

z/VM
7.3

*RSCS Networking
Planning and Configuration*



Note:

Before you use this information and the product it supports, read the information in [“Notices” on page 315.](#)

This edition applies to version 7, release 3 of IBM® z/VM® (product number 5741-A09) and to all subsequent releases and modifications until otherwise indicated in new editions.

Last updated: 2022-09-05

© **Copyright International Business Machines Corporation 1990, 2022.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

- Figures..... xi**

- Tables.....xv**

- About This Document..... xvii**
 - Intended Audience..... xvii
 - Syntax, Message, and Response Conventions..... xvii
 - Where to Find More Information.....xx
 - Links to Other Documents and Websites..... xx

- How to Send Your Comments to IBM.....xxi**

- Summary of Changes for z/VM: RSCS Networking Planning and Configuration.... xxiii**
 - SC24-6320-73, z/VM 7.3 (September 2022)..... xxiii
 - SC24-6320-01, z/VM 7.2 (July 2021)..... xxiii
 - SC24-6320-01, z/VM 7.2 (September 2020)..... xxiii
 - SC24-6320-00, z/VM 7.1 (September 2018)..... xxiii

- Part 1. Introduction..... 1**
 - Chapter 1. Introducing RSCS Networking..... 3
 - How RSCS Fits into Your z/VM System..... 3
 - How RSCS Manages Files.....4
 - Supported Nodes.....5
 - Types of Nodes.....6
 - Supported Link Types.....7
 - Non-SNA Connections.....7
 - SNA Connections.....8
 - TCP/IP Connections.....10
 - RSCS Configuration File.....11
 - RSCS Dynamic Authorization.....11
 - How RSCS Transfers Files.....11
 - List Processor Links.....12
 - Networking Links.....12
 - Notify Links.....13
 - Printer Links.....13
 - UFT Links.....13
 - Workstation Links.....13
 - RSCS Data Interchange Manager.....13
 - Summary of RSCS Communications.....13
 - Chapter 2. RSCS Environments..... 17
 - Single-system Environment.....17
 - 3270 Printers.....17
 - ASCII Devices.....17
 - Sharing Printers.....17
 - Sharing 3270 Printers.....18
 - Submitting Jobs from Remote Locations.....18
 - Advantages of RSCS in a Single-System Environment..... 19

Multisystem Environment.....	19
Data Exchanges.....	20
Sending Jobs to Other Places for Processing.....	21
Sending Output to Other Places.....	21
Advantages of RSCS in a Multisystem Environment.....	22
Chapter 3. RSCS Users.....	23
General Users.....	23
Users Authorized through RSCS Dynamic Authorization.....	23
Operators.....	23
Chapter 4. RSCS Data Interchange Manager.....	25
Introduction to RSCS Interchange.....	25
How RSCS Interchange Works.....	26
How Your Mail Is Sent to SMTP Users.....	27
Looking at z/VM-to-Network (SMTP) Mail.....	27
Looking at SMTP-to-z/VM Mail.....	28
Part 2. Planning and Configuration Guide.....	29
Chapter 5. System Planning for RSCS.....	31
Planning Checklist for System Configuration.....	31
VTAM Requirement.....	31
Preparing z/VM for RSCS.....	31
RSCS Virtual Machine.....	31
Identifying RSCS to z/VM.....	33
RACF Toleration.....	33
Joining the GCS Virtual Machine Group.....	34
Logging on the RSCS Virtual Machine Automatically.....	35
Preparing for Non-SNA Connections.....	36
Defining Non-SNA Connections.....	36
Requirements for Non-SNA Connections.....	37
Preparing for TCP/IP Connections.....	38
Defining TCP/IP Connections.....	38
Preparing for Networking Links.....	43
Requirements for Non-SNA Networking Links.....	43
Requirements for SNA Networking Links.....	44
Preparing VTAM for SNA Connections.....	44
SNA 3270 Printer Support.....	45
Cross-Domain Support.....	48
Definitions for SNA Networking Links.....	50
Virtual Storage Considerations.....	50
Calculating Virtual Storage.....	50
Defining the RSCS Virtual Storage Size.....	51
Considerations When Defining or Starting RSCS Links.....	51
Performance Considerations.....	52
Common Problems.....	52
z/VM Resource Considerations.....	52
Chapter 6. Planning for RSCS Configuration.....	55
Planning Checklist for RSCS Configuration.....	55
RSCS Configuration File.....	56
Understanding Identifier Naming Conventions.....	57
Other Conventions.....	59
Comments.....	59
AUTH Statement.....	60
CHANNELS Statement.....	60

DEST Statement.....	60
DUMP Statement.....	60
EXIT Statement.....	60
FCB Statement.....	61
FORM Statement.....	61
HIDECHARACTER Statement.....	61
IMBED Statement.....	62
ITRACE Statement.....	62
LANGUAGE Statement.....	62
LINKDEFINE Statement.....	62
LINKTYPE Statement.....	63
LOCAL Statement.....	64
MSGNOH Statement.....	65
OPFORM Statement.....	66
OPTION Statement.....	66
PARAM Statement.....	66
PORT Statement.....	67
RECOVERY Statement.....	67
REROUTE Statement.....	67
RETRY Statement.....	68
ROUTE Statement.....	68
SAFCLASS Statement.....	69
SETMSG Statement.....	70
SHADOWS Statement.....	70
SLOWDOWN Statement.....	70
TOLERANCE Statement.....	71
TRACEDEST Statement.....	71
UPARM Statement.....	71
RSCS Message File.....	71
Customizing RSCS.....	72
RSCS Exit Facility.....	72
Transmission Algorithms.....	72
ASCII Exits.....	72
Gateway Programming Interface.....	72
LPR Exits.....	72
LPD Exits.....	73
UFT Exits.....	73
UFTD Exits.....	73
Chapter 7. Planning for RSCS Operation.....	75
Planning Checklist for RSCS Operation.....	75
Operating the RSCS Virtual Machine.....	75
Operating Links.....	76
Initiating SNA Sessions.....	76
Preparing to Automate Operations with Execs.....	77
Automating Initialization.....	77
Automating Ongoing Operation.....	77
Shutting Down RSCS.....	78
Preparing to Automate Operations Using the RSCS Event Scheduler.....	78
Chapter 8. Planning for RSCS Administration.....	81
Planning Checklist for RSCS Administration.....	81
Tending to Remote Workstations and Printers.....	81
Preparing Network Information.....	82
Manipulating Files in Transit.....	82
Simplifying Jobs.....	82
Chapter 9. Planning for RSCS Diagnosis.....	83

Planning Checklist for RSCS Diagnosis.....	83
Diagnosing Problems Using RSCS Facilities.....	83
Using Tracing Facilities.....	83
Producing Dumps.....	84
Displaying Dump Information.....	84
Establishing Information-Collecting Procedures.....	84
Chapter 10. Configuring RSCS.....	85
Sample Files for Configuring RSCS.....	85
Setting Up the RSCS Configuration File.....	86
Defining Links and Routes.....	86
Using Second-Level Addressing to Avoid Routes.....	89
Coding Statements in the Configuration File.....	90
Setting Up the RSCS Events File.....	93
Creating an Events File.....	93
Automating Simple Events.....	95
Automating More Complex Tasks.....	96
Defining Holidays and Other Special Days.....	97
Displaying Events.....	97
Setting Up NOTIFY-Type Links.....	97
Defining a NOTIFY-Type Link.....	98
Defining TEMPLATE Files.....	98
Specifying Destination Identifiers.....	100
Notes for 3812, 3816, and 4224 Printers.....	101
RSCS Message Repositories.....	101
Using Exit Routines to Customize RSCS.....	101
Optional RSCS Servers.....	103
Chapter 11. Defining the RSCS Network Structure.....	105
Defining a Network with a Single System Node.....	105
Creating a Printer Pool.....	106
Defining a Network with Multiple System Nodes.....	106
Adding Another Link to a Node.....	107
Making Transitions in Links.....	107
Creating Multiple Paths.....	108
Creating an Alternate Path.....	108
Using Second-Level Routing to a Printer Pool.....	108
Defining a Large Network.....	108
Creating a Routing Group.....	109
Creating a Routing Group Within a Routing Group.....	109
Specifying Fanout Links.....	110
Chapter 12. Setting Up the RSCS Domain Name Server.....	113
RSCS Domain Name Server Virtual Machine.....	113
Running GETHOSTC.....	114
Chapter 13. Setting Up RSCS Dynamic Authorization.....	115
Planning Checklist for RSCS Dynamic Authorization.....	115
RSCS Dynamic Authorization Sample Files.....	115
RSCS Dynamic Authorization Server Virtual Machine.....	115
Preparing RSCS for RSCS Dynamic Authorization.....	116
Configuring RSCS Dynamic Authorization.....	116
Authorizing Locally Defined RSCS Commands.....	117
Chapter 14. Setting Up the RSCS Data Interchange Manager.....	119
Planning Checklist for RSCS Interchange.....	119
RSCS Interchange Sample Files.....	119
Planning for RSCS Interchange Exits.....	119

RSCS Interchange Server Virtual Machine.....	119
Preparing RSCS for RSCS Interchange.....	121
RSCS EXIT Statements for RSCS Interchange.....	121
RSCS IMBED Statement for RSCS Interchange.....	121
RSCS LINKTYPE Statement for RSCS Interchange.....	122
RSCS LINKDEFINE Statement for RSCS Interchange.....	122
RSCS PARM Statement for RSCS Interchange.....	122
Modifying the NOTHERE TEMPLATE.....	122
Preparing RACF for RSCS Interchange.....	123
Preparing SMTP for RSCS Interchange.....	123
Configuring RSCS Interchange.....	123
Defining Connections to SMTP and RSCS.....	123
Selecting Domain Names.....	124

Part 3. Reference.....127

Chapter 15. RSCS Configuration File Statements.....	129
Rules for Coding the RSCS Configuration File.....	129
Summary of RSCS Configuration File Statements.....	129
Comment.....	131
AUTH Statement.....	132
CHANNELS Statement.....	134
DEST Statement.....	135
DUMP Statement.....	136
EXIT Statement.....	137
FCB Statement.....	139
FORM Statement.....	140
HIDECHARACTER Statement.....	142
IMBED Statement.....	143
ITRACE Statement.....	145
LANGUAGE Statement.....	150
LINKDEFINE Statement.....	151
LINKTYPE Statement.....	158
LOCAL Statement.....	160
MSGNOH Statement.....	161
OPFORM Statement.....	162
OPTION Statement.....	163
PARM Statement.....	168
PARM Statement for an ASCII-Type Link.....	169
PARM Statement for a GATEWAY-Type Link.....	172
PARM Statement for a LISTPROC-Type Link.....	177
PARM Statement for an LPD-Type Link.....	181
PARM Statement for an LPR-Type Link.....	183
PARM Statement for an MRJE-Type Link.....	188
PARM Statement for an NJE-Type Link.....	193
PARM Statement for a NOTIFY-Type Link.....	201
PARM Statement for an RJE-Type Link.....	203
PARM Statement for an SNANJE-Type Link.....	208
PARM Statement for an SNARJE-Type Link.....	214
PARM Statement for an SNA3270P-Type Link.....	217
PARM Statement for a TCPASCII-Type Link.....	224
PARM Statement for a TCPNJE-Type Link.....	227
PARM Statement for a TN3270E-Type Link.....	234
PARM Statement for a UFT-Type Link.....	240
PARM Statement for a UFTD-Type Link.....	243
PARM Statement for a 3270P-Type Link.....	245
PORT Statement.....	251

