partitioned data set members
 defined **STL:112**

MSGDISK statement **LR:68**

MSGTRACE operand
 DEV (TCP/IP) statement **LR:133**
 DEV (terminal) statement **LR:186**
 LU (cross-domain) statement **LR:274**
 LU (type 2.1 node) statement **LR:232**
 LU (VTAMAPPL) statement **LR:118**
 TERM statement **LR:205**
 TP statement **LR:102**

MSGTRACE operand (A command) **OP:135**

MSGTXT command
 Log Compare Utility **GU:137**
 Loglist Utility **GU:79**
 Response Time Utility **GU:206**

MSGTXT command, ITPLSGEN **SGU:51**

MSGTXT decks
 capturing terminal data, methods of **SGU:72**
 problems and solutions **SGU:99**

MSGTXT statement **LR:386—388, PRI:15, STL:254**

coding **MGD:21**

operands
 CONCHAR **MGD:22**
 COUNT **MGD:22**
 PAD **MGD:22**
 TXTDLM **MGD:22**
 using **MGD:117**

MSGTXT statement, name
 defined **STL:20**
 naming conventions for **STL:20**
 MSGTXTID data field option **LR:441**
 MSGTXTID function **STL:351**
 MSGUTBL statement **DEF:275, LR:389, MGD:34, STL:30, STL:255**
 MSGUTBL statement, name
 defined **STL:20**
 naming conventions for **STL:20**
 MTRC command **GU:80**
 multichannel link (MCH) **DEF:199**
 multichannel name/address (MCH)
 activating scanner internal trace for **OP:152**
 canceling **OP:140**
 query network for **OP:44**
 starting activity for **OP:160**
 stopping activity for **OP:155**
 tracing characters on **OP:139**
 Multileaving Model 20 Workstation
 TPNS support for **PI:137**
 multiple device simulation exercise **PRI:33**
 multiple devices, using events to synchronize **MGD:140**
 multiple lines **DEF:182**
 multiple partitions **MGD:165**
 multiple PATH statements **PRI:34**
 multiple procedures exercise **PRI:29**
 multiple session selection with the SELECT
 Command **OP:87**
 multiple TYPE statements
 concatenating **STL:54**
 using **STL:53**
 multiple-domain network
 defined **PI:37**
 multiple-instance CPI-C TPs **SMP:125**
 MVS
 allocating SYSPRINT **OP:12**
 authorizing TPNS **PI:14**
 controlling messages in **OP:75**
 entering operator commands under **OP:24, OP:25**
 generating TPNS control program on **DEF:28—32**
 ITPS99TU format **OP:12**
 JCL for **OP:5—12**
 operating TPNS under **OP:5—12**
 return codes **OP:19**
 running TPNS on **PI:97**
 TPNS requirements for **PI:7**
 TSO, operating TPNS under **OP:12**
 TSO/ISPF, operating TPNS under **OP:14**
 MVS Programmed Cryptographic Facility (PCF)
 See encrypting/decrypting messages
 MWINDOW operand (MCH statement) **LR:167**

N

N (Activate/Deactivate TPNS Control Program SIT
 Trace) operator command **OP:152**
 N subcommand (F command) **OP:144**
 name field **MGD:18**
 Name field, coding conventions **LR:1**
 NAME operand
 BRANCH statement **LR:298**
 CALL statement **LR:302**
 NAMEHASH operand (NTWRK statement) **LR:74**
 names, MSGTXT and MSGUTBL statements
 defined **STL:20**
 naming conventions for **STL:20**
 naming conventions **PI:55**
 events **STL:92, STL:95**
 labels **STL:19**
 MSGTXT and MSGUTBL names **STL:20**
 named constants **STL:22**
 variables **STL:22**
 NC operand (A command) **OP:135**
 NCB control block **UE:71**
 NCP
 generating for real host ownership **DEF:65**
 modifying **PI:54**
 modifying for channel-attached Type 2.1
 nodes **DEF:64**
 modifying for SNA subarea simulations **DEF:63**
 modifying for TPNS **DEF:11, DEF:63—65**
 real host ownership of **DEF:65, DEF:130—133**
 simulating **DEF:65, DEF:124—133**
 time stamping messages **GU:244**
 using for direct-attach configuration **PI:54**
 NCP channel definition statement **LR:250**
 NCP operand (NTWRKLOG statement) **LR:80**
 NCP subarea, defining
 See SNA (Systems Network Architecture), defining
 NCP subarea resources
 NCPDD operand
 CHANL statement **LR:251**
 CNTLR statement **LR:254**
 NCTLEXIT operand
 discussion **UE:10**
 parameter list **UE:10**
 NCTLEXIT operand (NTWRK statement) **LR:74**
 NETEXIT operand
 discussion **UE:14**
 parameter list **UE:14**
 sample routine **UE:14**
 NETEXIT operand (NTWRK statement) **LR:75**
 NETID data field option (PU21 statement) **LR:243, LR:442**
 NETID function **STL:352**
 NETUSER operand (NTWRK statement) **LR:75**
 NETVIEW operand (O command) **OP:154**

- NetView Performance Monitor **PI:145**
- NetView Performance Monitor (NPM)
 - description **SGU:74**
 - reformatting data from **SGU:59, SGU:76**
- NetView program
 - automating **OP:98—103**
 - defining the operator interface with TPNS **PI:20—24**
 - definition requirements **PI:11**
 - help panels **PI:101**
 - resource list function **PI:102**
 - simulated device display function **PI:102**
 - terminal, issuing commands from **OP:85—98**
 - using CLISTs **PI:102**
 - using for communication network management **PI:143**
 - using the operator interface **OP:83—85**
 - using the operator interface with TPNS **PI:101**
 - using, description of **OP:83**
- NetView program, using operator interface between TPNS and **STL:99**
- network
 - canceled **OP:55**
 - console recovery, using **OP:56**
 - defining **LR:70**
 - initializing **OP:27, OP:146**
 - parameters, altering **OP:55, OP:126**
 - query **OP:28**
 - resetting **OP:54**
 - resources, canceling **OP:55, OP:140**
 - resources, displaying status of **OP:28—48, OP:156**
 - specifying separate log data set for **LR:80**
 - starting **OP:27, OP:160**
 - starting resources for **OP:50**
 - stopping **OP:27, OP:155**
 - stopping resources for **OP:50**
- network control block **UE:71**
- network control exit
 - discussion **UE:10**
 - parameter list **UE:10**
- network definition **DEF:1, DEF:11—26**
 - coding **MGD:80**
 - controlling intermessage delays with **STL:79**
 - example for VTAM application **MGD:194**
 - example of **STL:9**
 - including in STL input **STL:123**
 - integrating message generation decks with **MGD:189—202**
 - referencing STL programs in **STL:126**
- network definition example, basic **DEF:23**
- network definition statement **LR:70—79**
- network definition statements
 - ADJNODE **DEF:185**
 - APPCLU **DEF:89**
 - as input to the Script Generator Utility **PI:78**
 - CALLDATA **DEF:199**
 - CDRM **DEF:123**

- network definition statements (*continued*)
 - CHANL **DEF:124**
 - CMND **DEF:122**
 - CNTLR **DEF:124, DEF:172**
 - coding **PI:56**
 - COS **DEF:122**
 - COSEND **DEF:122**
 - COSTAB **DEF:122**
 - DEV **DEF:22, DEF:23, DEF:108, DEF:172**
 - DIST **MGD:191**
 - FE **DEF:260**
 - FILE **DEF:111**
 - FPORT **DEF:170**
 - IF **DEF:268**
 - INCLUDE **MGD:192**
 - LCH **DEF:200**
 - LINE **DEF:22, DEF:23, DEF:127, DEF:171**
 - LINK **DEF:126, DEF:172**
 - LLS **DEF:160, DEF:170**
 - LPORT **DEF:159**
 - LU **DEF:69, DEF:123, DEF:127, DEF:185**
 - MCH **DEF:199**
 - MODEEND **DEF:122, DEF:184**
 - MODEENT **DEF:122, DEF:184**
 - MODETAB **DEF:122, DEF:184**
 - MSGUTBL **DEF:275**
 - NTWRK **DEF:21, DEF:23**
 - NTWRKLOG **DEF:265**
 - operands for controlling intermessage delays
 - coding **STL:79**
 - DELAY operand **STL:79**
 - EMTRATE operand **STL:80**
 - IUTI operand **STL:79**
 - listed **STL:79**
 - THKTIME operand **STL:79**
 - UTI operand **STL:79**
 - order of **PI:57**
 - PADPARMS **DEF:199**
 - PATH **MGD:190**
 - PATH (for message decks) **DEF:272**
 - PATH (for network routing) **DEF:121, DEF:125**
 - PSUBVECT **DEF:184**
 - PU **DEF:126, DEF:127, DEF:172**
 - PU21 **DEF:172, DEF:184**
 - RATE **DEF:264, MGD:75**
 - related to message generation decks **PI:72**
 - RN **MGD:32**
 - SET **DEF:259**
 - SIDEEND **DEF:89**
 - SIDEENT **DEF:89**
 - SIDEINFO **DEF:89**
 - specifying programs used, operands and statements for
 - ATRDECK operand **STL:127**
 - FRSTTXT operand **STL:127**
 - PATH operand **STL:127**
 - PATH statement **STL:127**

network definition statements (*continued*)

SSCP **DEF**:120
TCPIP **DEF**:108
TERM **DEF**:22, **DEF**:23, **DEF**:171
TP **DEF**:90
UDIST **DEF**:275
using **PI**:56
UTBL **DEF**:275, **MGD**:34
UTI **MGD**:72
VC **DEF**:202
VTAMAPPL **DEF**:69
network definition statements, hierarchy **DEF**:23
network definitions **PRI**:4, **PRI**:9
 multiple device simulation **PRI**:34
 multiple procedures **PRI**:30
 simple simulation **PRI**:20
network level logic test statement **LR**:46—60
network link definition statement **LR**:260—263
network logic tests **DEF**:268—271
NETWORK operand (LU (type 2.1 node)
statement) **LR**:232
network options
 automatic terminal recovery **DEF**:273
 counters
 allocation for TPNS resources **DEF**:258
 altering values of **DEF**:259—260
 index **DEF**:258
 sequence **DEF**:258
 delays, message generation
 automatic UTI adjustment **DEF**:262
 BLKDLY operand **DEF**:264
 DELAY operand **DEF**:263
 EMTRATE **DEF**:263
 RATE statement **DEF**:264
 THKTIME operand **DEF**:264
 user time intervals (UTIs) **DEF**:261
 UTI adjustment **DEF**:262—263
 UTI statement **DEF**:261
 exit routines **DEF**:276
 future events **DEF**:260
 logging messages
 cross-domain **DEF**:267
 DEBUG option **DEF**:266
 inhibiting message logging **DEF**:267—268
 separate log data sets for networks **DEF**:265
 logic tests **DEF**:268—271
 message generation delays
 See network options, delays, message generation
 message logging
 See network options, logging messages
 message tracing **DEF**:274
 paths for message generation decks **DEF**:272
 random number generation **DEF**:272
 response-time statistics, online **DEF**:271
 scanning, terminal **DEF**:273
 start time **DEF**:260

network options (*continued*)

STL program tracing **DEF**:274
terminal recovery, automatic **DEF**:273
terminal scanning **DEF**:273
tracing messages **DEF**:274
tracing STL programs **DEF**:274
user data tables **DEF**:275
user exit routines **DEF**:276
UTBL statement **DEF**:275
network options, coding **PI**:68
network route definition statement **LR**:280—282
network-level exit
 discussion **UE**:14
 parameter list **UE**:14
 sample routine **UE**:14
network-level IF statements **STL**:88
network-level logic tests **MGD**:84
networks
 estimating storage for **PI**:89
 resources in **PI**:55
 simulating with TPNS **PI**:55
networks, defining **DEF**:11—26
new line key statement **LR**:390
new line key, simulating **LR**:390
NL data field option **LR**:442
NL function **STL**:353
NL statement **LR**:390, **STL**:56, **STL**:256
NLKEY operand (TERM statement) **LR**:205
NMVT **MGD**:156, **PI**:144
NOCDDLOG command **GU**:61
NOCNSL command **GU**:63
NOCTRC command **GU**:64
NODATA command **GU**:65
node
 NCP subarea, defining **LR**:252
 NCP subarea, defining link between **LR**:260
 NCP subarea, defining link to **LR**:258
 NCP subarea, defining logical unit in **LR**:264
 subarea, defining routes between **LR**:280
 Type 2.1, defining link between two **LR**:217
 Type 2.1, defining link station between **LR**:239
 Type 2.1, defining logical unit in **LR**:222
NODE command
 Log Compare Utility **GU**:138
 Response Time Utility **GU**:207
NODELAY command **SGU**:84
NODELAY command, ITPLSGEN **SGU**:53
NODSPLY command **GU**:67
NOFMT command **GU**:73
NOHDR command **GU**:81
NOINFO command **GU**:75
NOLIMIT command **SGU**:85
NOLIST command **SGU**:86
NOLOG command **GU**:77
NOMDR command **GU**:78

NOMTRC command **GU**:80
 non-display devices, changing parameters for **UE**:2
 nonadjacent domain
 defined **PI**:38
 nonswappable **PI**:15
 NONTWRK command **SGU**:87
 NOP statement **STL**:43, **STL**:257
 NOPDSOUT
 defined **STL**:114
 NORESP command **SGU**:89
 NORESP statement **STL**:87—88, **STL**:258
 NORR command **GU**:82
 NOSEQOUT
 defined **STL**:114
 NOSEQOUT command **SGU**:90
 NOSMSG command **GU**:235
 NOSOURCE
 defined **STL**:114
 NOSSCP command **SGU**:91
 NOSTRC command **GU**:87
 NOTIFY operand (CMND statement) **LR**:313
 NOTIFY03 operand (CMND statement) **LR**:314
 NOTIFYP operand (CMND statement) **LR**:314
 NOTIFYS operand (CMND statement) **LR**:314
 NOTPNS
 defined **STL**:114
 NOTRACE command
 Loglist Utility **GU**:92
 Response Time Utility **GU**:235
 NOVERIFY command **GU**:94
 NOVERIFY command, ITPLSGEN **SGU**:54
 NOVRPAC command **GU**:83
 NPM
 See NetView Performance Monitor (NPM)
 NPM/LU2 Reformatter Utility **PI**:82
 NSEQ data field option **LR**:442
 NSEQ operand (A command) **OP**:135
 NSW operand
 IF (message generation) statement **LR**:414
 IF (network configuration) statement **LR**:458
 SETSW statement **LR**:414
 NSW operand (A command) **OP**:136
 NTRWK command **SGU**:87
 Log Compare Utility **GU**:139
 Loglist Utility **GU**:84
 Response Time Utility **GU**:208
 NTRWK statement **DEF**:21, **DEF**:23, **LR**:70—79, **PRI**:20
 NTRWKLOG statement **DEF**:265, **LR**:80
 NULSUP operand (TERM statement) **LR**:205
 NUMCOLS function **STL**:65, **STL**:354
 NUMREC operand (FILE statement) **LR**:44
 NUMROWS function **STL**:65, **STL**:355