RECOVERY Statement.....	252
REROUTE Statement.....	253
RETRY Statement.....	257
ROUTE Statement.....	258
SAFCLASS Statement.....	261
SETMSG Statement.....	262
SHADOWS Statement.....	264
SLOWDOWN Statement.....	265
TOLERANCE Statement.....	266
TRACEDEST Statement.....	267
UPARM Statement.....	268
Chapter 16. RSCS Dynamic Authorization Configuration File Statements.....	269
Rules for Coding the RSCS Dynamic Authorization Configuration File.....	269
AUTHUSER Statement.....	270
FILEMODE Statement.....	271
KEEP Statement.....	272
LOG Statement.....	273
RSCS Statement.....	274
TELL Statement.....	275
TRACERSCSAUTH Statement.....	276
TRANSPPOOL Statement.....	277
Chapter 17. RSCS Data Interchange Manager Configuration File Statements.....	279
Rules for Coding the RSCS Interchange Configuration File.....	279
ADMIN Statement.....	280
AUTHORIZE Statement.....	281
DOMAIN Statement.....	282
EXIT Statement.....	283
FROM_FORMAT Statement.....	284
INFORM_USER Statement.....	285
LOG_DISK Statement.....	286
LOG_KEEP Statement.....	287
MSGNOH Statement.....	288
NICK_OVERRIDE Statement.....	289
NICKPRIV Statement.....	290
NICKTABLE Statement.....	291
RSCSLINK Statement.....	292
RSCSNAME Statement.....	293
SMTPNAME Statement.....	294
TEMP_KEEP Statement.....	295
Appendix A. Sample RSCS Configuration File.....	297
Appendix B. Logon Mode Table Entries.....	307
Logical Unit Sessions for Printer Links.....	308
Appendix C. Sample RSCS Dynamic Authorization Configuration File.....	311
Appendix D. Sample RSCS Data Interchange Manager Configuration File.....	313
Notices.....	315
Programming Interface Information.....	316
Trademarks.....	316
Terms and Conditions for Product Documentation.....	317
IBM Online Privacy Statement.....	317

Bibliography	319
Where to Get z/VM Information.....	319
z/VM Base Library.....	319
z/VM Facilities and Features.....	320
Prerequisite Products.....	322
Related Products.....	322
Additional Publications.....	323
 Index	 325

Figures

1. RSCS Transferring Spool Data over Telecommunications Lines.....	3
2. z/VM Use of the System Spool.....	4
3. RSCS Use of the System Spool.....	4
4. RSCS Spool Files Contain Tag Information.....	5
5. A Simple RSCS Network.....	5
6. Node Characteristics in a Simple Network.....	6
7. RSCS Communication across Non-SNA Links in a Network.....	8
8. RSCS Communication across SNA Links in a Network.....	9
9. RSCS Communication across TCP/IP Connections.....	10
10. Through RSCS, Several Users Can Share a Single 3270 Printer.....	18
11. RSCS Support in a Single-System Network.....	19
12. RSCS Allows Users on Different Systems to Exchange Messages and Files.....	21
13. Users Can Send Jobs to Other Systems for Processing.....	21
14. Users Can Send Output Files to Other Systems for Printing.....	22
15. Sending Mail through RSCS Interchange.....	25
16. RSCS Interchange in the NJE and IP Network Environments.....	26
17. Sending a z/VM Note to an SMTP User.....	27
18. Example of z/VM Mail Received by SMTP.....	27
19. Example of SMTP Mail Sent to z/VM through RSCS Interchange.....	28
20. Example of SMTP Mail Sent to z/VM Without Using RSCS Interchange.....	28
21. Sample RSCS Virtual Machine Definition.....	32
22. RSCS Entry in the SYSTEM NETID File.....	33
23. A Typical Virtual Machine Group.....	35

24. RSCS Entry in the PROFILE GCS Exec for the Recovery Machine.....	36
25. Defining Devices and Connections – Sample SYSTEM CONFIG.....	36
26. Examples of 37xx TCU Statements for Devices and Connections Used by RSCS.....	37
27. Requirements for Non-SNA Printers.....	37
28. Requirements for Non-SNA Remote Workstations.....	38
29. Requirements for TCP/IP Connections.....	39
30. Requirements for Non-SNA Networking Links (BSC).....	44
31. Requirements for Non-SNA Networking Links (CTCA or 3800).....	44
32. Requirements for SNA Networking Links (SNANJE-Type).....	44
33. Requirements for SNA Printers.....	47
34. Formula for Calculating Storage.....	51
35. Links for Direct Communication.....	64
36. Routes for Indirect Communication.....	69
37. A Simple Two-System Network.....	87
38. A Three-System Network as Defined to RSCS.....	88
39. A Three-System Network Defined to RSCS with TCP/IP Drivers.....	88
40. Alternative Paths in a NJE Network as Defined to RSCS.....	89
41. Sample Statements to Define a NOTIFY-Type Link.....	98
42. Sample UNKNOWN TEMPSAMP File.....	100
43. Sample Network with a Single System Node.....	105
44. Sample Network with Multiple System Nodes.....	107
45. Sample Large Network.....	109
46. Physical Network Connection.....	110
47. Sample RSCS Domain Name Server (RSCSDNS) Virtual Machine Definition.....	113
48. Sample RSCS Dynamic Authorization Server (RSCSAUTH) Virtual Machine Definition.....	115

49. Sample RSCS Interchange Server (XCHANGE) Virtual Machine Definition.....	120
50. An RSCS Configuration File With and Without Imbedded Files.....	144
51. System Nodes in a Sample Network.....	297
52. Sample Configuration File for Node NEWYORK.....	298
53. RU Buffer Size Specification Values.....	308

Tables

1. Examples of Syntax Diagram Conventions.....	xvii
2. RSCS Link Drivers and Associated Nodes.....	12
3. Summary of RSCS Link Types and Their Roles.....	14
4. Defining RSCS as a VTAM Application.....	45
5. Defining a Local SNA 3270 Printer to VTAM.....	46
6. Defining a Local Non-SNA 3270 Printer to VTAM.....	46
7. Defining a Local SNA RJE Workstation to VTAM.....	48
8. TEXT File Containing Logon Mode Table for RSCS.....	49
9. Categories of Configuration File Statements.....	57
10. TYPE Operand Specifications.....	87
11. Summary of Configuration File Statements.....	90
12. Number and Sequence of Statements.....	91
13. Variables Supported in NOTIFY Message Template.....	98
14. File Changes Required for RSCS Interchange.....	124
15. Summary of Configuration File Statements.....	129
16. How Non-List Processor Systems Handle List Processor Files.....	165
17. How Non-List Processor Systems Handle List Processor Files.....	173
18. How Non-List Processor Systems Handle List Processor Files.....	195
19. How Non-List Processor Systems Handle List Processor Files.....	210
20. Valid Combinations of FEATURE and TRANS Settings.....	222
21. How Non-List Processor Systems Handle List Processor Files.....	230
22. Valid Combinations of FEATURE and TRANS Settings.....	239
23. Valid Combinations of TRANS and FEATURE Settings.....	250

About This Document

This document describes tasks for planning and configuring IBM Remote Spooling Communications Subsystem (RSCS) Networking for z/VM. It also provides reference information on the configuration statements for the various types of RSCS servers.

Intended Audience

This information is primarily intended for personnel responsible for setting up an RSCS network on a z/VM system.

You should be familiar with z/VM, TCP/IP, IBM Systems Network Architecture (SNA), and 3270 architecture concepts and terminology. These subjects are not discussed in detail in this document, except when it is essential to the configuration of RSCS.

Syntax, Message, and Response Conventions

The following topics provide information on the conventions used in syntax diagrams and in examples of messages and responses.

How to Read Syntax Diagrams

Special diagrams (often called *railroad tracks*) are used to show the syntax of external interfaces.

To read a syntax diagram, follow the path of the line. Read from left to right and top to bottom.

- The \blacktriangleright — symbol indicates the beginning of the syntax diagram.
- The — \blacktriangleright symbol, at the end of a line, indicates that the syntax diagram is continued on the next line.
- The \blacktriangleright — symbol, at the beginning of a line, indicates that the syntax diagram is continued from the previous line.
- The — \blacktriangleright \blacktriangleleft symbol indicates the end of the syntax diagram.

Within the syntax diagram, items on the line are required, items below the line are optional, and items above the line are defaults. See the examples in [Table 1 on page xvii](#).

<i>Table 1. Examples of Syntax Diagram Conventions</i>	
Syntax Diagram Convention	Example
Keywords and Constants A keyword or constant appears in uppercase letters. In this example, you must specify the item KEYWORD as shown. In most cases, you can specify a keyword or constant in uppercase letters, lowercase letters, or any combination. However, some applications may have additional conventions for using all-uppercase or all-lowercase.	\blacktriangleright KEYWORD \blacktriangleleft

Table 1. Examples of Syntax Diagram Conventions (continued)

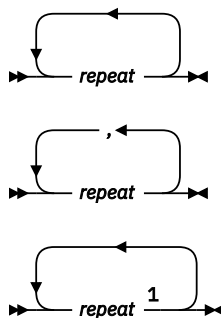
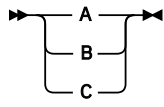
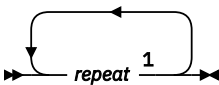
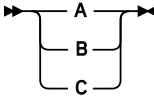
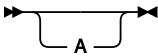
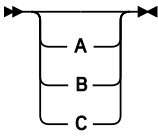
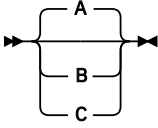
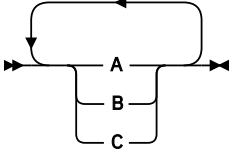

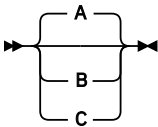
Syntax Diagram Convention	Example
<p>Abbreviations</p> <p>Uppercase letters denote the shortest acceptable abbreviation of an item, and lowercase letters denote the part that can be omitted. If an item appears entirely in uppercase letters, it cannot be abbreviated.</p> <p>In this example, you can specify KEYWO, KEYWOR, or KEYWORD.</p>	<p>▶▶ KEYWOrd ▶▶</p>
<p>Symbols</p> <p>You must specify these symbols exactly as they appear in the syntax diagram.</p>	<p>* Asterisk</p> <p>:</p> <p>Colon</p> <p>,</p> <p>Comma</p> <p>= Equal Sign</p> <p>- Hyphen</p> <p>() Parentheses</p> <p>.</p> <p>Period</p>
<p>Variables</p> <p>A variable appears in highlighted lowercase, usually italics.</p> <p>In this example, <i>var_name</i> represents a variable that you must specify following KEYWORD.</p>	<p>▶▶ KEYWOrd — <i>var_name</i> ▶▶</p>
<p>Repetitions</p> <p>An arrow returning to the left means that the item can be repeated.</p> <p>A character within the arrow means that you must separate each repetition of the item with that character.</p> <p>A number (1) by the arrow references a syntax note at the bottom of the diagram. The syntax note tells you how many times the item can be repeated.</p> <p>Syntax notes may also be used to explain other special aspects of the syntax.</p>	<p></p> <p></p> <p></p> <p>Notes:</p> <p>¹ Specify <i>repeat</i> up to 5 times.</p>
<p>Required Item or Choice</p> <p>When an item is on the line, it is required. In this example, you must specify A.</p> <p>When two or more items are in a stack and one of them is on the line, you must specify one item. In this example, you must choose A, B, or C.</p>	<p>▶▶ A ▶▶</p> <p></p>

Table 1. Examples of Syntax Diagram Conventions (continued)

Syntax Diagram Convention	Example
<p>Optional Item or Choice</p> <p>When an item is below the line, it is optional. In this example, you can choose A or nothing at all.</p> <p>When two or more items are in a stack below the line, all of them are optional. In this example, you can choose A, B, C, or nothing at all.</p>	 
<p>Defaults</p> <p>When an item is above the line, it is the default. The system will use the default unless you override it. You can override the default by specifying an option from the stack below the line.</p> <p>In this example, A is the default. You can override A by choosing B or C.</p>	
<p>Repeatable Choice</p> <p>A stack of items followed by an arrow returning to the left means that you can select more than one item or, in some cases, repeat a single item.</p> <p>In this example, you can choose any combination of A, B, or C.</p>	
<p>Syntax Fragment</p> <p>Some diagrams, because of their length, must fragment the syntax. The fragment name appears between vertical bars in the diagram. The expanded fragment appears in the diagram after a heading with the same fragment name.</p> <p>In this example, the fragment is named "A Fragment."</p>	 <p>A Fragment</p> 

Examples of Messages and Responses

Although most examples of messages and responses are shown exactly as they would appear, some content might depend on the specific situation. The following notation is used to show variable, optional, or alternative content:

xxx

Highlighted text (usually italics) indicates a variable that represents the data that will be displayed.

[]

Brackets enclose optional text that might be displayed.

{ }

Braces enclose alternative versions of text, one of which will be displayed.

|

The vertical bar separates items within brackets or braces.

...

The ellipsis indicates that the preceding item might be repeated. A vertical ellipsis indicates that the preceding line, or a variation of that line, might be repeated.

Where to Find More Information

For a list of documents that provide additional information on RSCS and other z/VM topics, see [“Bibliography” on page 319](#).

Links to Other Documents and Websites

The PDF version of this document contains links to other documents and websites. A link from this document to another document works only when both documents are in the same directory or database, and a link to a website works only if you have access to the Internet. A document link is to a specific edition. If a new edition of a linked document has been published since the publication of this document, the linked document might not be the latest edition.

How to Send Your Comments to IBM

We appreciate your input on this publication. Feel free to comment on the clarity, accuracy, and completeness of the information or give us any other feedback that you might have.

To send us your comments, go to [z/VM Reader's Comment Form \(https://www.ibm.com/systems/campaignmail/z/zvm/zvm-comments\)](https://www.ibm.com/systems/campaignmail/z/zvm/zvm-comments) and complete the form.

If You Have a Technical Problem

Do not use the feedback method. Instead, do one of the following:

- Contact your IBM service representative.
- Contact IBM technical support.
- See [IBM: z/VM Support Resources \(https://www.ibm.com/vm/service\)](https://www.ibm.com/vm/service).
- Go to [IBM Support Portal \(https://www.ibm.com/support/entry/portal/Overview\)](https://www.ibm.com/support/entry/portal/Overview).

Summary of Changes for z/VM: RSCS Networking Planning and Configuration

This information includes terminology, maintenance, and editorial changes. Technical changes or additions to the text and illustrations for the current edition are indicated by a vertical line (|) to the left of the change.

SC24-6320-73, z/VM 7.3 (September 2022)

This edition supports the general availability of z/VM 7.3. Note that the publication number suffix (-73) indicates the z/VM release to which this edition applies.

SC24-6320-01, z/VM 7.2 (July 2021)

This edition includes terminology, maintenance, and editorial changes.

SC24-6320-01, z/VM 7.2 (September 2020)

This edition supports the general availability of z/VM 7.2.

SC24-6320-00, z/VM 7.1 (September 2018)

This edition supports the general availability of z/VM 7.1.

Part 1. Introduction

Chapter 1. Introducing RSCS Networking

RSCS Networking for z/VM (commonly referred to as RSCS) is a networking program that enables users on one z/VM system to send messages, files, commands, and jobs to other users within a network. RSCS connects nodes (systems, devices, and workstations) using links. These links allow data, consisting mainly of CP spool files, to be transferred between these nodes: local and remote, adjacent and nonadjacent.

Remote

A separate system or device that you access through a telecommunication line.

Spooling

Storing and retrieving data on an auxiliary direct access storage device (DASD) so that it is convenient for later processing or output.

Communications

Interaction or information (data) exchange with others.

Subsystem

A system that runs under the control of another system.

Networking

Data processing systems and devices connected with other systems and devices.

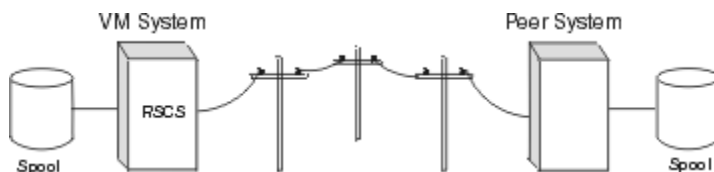


Figure 1. RSCS Transferring Spool Data over Telecommunications Lines

RSCS is a subsystem that can:

- Handle data being sent to, from, or through its z/VM system
- Store and retrieve input and output data files on the z/VM system spool
- Use communications equipment to transfer data between its z/VM system and remote users, devices, and other systems

Running under the Group Control System (GCS) component of z/VM, RSCS uses the spooling facilities of z/VM to store and retrieve data. z/VM handles data transfer *within* its system by means of spooling. RSCS extends z/VM spooling capabilities, handling data transfer *between* its z/VM system and outside sources. Data is stored on a spool after RSCS receives it and until RSCS can forward it to its destination. RSCS uses communications equipment to transfer data between the local z/VM system and other systems or remote locations.

The following topics provide an overview of how RSCS works and the type of connections that you can establish within an RSCS network.

How RSCS Fits into Your z/VM System

RSCS runs in a virtual machine and relies on GCS for supervisor services. The RSCS virtual machine must be defined as part of a GCS virtual machine group. GCS allows group members to share the following:

- A common storage area for information exchange
- Multitasking services
- General I/O services
- Other simulated-OS services

Introduction

For more information about GCS, see [z/VM: Group Control System](#).

How RSCS Manages Files

RSCS uses the z/VM system spool to manage file transfer. Like z/VM, RSCS uses the system spool for temporary file storage. [Figure 2 on page 4](#) shows z/VM using the system spool to store files moving within the local system (or node); that is, between its users and real devices.

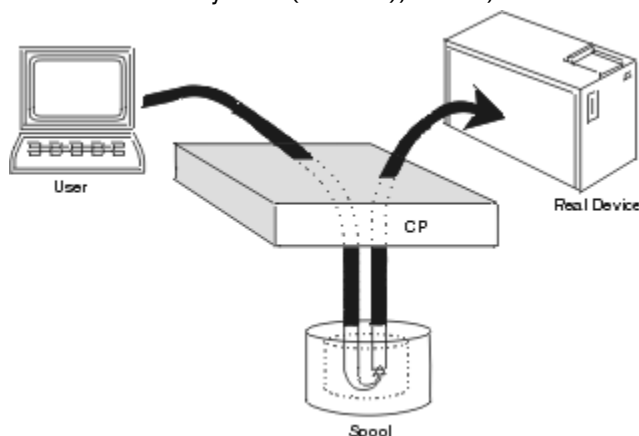


Figure 2. z/VM Use of the System Spool

RSCS extends z/VM spooling. It uses the spool to store files being transferred between its local system and remote users, devices, or systems. As [Figure 3 on page 4](#) shows, these files are spooled to RSCS for processing.

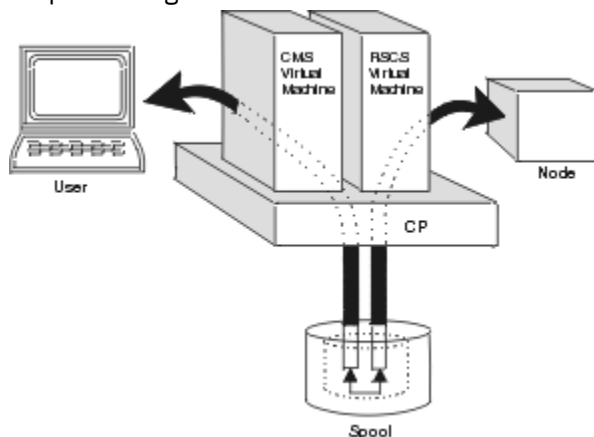


Figure 3. RSCS Use of the System Spool

To manage files spooled to remote nodes, RSCS relies on tag information. A spool file tag becomes a part of each data file spooled to RSCS. The tag contains information that describes where the file came from (origin information) and where it is going (destination information). [Figure 4 on page 5](#) shows an example of origin and destination information in a tag.

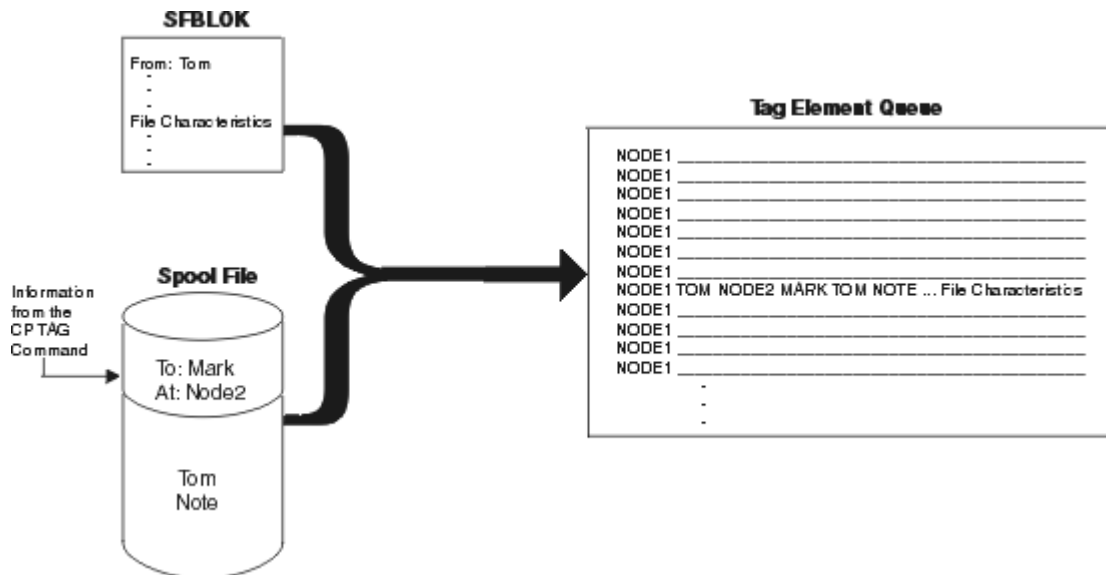


Figure 4. RSCS Spool Files Contain Tag Information

When RSCS receives a file, it records information from the spool file tag into a tag slot. RSCS uses the information in the tag slot to finish transferring the file to its destination.