O

O (Output Data/NetView Control) operator
 command **OP**:154
 OFFSET operand
 CURSOR statement **LR**:328
 SELECT statement **LR**:406
 ON operand (M command) **OP**:148
 ON SIGNED statement **STL**:96, **STL**:259
 ON statement **LR**:391, **MGD**:136
 ON/SIGNAL facility **MGD**:139
 ONEVENTS operand (DEACT statement) **LR**:339
 ONIN statement
 deactivating **STL**:83
 reference **STL**:82, **STL**:260
 testing **STL**:83
 using **STL**:82
 ONIN statement exercise **PRI**:83
 online response-time statistics (RSTATS) **DEF**:271
 ONOUT statement
 deactivating **STL**:83
 reference **STL**:82, **STL**:260
 testing **STL**:83
 using **STL**:82
 OPCMND statement **LR**:393, **MGD**:121, **PRI**:17,
 STL:101, **STL**:262
 operand field **MGD**:18
 operand names, entering **MGD**:20
 operand values, entering **MGD**:20
 operation
 execution parameters **OP**:17
 of TPNS **OP**:5
 specifying commands **OP**:21
 operational suggestions **UE**:38
 operator command exit
 discussion **UE**:11
 parameter list **UE**:11
 sample routine **UE**:12
 operator command statement **LR**:393
 operator commands **PRI**:17
 conventions **OP**:125
 entering at a console **PI**:99
 entering under MVS **OP**:24, **OP**:25
 entering under TSO **OP**:25
 entering under VM **OP**:26
 entering with OPCMND statement **PI**:100
 return codes on MVS **OP**:19
 return codes on VM **OP**:26
 specifying **OP**:21
 specifying resources with **PI**:99
 terms **OP**:123
 using **OP**:125—171, **PI**:98
 operator control command, specifying **LR**:393
 operator decisions, simulating **STL**:62
 operator interface between the NetView program and
 TPNS

- operator interface between the NetView program and TPNS (*continued*)
 - activating **OP:83**
 - deactivating **OP:84**
 - description of **OP:83**
 - routing messages **OP:84**
- operator reports
 - end of run reports **MGD:205, PI:109**
 - Inactivity Report **MGD:205, PI:110**
 - interval reports **MGD:204, PI:108**
 - trace reports **MGD:205, PI:109**
 - using **MGD:204, PI:107**
- operators
 - integer **STL:33**
 - relational **STL:49**
 - string **STL:25, STL:34**
- OPTFACL operand (CALLDATA statement) **LR:157**
- OPTIONS operand (NTWRK statement) **LR:75**
- OUTEXIT operand
 - discussion **UE:7**
 - parameter list **UE:7**
 - sample routine **UE:8**
- OUTEXIT operand (NTWRK statement) **LR:76**
- output
 - analyzing simulations with **MGD:203**
 - comparing 3270 display records **MGD:209**
 - determining response times **MGD:214**
 - formatting the log data set **MGD:206**
 - from RSTATS **PI:117**
 - from the Log Compare Utility **PI:113**
 - from the Loglist Utility **PI:111**
 - from the Response Time Utility **PI:115**
 - operator reports **PI:107**
 - types of **MGD:203, PI:107**
 - using operator reports **MGD:204**
 - using the log data set **MGD:206**
 - using the TPNS Display Monitor Facility **MGD:219**
- OUTPUT command, ITPLSGEN **SGU:51**
- output created by STL Translator
 - listed **STL:106**
 - MSGDD partitioned data set members **STL:112**
 - printed listing **STL:106**
 - sequential output data set, defining **STL:113**
 - temporary work data set **STL:114**
- Output Data/NetView control (O) operator
 - command **OP:154**
- output, specifying (ITPLSGEN) **SGU:51**
- overall selection control commands **GU:58**
- overriding script verification, ITPLSGEN **SGU:54**
- overriding user delays, ITPLSGEN **SGU:53**
- OWNER operand
 - LINE (peripheral link) statement **LR:259**
 - LINK statement **LR:263**

P

- P (Stop Network Resources) operator
 - command **OP:155**
- P command
 - Log Compare Utility **GU:156**
 - Loglist Utility **GU:85**
 - Response Time Utility **GU:209**
- P command, ITPLSGEN **SGU:54**
- ←< operator **STL:49**
- ←= operator **STL:49**
- > operator **STL:49**
- PA statement **LR:394**
- PA1-3 AID keys **STL:75**
- packet-switched network (PSN)
 - See X.25 packet-switched network (PSN)
- packets
 - See X.25 packet types
- PAD operand (DATASAVE statement) **LR:334**
- PAD operand (MSGTXT statement) **LR:387, MGD:22**
- PADPARMS operand (LCH statement) **LR:160**
- PADPARMS statement **DEF:199, LR:169—170**
- panels
 - Display Monitor Facility data stream **OP:111**
 - Display Monitor Facility selection **OP:107**
 - help, description of **OP:88**
 - RESLIST, with commands **OP:93**
 - RESLIST, without commands **OP:89**
- parallel sessions **DEF:74**
- parameter lists
 - controller-level exit **UE:30**
 - informational message-level exit **UE:17**
 - line error exit **UE:20**
 - message generation exit **UE:23**
 - message input exit **UE:4**
 - message output exit **UE:7**
 - network control exit **UE:10**
 - network-level exit **UE:14**
 - operator command exit **UE:11**
 - physical unit exit **UE:31**
 - user exit interface command exit **UE:22**
 - virtual route selection exit **UE:28**
 - descriptor block format **UE:29**
- PARM operand (EXIT statement) **LR:353**
- PARSESS operand **GU:233**
- PARSESS operand (LU (cross-domain) statement) **LR:274**
- partitions **SGU:80**
- passthrough network
 - See X.25 packet-switched network (PSN), defining X.25 network configurations
- PASSWD operand
 - APPCLU statement **LR:96**
 - VTAMAPPL statement **LR:124**
- PASSWD parameter **GU:233**

path names **PRI:21**
 PATH network definition operand, defining programs
 used **STL:127**
 PATH network definition statement
 CYCLIC operand **MGD:191**
 selecting decks **MGD:190**
 PATH network definition statement, defining programs
 used **STL:127**
 PATH operand **PRI:22**
 DEV (TCP/IP) statement **LR:133**
 DEV (terminal) statement **LR:186**
 LU (cross-domain) statement **LR:275**
 LU (type 2.1 node) statement **LR:232**
 LU (VTAMAPPL) statement **LR:118**
 TERM statement **LR:205**
 TP statement **LR:102**
 PATH operand (A command) **OP:136**
 PATH operand on network definition
 statements **MGD:190**
 path selection
 cyclic **MGD:191**
 probability distribution **MGD:191**
 PATH statement **LR:81—81, PRI:4, PRI:21**
 multiple PATH statements
 using to specify transaction mix **PI:71**
 PATH statement (for message decks) **DEF:272**
 PATH statement (for network routing) **DEF:121, DEF:125**
 PATH statement (network route definition) **LR:280**
 paths for message generation decks **DEF:272**
 PATHSEED operand (NTWRK statement) **LR:76**
 pause (QUIESCE) **PRI:17**
 PAUSE operand
 PU (cross-domain) statement **LR:285**
 PU (type 2.1 node) statement **LR:240**
 TERM statement **LR:205**
 PCF
 See encrypting/decrypting messages, MVS Programmed Cryptographic Facility (PCF)
 PCs
 See IBM Personal Computers (PCs)
 PERCENT command **GU:210**
 performance considerations **UE:39**
 performance tests
 conducting **PI:26**
 requirements **PI:26**
 peripheral link definition statement **LR:258—259**
 personal computers
 See IBM Personal Computers (PCs)
 PF (Program Function) keys **SGU:135**
 PF statement **LR:395**
 PF1-24 AID keys **STL:75**
 physical configuration
 CPI-C transaction program **PI:42, PI:44, PI:46**
 defined **PI:38**
 description of **DEF:11**
 physical configuration (*continued*)
 direct-attach **PI:42, PI:53**
 example **PI:42**
 link-attach **PI:42, PI:50**
 related to logical configuration **PI:38**
 TCP/IP **PI:42, PI:48**
 types
 direct-attach **DEF:12, DEF:19**
 link-attach **DEF:12, DEF:16**
 TCP/IP **DEF:12, DEF:15**
 VTAMAPPL **DEF:12, DEF:13**
 VTAMAPPL **PI:41, PI:44**
 physical unit control block **UE:76**
 physical unit definition statement **LR:283—286**
 physical unit exit
 discussion **UE:30**
 parameter list **UE:31**
 return codes **UE:31**
 physical unit Type 2.1 **LR:243—244**
 PID operand (JUMP statement) **LR:380**
 PIU trace, command **OP:161**
 planning for message generation
 designing message generation decks **MGD:9**
 determining message content **MGD:9**
 documenting message generation decks **MGD:10**
 identifying special requirements **MGD:8**
 integrating decks with network definitions **MGD:10**
 structuring message generation decks **MGD:9**
 testing TPNS scripts **MGD:10**
 understanding simulation objectives **MGD:7**
 understanding what you are testing **MGD:7**
 planning STL programs
 message content, determining **STL:14**
 procedures, structuring **STL:14**
 programs, structuring **STL:13**
 steps for program development **STL:13**
 PLENG operand (DATASAVE statement) **LR:335**
 POLL operand
 DEV (terminal) statement **LR:186**
 TERM statement **LR:206**
 polling and addressing **MGD:157**
 PORT operand (DEV (TCP/IP) statement) **LR:134**
 PORT operand (DEV (TCPIP) statement) **DEF:109**
 POS operand (DATASAVE statement) **LR:335**
 position cursor statement **LR:327**
 POST operand (A command) **OP:136**
 POST operand (EVENT statement) **LR:351**
 POST statement **STL:91, STL:263**
 POSTED function **STL:93, STL:356**
 posting an event **LR:351**
 defined **STL:92**
 determining if an event is posted **STL:93**
 using POST statement **STL:92**
 PRCVPAC operand (MODEENT statement) **LR:64**
 preprocessing the network **PRI:4, PRI:9**
 correcting errors **PRI:22**

preprocessing the network (*continued*)
 error listing **PRI:23**
 how to **PRI:17, PRI:22**
 listing with no errors **PRI:22**
 preprocessing TPNS scripts
 See TPNS scripts, preprocessing
 preprocessing with no errors **PRI:22**
 preprocessor
 See *also* TPNS scripts, preprocessing
 compared to ITPSYSIN **PI:89**
 Cross Reference Report, example of **GU:20**
 Cross Reference Report, using **GU:20**
 description of **GU:13**
 EXEC, example of **GU:25**
 execution parameters **GU:22**
 input **GU:13**
 JCL, example of **GU:23**
 output **GU:14, PI:88**
 output, examples of **GU:16—19**
 PREP statement **GU:14**
 return codes **GU:26**
 running **GU:21**
 TPNS/ISPF Interface, invoking **GU:22**
 TSO CLIST, example of **GU:24**
 using **PI:88**
 preprocessor error listing **PRI:23**
 primary resource selection control commands **GU:58**
 PRINT AID key **STL:75**
 print control block **UE:72**
 print key statement **LR:396**
 print key, simulating **LR:396**
 PRINT statement **LR:396**
 printed listing
 contents of by column **STL:108**
 defined **STL:106**
 printer
 dumping control blocks to **OP:141**
 dumping trace tables to **OP:161**
 printers, simulating **MGD:161**
 printing TPNS output
 requirements **PI:10**
 PRIPROT operand (MODEENT statement) **LR:64**
 probability distribution, path selection with **MGD:191**
 probability distributions, selecting table entries
 with **MGD:36**
 problems
 classifying **OP:115**
 hardware **OP:115**
 isolating
 incorrect data **OP:117**
 line errors **OP:119**
 loops **OP:117**
 missing message traffic **OP:117**
 no outstanding WTOR **OP:119**
 program checks **OP:116**
 reporting **OP:120**
 problems (*continued*)
 software **OP:115**
 user **OP:116**
 procedure
 defined **STL:17, STL:124**
 procedure, STL **PRI:5**
 multiple procedures **PRI:33**
 procedures from different STL programs, combining
 declaring variables for **STL:130**
 example of **STL:132**
 numbering IF statements for **STL:131**
 reasons for **STL:130**
 Process Actual statistics **OP:57**
 PROCESS command **GU:211**
 process commands, description of **GU:104**
 Process System statistics **OP:57**
 PROCESS, ACTUAL operand **GU:171, GU:211**
 PROCESS, SYSTEM operand **GU:171, GU:211**
 processing logic texts
 activating **MGD:100**
 deactivating **MGD:101**
 Product Identifier Subvector statement **LR:238**
 profile, ISPF editor **PRI:14, PRI:94**
 program
 See STL program
 program access key statement **LR:394**
 program access key, simulating **LR:394**
 program development, facilitating **STL:16**
 Program Directory
 See TPNS Program Directory
 Program Function (PF) keys **SGU:135**
 program function key statement **LR:307, LR:395**
 Program Function keys, simulating **LR:307, LR:395**
 PROGRAM=name
 defined **STL:115**
 programmed symbols (PS) **MGD:164**
 programs, structuring
 See structuring STL programs
 PROTMSG operand
 DEV (TCP/IP) statement **LR:134**
 DEV (terminal) statement **LR:186**
 LU (cross-domain) statement **LR:275**
 LU (type 2.1 node) statement **LR:232**
 LU (VTAMAPPL) statement **LR:118**
 TERM statement **LR:206**
 PRT control block **UE:72**
 PRTLNCNT=nnn
 defined **STL:115**
 PRTMSG operand
 DEV (terminal) statement **LR:187**
 LU (cross-domain) statement **LR:275**
 LU (type 2.1 node) statement **LR:233**
 TERM statement **LR:206**
 PRTSPD operand
 DEV (terminal) statement **LR:187**
 LU (cross-domain) statement **LR:275**

PRTSPD operand (*continued*)
 LU (type 2.1 node) statement **LR:233**
 LU (VTAMAPPL) statement **LR:118**
 TERM statement **LR:207**
 PRTSPD operand (A command) **OP:136**
 PS operand
 DEV (TCP/IP) statement **LR:134**
 DEV (terminal) statement **LR:187**
 LU (cross-domain) statement **LR:275**
 LU (type 2.1 node) statement **LR:233**
 LU (VTAMAPPL) statement **LR:118**
 PSEQACT operand (CMND statement) **LR:314**
 PSEQVAL operand (CMND statement) **LR:315**
 PSERVIC operand (MODEENT statement) **LR:64**
 PSN (packet-switched network)
 See X.25 packet-switched network (PSN)
 PSNDPAC operand (MODEENT statement) **LR:65**
 PSUBVECT statement **DEF:184, LR:238**
 PTN control block **UE:74**
 PU control block **UE:76**
 PU statement **DEF:126, DEF:127, DEF:172, 242, LR:283**
 PU21 group **DEF:184**
 PU21 logical unit definition statement **LR:222—237**
 PU21 physical unit definition statement **LR:239—242**
 PU21 scripts
 boundary channel attached **SMP:75**
 with dependent LUs **SMP:50**
 with LU Type 6.2 **SMP:52**
 PU21 simulation **LR:213—244**
 PU21 statement **DEF:172, DEF:184, LR:243—244**
 PU21ID data field option **LR:442**
 PU21ID function **STL:358**
 PUEXIT operand
 discussion **UE:30**
 parameter list **UE:31**
 return codes **UE:31**
 PUEXIT operand (NTRWK statement) **LR:76**
 PUID data field option **LR:442**
 PUID function **STL:357**
 PUNAME operand
 LINE (type 2.1 node) statement **LR:218**
 LINK statement **LR:263**
 PUTYPE operand (PU (cross-domain) statement) **LR:286**
 PVC operand (MCH statement) **LR:167**

Q

Q (Query Network Resources) operator command **OP:156**
 Q subcommand (F command) **OP:144**
 QSIGNAL operand (EVENT statement) **LR:352**
 QSIGNAL statement **STL:95, STL:265**
 qsignaling an event
 defined **STL:92**

qsignaling an event (*continued*)
 using QSIGNAL statement **STL:95**
 query
 APPCLU **OP:33, OP:36**
 device **OP:36**
 LCH **OP:45**
 line, link, channel, TCP/IP, or VTAM **OP:33**
 LLS **OP:42**
 LPORT or FPORT **OP:41**
 MCH **OP:44**
 network **OP:29**
 save areas **OP:48**
 subarea node **OP:31**
 terminal **OP:35**
 transaction program **OP:36**
 Type 2.1 node **OP:46**
 user areas **OP:48**
 Query Network Resources (Q) operator command **OP:156**
 QUIESCE indicator **MGD:65**
 QUIESCE operand
 DEV (TCP/IP) statement **LR:135**
 DEV (terminal) statement **LR:188**
 LU (cross-domain) statement **LR:276**
 LU (type 2.1 node) statement **LR:233**
 LU (VTAMAPPL) statement **LR:119**
 TERM statement **LR:207**
 TP statement **LR:102**
 QUIESCE operand (A command) **OP:136**
 QUIESCE statement **LR:397, MGD:65, PRI:17, STL:266**
 QUIESCE UNTIL POSTED
 defined **STL:93**
 example of **STL:93**
 using with ONIN and ONOUT **STL:94**
 QUIESCE UNTIL SIGNED
 defined **STL:96**
 QUIESCE UNTIL statement
 defined **STL:80**
 used to define asynchronous condition **STL:82**
 using **STL:89, STL:266**

R

R (Reset Network) operator command **OP:159**
 R operand (A command) **OP:136**
 RANDOM function **STL:60, STL:359**
 random number **LR:83**
 random number generation **DEF:272**
 random numbers
 as part of message text **STL:60**
 functions for generating **STL:60**
 generating messages dynamically with **MGD:32**
 represented by EBCDIC string **STL:61**
 using \$RNUM\$ to specify a range of **MGD:32**
 random selection of table entries **MGD:36**

RATE network definition statement **MGD:75**
 RATE statement **DEF:264, LR:82**
 rate table statement **LR:82**
 rate tables, generating **DEF:287—290**
 reading the log data set **UE:43**
 real host ownership (RHO)
 network definition example **DEF:131—133**
 of simulated NCP **DEF:130—133**
 performing simulations **DEF:130—131**
 types of simulations **DEF:130**
 real host ownership of a simulated NCP
 subarea **SMP:80**
 real host ownership of NCP subarea model **SMP:348**
 real host ownership simulation **LR:16**
 RECALL data field option **LR:442**
 receive (RECV) records **SGU:40**
 receiving messages
 during intermessage delay **STL:81**
 updating terminal's display buffer **STL:81**
 RECFM operand (FILE statement) **LR:45**
 RECFMS **PI:144**
 RECLen operand (FILE statement) **LR:45**
 record
 informational **GU:35**
 LAN resource **GU:34**
 log **GU:35**
 log data set, types of **GU:31**
 log display attribute table header **GU:45**
 Loglist Utility log header **GU:42**
 Loglist Utility log, types of **GU:46**
 MSGTRACE **GU:35**
 SNA resource **GU:31**
 STL trace **GU:36**
 TCP/IP resource **GU:33**
 X.25 resource **GU:33**
 record selection control commands **GU:58**
 reformatting trace output
 with ITPLU2RF **SGU:59**
 with ITPVTBRF **SGU:76**
 register linkage **UE:2**
 REGISTER operand **DEF:187**
 REGISTER operand (LU (type 2.1 node) statement) **LR:234**
 regression tests
 advantages of using TPNS for **PI:25**
 conducting **PI:25**
 relational operators **STL:48, STL:49**
 RELEASE operand (A command) **OP:136**
 releasing terminals **PRI:76**
 REMDTE operand (A command) **OP:136**
 REMDTE operand (LCH statement) **LR:160**
 remote job entry (RJE) scripts
 with 3770 multiple logical units **SMP:36**
 with 3770 single logical unit **SMP:40**
 with 3770 SNA **SMP:33**
 with 3780 BSC **SMP:34**
 remote job entry (RJE) scripts (*continued*)
 with Model 20 work station **SMP:30**
 REPEAT function **STL:54, STL:360**
 repeat previous function **GU:231**
 repetition function **GU:230, GU:231**
 REPLYTO operand (LLS statement) **LR:144, LR:154**
 REPORT command **GU:157, GU:212**
 REPORT command, ITPSGEN **SGU:88**
 REPORT control command, ITPSDL **SGU:145**
 REPORT control command, ITPTCL **SGU:137**
 REPORT operand (NTWRK statement) **LR:76**
 reporting problems **OP:120**
 reports
 See TPNS operator reports
 REQMS command **PI:144**
 Request Maintenance Statistics (REQMS) **PI:144**
 request/response header statement **LR:401—402**
 request/response unit (RU)
 in subarea simulations **DEF:153—155**
 in terminal simulations **DEF:150**
 segmented **DEF:153**
 requirements, special
 See special requirements
 requirements, system
 discussion **PRI:7**
 ITPECHO JCL **PRI:7**
 TPNS/ISPF Interface **PRI:8**
 reserved variable names **PRI:44**
 reserved variables
 in asynchronous conditions **STL:24**
 use of **STL:24**
 reserved words
 defined **STL:19**
 listed **STL:391**
 RESET (reset event) statement **STL:93, STL:267**
 RESET (reset key) statement **STL:269**
 reset key, simulating **LR:398**
 RESET operand (A command) **OP:137**
 RESET operand (EVENT statement) **LR:352**
 RESET statement **LR:398**
 coding example **MGD:120**
 using **MGD:120, MGD:158**
 resetting an event **LR:351**
 RESLIST command
 description of **OP:89**
 display with commands, sample of **OP:93**
 display without commands, sample of **OP:89**
 entering commands from **OP:92**
 exiting **OP:96**
 invoking **OP:93**
 messages **OP:96**
 PF keys for moving through **OP:93**
 redisplaying **OP:96**
 resource name, type combinations **OP:94**
 screen format of **OP:89**
 scrolling **OP:93**

RESOURCE operand **DEF:69, DEF:82, DEF:127**
 resource list function **OP:89, PI:102**
 RESOURCE operand
 CMND statement **LR:315**
 DEV (terminal) statement **LR:188**
 LU (cross-domain) statement **LR:276**
 LU (type 2.1 node) statement **LR:234**
 LU (VTAMAPPL) statement **LR:119**
 TERM statement **LR:207**
 resource selection control commands **GU:58**
 resources
 specifying with operator commands **PI:99**
 used for TPNS test **PI:33**
 resources, TPNS, allocating **STL:31, STL:131**
 RESP command **SGU:89**
 RESP operand
 CMND statement **LR:316**
 IF (message generation) statement **LR:371**
 IF (network configuration) statement **LR:53**
 TEXT statement **LR:426**
 RESP operand on the IF statement **MGD:153**
 RESP statement **LR:399**
 coding example **MGD:120, MGD:152**
 using **MGD:120, MGD:152**
 response listing file **GU:168**
 response time accuracy **GU:171**
 Response Time Frequency Distribution **GU:169**
 response time monitor customization statement **LR:84**
 response time reports
 description of **GU:164**
 levels of **GU:163**
 reading **GU:163—170**
 Response Time Utility **MGD:214**
 * (comment) command **GU:224**
 APPCLU command **GU:191**
 BTRANS command **GU:192**
 CGRAPH command **GU:195**
 command syntax **GU:190**
 compared to RSTATS **PI:117**
 control commands **GU:189**
 Cumulative Response Time Distribution **GU:169**
 Cumulative Response Time Distribution, example of **GU:186**
 defining transactions **PI:115**
 description of **GU:163**
 END command **GU:196**
 ETRANS command **GU:197**
 EXEC, example of **GU:181**
 execution parameters **GU:177**
 EXIT command **GU:200**
 EXTERM command **GU:201**
 for communication network management **PI:146**
 GRAPH command **GU:202**
 HEADER command **GU:203**
 JCL, example of, on disk **GU:180**
 JCL, example of, on tape **GU:179**

Response Time Utility (*continued*)
 JCL, using **GU:178**
 LENGTH command **GU:204**
 LINE command **GU:205**
 MSGTXT command **GU:206**
 NODE command **GU:207**
 NTWRK command **GU:208**
 output **PI:115**
 output, examples of **GU:181—187**
 output, list of reports **GU:163—170**
 P command **GU:209**
 PERCENT command **GU:210**
 PROCESS command **GU:211**
 REPORT command **GU:212**
 response listing file **GU:168**
 Response Time Frequency Distribution **GU:169**
 Response Time Frequency Distribution, example of **GU:185**
 return code **GU:187**
 RUN command **GU:214**
 running **PI:115**
 storage requirements **GU:176**
 TCPIP command **GU:215**
 TERM command **GU:216**
 terminal report, example of **GU:183**
 TGRAPH command **GU:218**
 TIME command **GU:219**
 Time Graph of Responses **GU:169**
 Time Graph of Responses, example of **GU:187**
 TPNS/ISPF Interface, running under **GU:177**
 TPRINT command **GU:220**
 transaction record listing **GU:167**
 transaction records listing, example of **GU:184**
 TRUNC command **GU:221**
 TSO CLIST, example of **GU:180**
 TSO, running under **GU:180**
 UNLOCK command **GU:222**
 user exit routines **GU:178**
 using **PI:114**
 VM, running on **GU:181**
 VTAMAPPL command **GU:223**
 Response Time Utility exit routines **UE:47**
 Response Time Utility exits, coding
 discussion **UE:47**
 fields corresponding to parameter list **UE:48**
 parameter list **UE:47**
 register linkage **UE:47**
 sample exit routine **UE:48**
 response time vector table **UE:77**
 response times
 analyzing **GU:163**
 calculating **GU:171**
 Cumulative Response Time Distribution **GU:169**
 determining **MGD:214, PI:114**
 determining with the Response Time Utility **MGD:214**

response times (*continued*)
 determining with the Response-Time Statistics facility **MGD:220**
 device dependencies **GU:172**
 monitoring **PI:145**
 Response Time Frequency Distribution **GU:169**
 rules for calculating **GU:171**
 Time Graph of Responses **GU:169**
 transactions as delimiters **GU:173**
 using RSTATS for **PI:104, PI:116**
 using the Response Time Utility for **PI:114**
 response unit (RU) **PRI:84**
 Response-Time Monitor
 description **PI:145**
 Response-Time Statistics facility (RSTATS) **MGD:220**
 response-time statistics, online **DEF:271**
 Restart Message Logging (E) operator
 command **OP:81, OP:142**
 restarting ITPECHO **PRI:8**
 restarting message logging **OP:81**
 restrictions on SUBSTR function in asynchronous conditions **STL:91**
 RETRYCT operand
 LLS statement **LR:144, LR:154**
 MCH statement **LR:167**
 RETRYTO operand (MCH statement) **LR:168**
 return codes **UE:2**
 ITPBTS **SGU:189**
 ITPIDC **SGU:45**
 ITPLU2RF **SGU:68**
 ITPSGEN **SGU:96**
 ITPVTBRF **SGU:77**
 on VM **OP:26**
 return codes, STL Translator **STL:120**
 return from subroutine statement **LR:400**
 RETURN statement **LR:400, MGD:124, PRI:57, STL:42, STL:270—270**
 RH **STL:24**
 RH statement **LR:401—402**
 coding example **MGD:154**
 using **MGD:149**
 using during SNA simulations **MGD:121**
 RH, building **LR:401**
 RH, modifying **STL:72, STL:274**
 RHO
 See real host ownership (RHO)
 RIGHT function **STL:361**
 RIGHT function (DATASAVE statement) **LR:332**
 RIGHT function on the DATASAVE statement **MGD:46**
 RIGHT operand (CURSOR statement) **LR:328**
 RMTLLS operand (LLS statement) **LR:145**
 RMTSAP operand (LLS statement) **LR:154**
 RN network definition statement **MGD:32**
 RN statement **LR:83**
 RNUM data field option **LR:444**
 RNUM function **STL:60, STL:362**
 ROLLDOWN AID key **STL:75**
 rolldown key statement **LR:403**
 rolldown key, simulating **LR:403**
 ROLLDOWN statement **LR:403**
 ROLLUP AID key **STL:75**
 rollup key statement **LR:404**
 rollup key, simulating **LR:404**
 ROLLUP statement **LR:404**
 routing TPNS messages **OP:58**
 ROW operand
 CURSOR statement **LR:328**
 SELECT statement **LR:406**
 ROWCOL function **STL:363**
 RSP control block **UE:77**
 RSP fields corresponding to parameter list **UE:48**
 RSTATS
 See also network options
 See also Response-Time Statistics facility (RSTATS)
 activating **PI:117**
 compared to the Response Time Utility **PI:117**
 for communication network management **PI:145**
 output **PI:117**
 using **PI:104, PI:116**
 RSTATS operand
 DEV (TCP/IP) statement **LR:135**
 DEV (terminal) statement **LR:188**
 LU (cross-domain) statement **LR:277**
 LU (type 2.1 node) statement **LR:235**
 LU (VTAMAPPL) statement **LR:120**
 TERM statement **LR:208**
 RSTATS operand (A command) **OP:137**
 RSTATS Query (W) operator command **OP:165**
 RSTATS, Response-Time Statistics feature
 description of **OP:56**
 output **OP:56**
 Process Actual **OP:57**
 Process System **OP:57**
 resetting **OP:58**
 RSTATS operand **OP:58**
 RTHRPUT operand (LCH statement) **LR:160**
 RTM (Response Time Monitor)
 See IBM 3274 Response Time Monitor (RTM)
 RTM operand
 PU (cross-domain) statement **LR:286**
 TERM statement **LR:208**
 RTM statement **LR:84**
 RTR operand
 DEV (terminal) statement **LR:189**
 LU (cross-domain) statement **LR:277**
 LU (type 2.1 node) statement **LR:235**
 LU (VTAMAPPL) statement **LR:120**
 TERM statement **LR:208**
 RU **STL:24**
 chaining to support, with TPNS Display Monitor Facility **OP:113**

RU (*continued*)
 unsolicited, transmitting **OP:162**
 RU (request/response unit)
 See request/response unit (RU)
 run **PRI:4**
 run a network **PRI:25**
 RUN command
 Log Compare Utility **GU:158**
 Loglist Utility **GU:86**
 Response Time Utility **GU:214**
 RUN command, ITPLSGEN **SGU:55**
 run STL **PRI:17**
 run time user exits, coding
 addressing mode considerations **UE:2**
 changing device parameters **UE:2**
 compatible exits **UE:38**
 discussion **UE:1**
 exit interface routine **UE:31**
 exit routines **UE:1**
 JCL and EXEC definitions **UE:37**
 operational suggestions **UE:38**
 performance considerations **UE:39**
 register linkage **UE:2**
 return codes **UE:2**
 run TPNS **PRI:23**
 running ITPLSGEN
 specifying execution parameters **SGU:49**
 using CLISTs **SGU:48**
 using EXECs **SGU:49**
 using JCL **SGU:48**
 using the TPNS/ISPF interface **SGU:49**
 running ITPLU2RF
 specifying execution parameters **SGU:61**
 using CLISTs **SGU:61**
 using EXECs **SGU:61**
 using JCL **SGU:60**
 using the TPNS/ISPF interface **SGU:61**
 running TPNS
 analyzing test results **PI:107**
 as a permanent task **PI:103**
 automatically **PI:100**
 controlling and monitoring **PI:98**
 estimating storage requirements **PI:89**
 from a console **PI:99**
 on MVS **PI:97**
 on VM **PI:98**
 running and analyzing a sample test **PI:87**
 specifying resources **PI:99**
 under TSO **PI:97**
 using online response-time monitor **PI:104**
 using OPCMND statement **PI:100**
 using the operator interface with the NetView
 program **PI:101**
 using the TPNS Display Monitor Facility **PI:102**
 using TPNS operator commands **PI:98**

running TPNS tests
 See running TPNS
 RUSIZES operand (MODEENT statement) **LR:65**