Supported Nodes

Networks consist of systems and input/output (I/O) devices connected by communications equipment. The connections between various points in a network are called links. The points at which links meet or end are called nodes. Figure 5 on page 5 shows a simple network.

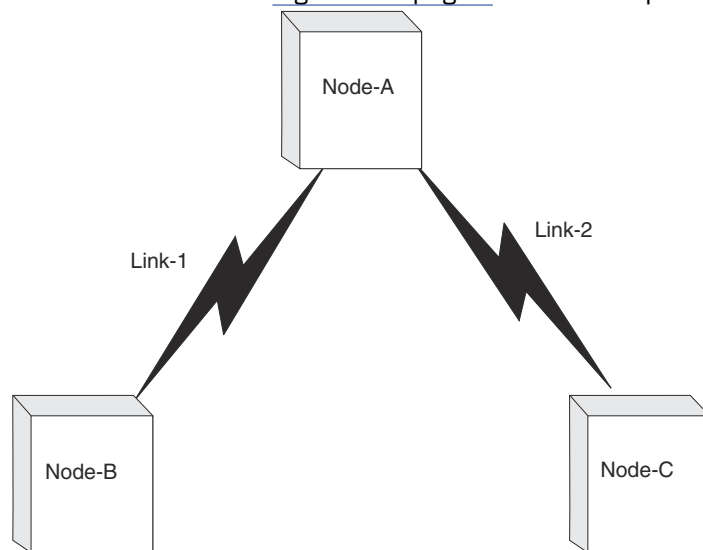


Figure 5. A Simple RSCS Network

Nodes in an RSCS network can be devices, such as workstations and printers, or systems, including z/VM and non-z/VM peer systems. A peer is an equivalent system with which data can be mutually exchanged. In a network, nodes are considered to be local or remote.

A local node is the system you are using at that particular time. Because RSCS runs on z/VM, its local node is a z/VM system. A remote node is another system connected to your local system by communications equipment over RSCS-defined links. Remote nodes are either adjacent or nonadjacent.

An adjacent node is one that RSCS can communicate with directly. In other words, RSCS can transfer data to it without going through another node first. Conversely, for nonadjacent nodes the communication is

Introduction

indirect; data passes from RSCS to an intermediate (adjacent) node before the nonadjacent node receives it.

Figure 6 on page 6 shows the distinction between remote and local and adjacent and nonadjacent nodes.

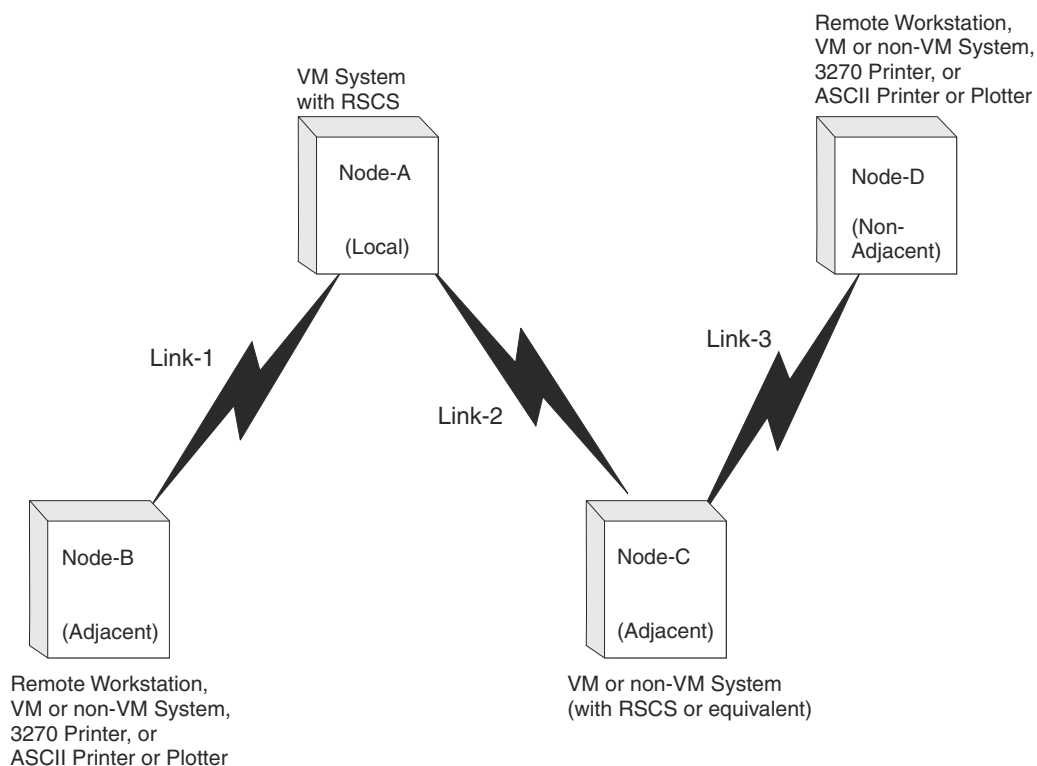


Figure 6. Node Characteristics in a Simple Network

In Figure 6 on page 6:

- Node-A is the local node.
- Node-B, Node-C, and Node-D are remote nodes.
- Node-B and Node-C are adjacent to Node-A. They are directly connected by Link-1 and Link-2 respectively.
- Node-A and Node-D are nonadjacent nodes that are indirectly connected by Link-2, adjacent Node-C, and Link-3.

Types of Nodes

A node is either a system node or a station node. A station node can originate and receive information. It can be either a computer, workstation, or printer. A system node, however, must be a computer. Besides originating and receiving information, system nodes can also relay information between two other nodes. If, because of its position in a network, a node relays information, it is sometimes called a store-and-forward node.

In Figure 6 on page 6, Node-C represents a store-and-forward node. It must be a system node because it must relay information between nodes. If a node cannot relay information because of its position in a network, it is sometimes called an end node. In Figure 6 on page 6, both B and D represent end nodes. They can be either system nodes or station nodes.

RSCS can communicate with system nodes that are running under the control of network job entry (NJE) compatible subsystems, such as:

- JES2

- JES3
- RSCS
- VSE/POWER
- AS/400 Communications Utilities
- Products that provide NJE functions for Linux® or AIX®

For more information about NJE, see *z/OS: Network Job Entry (NJE) Formats and Protocols* ([https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa320988/\\$file/hasa600_v2r5.pdf](https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa320988/$file/hasa600_v2r5.pdf)).

RSCS can communicate with station nodes that are:

- ASCII printers or plotters
- Computers running under the control of a system that can provide a multi-leaving protocol
- IBM 3270 Information Display System Printers
- Line printer daemons and clients in a TCP/IP network
- Unsolicited File Transfer (UFT) daemons and clients in a TCP/IP network
- Workstations running under the control of remote job entry (RJE)

Supported Link Types

A link in an RSCS network is a connection between two adjacent nodes. RSCS transfers data between its system and other nodes over links. Each link is associated with a programming routine, called a driver, that manages the transmission and reception of files, messages, and commands over the link.

The way that a driver manages the data is called a protocol. All file transmission between networking nodes uses NJE protocol. 3270 printers use 3270 data streams, workstations use RJE protocol, and ASCII printers use data streams appropriate to that printer. Systems Network Architecture (SNA) provides one set of protocols that governs communications on links. TCP/IP links use sockets.

The method RSCS uses for sending data to a node varies, depending on the type of connection that is used to establish a link. As the following topics describe, RSCS can support non-SNA, SNA, and TCP/IP connections.

Non-SNA Connections

Non-SNA connections between RSCS nodes are usually established using binary synchronous communication (BSC) or channel-to-channel (CTC) type lines. These link are associated with virtual line addresses. RSCS has direct control of I/O activity on the link and must respond to all error conditions that occur on the link.

Over non-SNA connections, RSCS communicates directly with adjacent nodes and indirectly with nonadjacent nodes. Direct communication means that RSCS determines where to move data and how to get it there. Indirect communication means that RSCS sends a file to an intermediate system (adjacent node) where it is stored temporarily and then forwarded on to its destination. This destination could be the nonadjacent node or another intermediate system. This method of storing files, between the time they are sent and the time they are received, is called store-and-forward transmission.

Introduction

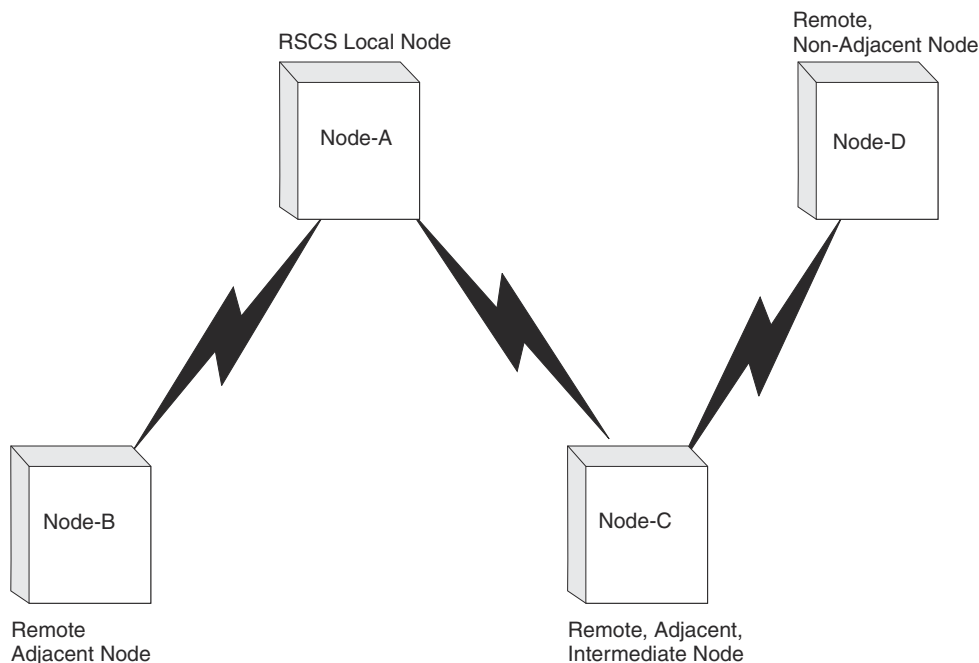


Figure 7. RSCS Communication across Non-SNA Links in a Network

For example, in Figure 7 on page 8, RSCS resides on the system at Node-A. From Node-A, RSCS can communicate with Node-B, Node-C, and Node-D. When RSCS communicates with Node-B or Node-C, the communication is direct. Communication with Node-D is indirect. Because the communication is indirect, the store-and-forward transmission method is used. This means that when a file is sent from Node-A to Node-D it temporarily is stored at Node-C. Then, the networking program at Node-C (which may or may not be another RSCS system) forwards the file on to Node-D.