S

S (Start Network Resources) operator
 command **OP:160**
 S operand (A command) **OP:137**
 SALIMIT operand
 CNTLR statement **LR:254**
 SSCP statement **LR:289**
 sample exits
 EXIT operand **UE:52**
 EXIT statement **UE:24**
 sample EXIT statements **UE:27**
 INEXIT operand **UE:5**
 INFOEXIT operand **UE:17**
 LNEREXIT operand **UE:20**
 Loglist Utility exit **UE:43**
 NETEXIT operand **UE:14**
 OUTEXIT operand **UE:8**
 Response Time Utility exit **UE:48**
 UCMDEXIT operand **UE:12**
 sample installation networks
 ITPECHO **SMP:3**
 MVS directions for INSTALL1 **SMP:4**
 MVS directions for INSTALL2 **SMP:13**
 single-domain lines simulation **SMP:12**
 VM directions for INSTALL1 **SMP:6**
 VM directions for INSTALL2 **SMP:14**
 VTAM application **SMP:4**
 VTAM application program, description of **SMP:3**
 sample loglists
 CMS with 3270 BSC **SMP:261**
 INSTALL1 network **SMP:233**
 INSTALL2 network **SMP:245**
 real host ownership **SMP:275**
 TPNS as an application **SMP:256**
 type 2.1 node with independent LUs **SMP:293**
 sample output, Loglist Utility
 See STL trace output, Loglist Utility
 sample tests
 running and analyzing **PI:87**
 transaction mix in **PI:72**
 SAV control block **UE:78**
 save and user areas
 See also user areas
 converting DBCS data in a message from **MGD:47**
 generating messages dynamically with **MGD:42—50**
 inserting data into a message from **MGD:48**
 manipulating data in a message from **MGD:46**
 placing data in, with the DATASAVE
 statement **MGD:43**
 save area control block **UE:78**

- save areas **STL:22**
 - querying **OP:48**
- save data statement **LR:330—337**
- SAVEAREA operand
 - DEV (TCP/IP) statement **LR:135**
 - DEV (terminal) statement **LR:189**
 - LU (cross-domain) statement **LR:277**
 - LU (type 2.1 node) statement **LR:235**
 - LU (VTAMAPPL) statement **LR:120**
 - TERM statement **LR:208**
- saving data **LR:330**
- SAY statement **PRI:75, STL:99, STL:271**
- SB2DBCS function **STL:364**
- SB2DBCS function (DATASAVE statement) **LR:332**
- SB2DBCS function on the DATASAVE statement **MGD:47**
- SB2MDBCS function **STL:365**
- SB2MDBCS function (DATASAVE statement) **LR:332**
- SB2MDBCS function on the DATASAVE statement **MGD:47**
- SCAN **DEF:273**
- SCAN operand
 - IF (message generation) statement **LR:371**
 - IF (network configuration) statement **LR:53**
 - NTWRK statement **LR:77**
- SCANCNTR operand on the IF statement **MGD:98**
- SCANCTR operand
 - IF (message generation) statement **LR:372**
 - IF (network configuration) statement **LR:54**
- scanner internal trace (SIT)
 - activating **OP:152**
 - output **OP:71**
- scanning, terminal **DEF:273**
- schedule
 - example **PI:35**
- SCREEN **STL:24**
- Screen Data File (SDF) **SGU:103**
- Screen Definition Facility II (SDF II) **PI:80, SGU:103, SGU:113**
- SCREEN reserved variable **PRI:44**
- script
 - See TPNS scripts
- SCRIPT control command **SGU:163**
- script generating utilities
 - creating message generation decks with **MGD:4**
 - Interactive Data Capture **MGD:4**
 - ITPBTS **MGD:5, PI:82**
 - Script Generator Utility **MGD:4, PI:76**
 - TPNS Application Test Processor Utility (TPNS/ATP) **MGD:5**
 - TPNS/ATP **PI:80**
- Script Generator Utility **MGD:4**
 - capturing terminal traffic **PI:78**
 - defining network for **PI:78**
 - ITPSGEN **PI:76**
 - obtaining system traces **PI:77**

- Script Generator Utility (*continued*)
 - reformatting trace output **PI:78**
 - running ITPSGEN **PI:78**
 - sorting trace data **PI:78**
 - using **PI:76**
- Script Generator Utility, description **SGU:71, SGU:94**
- scripts
 - See TPNS scripts
- scripts, debugging with TPNS Display Monitor Facility **OP:105**
- scroll key statement **LR:405**
- scroll key, simulating **LR:405**
- SCROLL operand (TERM statement) **LR:209**
- SCROLL statement **LR:405, MGD:165, STL:272**
- scrolling **MGD:165**
- SDF (Screen Data File) **SGU:103**
- SDF II
 - See Screen Definition Facility II (SDF II)
- SDLC (synchronous data link control)
 - commands supported **DEF:149—150**
 - responses supported **DEF:149—150**
 - TPNS support in SNA simulations **DEF:117**
- SDLC dial lines **DEF:117**
- SDLC switched lines **DEF:117**
- SEARCH control command **SGU:146**
- SECDELAY operand (LINE (type 2.1 node) statement) **LR:221**
- secondary resource selection control
 - commands **GU:58**
- SECPROT operand (MODEENT statement) **LR:65**
- SELECT command **GU:140**
- SELECT command, multiple session selection with **OP:87**
- SELECT control command, ITPSDL **SGU:147**
- SELECT statement **LR:406, STL:44, STL:273**
- SELECT statement exercise **PRI:53**
- selecting table entries
 - randomly with a probability distribution **MGD:36**
 - with the \$UTBL\$ data field option **MGD:35**
- selection commands, description of **GU:103**
- selector pen **SGU:80**
- selector pen detect statement **LR:406**
- selector pen, simulating **LR:406**
- self-checking scripts
 - creating **PI:84—86**
 - creating with logic tests **MGD:111—115**
 - detecting problems without **PI:85**
 - logic example **MGD:112**
 - using **PI:84**
 - verifying simulations without **MGD:111**
 - with message-level logic tests **MGD:113**
 - with network-level logic tests **MGD:113**
 - writing **PI:84**
- SEND AID key **STL:75**
- send key statement **LR:407**

send key, simulating **LR:407**
 send message key statement **LR:409**
 SEND MSG key, simulating **LR:409**
 SEND statement **LR:407, MGD:144**
 sending messages
 See messages, transmitting
 SENDLINE AID key **STL:75**
 sendline key statement **LR:408**
 sendline key, simulating **LR:408**
 SENDLINE statement **LR:408**
 SENDMSG AID key **STL:75**
 SENDMSG statement **LR:409**
 SENDSABM operand (MCH statement) **LR:168**
 SENSE operand
 CMND statement **LR:316**
 RESP statement **LR:399**
 SENSE operand on the CMND statement **MGD:150**
 SEQ data field option **LR:444**
 SEQ operand
 DEV (TCP/IP) statement **LR:135**
 DEV (terminal) statement **LR:189**
 LU (cross-domain) statement **LR:277**
 LU (type 2.1 node) statement **LR:235**
 LU (VTAMAPPL) statement **LR:120**
 TERM statement **LR:209**
 TP statement **LR:102**
 SEQOUT
 See sequential output data set
 SEQOUT command **SGU:90**
 sequence counters
 See also network options, counters
 data field options for **MGD:39**
 generating messages with **MGD:39—40**
 sequential output data set
 defined **STL:113**
 sequential processing, altering **MGD:123**
 SERVADDR operand (A command) **OP:137**
 SERVADDR operand (DEV (TCP/IP) statement) **LR:136**
 session
 defined **PI:37**
 session, establishing **PRI:15**
 SESSIONS operand (CNOS operand (APPCLU statement)) **LR:96**
 sessions, initiating for SNA terminals **MGD:154**
 SESSNO data field option **LR:444**
 SESSNO function **STL:366**
 set attribute character statement **LR:297**
 set counters statement **LR:410**
 SET statement **DEF:259, LR:410, MGD:125**
 set switches statement **LR:414**
 set UTI statement **LR:416**
 SETRH statement **STL:72, STL:274**
 SETSW statement **LR:414, MGD:125**
 SETTH statement **STL:72, STL:276**
 setting attribute character on display screen **LR:297**
 setting counters **LR:410**
 setting switches **LR:46**
 setting up a future action **LR:391**
 setting up asynchronous conditions **STL:90**
 SETUTI statement **LR:416, MGD:73**
 shared variables **PRI:43, STL:23**
 SHOW command **OP:97**
 SHOW statement **LR:417, MGD:122, STL:99, STL:278**
 SIDEEND statement **DEF:89, LR:86**
 SIDEENT statement **DEF:89, LR:87**
 SIDEINFO operand (APPCLU statement) **LR:97**
 SIDEINFO statement **DEF:89, LR:89**
 SIGNAL operand (A command) **OP:137**
 SIGNAL operand (EVENT statement) **LR:352**
 SIGNAL statement **STL:95, STL:279**
 signaling an event
 defined **STL:92**
 using SIGNAL statement **STL:95**
 signalling action on an event **LR:351**
 SIGNORE control command **SGU:166**
 SIGNON control command **SGU:161**
 simple conditions
 See conditions, simple
 simple DO statement **STL:45, STL:224**
 simple DO statement groups **PRI:49**
 example **PRI:49**
 explanation **PRI:51**
 Simple TCP Client **DEF:105, DEF:114**
 Simple TCP Client configuration
 communication controller for **PI:12**
 network definition statement for **PI:61**
 requirements **PI:11**
 use of **PI:42**
 using **PI:49**
 Simple TCP Client example
 message generation decks **SMP:115**
 network definition statements **SMP:114**
 sample network **SMP:114**
 STL procedures **SMP:116**
 Simple TCP Client model **SMP:386**
 Simple TCP Client support **PI:122**
 simulated device display function **PI:102**
 simulated resource type codes
 device types **UE:53**
 terminal types **UE:53**
 simulated resource type codes, list of **GU:239**
 simulating
 3270 Data Analysis/APL Character Set **MGD:162**
 3270 extended functions **MGD:163**
 3274 Local Clear key **MGD:160**
 3278 Magnetic Stripe Reader **MGD:161**
 APL/Text Character Set **MGD:163**
 cursor movement **MGD:159**
 Delete key **MGD:159**
 Enter key **MGD:145**

simulating (*continued*)

- errors in 3270 BSC terminal **MGD:161**
- errors in 3270 SDLC terminal **MGD:161**
- errors in IBM 5250 terminal **MGD:180**
- errors in LU2 terminal **MGD:161**
- errors in SNA devices **MGD:152**
- Insert key **MGD:159**
- printers **MGD:161**
- Tab key **MGD:145**

simulating keyboard actions

See keyboard actions, simulating

simulation statements, list of **LR:292, LR:293, LR:294, LR:295**

simulations

See logical configuration, types

single-domain line definition **LR:192—194**

single-domain lines simulation sample **SMP:12**

single-domain simulation

defining line for X.25 logical channel **LR:192**

single-instance CPI-C TP simulation sample **SMP:117**
SIT

See scanner internal trace (SIT)

SIT trace report **PI:109**

SLR ALL **GU:169**

SMF **GU:169**

SMSG command **GU:235**

SNA

command, building **LR:308**

logical error, simulating **LR:399**

request/response header, modifying **LR:401**

simulation statements, list of **LR:292**

transmission header, modifying **LR:428**

SNA (Systems Network Architecture)

combining terminal and subarea simulations

- hierarchy of statements **DEF:136**

- network definition example **DEF:136—140**

defining host subarea resources **DEF:120—124**

defining NCP subarea resources **DEF:124—128**

defining subarea resources

- host channel interface **DEF:124**

- host logical unit **DEF:123**

- host routing **DEF:121**

- host session control **DEF:122—123**

- host subarea **DEF:120**

- NCP link interface **DEF:126—127**

- NCP physical and logical units **DEF:127—128**

- NCP routing **DEF:125**

- NCP subarea **DEF:124**

- remote host subarea **DEF:123**

defining terminals **DEF:20—24, DEF:118**

ENA (extended network addressing) **DEF:141—142**

extended network addressing **DEF:141—142**

extended subarea addressing

- hashing table for **DEF:142**

- specifying **DEF:141**

- TPNS support **DEF:141—142**

SNA (Systems Network Architecture) (*continued*)

network definition statements (subareas)

- CDRM **DEF:123**

- CHANL **DEF:124**

- CNTLR **DEF:124**

- COSTAB, COS, COSEND **DEF:122**

- LINE **DEF:127**

- LINK **DEF:126**

- LU **DEF:123, DEF:127**

- MODETAB, MODEENT, MODEEND **DEF:122**

- PATH **DEF:121, DEF:125**

- PU **DEF:126, DEF:127**

- SSCP **DEF:120**

network definition statements (terminals)

- DEV **DEF:22, DEF:118**

- LINE **DEF:22, DEF:118**

- NTWRK **DEF:21, DEF:118**

- TERM **DEF:22, DEF:118**

real host ownership

See real host ownership (RHO)

RHO

See real host ownership (RHO)

simulating resources and subareas **DEF:117—155**

simulating subareas

- activation of resources **DEF:145—148**

- CDRM-CDRM session activation **DEF:146**

- CDRM-CDRM session deactivation **DEF:147**

- deactivation of resources **DEF:145—148**

- defining **DEF:118—142**

- INIT-OTHER **DEF:148—148**

- LU-LU session activation **DEF:147—147**

- LU-LU session deactivation **DEF:148**

- request/response units (RUs) **DEF:153—155**

- resources simulated **DEF:118—119**

- SDLC commands **DEF:149—150**

simulating terminals

- activation of terminals **DEF:142—145**

- deactivation of terminals **DEF:142—145**

- defining **DEF:118**

- LU-LU session activation **DEF:143—144**

- LU-LU session deactivation **DEF:145**

- request/response units (RUs) **DEF:150**

- SDLC commands **DEF:149—150**

TPNS simulation support

- activation and deactivation of

 - resources **DEF:142—148**

- request/response units **DEF:150—155**

- SDLC commands and responses **DEF:149—150**

- SDLC lines **DEF:117**

SNA command statement **LR:308—317**

SNA commands **DEF:150**

SNA logon examples **MGD:155**

SNA LUs

- initiating (ILU) **STL:66**

- logging off **STL:69**

- logging on **STL:66**

SNA LUs (*continued*)

- primary **STL:66**
- secondary **STL:66**
- SNA messages, modifying **MGD:149**
- SNA resource record **GU:31**
- SNA response statement **LR:399**
- SNA subareas, defining
 - See SNA (Systems Network Architecture), defining subarea resources
- SNA terminals
 - generating messages for **MGD:149**
 - initiating sessions for **MGD:154**
- SNA, changing parameters for **UE:2**
- SNACMND statement **STL:280**
- SNASCOPE operand
 - IF (message generation) statement **LR:372**
 - IF (network configuration) statement **LR:54**
- SNF operand (TH statement) **LR:428**
- SON operand (CMND statement) **LR:316**
- sorting trace data **SGU:77**
- SOURCE operand (M command) **OP:149**
- SPACE operand (MSGDISK statement) **LR:68**
- special requirements
 - message generation statements, including **STL:13**
 - products, programs, and resources **STL:12**
 - running the test **STL:12**
- specifying
 - first message generation deck **MGD:192**
 - message generation decks for error recovery **MGD:193**
- specifying a device, ITPLSGEN **SGU:52**
- specifying input, ITPLSGEN **SGU:51**
- specifying message generation decks **LR:61**
- specifying output, ITPLSGEN **SGU:51**
- specifying rate table member **LR:82**
- SPOLL operand (FPORT statement) **LR:151**
- SRCVPAC operand (MODEENT statement) **LR:65**
- SSCOS operand (SSCP statement) **LR:289**
- SSCP command **SGU:91**
- SSCP statement **DEF:120, LR:287—289**
- SSCP subarea definition statement **LR:287—289**
- SSCPID data field option **LR:444**
- SSCPID function **STL:367**
- SSEQACT operand (CMND statement) **LR:316**
- SSEQVAL operand (CMND statement) **LR:317**
- SSNDPAC operand (MODEENT statement) **LR:65**
- START command **GU:141**
- Start Network Resources (S) operator
 - command **OP:160**
- start time **DEF:260**
- start-stop communication terminals
 - leased communication facilities **PI:135**
 - switched communication facilities **PI:135**
- starting a network **PRI:4, PRI:9**
 - how to **PRI:27**
- starting and stopping
 - network resources **OP:28**
 - TPNS **OP:27**
- starting IDC
 - defining the job stream **SGU:6**
 - running IDC as a batch job **SGU:6**
 - running IDC as a started procedure **SGU:7**
 - running IDC from a CLIST **SGU:7**
 - running IDC from GCS **SGU:8**
 - running IDC from the TPNS/ISPF interface **SGU:6**
 - specifying execution parameters **SGU:9**
- starting ITPLSGEN **SGU:55**
- starting to use TPNS exercise **PRI:13**
- starting TPNS **PRI:4, PRI:9**
 - how to **PRI:23**
- statement
 - assignment statements, use of **STL:18, STL:32**
 - continuation of **STL:19**
 - declarative statements, use of **STL:17, STL:18**
 - defined **STL:17**
 - elements of **STL:17**
 - expressions in **STL:18**
 - functions, use of **STL:18, STL:35**
 - keyword statements, use of **STL:18, STL:32**
 - message generation statements, including in STL program **STL:37**
 - types of **STL:18**
- statement correlation records, creating
 - using the @PROGRAM statement **STL:140**
 - using the PROGRAM execution parameter **STL:140**
- statement field **MGD:18**
- statements
 - 3101 simulation, list of **LR:292**
 - 3270 simulation, list of **LR:292**
 - 3290 simulation, list of **LR:293**
 - 3641 simulation, list of **LR:294**
 - 3643 simulation, list of **LR:294**
 - 5250 simulation, list of **LR:294**
 - 8775 simulation, list of **LR:295**
 - adjacent node definition **LR:216**
 - ADJNODE **LR:216**
 - APPCLU **LR:94**
 - ATTR **LR:297**
 - BRANCH **LR:298**
 - BTAB **LR:299**
 - CALC **LR:300**
 - CALL **LR:302**
 - CALL REQUEST data definition **LR:157**
 - CALLDATA **LR:157**
 - CANCEL **LR:303**
 - CDRM **LR:248**
 - CHANL **LR:250**
 - CHARSET **LR:304**
 - class of service table entry **LR:255**
 - CLEAR **LR:305**
 - CLEARPTN **LR:306**

statements (*continued*)

CMACCP LR:318
 CMALLC LR:320
 CMC FM LR:320
 CMC FMD LR:320
 CMD LR:307
 CMDEAL LR:320
 CMECS LR:320
 CMECT LR:320
 CMEMN LR:320
 CMEPLN LR:320
 CMESL LR:320
 CMFLUS LR:320
 CMINIT LR:320
 CMND LR:308
 CMPTR LR:320
 CMRCV LR:320
 CMSCT LR:320
 CMSDT LR:321
 CMSED LR:321
 CMSEND LR:320
 CMSERR LR:320
 CMSF LR:321
 CMSFM5 LR:321
 CMSLD LR:321
 CMSMN LR:321
 CMSPLN LR:321
 CMSPTR LR:321
 CMSRC LR:321
 CMSRT LR:320, LR:321
 CMSSL LR:321
 CMSST LR:321
 CMSTPN LR:321
 CMTRTS LR:321
 CNTLR LR:252
 COLOR LR:325
 COS LR:255
 COSEND LR:256
 COSTAB LR:257
 CTAB LR:326
 CURSOR LR:327
 CURSRSEL LR:329
 DATASAVE LR:330
 DEACT LR:338
 DELAY LR:340
 DELETE LR:342
 DIST LR:42
 DUP LR:343
 ENDTXT LR:344
 ENTER LR:345
 EREOF LR:346
 EREOS LR:347
 ERIN LR:348
 ERROR LR:349
 ESC LR:350
 EVENT LR:351

statements (*continued*)

EXIT LR:353
 FE LR:43
 FILE LR:44
 FLDADV LR:355
 FLDBKSP LR:356
 FLDMINUS LR:357
 FLDPLUS LR:358
 FM LR:359
 format control LR:459
 FPORT LR:150
 general definition, list of LR:291
 HELP LR:360
 HIGHLIGHT LR:361
 HOME LR:362
 IBM Token-Ring Network port definition LR:146
 IF (message generation) LR:363
 IF (network configuration) LR:46
 INCLUDE LR:61
 INSERT LR:379
 JUMP LR:380
 LABEL LR:381
 LCH LR:158
 LCLEAR LR:382
 LINE LR:258
 LINE (Single Domain) LR:192
 LINE (type 2.1 node) LR:217
 LINK LR:260
 LLS (LAN) LR:142, LR:152
 LOG LR:383
 LPORT LR:146
 LU (cross-domain) LR:264
 LU (type 2.1 node) LR:222
 LU (VTAMAPPL) LR:109
 MCH LR:163
 MODEEND LR:62
 MODEENT LR:63
 MODETAB LR:67
 MONITOR LR:385
 MSGDISK LR:68
 MSGTXT LR:386
 MSGUTBL LR:389
 network log data set LR:80
 NL LR:390
 NTRWK LR:70
 NTRWKLOG LR:80
 ON LR:391
 OPCMND LR:393
 operands for, summary of LR:31—31
 PA LR:394
 PADPARMS LR:169
 PATH LR:81
 PATH distribution LR:42
 PF LR:395
 PRINT LR:396
 PSUBVECT LR:238

statements (*continued*)

PU LR:239, LR:283

PU21 LR:243

QUIESCE LR:397

RATE LR:82

RESET LR:398

RESP LR:399

RETURN LR:400

RH LR:401

RN LR:83

ROLLDOWN LR:403

ROLLUP LR:404

RTM LR:84

SCROLL LR:405

SELECT LR:406

SEND LR:407

SENDLINE LR:408

SENDMSG LR:409

SET LR:410

SETSW LR:414

SETUTI LR:416

SHOW LR:417

SIDEEND LR:86

SIDEENT LR:87

SIDEINFO LR:89

SNA simulation, list of LR:292, LR:295

SSCP LR:287

STOP LR:418

STRIPE LR:419

SYSREQ LR:420

TAB LR:421

TCPIP LR:138

TERM LR:195

TEXT LR:422, LR:427

TH LR:428

TP LR:99

UDIST LR:90

UTBL LR:91

UTI LR:92

VC LR:171

VTAMAPPL LR:123

WAIT LR:430

WTO LR:433

statements for controlling intermessage delays STL:80

statements, asynchronous

See asynchronous statements

statements, asynchronous subset

See asynchronous subset statements

statements, continuation of STL:19

statements, network definition

See network definition statements

STATION operand (MCH statement) LR:168

STATUS operand

ERROR statement LR:349

IF (message generation) statement LR:373

STATUS=HOLD operand on the IF

statement MGD:102

STCPORT operand (TCPIP statement) LR:139

STHRPUT operand (LCH statement) LR:161

STIME operand (NTWRK statement) LR:77

STL

See Structured Translator Language

See Structured Translator Language (STL)

STL (Structured Translator Language) PRI:5, PRI:13

STL command, ITPLSGEN SGU:51

STL comments

multiple line comments PRI:45

single line comments PRI:45

STL functions PRI:73

CCOL function PRI:73

CHAR function PRI:73

CROW function PRI:73

INDEX function PRI:73

LENGTH function PRI:73

LUID function PRI:73

SUBSTR function PRI:73

TOD function PRI:73

STL input PRI:5, PRI:9

calling other procedures PRI:57

DO groups PRI:49

functions PRI:74

IF statements PRI:47

multiple procedures PRI:29

ONIN statements PRI:84

SELECT groups PRI:53

simple simulation PRI:15

user tables PRI:80

STL named constant declarations

CPI-C verb parameters STL:396

STL procedure PRI:5

See also procedure

multiple procedures PRI:33

STL procedures

for 5250 SMP:50

for ALC networks SMP:101

for CICS

with 3270 BSC and SNA SMP:17

with 3600 SMP:18

with 3767 SNA SMP:18

for CMS

with 2741 SMP:19

with 3270 BSC SMP:21

with 3270 SNA SMP:23

for CPI-C

with multiple-instance TPs SMP:129

with single-instance TPs SMP:122

for frame relay networks SMP:107

for FTP networks SMP:113

for IMS

with 3270 BSC and SNA SMP:27

with 3274 SNA SMP:27

with 3600 SMP:28

- STL procedures (*continued*)
 - for IMS (*continued*)
 - with TWX **SMP:29**
 - for ITPECHO **SMP:8**
 - for real host ownership **SMP:82**
 - for RJE
 - with 3770 multiple logical units **SMP:38**
 - with 3770 single logical unit **SMP:43**
 - with 3770 SNA **SMP:34**
 - with 3780 BSC **SMP:35**
 - with Model 20 work station **SMP:31**
 - for Simple TCP Client networks **SMP:116**
 - for Telnet 3270 networks **SMP:110**
 - for TPNS as an application **SMP:88**
 - for TSO
 - VTAM, with 3270 SNA, 3271 SDLC, 3767 **SMP:48**
 - VTAM, with 3767 SNA **SMP:46**
 - with 2741 **SMP:45**
 - for Type 2.1 node (PU21)
 - boundary channel attached **SMP:76**
 - with dependent LUs **SMP:51**
 - with LU Type 6.2 **SMP:63**
 - modifying the TPNS BIND response **SMP:94**
- STL program
 - defined **STL:6, STL:124**
 - elements of **STL:17**
 - example **PI:75**
 - examples of **STL:9**
 - procedures in **PI:74**
 - referencing in network definition **STL:126**
 - related to network definitions **STL:6**
 - using the STL Translator with **PI:75**
- STL program name, defining (ITPLSGEN) **SGU:51**
- STL program tracing **DEF:274**
- STL program, generating (ITPLSGEN) **SGU:51**
- STL programs, defining order of use
 - ATRDECK operand **STL:127**
 - FRSTTXT operand **STL:127**
 - PATH operand **STL:127**
 - PATH statements **STL:127**
- STL programs, documenting
 - See documenting STL programs
- STL programs, planning
 - See planning STL programs
- STL programs, structuring
 - See structuring STL programs
- STL programs, testing
 - See testing STL programs
- STL reserved words
 - defined **STL:19**
 - listed **STL:391**
- STL statement
 - See statement
- STL statements for controlling intermessage delays **STL:80**
- STL trace output, Loglist Utility
 - reading **STL:142**
 - sample output, explanation of **STL:142**
- STL trace records **GU:36, UE:43**
 - logging **STL:141**
 - printing **STL:141**
 - statement correlation records, obtaining **STL:140**
 - using to test TPNS scripts **PI:83**
- STL trace records exercise **PRI:61**
- STL Translator **PRI:17**
 - defined **STL:7, STL:103**
 - output from **PI:75**
 - return codes **STL:120**
 - running **STL:114**
 - running using a TSO CLIST **STL:117**
 - running using an EXEC **STL:118**
 - running using JCL **STL:116**
 - using **PI:75**
- STL Translator execution parameters
 - NOPDSOUT **STL:114**
 - NOSEQOUT **STL:114**
 - NOSOURCE **STL:114**
 - NOTPNS **STL:114**
 - PROGRAM=name **STL:115**
 - PRTLNCNT=nnn **STL:115**
 - using **STL:114**
- STL Translator return codes **STL:120**
- STL variable declarations
 - CPI-C verb parameters **STL:395**
- STL Variable Declarations for CPI-C Verb Parameters **STL:395**
- STL variables
 - classes **PRI:43**
 - reserved names **PRI:44**
 - types **PRI:43—45**
- STLMEM operand (MSGTXT statement) **LR:387**
- STLTRACE network definition operand **STL:141**
- STLTRACE operand **PRI:61**
 - DEV (TCP/IP) statement **LR:136**
 - DEV (terminal) statement **LR:189**
 - LU (cross-domain) statement **LR:277**
 - LU (type 2.1 node) statement **LR:235**
 - LU (VTAMAPPL) statement **LR:120**
 - TERM statement **LR:209**
 - TP statement **LR:102**
- STLTRACE operand (A command) **OP:137**
- stop execution (QUIESCE) **PRI:17**
- Stop Network Resources (P) operator
 - command **OP:155**
- STOP statement **LR:418, MGD:64**
- stopping Display Monitor **PRI:26**
- stopping TPNS and the network **PRI:9, PRI:37**
- storage
 - estimating requirements **PI:89**
 - for additional terminals **PI:96**
 - for TPNS control program **PI:96**

- storage (*continued*)
 - host processor virtual **PI:89**
 - network size **PI:89**
 - sample estimate of network groups **PI:90**
 - sample network for estimating **PI:90—96**
- storage for TPNS scripts **DEF:26**
- STRC command **GU:87**
- stress tests
 - advantages of using TPNS for **PI:27**
 - conducting **PI:26**
 - requirements **PI:27**
- string
 - condition, defined **STL:50**
 - constant, defined **STL:24**
 - expression, defined **STL:34**
 - operators **STL:34**
 - variable, defined **STL:22**
- STRING statement **STL:28, STL:285**
- string variables **PRI:43**
- STRIPE statement **LR:419, MGD:161, STL:59, STL:286**
- structured flow-of-control statements
 - CALL statement
 - asynchronous subset statement **STL:84**
 - structured flow-of-control statement **STL:42**
 - DO FOREVER statement groups **STL:46**
 - DO statement group execution, control of **STL:47**
 - DO statement groups, types of **STL:45**
 - DO WHILE statement groups **STL:46**
 - IF/THEN/ELSE statement groups **STL:43**
 - iterative DO statement groups **STL:47**
 - SELECT statement groups **STL:44**
 - simple DO statement groups **STL:45**
 - use of **STL:41**
- Structured Translator Language
 - described **STL:5**
 - using message generation decks with **STL:7**
 - when to use **STL:7**
- Structured Translator Language (STL) **MGD:4, PRI:5, PRI:13**
 - advantages of using **PI:74**
 - using **PI:74**
- structuring STL input
 - avoiding procedure call misuse **STL:125**
 - coding multiple programs **STL:124**
 - naming programs, procedures, and user tables **STL:126**
 - organizing STL programs **STL:124**
- structuring STL programs
 - calling procedures in separate source data sets **STL:125**
 - containing multiple procedures **STL:129**
 - related to PATH statement entries **STL:129, STL:130**
 - using separate source data sets **STL:128**
- SUBAHASH operand (NTWRK statement) **LR:78**
- subarea
 - defined **PI:37**
- SUBAREA operand
 - CDRM statement **LR:248**
 - CNTRLR statement **LR:254**
 - PU (cross-domain) statement **LR:286**
 - SSCP statement **LR:287**
- subarea simulation
 - activating resources **DEF:145—148**
 - adjacent domain **DEF:119**
 - CDRM session, activating **DEF:146**
 - CDRM session, deactivating **DEF:147**
 - components simulated **DEF:118—119**
 - deactivating resources **DEF:145—148**
 - defined **PI:37**
 - defining resources **DEF:118—140**
 - encrypting/decrypting messages in **DEF:298—305**
 - host subarea, defining **DEF:120—124**
 - LU-LU session, activating **DEF:147—147**
 - LU-LU session, deactivating **DEF:148**
 - multiple-domain network **PI:52**
 - NCP subarea, defining **DEF:124—128**
 - nonadjacent domain **DEF:119**
 - physical configuration for **PI:53**
 - single-domain network **PI:52**
 - SNA commands supported **DEF:154—155**
 - statements for **PI:63—65**
 - time stamping messages **GU:244**
 - X.25 network example **DEF:208—211**
 - X.25 passthrough network example **DEF:210**
- subareas, defining
 - See* SNA (Systems Network Architecture), defining subarea resources
- SUBSTR function **PRI:73**
 - in asynchronous conditions **STL:90—91, STL:368—369**
 - use of **STL:36, STL:368—369**
- Summary Report **GU:112, PI:114**
- Summary Report (SNA 3270 Reformatter Utility) **SGU:62**
- SUSPEND statement **PRI:16, STL:81, STL:287—288**
- SW operand
 - IF (message generation) statement **LR:414**
 - IF (network configuration) statement **LR:458**
 - SETSW statement **LR:414**
- SW operand (A command) **OP:137**
- switch tests **GU:39**
- switched lines (BSC2) **DEF:245**
- switched lines (SDLC) **DEF:117**
- switched lines (TWX) **DEF:241**
- switches **STL:22**
- switches, setting **LR:414**
- switches, setting with the SETSW statement **MGD:125**
- SWSUBFLD operand (PSUBVECT statement) **LR:238**
- synchronizing
 - with control commands **GU:106**