RSCS recognizes the following types of non-SNA links:

ASCII

A link to a remote or local ASCII printer or plotter using an ASCII control unit

GATEWAY

A link using an installation-defined protocol

LISTPROC

A list processor

MRJE

A link to a multi-leaving remote job entry workstation using a BSC adapter

NJE

A link to a remote network job entry system using a 3088, CTC adapter, ESCON or FICON® CTC, or BSC adapter

NOTIFY

A note generator

RJE

A link to a remote job entry workstation using a BSC adapter

3270P

A link to a remote or local 3270 Information Display System Printer that is channel-attached or using a BSC adapter

SNA Connections

For SNA connections, RSCS deals with *where* and calls on another product, VTAM®, to deal with *how*. SNA connections, which are also called session drivers, are VTAM-controlled devices that are logically connected to RSCS; they perform data transfer on SNA links. VTAM directly processes all I/O and errors

that may occur; it then provides RSCS with information about these transactions. These connections are also referred to as LU_LU (logical unit-to-logical unit) sessions.

RSCS communicates directly with both adjacent and nonadjacent nodes through VTAM by means of SNA links. On SNA connections, VTAM takes care of physically moving data between its origin and destination. To accomplish this, VTAM must have an accurate view of the network's physical configuration so that it can select paths for moving the data between nodes. RSCS, however, does not become involved with the physical movement of data on SNA links. While VTAM identifies the paths, RSCS simply identifies the destinations.

Because RSCS and VTAM divide the work involved with data transmission, they have different physical and logical views of nodes connected by SNA links. For example, nodes that appear physically nonadjacent to VTAM seem adjacent to RSCS. As [Figure 8 on page 9](#) shows, through VTAM, RSCS can establish direct communications with nodes that are both physically adjacent and physically nonadjacent.

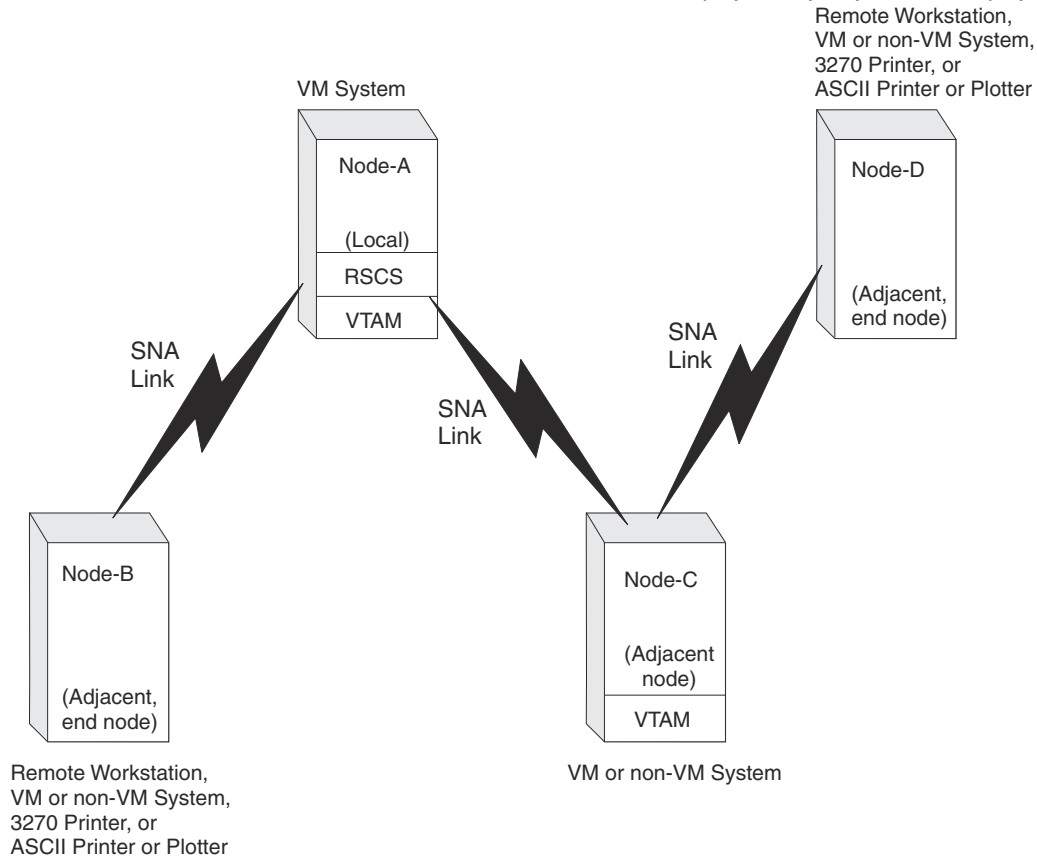


Figure 8. RSCS Communication across SNA Links in a Network

In [Figure 8 on page 9](#), RSCS resides on the system at Node-A. From Node-A, RSCS can communicate with Node-B, Node-C, and Node-D. If this were a non-SNA network (like the one in [Figure 7 on page 8](#)), RSCS would be communicating directly with Node-B and Node-C and indirectly with Node-D. However, this is an SNA network with VTAM residing on Node-A and Node-C. Because of VTAM, RSCS can communicate directly with all the nodes in this network. For example, a file sent from Node-A, destined for Node-D, is transmitted directly to Node-D with VTAM's help. It is not temporarily stored at Node-C as with non-SNA links.

RSCS recognizes the following types of SNA links:

SNANJE

A link to a remote network job entry system

SNARJE

A link to a remote job entry workstation

SNA3270P

A link to a remote or local 3270 Information Display System Printer

TCP/IP Connections

RSCS provides several links that enable it to interact with printers, workstations, or peer NJE nodes within a TCP/IP network. RSCS recognizes the following types of TCP/IP connections:

LPD

A connection that enables RSCS to receive print files from a line printer remote (LPR) client in a TCP/IP network

LPR

A connection that sends print files to a line printer daemon in a TCP/IP network

TCPASCII

A connection to an ASCII printer or plotter attached to a terminal server

TCPNJE

A connection to a remote peer NJE system in a TCP/IP network

TN3270E

A link to a remote 3270 Information Display System Printer that is attached within a TCP/IP network

UFT

A connection used to send files to a UFT client in a TCP/IP network

UFTD

A connection that enables RSCS to receive files from a UFT server in a TCP/IP network for distribution in the RSCS network

These TCP/IP connections use sockets to establish the protocol layer to an IP network; these connections are associated with dotted decimal addresses or fully qualified host names. This network provides connectivity through routers and gateways to a client, server, or NJE peer system in a TCP/IP network. As [Figure 9 on page 10](#) shows, TCP/IP services the connections to the IP network; it also provides RSCS with information about the flow of data.

Attention: IPv6 Not Supported

RSCS supports only IPv4 addressing; it does not support IPv6.

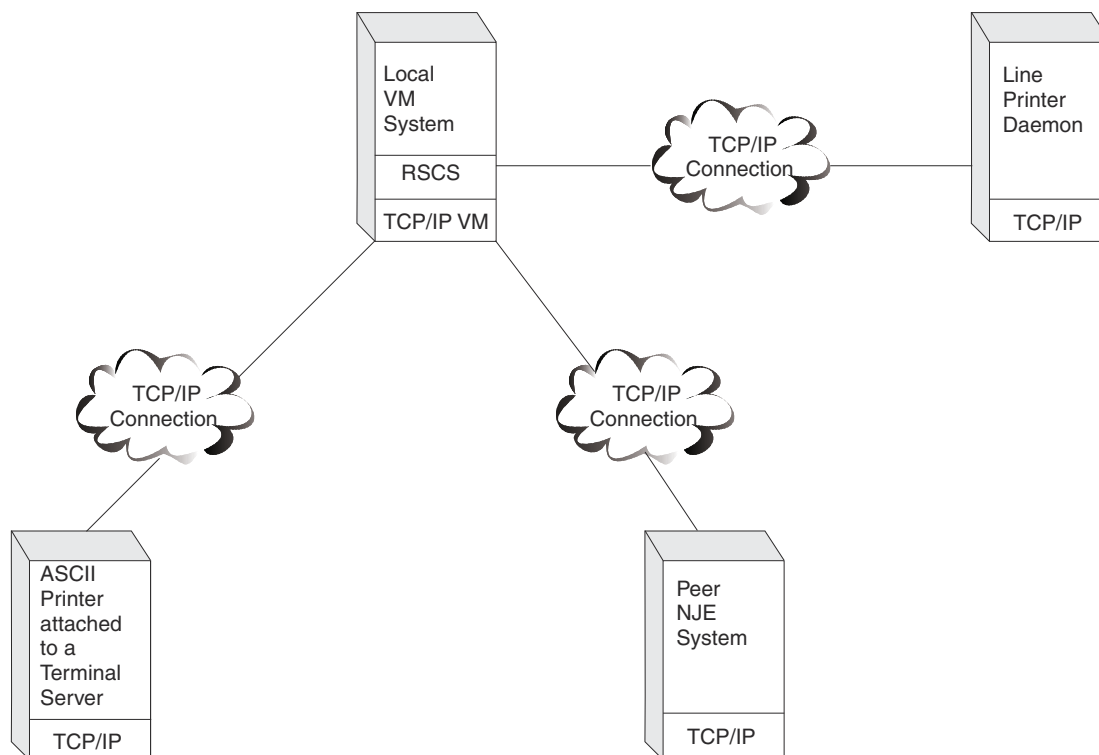


Figure 9. RSCS Communication across TCP/IP Connections

RSCS Domain Name Server

The RSCS domain name server is an optional facility that performs domain name system (DNS) searches on behalf of RSCS to translate fully qualified host names (domain name addresses) to dotted decimal IP addresses in a TCP/IP network. For more information, see [Chapter 12, “Setting Up the RSCS Domain Name Server,”](#) on page 113.