- synchronizing (*continued*)
 - with selection commands **GU:107**
- synchronizing multiple devices with events **MGD:140**
- synchronous data link control (SDLC)
 - See SDLC (synchronous data link control)
- Synchronous Data Link Control (SDLC) line disciplines
 - TPNS support for **PI:137**
- synchronous statements **PRI:47**
- SYNCPPOINT command **GU:142**
- syntax conventions
 - for data field options **MGD:30**
 - for message generation statements **MGD:17**
- syntax errors, correcting
 - See errors, correcting, syntax
- syntax rules for STL **STL:18**
- syntax testing **MGD:10**
- SYSPRINT
 - allocating the data set **OP:12**
- SYSREQ AID key **STL:75**
- SYSREQ key, simulating **LR:420**
- SYSREQ statement **LR:420, STL:289**
- system request statement **LR:420**
- system requirements
 - discussion **PRI:7**
 - ITPECHO JCL **PRI:7**
 - TPNS/ISPF Interface **PRI:8**
- system requirements for TPNS **PI:7**
- system services control point (SSCP)
 - defined **PI:37**
- system under test
 - defined **PI:38**
- system under test, definition of **DEF:1**
- Systems Network Architecture (SNA)
 - See SNA (Systems Network Architecture)

T

- T (Dispatcher and Channel Traces) operator
 - command **OP:161**
- TAB data field option **LR:445**
- TAB function **STL:370**
- tab key statement **LR:421**
- tab key, simulating **LR:421, MGD:145**
- TAB statement **LR:421, MGD:145, STL:56, STL:290**
- tab statement, conditional **LR:326**
- TABLEI operand (DATASAVE statement) **LR:335**
- TABLEO operand (DATASAVE statement) **LR:336**
- tables exercise **PRI:79**
- TADDR operand
 - LINE (type 2.1 node) statement **LR:221**
 - LINK statement **LR:263**
- TAG, event **STL:92, STL:95**
- TC operand (A command) **OP:137**
- TCF (Testcase File) **SGU:103**
- TCP/IP **DEF:105**
 - See also FTP

- TCP/IP (*continued*)
 - See also Simple TCP Client
 - See also Telnet 3270
 - See also Telnet 3270 network
 - coding network definition **DEF:107**
 - example network definition **DEF:108**
 - defining application configurations **DEF:106**
 - logical configuration **DEF:106**
 - physical configuration **DEF:107**
 - defining clients **DEF:108**
 - general simulation characteristics **DEF:109**
 - defining connections **DEF:108**
 - network definition statements
 - DEV **DEF:107, DEF:108**
 - TCPIP **DEF:107, DEF:108**
 - simulating clients **DEF:105**
 - using client simulation **DEF:107**
 - using connection protocol **DEF:105**
- TCP/IP connection (TCPIP)
 - canceled **OP:140**
 - JCL for **OP:11**
 - listing addresses of **OP:147**
 - query network for **OP:33, OP:156**
 - starting activity for **OP:160**
 - stopping activity for **OP:155**
- TCP/IP resource records **GU:33**
- TCPIP
 - See TCP/IP connection (TCPIP)
- TCPIP command
 - Log Compare Utility **GU:143**
 - Loglist Utility **GU:88**
 - Response Time Utility **GU:215**
- TCPIP statement **DEF:108, LR:138—139, PI:61**
- TCPIPID data field option **LR:445**
- TCPIPID function **STL:371**
- TCPNAME operand (TCPIP statement) **LR:139**
- Teleprocessing Network Simulator
 - See TPNS
- Teleprocessing Network Simulator (TPNS)
 - defined **STL:5**
 - description of **OP:1, SMP:1**
 - operating systems supported **STL:5**
 - operation of **OP:5**
- Telnet 3270
 - logging off **STL:70**
 - logging on **STL:70**
- Telnet 3270 clients, simulating
 - time stamping messages **GU:244**
- Telnet 3270 configuration
 - communication controller for **PI:12**
 - network definition statements for **PI:61**
 - requirements **PI:11**
 - use of **PI:42**
 - using **PI:49**
- Telnet 3270 example
 - message generation decks **SMP:109**

- Telnet 3270 example (*continued*)
 - network definition statements **SMP:109**
 - sample network **SMP:109**
 - STL procedures **SMP:110**
- Telnet 3270 model **SMP:379**
- Telnet 3270 network
 - simulating clients **DEF:105, DEF:110**
 - simulation characteristics
 - 3270 **DEF:110**
 - display **DEF:110**
- temporary work data set
 - defined **STL:114**
- TERM command
 - Log Compare Utility **GU:144**
 - Loglist Utility **GU:66, GU:89**
 - Response Time Utility **GU:216**
- TERM statement **DEF:22, DEF:23, DEF:171, LR:195, PI:63**
- TERMIN data field option **LR:445**
- TERMIN function **STL:372**
- terminal
 - NetView, issuing commands from **OP:85—97**
 - query network for **OP:35**
 - simulated, displaying **OP:145**
- terminal control block **UE:79**
- terminal counters **LR:457**
- terminal definition statement **LR:195—211**
- terminal recovery, automatic **DEF:273**
- terminal scanning **DEF:273**
- terminal simulation
 - defined **PI:37**
 - multiple-domain network **PI:49**
 - physical configuration for **PI:50**
 - single-domain network **PI:49**
 - statements for **PI:62**
 - time stamping messages **GU:244**
- terminal simulations, list of **LR:453—455**
- Terminal Status Query (G) operator command **OP:145**
- terminal traffic system, capturing **SGU:73**
- terminal traffic, single, capturing **SGU:72**
- terminal type codes **UE:53**
- terminal types supported by ITPSGEN **SGU:80**
- terminal, defining **LR:195**
- terminals
 - storage requirements **PI:96**
 - supported by TPNS **PI:135**
- terminals, defining
 - See* SNA (Systems Network Architecture), defining terminals
- TERMSELF statement **STL:69, STL:291**
- test objectives
 - establishing **PI:29**
- test plan **DEF:11**
 - creating **PI:29**
 - designing **PI:30**
 - entry and exit criteria **PI:35**
- test plan (*continued*)
 - example **PI:30**
 - introduction **PI:33**
 - parts of **PI:32**
 - resources **PI:33**
 - schedule for conducting test **PI:35**
 - specifications **PI:34**
 - statement of objectives **PI:29, PI:33**
 - test procedures **PI:34**
 - writing **PI:32**
- test plan for simulation, understanding **STL:11**
- test system
 - See* system under test, definition of
- test under mask comparison **STL:51**
- TESTCASE control command **SGU:138**
- Testcase File (TCF) **SGU:103**
- testing application development
 - testing VTAM applications **PI:45**
 - using TPNS as an application prototype **PI:45**
- testing asynchronous conditions
 - See* asynchronous conditions, testing
- testing for event completion, IF (network configuration)
 - statement **LR:46**
- testing ONIN statements
 - See* ONIN statement, testing
- testing ONOUT statements
 - See* ONOUT statement, testing
- testing STL programs
 - logic testing **STL:15, STL:140**
 - syntax testing **STL:15, STL:133**
- TESTREQ key, simulating **LR:420**
- tests
 - analyzing results **PI:107**
 - capacity planning **PI:27**
 - costs of **PI:27**
 - counter **GU:40**
 - cursor position **GU:39**
 - data **GU:37**
 - deciding what to test **PI:27**
 - describing procedures for **PI:34**
 - establishing objectives for **PI:29, PI:33**
 - event **GU:38**
 - function **PI:25**
 - performance **PI:26**
 - regression **PI:25**
 - running and analyzing a sample **PI:87**
 - running TPNS **PI:97**
 - stress **PI:26**
 - switch **GU:39**
 - types of **PI:25**
 - using output from **PI:107**
- text entry, simulating
 - See* keyboard actions, simulating, text entry
- TEXT operand
 - DATASAVE statement **LR:336**
 - IF (message generation) statement **LR:373**

TEXT operand (*continued*)

- IF (network configuration) statement **LR:55**
- MSGTXT statement **LR:387**

TEXT operand on the IF statement **MGD:88**

text operands, continuation of **LR:3**

TEXT statement **LR:422—427**

- coding **MGD:23**
- generating messages with **MGD:29—51**
- summarizing message generation with **MGD:50**

TEXTSEED operand (NTWRK statement) **LR:78**

TEXTTO operand (PADPARMS statement) **LR:170**

TGN operand

- PU (cross-domain) statement **LR:286**
- PU (type 2.1 node) statement **LR:241**

TGRAPH command **GU:218**

TH **STL:24**

TH statement **LR:428**

- coding example **MGD:120, MGD:153**
- using **MGD:149**
- using during SNA simulations **MGD:120**

TH, modifying **STL:72, STL:276**

THEN operand

- IF (message generation) statement **LR:374**
- IF (network configuration) statement **LR:55**
- ON statement **LR:392**

THEN operand on the IF statement

- E (execute) on **MGD:94**
- IGNORE action on **MGD:97**
- on multiple IF statements **MGD:96**
- use of **MGD:92**
- VERIFY—(data) on **MGD:95**

THKTIME network definition operand, using to control intermessage delay **STL:79**

THKTIME operand

- DEV (TCP/IP) statement **LR:136**
- DEV (terminal) statement **LR:189**
- LU (cross-domain) statement **LR:278**
- LU (type 2.1 node) statement **LR:236**
- LU (VTAMAPPL) statement **LR:121**
- TERM statement **LR:209**

THKTIME operand on the NTWRK statement **MGD:71**

THROTTLE operand

- DEV (terminal) statement **LR:190**
- LU (cross-domain) statement **LR:278**
- LU (type 2.1 node) statement **LR:236**
- LU (VTAMAPPL) statement **LR:121**
- TERM statement **LR:210**

TIC traces **PI:109**

TIME command

- Loglist Utility **GU:90**
- Response Time Utility **GU:219**

TIME command, ITPSGEN **SGU:92**

Time Graph of Responses **GU:169**

time graph parameters **GU:218**

time of day (TOD) function

- See* TOD function

TIME operand

- DELAY statement **LR:340**
- EVENT statement **LR:352**
- FE statement **LR:43**
- WAIT statement **LR:431**

time sharing option (TSO) scripts

- SPF, with 3270 SNA **SMP:50**
- VTAM, with 3270 SNA, 3271 SDLC, 3767 **SMP:47**
- VTAM, with 3767 SNA **SMP:45**
- with 2741 **SMP:45**

time stamps

- description of **GU:242**
- for message data records **GU:243**
- for records that are not message data records **GU:243**
- with local area network simulation **GU:245**
- with subarea simulation **GU:244**
- with Telnet 3270 or FTP client simulation **GU:244**
- with terminal simulation **GU:244**
- with TWX and 2741 terminal simulation **GU:245**
- with VTAMAPPL simulation **GU:243**

TIMER operand (M command) **OP:150**

TNPORT operand (TCPIP statement) **LR:139**

TOD data field option **LR:445**

TOD function **PRI:73, STL:61, STL:373**

token-ring adapter

- defining TPNS resources with **DEF:159**
- using for TPNS simulations **DEF:157**

TP **STL:211**

- See also* transaction program
- See also* transaction program (TP)

TP command **GU:91**

TP statement **DEF:90, LR:99**

- CPI-C transaction program configuration **PI:60**

TPID data field option **LR:445**

TPID function **STL:374**

TPINSTNO data field option **LR:445**

TPINSTNO function **STL:375**

TPNAME operand

- SIDEENT statement **LR:87**
- SIDEINFO operand (APPCLU statement) **LR:97**
- TP statement **LR:102**

TPNS

- See also* Teleprocessing Network Simulator (TPNS)
- authorizing **PI:14**
- checklist for installing **PI:24**
- communication controller requirements **PI:12**
- controlling and monitoring **PI:97**
- defining the operator interface with NetView **PI:20**
- defining the simulated network **PI:55**
- description **PI:1**
- general requirements **PI:10**
- host processor requirements for **PI:1**
- making nonswappable **PI:15**
- MVS requirements for **PI:7**
- types of resources simulated **PI:2**

TPNS (*continued*)

- using **PI:1**
- VM requirements for **PI:10**
- TPNS Application Test Processor (TPNS/ATP) **MGD:5**
 - advantages of using **PI:81**
 - documentation processors **PI:81**
 - editor **PI:81**
 - facilities **PI:80**
 - installation requirements **PI:11**
 - running **PI:80**
 - script generator **PI:81**
 - using **PI:80**
- TPNS CLISTs **PI:133**
- TPNS command, ITPLSGEN **SGU:52**
- TPNS comment command **SGU:176**
- TPNS communication controller, loading **OP:168**
- TPNS console messages, inhibiting logging of **OP:80**
- TPNS control blocks, dumping **OP:50**
- TPNS control program
 - control program output **DEF:31**
 - description **DEF:27**
 - entering macros **DEF:27, DEF:28**
 - estimating storage for **PI:96**
 - generating **DEF:27—65, PI:54**
 - generation example **DEF:60**
 - generation macros
 - GENEND **DEF:59**
 - TPNS1050 **DEF:54**
 - TPNS2741 **DEF:57**
 - TPNSACAL **DEF:38**
 - TPNSALC **DEF:41**
 - TPNSBSC1 **DEF:43**
 - TPNSBSC2 **DEF:43**
 - TPNSBSC3 **DEF:43**
 - TPNSFDXS **DEF:46**
 - TPNSGEN **DEF:35**
 - TPNSHDXS **DEF:46**
 - TPNSLPRT **DEF:49**
 - TPNSMCAL **DEF:38**
 - TPNSTWX **DEF:51**
 - TPNSTWXL **DEF:51**
 - generation stages, overview **DEF:27**
 - MVS, generating on **DEF:28—32**
 - return codes **DEF:32**
 - running the control program generator **DEF:222**
 - on MVS **DEF:28**
 - on VM **DEF:30**
 - specifying execution parameters **DEF:31**
 - with the TPNS/ISPF Interface **DEF:30**
 - time stamping messages **GU:244**
 - using **PI:51**
 - VM, generating on **DEF:28**
- TPNS control program generation macros
 - coding conventions **DEF:33**
 - GENEND 58, **DEF:59**
 - TPNS1050 **DEF:54**

TPNS control program generation macros (*continued*)

- TPNS2741 **DEF:57**
- TPNSACAL **DEF:38**
- TPNSALC **DEF:41**
- TPNSBSC1 **DEF:43**
- TPNSBSC2 **DEF:43**
- TPNSBSC3 **DEF:43**
- TPNSFDXS **DEF:46**
- TPNSGEN **DEF:35**
- TPNSHDXS **DEF:46**
- TPNSLPRT **DEF:49**
- TPNSMCAL **DEF:38**
- TPNSTWX **DEF:51**
- TPNSTWXL **DEF:51**
- TPNS Display Monitor Facility **MGD:169, MGD:219**
 - BIND images supported **OP:113**
 - chaining to support RU **OP:113**
 - data stream display **OP:105**
 - controlling **OP:112**
 - data stream display fields **OP:111**
 - data stream display messages **OP:112**
 - debugging scripts with **OP:105**
 - debugging TPNS scripts **PI:103**
 - demonstrating products **PI:103**
 - displaying screen images and data flows **OP:108, OP:110**
 - installation requirements **PI:11**
 - logging on **OP:107**
 - monitoring data streams **OP:110**
 - monitoring resources **PI:103**
 - selection panel **OP:107**
 - specifying M operator command operands **OP:107, OP:148**
 - specifying source for building data stream **OP:108**
 - specifying the resource name **OP:108, OP:110**
 - starting monitoring with M operator command **OP:106, OP:148**
 - updating monitoring display **OP:108**
 - using **PI:102**
 - using, description of **OP:105**
- TPNS Display Monitor Facility, using **STL:99**
- TPNS library, setting up **SGU:105**
- TPNS line types
 - See line types simulated by TPNS
- TPNS message generation statements in STL programs **STL:37**
- TPNS operator reports
 - end of run **OP:67**
 - inactivity **OP:73**
 - interval **OP:61—67**
 - line trace **OP:69**
 - LLS Interval **OP:66**
 - LLS Resource End of Run **OP:68**
 - scanner internal trace (SIT) **OP:71**
 - types, list of **OP:61**
 - X.25 Resource End of Run **OP:68**

TPNS operator reports (*continued*)
 X.25 Resource Interval **OP:64**
 TPNS output
 See output
 TPNS preprocessor **MGD:201**
 See also TPNS scripts, preprocessing
 TPNS Primer Information List **PRI:95**
 TPNS Program Directory
 obtaining **PI:13**
 printing **PI:14**
 TPNS resources, allocating **STL:31, STL:131**
 TPNS sample data set **PI:131**
 TPNS scripts
 coding **PI:55**
 coding method **DEF:2**
 coding steps (checklist) **DEF:8**
 coding with delimiters **MGD:67**
 coding with intermessage delays **MGD:79**
 debugging with the TPNS Display Monitor Facility **PI:103**
 defined **PI:1**
 description of **DEF:1**
 developing **MGD:11**
 example **DEF:3**
 example for VTAM application **MGD:198**
 methods for testing **PI:83—86**
 preprocessing **DEF:25, PI:87**
 samples **MGD:229—255**
 self-checking **PI:84—86**
 storage for **DEF:26**
 storing **MGD:201**
 testing **MGD:10**
 TPNS System Requirements Checklist **PRI:94**
 TPNS test plan
 See test plan
 TPNS tests
 configuring your system **PI:4**
 creating message generation decks **PI:4**
 defining the simulated network **PI:4**
 planning for **PI:3**
 running the test **PI:5**
 sequence for conducting **PI:3**
 using TPNS output **PI:5**
 TPNS USER terminal **DEF:249, MGD:181**
 TPNS, running
 See running TPNS
 TPNS/ATP
 See Application Test Processor (TPNS/ATP)
 See TPNS Application Test Processor (TPNS/ATP)
 TPNS/ATP Attention Identifier Definition Menu **SGU:131**
 TPNS/ATP Data Field Options Menu **SGU:132**
 TPNS/ATP Datasets and Parameter Definition Menu **SGU:120**
 TPNS/ATP editor
 adding test data **SGU:130**
 TPNS/ATP editor (*continued*)
 application screen image, displaying **SGU:130**
 ATPSCRDF **SGU:130**
 data field options, displaying **SGU:132**
 deleting test data **SGU:130**
 description **SGU:116**
 help facility **SGU:134**
 invoking **SGU:118, SGU:119**
 keyboard action, selecting **SGU:132**
 menu and panel flow **SGU:134**
 saving test data **SGU:130**
 screen definition **SGU:117**
 screens, choosing version to define **SGU:129**
 testcase definition **SGU:117**
 testcase, selecting **SGU:124**
 TPNS/ATP Editor Selection Menu **SGU:122**
 TPNS/ATP Primary Option Menu **SGU:119**
 TPNS/ATP Screen Selection Menu **SGU:129**
 TPNS/ATP Testcase Definition Menu **SGU:125**
 TPNS/ATP Testcase Definition Panel **SGU:126**
 TPNS/ATP Testcase Selection Menu **SGU:124**
 TPNS/ISPF Interface **PRI:8**
 invoking **GU:5**
 overview **GU:5**
 setting up **GU:5**
 TPNS/ISPF Interface model networks
 CPI-C **SMP:396**
 direct-attached cross-domain **SMP:315**
 Ethernet **SMP:323**
 frame relay **SMP:389**
 FTP **SMP:383**
 full-duplex and half-duplex **SMP:356**
 IBM Token-Ring **SMP:363**
 link-attached cross-domain **SMP:328**
 real host ownership of NCP subarea **SMP:348**
 Simple TCP Client **SMP:386**
 Telnet 3270 **SMP:379**
 Type 2.1 node **SMP:336**
 VTAM application **SMP:371**
 X.25 **SMP:374**
 TPNS/ISPF Interface, installing
 allocating data sets **PI:19**
 ISPF/PDF Primary Option Menu changes **PI:18**
 library setup **PI:19**
 TPNS1050
 See TPNS control program generation macros
 TPNS2741
 See TPNS control program generation macros
 TPNSACAL
 See TPNS control program generation macros
 TPNSALC
 See TPNS control program generation macros
 TPNSBSC1
 See TPNS control program generation macros
 TPNSBSC2
 See TPNS control program generation macros

TPNSBSC3
 See TPNS control program generation macros
 TPNSFDXS
 See TPNS control program generation macros
 TPNSGEN
 See TPNS control program generation macros
 TPNSHDXS
 See TPNS control program generation macros
 TPNSIMM command processor **OP:98**
 TPNSLPRT
 See TPNS control program generation macros
 TPNSMCAL
 See TPNS control program generation macros
 TPNSTWX
 See TPNS control program generation macros
 TPNSTWXL
 See TPNS control program generation macros
 TPREPEAT operand (TP statement) **LR:103**
 TPRINT command **GU:220**
 TPSTATS operand (TP statement) **LR:103**
 TPSTIME operand (TP statement) **LR:104**
 TPTYPE operand (TP statement) **LR:104**
 TPUTNEG operand (MCH statement) **LR:168**
 trace
 BTU **OP:161**
 channel **OP:49**
 control program **OP:49**
 dispatcher **OP:49, OP:161**
 line **OP:49, OP:139**
 PIU **OP:161**
 scanner internal (SIT) **OP:50, OP:152**
 tables **OP:161**
 TRACE command
 Loglist Utility **GU:92**
 Response Time Utility **GU:235**
 trace data records **UE:43**
 trace reports **MGD:205**
 line trace report **PI:109**
 SIT trace report **PI:109**
 using **PI:109**
 tracing a sample STL program
 Loglist Utility output, explanation of **STL:146**
 sample network definition used **STL:145**
 sample printed listing obtained **STL:146**
 sample STL program used **STL:145**
 tracing messages **DEF:274**
 tracing STL exercise **PRI:61**
 tracing STL programs **DEF:274**
 tracing system activity
 where to trace **SGU:75**
 with Generalized Trace Facility (GTF) **SGU:74**
 with NetView Performance Monitor (NPM) **SGU:74**
 with user-written traces **SGU:74**
 with VTAM Buffer Trace **SGU:74**
 TRAN operand (PADPARMS statement) **LR:170**
 TRANS operand (TERM statement) **LR:210**
 TRANSACT control command, ITPTCL **SGU:139**
 TRANSACT control command, ITPXREF **SGU:143**
 transaction processing
 BTRANS and ETRANS commands **GU:173**
 examples **GU:174—176**
 starting **GU:173**
 to compute response time **GU:173**
 transaction program
 See also TP
 instance **PI:121**
 transaction program (TP) **MGD:127**
 defined **PI:37**
 transaction record listing **GU:167**
 transaction type **GU:173**
 transactions
 deciding what to test **PI:70**
 defined **PI:70**
 defining to the Response Time Utility **PI:115**
 mix **PI:71**
 rate of **PI:72**
 TRANSDLY operand (LCH statement) **LR:161**
 TRANSLATE function **STL:376**
 TRANSLATE function (DATASAVE statement) **LR:332**
 TRANSLATE function on the DATASAVE
 statement **MGD:46**
 translating STL input **PRI:5, PRI:9**
 how to **PRI:17**
 translation **PRI:5, PRI:9**
 how to **PRI:17**
 translator
 See STL Translator
 transmission header statement **LR:428**
 transmit (XMIT) records **SGU:40**
 Transmit Interrupt
 conditions for continuing after **STL:78**
 defined **STL:77**
 statements causing **STL:77**
 TRANSMIT statement **PRI:16, STL:75—76, STL:292**
 transmitting messages
 statements used to transmit messages,
 listed **STL:77**
 using AID keys **STL:75**
 using TRANSMIT statement **STL:75**
 trigger field AID **SGU:80**
 TRM control block **UE:79**
 TRUNC command **GU:221**
 TSEQ data field option **LR:445**
 TSEQ operand (A command) **OP:138**
 TSO
 authorizing TPNS under **PI:15**
 entering operator commands under **OP:25**
 operating TPNS under **OP:12**
 running TPNS under **PI:97**
 TSPROF operand (MODEENT statement) **LR:65**

TSW operand
 IF (message generation) statement **LR:414**
 IF (network configuration) statement **LR:458**
 SETSW statement **LR:414**
 TSW operand (A command) **OP:138**
 TWX
 IMS example **SMP:28**
 TWX dial lines **DEF:241**
 TWX support **STL:76, STL:292**
 TWX switched lines **DEF:241**
 TWXREOM operand
 TERM statement **LR:210**
 TEXT statement **LR:426**
 TWXSEOM operand
 MSGTXT statement **LR:388**
 TEXT statement **LR:426**
 TXTDLM operand (MSGTXT statement) **LR:388**
 TXTDLM operand on the MSGTXT statement **MGD:22**
 Type 2.1 link definition **LR:217—221**
 Type 2.1 node model **SMP:336**
 type 2.1 node scripts
 boundary channel attached **SMP:75**
 with dependent LUs **SMP:50**
 with LU Type 6.2 **SMP:52**
 Type 2.1 node simulation **LR:243—244**
 Type 2.1 nodes
 boundary channel-attached Type 2.1 nodes
 defining **DEF:190**
 modifying NCP for **DEF:64**
 physical configuration for **DEF:12**
 TPNS simulation support **DEF:183**
 network definition example **DEF:193—196**
 network definition statements
 ADJNODE/LINE/PU **DEF:185**
 LU **DEF:185**
 MODETAB, MODEENT, MODEEND **DEF:184**
 PSUBVECT **DEF:184**
 PU21 **DEF:184**
 simulating **DEF:181—196**
 TPNS simulation support **DEF:181—183**
 Type 2.1 nodes, statements for simulating **PI:66**
 type codes, simulated resources **GU:239**
 TYPE operand
 DEV (TCP/IP) statement **LR:136**
 DEV (terminal) statement **LR:190**
 FILE statement **LR:44**
 IF (message generation) statement **LR:377**
 IF (network configuration) statement **LR:58**
 MODEENT statement **LR:66**
 RH statement **LR:402**
 TERM statement **LR:210**
 TYPE operand on TCP/IP DEV **DEF:110**
See also TCP/IP, network definition statements
 TYPE statement **PRI:16, STL:53—60, STL:296**
 types
 constant **STL:24**

types (*continued*)
 variable **STL:22**
 types, variable **PRI:43—45**
 typographic conventions **STL:xv**

U

U (Unsolicited RU Transmission) operator
 command **MGD:156, OP:162**
 U operand (A command) **OP:138**
 UASIZE operand
 DEV (TCP/IP) statement **LR:136**
 DEV (terminal) statement **LR:191**
 LU (cross-domain) statement **LR:278**
 LU (type 2.1 node) statement **LR:236**
 LU (VTAMAPPL) statement **LR:121**
 UCD operand (TP statement) **LR:104**
 UCMDEXIT operand
 discussion **UE:11**
 parameter list **UE:11**
 sample routine **UE:12**
 UCMDEXIT operand (NTRWK statement) **LR:78**
 UDIST network definition statement **STL:63**
 UDIST statement **DEF:275, LR:90**
 UNBUFPRRT operand (DEV statement) **LR:191**
 unconditional delimiters
 interrupting message generation with **MGD:56—65**
 QUIESCE **MGD:65**
 STOP **MGD:64**
 WAIT **MGD:56**
 undeclared variables, errors from
 discussion **PRI:44**
 error messages **PRI:44**
 understanding DSPY records
 comparing, multiple devices **GU:102**
 comparing, one device **GU:101**
 understanding STL variables
 classes **PRI:43**
 reserved names **PRI:44**
 types **PRI:43—45**
 UNITSZ operand
 CHANL statement **LR:251**
 LINE (type 2.1 node) statement **LR:221**
 UNLOCK command **GU:222**
 unshared variables **PRI:43, STL:23**
 unsolicited data, generating from SNA physical
 units **MGD:156**
 unsolicited monitor data **PI:144**
 UOM operand
 DEV (TCP/IP) statement **LR:137**
 DEV (terminal) statement **LR:191**
 LU (cross-domain) statement **LR:278**
 LU (type 2.1 node) statement **LR:236**
 LU (VTAMAPPL) statement **LR:121**
 UP operand
 CURSOR statement **LR:328**

UP operand (*continued*)
 SCROLL statement **LR:405**
 UPCASE command **GU:93**
 UPDATE operand (M command) **OP:150**
 UPPERCASE command **GU:159**
 URC operand (CMND statement) **LR:317**
 user areas
 See also save and user areas
 converting DBCS data in a message from **MGD:47**
 generating messages dynamically with **MGD:42—50**
 inserting data into a message from **MGD:48**
 placing data in, with the DATASAVE
 statement **MGD:43**
 querying **OP:48**
 user data table (member) statement **LR:389**
 user data table statement **LR:91**
 user data table, building **LR:91**
 user data tables **DEF:275**
 user delays, defining (ITPLSGEN) **SGU:52**
 user delays, overriding (ITPLSGEN) **SGU:53**
 user exit interface command exit
 discussion **UE:21**
 parameter list **UE:22**
 user exit routines **DEF:276**
 user exit routines, Response Time Utility **GU:178**
 user exit statement **LR:353**
 user exits **STL:39, STL:297**
 user tables
 comparing entries using UTBLSCAN
 function **STL:64**
 declaring **STL:29**
 defining **MGD:34, STL:62**
 defining with UTBL network definition
 statement **MGD:34**
 described **STL:62**
 generating messages dynamically with **MGD:33—38**
 selecting entries in
 with UTBL function **STL:63**
 with UTBLMAX function **STL:63**
 user tables exercise **PRI:79**
 USER terminal
 See TPNS USER terminal
 user time interval **PRI:21**
 User Time Interval statement **LR:92**
 user time intervals
 altering with the A (Alter) operator
 command **MGD:74**
 defined **MGD:69**
 referencing multiple **MGD:72**
 specifying multiple **MGD:72**
 USERAREA operand
 DEV (TCP/IP) statement **LR:137**
 DEV (terminal) statement **LR:191**
 LU (cross-domain) statement **LR:279**
 LU (type 2.1 node) statement **LR:237**
 LU (VTAMAPPL) statement **LR:122**
 USERAREA operand (*continued*)
 TERM statement **LR:211**
 TP statement **LR:105**
 USERDATA operand (CALldata statement) **LR:157**
 USEREXIT statement **STL:39, STL:297**
 using CPI-C verbs **STL:39**
 using Loglist control commands **PRI:64**
 UTBL **DEF:275**
 UTBL data field option **LR:445**
 UTBL distribution statement **LR:90**
 UTBL function **PRI:79, STL:63—63, STL:376—378**
 UTBL network definition statement **MGD:34**
 UTBL operand
 IF (message generation) statement **LR:377**
 IF (network configuration) statement **LR:59**
 UTBL operand on the IF statement **MGD:97**
 UTBL statement **DEF:275, LR:91**
 UTBLCNTR operand
 IF (message generation) statement **LR:378**
 IF (network configuration) statement **LR:59**
 UTBLCNTR operand on the IF statement **MGD:97**
 UTBLMAX function **STL:63, STL:379**
 UTBLMAX operand (SET statement) **LR:413**
 UTBLSCAN function **STL:64, STL:380—380**
 UTBLSEED operand (NTWRK statement) **LR:78**
 UTI network definition operand, using to control inter-
 message delay **STL:79**
 UTI network definition statement **MGD:72**
 UTI operand **PRI:21**
 DELAY statement **LR:341**
 NTWRK statement **LR:78**
 SETUTI statement **LR:416**
 WAIT statement **LR:432**
 UTI operand (A command) **OP:138**
 UTI statement **LR:92, STL:299**
 UTI, setting **LR:416**
 UXOC EXIT operand
 discussion **UE:21**
 parameter list **UE:22**
 UXOC EXIT operand (NTWRK statement) **LR:79**