RSCS Configuration File

The configuration file tells RSCS about the network hardware and software arrangement. Statements in the configuration file describe (among other things):

- The name of the z/VM system where RSCS is located (local node)
- The names of links to adjacent nodes
- Communication paths (routes) to nonadjacent nodes through adjacent nodes
- How to communicate with the nodes (based on whether they are 3270 printers, workstations, peer systems, or ASCII devices)
- The names of RSCS operators, who can issue RSCS commands and control RSCS links

RSCS loads the configuration file during its initialization process. RSCS then builds internal tables based on information from the configuration file statements that govern operations at the network. Part 2 of this document contains information about how to use this file to configure your RSCS system.

If your network should change after RSCS has been initialized, you have two ways to tell RSCS about these changes. You could change the configuration file and initialize RSCS again, but if the change is temporary, this is not necessary. Instead, you can use certain commands that dynamically add, delete, and redefine links between nodes while RSCS is still running. These commands cause changes in the internal tables that, in turn, let RSCS quickly make changes to the network's configuration without shutting down and starting up again. These commands are described in [z/VM: RSCS Networking Operation and Use](#).

RSCS Dynamic Authorization

RSCS dynamic authorization (RSCSAUTH) is an optional facility that allows you to authorize users for RSCS commands and links without updating the RSCS configuration file. RSCSAUTH maintains the authorization information in a set of tables. You can use RSCSAUTH to authorize a user for any set of commands and links. You can add, change, or remove authorizations without restarting the RSCS server.

RSCSAUTH is authorized to issue all RSCS commands. If a user is authorized for a command through RSCSAUTH, the user issues the command to the RSCSAUTH server instead of the RSCS server. RSCSAUTH checks the authorization tables to validate the user's authority, then issues the command to the RSCS server. The RSCS server executes the command and returns the response to RSCSAUTH, which passes the response back to the user who originated the command.

For more information, see [Chapter 13, “Setting Up RSCS Dynamic Authorization,”](#) on page 115.

How RSCS Transfers Files

RSCS transfers files using link drivers. While a link driver is active, it handles data transfer between its link and the z/VM system spool. The link drivers use tag slot information to complete file transfer.

There are three categories of link drivers: line drivers, session drivers, and TCP/IP connections. [Table 2 on page 12](#) shows the various link drivers and their associated nodes.

Line drivers perform data transfer on non-SNA links. They are called line drivers because they handle actual I/O on communications lines. They transmit and receive files over these lines between nodes associated with specific drivers.

Introduction

Session drivers perform data transfer on SNA links. They are called session drivers because they enter into LU_LU (logical unit-to-logical unit) sessions with VTAM. They send and receive files, through VTAM, between nodes associated with specific drivers.

TCP/IP connections perform data transfer to peer NJE nodes and devices that reside in a TCP/IP network. They are called connections because the links use sockets to establish the protocol layer to an IP network; these connections are associated with dotted decimal addresses or fully qualified host names in a TCP/IP network. TCP/IP services the connections to the IP network and provides RSCS with information about the flow of data.

Table 2. RSCS Link Drivers and Associated Nodes

Driver Name	Driver Type	Associated Node
ASCII	Line	Remote or local ASCII printers
GATEWAY	Line	Defined by your installation
LISTPROC	Line	None
LPD	TCP/IP	LPR client within a TCP/IP network
LPR	TCP/IP	Line printer daemon within a TCP/IP network
MRJE	Line	Remote multi-leaving workstations
NJE	Line	Remote (peer) systems
NOTIFY	Line	None
RJE	Line	Remote workstations
SNANJE	Session	Remote (peer) systems using VTAM
SNARJE	Session	RJE workstation using VTAM
SNA3270P	Session	Remote or local 3270 printers using VTAM
TCPASCII	TCP/IP	Remote ASCII printers in a TCP/IP network
TCPNJE	TCP/IP	Remote (peer) systems within a TCP/IP network
TN3270E	TCP/IP	Remote 3270 printer attached within a TCP/IP network.
UFT	TCP/IP	UFT daemon within a TCP/IP network
UFTD	TCP/IP	UFT client within a TCP/IP network
3270P	Line	Remote or local 3270 printers

List Processor Links

The list processor link driver, LISTPROC, is an RSCS facility that you can use to process a spool file containing a list of destinations. A list processor enables a file with a distribution list to be transmitted over a network to the destinations listed in the distribution list using the minimum number of copies of the file.

The list processor link driver operates on a specially-defined link. Unlike other types of links, RSCS will not let you delete it once you define it. RSCS also prevents you from defining any other LISTPROC-type links by ignoring the new definition. The existing RSCS configuration file statements and commands define and start the link ID and line driver. Internally, the list processor appears to RSCS as an NJE line driver driving a line wrapped back upon itself. Data is read from the spool by the transmitting side of the line driver and written back into new spool files by the receiving side of the line driver.

Networking Links

The networking link drivers (GATEWAY, LISTPROC, NJE, SNANJE, and TCPNJE) use a special technique called multistreaming for transferring files. Multistreaming allows more than one file to be transmitted concurrently over the same link. For example, if files A and B are being transmitted at the same time, a portion of file A will be transmitted, then a portion of file B, followed by another portion of file A. This

type of intermixing can improve traffic flow on the network. With multistreaming, small files are not kept waiting while a large file is being transmitted. Both large and small files are transmitted at the same time.

The LISTPROC, NJE, SNANJE, and TCPNJE link drivers support up to 7 transmission streams and the GATEWAY link driver supports up to 32 transmission streams.

Notify Links

The NOTIFY link driver is also a special type of link. It is designed to store files, generate notes, and purge files. In other RSCS documents, we show examples of how you can use a NOTIFY link to handle misdirected files in your network. Using the ROUTE configuration file statement or command, you can direct files with unknown user IDs and node IDs to a NOTIFY link. You can define the link to hold files for a specified period of time, create a note reflecting the problem, and purge the file when the period of time has elapsed.

Printer Links

The printer link drivers (ASCII, LPD, LPR, SNA3270P, TCPASCII, TN3270E and 3270P) enable RSCS to communicate with a variety of printer and plotter devices. RSCS can support IBM 3270 Information Display Printers, ASCII printers or plotters, or a print daemon in a TCP/IP network. The SNA3270P-type, TN3270E-type, and 3270P-type links permit the production of GDDM (Graphical Data Display Manager) Extended 3270 data stream output or IPDS (Intelligent Printer Data Stream) output through GDDM or PSF/VM (Print Services Facility/VM). The ASCII-type and TCPASCII-type links send data streams of ASCII characters and control sequences to ASCII printers and plotters. In addition, the LPR-type and LPD-type links enable RSCS to send data streams of ASCII characters to a line printer daemon and receive them from LPR clients, so that print files can be exchanged between a TCP/IP network and a NJE network.

UFT Links

The UFT-type and UFTD-type links enable RSCS to send files to a UFT daemon and receive them from a UFT client allowing for the exchange of files between a TCP/IP network and a NJE network.

Workstation Links

Each workstation link (RJE, MRJE, and SNARJE) enables RSCS to communicate with one remote workstation in the network. The communication protocols RSCS uses varies according to the type of link that is established. RJE-type links use binary synchronous communications (BSC) to communicate with a single workstation. An MRJE-type link uses a multi-leaving protocol that enables it to operate in host mode or in workstation mode. For SNARJE-type links, SNA protocols are used to communicate with a workstation.

RSCS Data Interchange Manager

In addition to the various types of links that RSCS supports, RSCS includes an optional facility called the RSCS Data Interchange Manager (RSCS Interchange). RSCS Interchange handles the exchange of mail between z/VM users and users in a TCP/IP network environment who use Simple Mail Transfer Protocol (SMTP) mail functions.

SMTP is an electronic mail protocol that supports client (sender) and server (receiver) functions. The SMTP mail format is different from the NJE format used by z/VM, such as the CMS NOTE command. However, RSCS Interchange allows users with these different mail formats to communicate with each other using their usual method of sending notes.

For more information, see [Chapter 4, “RSCS Data Interchange Manager,”](#) on page 25.

Summary of RSCS Communications

Table 3 on page 14 summarizes the types of links that RSCS can support in your network. The table includes the type of communication the links enable and the types of nodes they connect.

Table 3. Summary of RSCS Link Types and Their Roles

Link Category	Function	Link Type	Driver Type	Node and Communication
List Processor	Provides mechanisms to send one file to many destinations using the minimum number of physical transmissions.	LISTPROC	Line	None
Networking	Provides for communication between RSCS and an adjacent NJE-compatible subsystem.	GATEWAY	Line	Allows a connection to a NJE-compatible subsystem using protocols defined by your installation.
		NJE	Line	Each link controls one communication adapter, for a BSC line, a CTCA, an ESCON or FICON CTCA, or a 3088.
		SNANJE	Session	Allows logical unit, type 0 sessions with SNA systems. A SNANJE link uses a full duplex protocol; both session partners can simultaneously send and receive.
		TCPNJE	TCP/IP connection	Allows communications, using standard TCP/IP sockets, with an NJE peer node that resides in a TCP/IP network or in a z/OS® or VSE system.
Notify	Generates and sends notes to users when RSCS finds a problem with a file. If a file cannot be delivered, Notify informs its originator and holds the file until it can be processed. Notify purges the file after a time limit expires.	NOTIFY	Line	None

Table 3. Summary of RSCS Link Types and Their Roles (continued)

Link Category	Function	Link Type	Driver Type	Node and Communication
Printer	Connects a system node and a station node (an IBM 3270 Information Display System printer, an ASCII printer or plotter, or a line printer daemon).	ASCII	Line	Transmits data streams of ASCII characters and control sequences to ASCII printers and plotters connected to a host z/VM system.
		LPD	TCP/IP connection	Enables RSCS to receive ASCII print data streams from an LPR client in a TCP/IP network and convert to EBCDIC.
		LPR	TCP/IP connection	Enables RSCS to send EBCDIC print data streams to an ASCII line printer daemon in a TCP/IP network.
		SNA3270P	Session	Transmits files from a host z/VM system to a 3270 printer through VTAM using types 0, 1, or 3 logical unit protocols.
		TCPASCII	TCP/IP connection	Transmits data streams of ASCII characters and control sequences to ASCII printers and plotters that are attached to a terminal server within a TCP/IP network.
		TN3270E	TCP/IP connection	Transmits files from a host z/VM system to a 3270 printer, which is attached within a TCP/IP network.
		3270P	Line	Transmits files from a host z/VM system to a 3270 printer, which is locally channel-attached or is remote through a BSC adapter.
UFT	TCP/IP connection using the unsolicited file transfer protocol to send files.	UFT	TCP/IP connection	Enables RSCS to send data streams to a UFT daemon in a TCP/IP network.
UFTD	TCP/IP connection using the unsolicited file transfer protocol to receive files.	UFTD	TCP/IP connection	Enables RSCS to receive data streams from a UFT client in a TCP/IP network.

Table 3. Summary of RSCS Link Types and Their Roles (continued)

Link Category	Function	Link Type	Driver Type	Node and Communication
Workstation	Connects a system node (or host) and a station node that is a workstation or another computer that is acting as a workstation.	MRJE	Line	Provides multi-leaving protocol for a remote workstation or as a remote workstation, using a BSC adapter.
		RJE	Line	Provides BSC line protocol only to a single remote workstation.
		SNARJE	Session	Transmits files between a host z/VM system and an SNA RJE workstation using LU T1 sessions.