V

V (Activate/Deactivate Cross-Domain Resources) oper-
 ator command **OP:163**
 VALUE operand (CALC statement) **LR:301**
 variable classes
 declaring shared variables **PRI:44**
 declaring unshared variables **PRI:44**
 shared variables **PRI:43**
 unshared variables **PRI:43**
 variable data
 coding **LR:437**
 in TEXT statement **LR:437**
 list of options **LR:437**

- variable declarations and comments exercise **PRI:43**
- variable dictionary
 - example **STL:136**
 - reading **STL:137**
 - using to find errors **STL:136**
 - using with STL trace facility
 - using @PROGRAM statement **STL:113**
 - using PROGRAM execution parameter **STL:113**
- variable event names, using **MGD:137**
- variable types **PRI:43—45**
- variables
 - classes
 - shared, defined **STL:23**
 - unshared, defined **STL:23**
 - defined **STL:21**
 - naming conventions for **STL:22**
 - reserved
 - in asynchronous conditions **STL:24**
 - use of **STL:24**
 - types
 - bit, defined **STL:22**
 - integer, defined **STL:22**
 - string, defined **STL:22**
- VC statement **DEF:202, LR:171**
- VCTYPE operand (LCH statement) **LR:161**
- Verification Reports
 - counter tests, contents for **GU:40**
 - cursor position tests, contents for **GU:39**
 - data tests, contents for **GU:37**
 - event tests, contents for **GU:38**
 - example of **GU:36**
 - fields of condition, expected value, and actual **GU:36**
 - reading **GU:36**
 - switch tests, contents for **GU:39**
- Verification Summary Reports
 - example of **GU:42**
 - reading **GU:41**
- verification, overriding (ITPLSGEN) **SGU:54**
- VERIFY
 - THEN operand, IF (message generation) statement **LR:376**
 - THEN operand, IF (network configuration) statement **LR:58**
- VERIFY command **GU:94**
- VERIFY command, ITPLSGEN **SGU:53**
- verify data records **UE:43**
- VERIFY statement **STL:100, STL:300**
- VERIFY—(data) for THEN and ELSE on the IF statement **MGD:95**
- verifying scripts, ITPLSGEN **SGU:53**
- verifying simulations without self-checking scripts **MGD:111**
- VIEW operand (M command) **OP:151**
- virtual circuit **DEF:202**
- virtual route selection exit
 - descriptor block format **UE:29**
 - discussion **UE:28**
 - parameter list **UE:28**
- virtual storage
 - estimating **PI:89**
- virtual storage for the Response Time Utility **GU:176**
- VM
 - authorizations, required **PI:10**
 - authorizing TPNS **PI:17**
 - DIAG98 option **PI:17**
 - entering operator commands under **OP:26**
 - generating TPNS control program on **DEF:28**
 - logging messages on separate virtual machine **OP:77—80**
 - operating TPNS under **OP:14**
 - real input/output (I/O) requirements **PI:10**
 - return codes **OP:26**
 - running TPNS on **PI:98**
 - TCP/IP requirements **PI:11**
 - TPNS requirements **PI:10**
 - VTAM requirements **PI:11**
- VM files **PI:133**
- VMID operand (NTWRKLOG statement) **LR:80**
- VMODULE operand (MCH statement) **LR:168**
- VR operand
 - COS statement **LR:255**
 - PATH statement **LR:282**
- VREXIT operand
 - descriptor block format **UE:29**
 - discussion **UE:28**
 - parameter list **UE:28**
- VREXIT operand (NTWRK statement) **LR:79**
- VRPOOL operand
 - CNTLR statement **LR:254**
 - SSCP statement **LR:289**
- VRPWS operand (PATH statement) **LR:282**
- VSAM data set, setting up **SGU:105**
- VTAM
 - application program password, defining **LR:123**
 - application program sample, description of **SMP:3**
 - application program, defining symbolic name **LR:123**
 - application sample **SMP:4**
 - applications
 - See VTAM applications
 - logical unit half-sessions, defining **LR:109**
 - VTAM Encrypt/Decrypt program
 - See encrypting/decrypting messages
 - VTAM API (VTAM Application Program Interface)
 - See VTAM applications, Application Program Interface
 - VTAM application model **SMP:371**
 - VTAM Application Program Interface (VTAM API)
 - See VTAM applications, Application Program Interface

- VTAM application simulation
 - physical configuration for **PI:44**
- VTAM applications
 - Application Program Interface
 - simulating LUs accessing **DEF:67**
 - TPNS using **DEF:67**
 - CICS/VS definitions, coordinating with TPNS, VTAM definitions **DEF:72**
 - coding considerations
 - for TPNS VTAMAPPL **DEF:70**
 - for VTAM APPL **DEF:70**
 - coordinating TPNS, VTAM, and subsystem definitions **DEF:71**
 - defining resources **DEF:68—73**
 - IMS/VS definitions, coordinating with TPNS, VTAM definitions **DEF:73**
 - index counters for VTAMAPPL and LU **DEF:258**
 - initiating sessions
 - by TPNS **DEF:81**
 - by VTAM **DEF:82**
 - network definition example **DEF:73—81**
 - network definition statements
 - LU **DEF:69**
 - VTAMAPPL **DEF:69**
 - sequence counters for VTAMAPPL and LU **DEF:258**
 - simulating **DEF:67**
 - simulating LUs accessing **DEF:67**
 - terminating sessions
 - by TPNS **DEF:81**
 - by VTAM **DEF:82**
 - TPNS simulation support **DEF:67—68**
- VTAM Buffer Trace **SGU:74**
- VTAM Encrypt/Decrypt program **DEF:291**
 - See also* encrypting/decrypting messages
- VTAMAPID function **STL:381**
- VTAMAPPL **PRI:5**
 - canceling **OP:140**
 - JCL for **OP:11**
 - query network for **OP:33**
 - starting activity for **OP:160**
 - stopping activity for **OP:155**
- VTAMAPPL command
 - Log Compare Utility **GU:145**
 - Loglist Utility **GU:95**
 - Response Time Utility **GU:223**
- VTAMAPPL configuration
 - See also* physical configuration, types
 - communication controller for **PI:12, PI:44**
 - network definition statements for **PI:58**
 - requirements **PI:10**
 - use of **PI:41**
 - using **PI:44, PI:45**
- VTAMAPPL logical unit definition statement **105, LR:109—122**

- VTAMAPPL simulation **PRI:19**
 - time stamping messages **GU:243**
- VTAMAPPL simulation statement **LR:123**
- VTAMAPPL statement **DEF:69, LR:123—124, PI:59, PRI:21**
- VTAMAPPLID data field option **LR:447**
- VWINDOW operand (LCH statement) **LR:162**

W

- W (RSTATS Query) operator command **OP:165**
- wait for response statement **LR:430—432**
- WAIT indicator **MGD:61**
 - preserving over asynchronous IF statements **MGD:62**
- WAIT statement **LR:430—432, STL:301—302**
 - coding **MGD:23**
 - operands
 - EVENT **MGD:57**
 - TIME **MGD:58**
 - UTI **MGD:60**
 - using **MGD:56**
- WAIT UNTIL POSTED
 - defined **STL:93**
 - example of **STL:93**
 - using with ONIN and ONOUT **STL:94**
- WAIT UNTIL SIGNALLED
 - defined **STL:96**
- WAIT UNTIL statement
 - defined **STL:80**
 - used to define asynchronous condition **STL:82**
 - using **STL:89**
- WAIT/POST facility **MGD:138**
- Western Union Model 33/35 TWX Terminal
 - generating messages for **MGD:173**
 - logic testing **DEF:242, MGD:174**
 - network definition example **DEF:242**
 - simulating **DEF:241—242, MGD:173**
 - with different line types **DEF:241**
- WHEN operand
 - IF (message generation) statement **LR:378**
 - IF (network configuration) statement **LR:60**
- WHEN operand on the IF statement **MGD:88**
- work variable **STL:137**
- WORKSET operand (MSGDISK statement) **LR:69**
- worksheets
 - TPNS Primer Information List **PRI:95**
 - TPNS System Requirements Checklist **PRI:94**
- WRITE command **GU:230**
 - write data to console statement **LR:433**
 - write data to log statement **LR:383—384**
 - writing data to operator console **LR:433**
 - writing Verify records **STL:100**
- WTO statement **LR:433—433, MGD:122**

X

X (X.25 Command) operator command **OP:166**
X operand (A command) **OP:138**
X subcommand (F command) **OP:144**
X.25 diagnostic fields **DEF:308—310**
X.25 logical channel definition statement **LR:158—162**
X.25 model **SMP:374**
X.25 multichannel link definition statement **LR:163—168**
X.25 networks
 message logging **GU:249**
X.25 packet switching
 TPNS support for network resources **PI:138**
 TPNS support of line discipline **PI:139**
X.25 packet types **DEF:307**
X.25 packet-switched network (PSN)
 adding X.25 resources to TPNS networks **DEF:202**
 data circuit-terminating equipment (DCE) **DEF:197**
 data terminal equipment (DTE) **DEF:197**
 DCE (data circuit-terminating equipment) **DEF:197**
 defining X.25 network configurations
 SNA subareas adjacent to real X.25 PSN **DEF:210**
 X.25 PSN and adjacent SNA subareas **DEF:209**
 X.25 PSN and terminals **DEF:205**
 X.25 PSN passthrough network between real SNA subareas and system under test **DEF:210**
 X.25 PSN passthrough network between real terminals and system under test **DEF:206**
 X.25 resources converted from non-X.25 **219**
 X.25 terminals **DEF:207**
 defining X.25 network resources **DEF:199—203**
 diagnostic fields **DEF:308—310**
 DTE (data terminal equipment) **DEF:197**
 network definition examples
 X.25 resources converted from non-X.25 **219**
 X.25 resources in same domain as system under test **DEF:203—207**
 X.25 resources with adjacent SNA subareas **DEF:208—211**
 network definition statements
 CALLDATA **DEF:199**
 LCH **DEF:200**
 MCH **DEF:199**
 PADPARMS **DEF:199**
 VC **DEF:202**
 options
 not supported by TPNS **DEF:221**
 supported by TPNS **DEF:219**
 packet types **DEF:307**
 packets and diagnostics **DEF:307—310**
 PAD (TWX packet assembly/disassembly) **DEF:222**
 protocols above X.25 **DEF:221**
 simulating **DEF:197—222**
 special considerations **DEF:198**

X.25 packet-switched network (PSN) (*continued*)

 TPNS control program for **DEF:222**
 TPNS simulation support **DEF:197—198**
 TPNS USER terminal **DEF:222**
 TWX assembly/disassembly (PAD) **DEF:198, DEF:222**
X.25 packet-switched networks, statements for simulating **PI:66**
X.25 Resource End of Run Report **OP:68**
X.25 Resource Interval Report **OP:64**
X.25 resource records **GU:33**
X.25 resources, controlling **OP:50**
X.25 simulation
 defining logical channel for **LR:158**
 defining multichannel link **LR:163**
 logical channel, defining single-domain line for **LR:192**
X.25 virtual circuit definition statement **LR:171**
X3PARMS operand (PADPARMS statement) **LR:170**
XID operand
 PU (type 2.1 node) statement **LR:241**
 TERM statement **LR:211**
XIDRSLV operand (LLS statement) **LR:145, LR:154**
XPRTY operand (LLS statement) **LR:145**

Y

Y (Communication Controller Control) operator command **OP:168**
YEAR data field option **LR:447**
YEAR function **STL:382**

Z

Z (Closedown) operator command **OP:169**
ZEND command **PRI:17, PRI:37**
ZEND operator command **MGD:121**

Communicating Your Comments to IBM

TPNS
Teleprocessing Network Simulator
Master Index
Version 3 Release 5
Publication No. GC31-6059-03

If you especially like or dislike anything about this book, please use one of the methods listed below to send your comments to IBM. Whichever method you choose, make sure you send your name, address, and telephone number if you would like a reply.

Feel free to comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. However, the comments you send should pertain to only the information in this manual and the way in which the information is presented. To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

If you are mailing a readers' comment form (RCF) from a country other than the United States, you can give the RCF to the local IBM branch office or IBM representative for postage-paid mailing.

- If you prefer to send comments by mail, use the RCF at the back of this book.
- If you prefer to send comments by FAX, use this number:
United States and Canada: **1-800-338-2714**
- If you prefer to send comments electronically, use this network ID:
 - IBM Mail Exchange: **USIB1LTD at IBMMAIL**
 - IBMLink: **TPNS at RALVM7**
 - Internet: **TPNS@VNET.IBM.COM**

Make sure to include the following in your note:

- Title and publication number of this book
- Page number or topic to which your comment applies.

Help us help you!

TPNS
Teleprocessing Network Simulator
Master Index
Version 3 Release 5
Publication No. GC31-6059-03

We hope you find this publication useful, readable and technically accurate, but only you can tell us! Your comments and suggestions will help us improve our technical publications. Please take a few minutes to let us know what you think by completing this form.

Overall, how satisfied are you with the information in this book?	Satisfied	Dissatisfied
	☐	☐

How satisfied are you that the information in this book is:	Satisfied	Dissatisfied
Accurate	☐	☐
Complete	☐	☐
Easy to find	☐	☐
Easy to understand	☐	☐
Well organized	☐	☐
Applicable to your task	☐	☐

Specific Comments or Problems:

Please tell us how we can improve this book:

Thank you for your response. When you send information to IBM, you grant IBM the right to use or distribute the information without incurring any obligation to you. You of course retain the right to use the information in any way you choose.

Your Internet Address: _____

_____ Name	_____ Address
---------------	------------------

_____ Company or Organization	_____
----------------------------------	-------

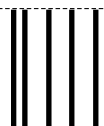
_____ Phone No.	_____
--------------------	-------



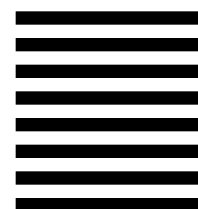
Fold and Tape

Please do not staple

Fold and Tape



NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES



BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

TPNS Development and Support
Department LWZA
International Business Machines Corporation
PO BOX 12195
RESEARCH TRIANGLE PARK NC 27709-9990



Fold and Tape

Please do not staple

Fold and Tape



File Number: S370/4300/30XX-50
Program Number: 5688-121



Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.



GC31-6059-03