Chapter 2. RSCS Environments

RSCS can be used in either a single-system or a multisystem (network) environment. The configuration of a network depends both on the size and the nature of the business.

Single-system Environment

Typically, a small business operates in a single-system network. This network consists of a single, main computer that is centrally located and has a number of local and remote devices connected to it. Remote devices are those connected to the system by communications equipment. Remote connection allows for geographic separation between the system and the device. (They might be on different floors, in different buildings, or in different cities.) Remote users are those who access the system over remote connections.

In a single-system environment, RSCS running under z/VM can have output printed on a 3270 Information Display System printer or an ASCII device that several users share (regardless of input method or location).

3270 Printers

3270 printers are those printers that connect to the system by the same type of control unit as 3270 terminals. This group includes models that produce both alphanumeric and graphic (monochrome and color) output.

These printers range from basic function, low-cost models to more costly, extended function models. Basic function, low-cost 3270 printers help businesses provide their employees with timely printed output. Their basic function provides for day-to-day working copy. The low cost makes it feasible for a business to own several and distribute them among individual departments that are located away from the system, near users. Members in or near the department can share the printer and the benefits of having their output printed when and where it is convenient. The more costly, extended function 3270 printers provide for the special needs (graphics output, for example) of a business.

ASCII Devices

RSCS can communicate with ASCII devices connected to an IBM 7171 ASCII Device Attachment Control Unit or a 9370 ASCII Subsystem Controller in transparency mode. RSCS can also communicate with ASCII devices that are attached to a terminal server in a TCP/IP network.

With this RSCS support, user programs can transmit data streams of ASCII characters and control sequences rather than EBCDIC data streams to ASCII printers and plotters. RSCS uses the transparency mode of the controller to allow the user's program to exercise the many features of an ASCII device such as type quality or type size which are not supported by the 3270 emulation provided by the controller.

Sharing Printers

Scheduling and managing real devices is a necessary part of data processing. These are services z/VM gives to its users. z/VM controls real devices; users do not. Users do not have to wait for real devices to become available. They work as they need to, using virtual devices on their virtual machines.

Along with scheduling and managing real devices, the system provides for the more efficient shared use of those devices. They are not controlled by individual users, but by the system to work for everyone. Sharing is possible, under z/VM, because of spooling.

When users "print" output on their virtual printers, for example, they are creating output files which the system stores on the spool. The spool, which has a large capacity for holding data, can hold output files from many different users for the same printer. When the required real printer is available, the system selects which file to print next. Only then does the spool file become printed on paper.

Environments

The system manages the flow of files to real printers. Users are concerned only with their virtual printers, which are always available to them. They can work continuously, run jobs that produce output at any rate, in any order. The system controls the real printers. It schedules the actual printing of an output file on a real printer such as a 1403, 3800, 4245, or 4248 printer. In this way, many users can share a single real printer.

Sharing 3270 Printers

z/VM spooling cannot support 3270 printers; however, RSCS can support these printers. Businesses that want the flexibility and convenience offered by 3270 printers can support them through RSCS.

RSCS supports both remote and local 3270 printers and manages them, through the system spool. RSCS extends the z/VM spooling capabilities to include 3270 printers.

The system controls other types of printers, but leaves control of 3270 printers to RSCS. Users direct their output for a 3270 printer to RSCS. When they "print" output, the system stores it in a spool file for RSCS. When the required real 3270 printer is available, RSCS selects which file to print next and sends it along to the printer. Only then does the spool file become printed on paper. [Figure 10 on page 18](#) shows several users sharing a single real (RSCS-controlled) 3270 printer.

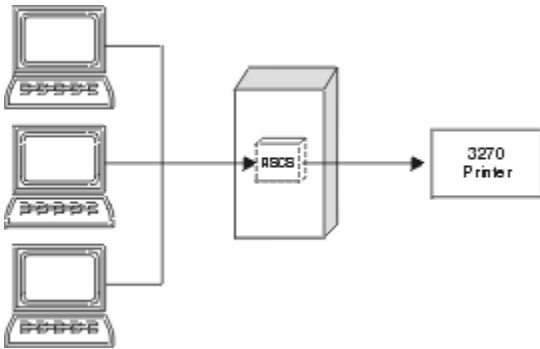


Figure 10. Through RSCS, Several Users Can Share a Single 3270 Printer

Submitting Jobs from Remote Locations

The following topics describe how you can use RSCS to submit different types of jobs from remote locations in your network.

Remote Workstations

In RSCS terms, a remote workstation consists of one or more devices through which users can send input to and receive output from a host system. The host is the system used for processing jobs. It may be the same system to which a remote workstation is physically connected. Or, in a multisystem network, the host may be another system in the network. This process is called remote job entry, and so the remote workstations are often called RJE workstations.

Input and output are in the form of files. The files are created before they are sent to or from the remote workstation. Input files can be jobs and data that users have created or programs they have written. Output files can be formatted reports or results from compiling programs.

Using a remote workstation, a user can send an input file to the host system for processing. The host creates an output file as the result of its processing and returns that file to the user. RSCS moves the data from the user to the system and from the system to the user through the remote workstation.

Remote Workstation Location

A remote workstation can be anywhere. It does not have to be near the system, but it can be situated, instead, near its users—where it is convenient for their work. Remote workstations might be located in individual users' offices. So, without leaving their work area users can:

- Create files (or use existing files)

- Send jobs to the system for processing
- Do other work at their desks while jobs are being processed
- Receive output when job processing is done
- Use results immediately for the next phase of their work

Or, a remote workstation might be centrally located, where it is convenient for several people to share—in the same building and on the same floor with its users.

Advantages of RSCS in a Single-System Environment

In a single-system environment, RSCS offers greater efficiency, productivity, and flexibility.

ASCII printer and plotter support allows users the advantage of using ASCII data streams rather than EBCDIC data streams. RSCS supports these devices through the ASCII-type and TCPASCII-type links. This support allows users to take advantage of ASCII printer type size and quality which 3270 emulation does not support. RSCS can also act as a printer server between workstations and other remote hosts and IP printers.

3270 printers complement rather than replace other (system-controlled) printers. They do work that other printers cannot do, allowing the other printers to be used more efficiently—for high-volume work and high-quality copy.

RSCS 3270 printer support means that:

- Users do not have to go after printouts or have them delivered from some other location. RSCS moves output data to a printer near users, so they can retrieve it as soon as it is printed.
- Neither the system nor individual users need to be involved with managing the shared use of a printer. RSCS can handle several users' output to a single device.

Remote workstations let employees use their company's computer to do their work even if they are geographically removed from it.

- Users do not have to carry (or have someone else carry) data to and from the computer. RSCS handles that—moving the data between users and the system. People stay where they are, where they work. Only the data moves.
- Users can work on the system whenever and as often as needed. Users can get results, make corrections, and try again in the same day. Or, they can get results and move on to the next task.

RSCS gives you the flexibility to align your resources according to the needs of your business. It lets you use devices more efficiently, which saves employees' time and energy for other activities. This means improved productivity in your employees and your business. [Figure 11 on page 19](#) shows RSCS supporting Remote Job Entry and users sharing an ASCII printer and a 3270 printer.

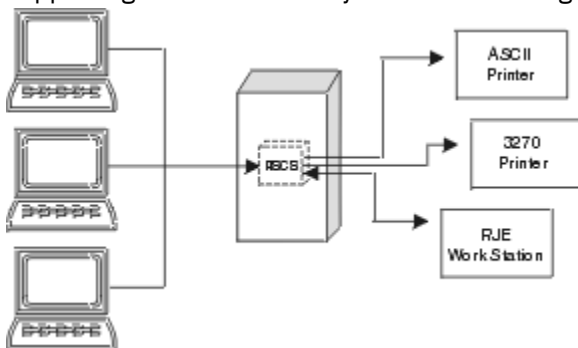


Figure 11. RSCS Support in a Single-System Network

Multisystem Environment

A multisystem environment expands the benefits of a single-system environment. Typically, a large business or one that has more than one geographic location operates in a multisystem network. This

Environments

network consists of two or more systems connected to each other physically (by communications equipment) and logically (in a defined relationship, as peer or host, to other systems).

The systems may be at the same or at different geographic locations. To increase processing power, a business may choose to add a second processor at a site and connect it to their existing processor (rather than to replace their existing processor with one that has a larger processing capacity). Besides being connected to one another, each system in this network may have several local and remote devices connected to it.

In a multisystem network, because the systems are interconnected, data can be moved through and between them—from any system, to any system. RSCS running under z/VM (in addition to what it can do in a single-system environment) can do networking.

To users, networking means they can:

- Exchange data with users on the same system
- Exchange data with systems and users at other locations
- Send jobs to other systems for processing
- Direct processed output to devices, such as printers and punches, that are connected to another system

Data Exchanges

Exchange means two-way communication. The data RSCS handles falls into two categories — messages and files. Although messages are usually smaller pieces of data, the main distinction between the two is how RSCS handles them.

Messages are not spooled; they are sent directly. If a message cannot be sent or its recipient is not there, the message is not delivered. (This is like calling someone on the telephone when the line is busy or there's no answer.) RSCS tells the sender when it cannot deliver the message. RSCS handles commands like messages.

Files are spooled, so they can always be delivered. If a file cannot be sent right away, it is held until it can be sent. When it arrives at its destination, it is placed in a user's virtual reader. So, it is delivered whether or not the recipient is there. (This is like sending someone a letter. It will be delivered to the mailbox even if no one is home.)

RSCS handles user exchanges and operational exchanges. The following topics describe both, starting with the more frequent user exchanges.

User Exchanges

z/VM has the facilities to allow its users to exchange messages and files with other users on the *same* system. RSCS allows users to exchange data as easily with users on *other* systems and locations. RSCS moves data from one place to another in either direction—from system to system, user to user.

User exchanges can be business correspondence (electronic mail).

- Users can send data they've created to an associate at another regional office.
- Users can send weekly reports to a supervisor at a central location.

User exchanges can also be used for short reminders and notices (instead of telephone messages).

All of these exchanges can be done between users on different systems and at different locations.



Figure 12. RSCS Allows Users on Different Systems to Exchange Messages and Files

Operational Exchanges

Operational exchanges deal with system operations. z/VM provides the Programmable Operator Facility (PROP), which can be programmed to handle a number of routine system operations. Systems can run largely unattended. Sometimes they may need a person for special and manual operations, but they do not need the constant attention of a skilled operator.

RSCS adds the option of geographic separation between system and operator. Using PROP, a business can have an operator of one system oversee the operation of a second system at a remote location. RSCS sends information on special operations from PROP on the remote system to the operator and sends the operator's responses back to the remote system. Using RSCS in this way, it is possible for one operator to oversee the operation of several systems, even in different cities or states.

Sending Jobs to Other Places for Processing

Sending jobs to other places for processing is like sending jobs from a remote workstation, only broader in scope. Users can send jobs not only to their own system, but to and from other systems in the network. This makes resources available to users wherever they are in the network. They can take advantage of the following items:

- A special configuration or piece of equipment connected to another system
- Programs or applications that are on other systems
- Greater processing capacity on another system
- Data from common data bases maintained on other systems

RSCS puts users in touch with the resources they need by moving data to and from the selected resource. The resources of a business are available to more of its employees to help them do their jobs.

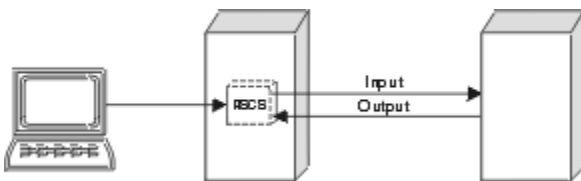


Figure 13. Users Can Send Jobs to Other Systems for Processing

Sending Output to Other Places

The other places in this case are other systems (other than the one the user is logged onto) or a device connected to another system. The output may be destined for a printer, a punch, or another user.

When printed output is needed, a user can send the output to a user on another system for printing. That is, instead of printing the output and mailing it, it is mailed first (using RSCS) and printed at its destination.

Or, for another example, a specialized printer in a multisystem network may be connected to another system. RSCS can send the data to the specialized printer on the other system.

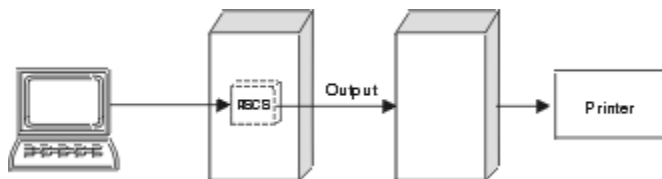


Figure 14. Users Can Send Output Files to Other Systems for Printing

Advantages of RSCS in a Multisystem Environment

In a multisystem environment, RSCS offers greater efficiency, productivity, and flexibility. Earlier, you learned the benefits RSCS offers in a single-system environment through its 3270 printer, remote workstation, or ASCII support. Any system in the network with which RSCS can communicate can be used by your employees.

RSCS gives you many options in using your computing resources. Businesses can distribute computing systems, designed to the specific needs of a given department. Multisystem environments offer:

- Specialized device configuration
- Large processing capacity
- Special set of applications and programs

Any of your employees can have access to the resources within your business network from their own location. However, you control their access to your resources at a corporate or site level.

Employees can get data they need from other systems. When a piece of work passes from one phase to another, moving as it does from department to department, RSCS can transfer data from one employee to the next, from one system to another system. Employees can:

- Correspond electronically
- Use programs on another system to process their jobs
- Send jobs from a remote workstation at their location to the local or to a remote system
- Direct output to RSCS-controlled 3270 printers or ASCII devices from jobs they have submitted to either local or remote systems
- Send or receive output for other system-controlled printers through RSCS
- Send an output file they have created for another employee to print on that employee's system

You can buy and locate resources (processors, computer programs, I/O devices) to fit your business needs—by departments or regions. Because these resources can be shared, you can distribute the workload of your business and improve your employees access to these resources. This can lead to greater efficiency and productivity in your business.

Chapter 3. RSCS Users

There are three categories of RSCS users: general users, users authorized through RSCS dynamic authorization, and operators.

General Users

Most users are not aware that they are using RSCS because they use it indirectly. When a z/VM user enters a TELL, NOTE, or SENDFILE command and the destination is not the local system, CMS uses RSCS to send the data. These and other z/VM commands that use the network do not require users to know about RSCS.

Users can use RSCS commands to transmit messages and to inquire about the status of specific files, links, and systems. Users can also control files that have been given to RSCS to transmit over the network or that RSCS is transmitting to them. *z/VM: RSCS Networking Operation and Use* tells you how to use these commands.

Users Authorized through RSCS Dynamic Authorization

A user can be authorized through RSCSAUTH for any set of RSCS commands, RSCS links, and RSCSAUTH commands. A user who is authorized through RSCSAUTH has authority equivalent to an alternate operator for the authorized RSCS commands and links.

Operators

There are several types of RSCS operators, which are determined by their authority level and the type of work they do. One person can perform several roles.

RSCS recognizes the following types of operators:

RSCS Console Operator

This person is logged on to the RSCS server virtual machine (RSCS) and controls your local RSCS node. For high-traffic nodes, the RSCS console is constantly receiving messages, making it difficult for a dedicated operator to work. For this reason, many installations run RSCS as a disconnected virtual machine and designate several users as system-authorized alternate operators.

System-Authorized Alternate Operator

This person (defined in the RSCS configuration file) has unlimited control over your local RSCS node.

Link-Authorized Alternate Operator

This person (defined in the RSCS configuration file) is authorized to control a single link and the information that passes over that link to and from a single workstation or printer. Link operators usually are also workstation or printer operators but have added capability. Depending on the particular model of workstation or printer in use, its operator may have little or no ability to communicate with RSCS. However, if operators are also link operators, they can easily process commands and have almost as much ability to control the link and the information that passes over it as the local RSCS operator does.

Remote Workstation Operator

This person controls a remote workstation that is part of the RSCS network. That is, the remote workstation is a separate remote node and is connected to the local node by an RSCS link.

RSCSDNS Console Operator

This person is logged on to the RSCS domain name server virtual machine (RSCSDNS) and has unlimited control over the server.

RSCSAUTH Console Operator

This person is logged on to the RSCS dynamic authorization server virtual machine (RSCSAUTH) and has unlimited control over the server. However, this server runs disconnected, so operation is generally handled by the RSCSAUTH administrator.

RSCSAUTH Administrator

This person (defined in the RSCSAUTH configuration file) is fully authorized for the local RSCSAUTH server.

XCHANGE Console Operator

This person is logged on to the RSCS Interchange server virtual machine (XCHANGE) and has unlimited control over the server. However, this server runs disconnected, so operation is generally handled by privileged users.

RSCS Interchange Privileged User

This person (defined in the XCHANGE configuration file) is authorized to issue privileged RSCS Interchange commands.

Chapter 4. RSCS Data Interchange Manager

RSCS Data Interchange Manager (RSCS Interchange) handles the exchange of mail between z/VM and IP networks.

Introduction to RSCS Interchange

RSCS Interchange serves as a bridge for exchanging mail between networks using an NJE facility, such as z/VM, and networks using the Simple Mail Transfer Protocol (SMTP), such as an internet. Users in these different networks have different mail formats. RSCS Interchange allows users with different mail formats to communicate with each other with their usual method of sending notes.

For example, if you use the CMS NOTE command to write and send notes from a z/VM system, you can use the same method to write and send a note to someone in a computer environment that uses SMTP.

The following illustration, [Figure 15 on page 25](#), shows how mail flows through RSCS Interchange between z/VM users and SMTP users.

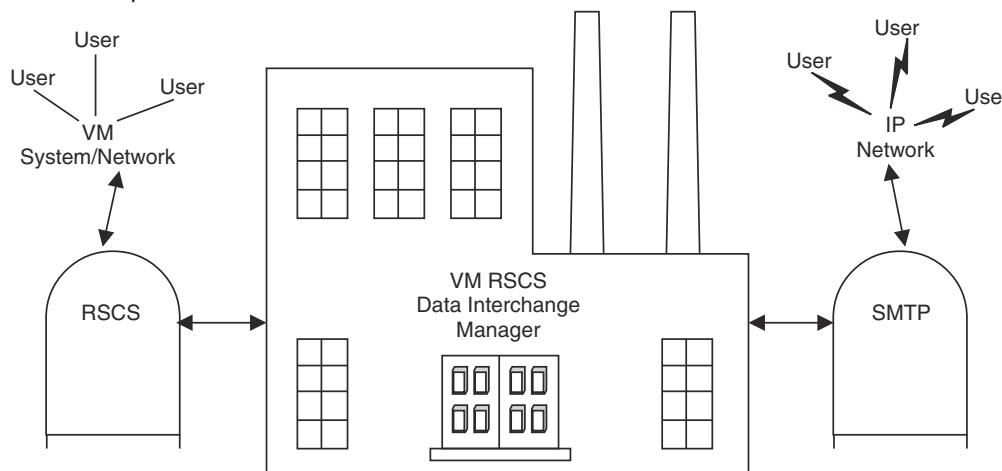


Figure 15. Sending Mail through RSCS Interchange

SMTP is an electronic mail protocol with both client (sender) and server (receiver) functions. RSCS Interchange handles mail traffic destined for SMTP from z/VM or other NJE systems, and mail traffic destined for z/VM or other NJE systems from SMTP.

Your electronic mail software created the mail, which in turn uses a network to send the mail to its destination. The network is a set of two or more computers or workstations linked in a way that lets data be transmitted between them.

Networks that can exchange mail with RSCS Interchange can include:

- Internet, which is a logical collection of networks supported by gateways, bridges, hosts, and various layers of protocols
- Local Area Network (LAN) attached workstations
- Hosts, which are computers connected to a network, which provides an access point to that network

The functional unit that interconnects a local data network with another network having different protocols is called the gateway. With RSCS Interchange, if you are a z/VM user writing a note to someone in another network, the data exchange between networks through a gateway is transparent to you.

How RSCS Interchange Works

The system administrator or operator sets up the RSCS Interchange server to make it work for the users. For more information, see [Chapter 14, “Setting Up the RSCS Data Interchange Manager,”](#) on page 119.

Typical system administrator tasks include:

- Setting up the RSCS Interchange configuration file
- Setting up and maintaining special RSCS Interchange nicknames
- Installing the RSCS Interchange server
- Setting up communications between RSCS and RSCS Interchange
- Setting up communications between SMTP and RSCS Interchange

The configuration file defines the setup of the RSCS Interchange server. In the configuration file, you specify statements to define product requirements, such as:

- The name or identification of the RSCS Interchange server to the SMTP virtual machine (RSCSNAME statement)
- The SMTP domain in which the RSCS Interchange server, SMTP, and domain name server communicate (DOMAIN statement)
- The name that links the RSCS virtual machine to RSCS Interchange (RSCSLINK statement)

Nicknames identify the SMTP users that you want to communicate with through RSCS Interchange. RSCS Interchange contains a nickname table where these nicknames and corresponding SMTP addresses reside. These nicknames are not the same as the nicknames used for CMS NAMES files.

To send a note to someone on another network from yours, you create the note with the method you currently use, with an address of:

```
userid at nodeid
```

You must know the user's address, however. The format of a typical SMTP address is:

```
userid@hostname.domain
```

where the domain might be:

```
hurrah.col.edu
```

If you send a note to someone and RSCS Interchange does not recognize it as a CMS note or mail file, the note is returned to you. The RSCS Interchange server notifies you about the error.

[Figure 16 on page 26](#) shows how RSCS Interchange works in a z/VM or NJE system to provide the link between that network or system and the TCP/IP network.

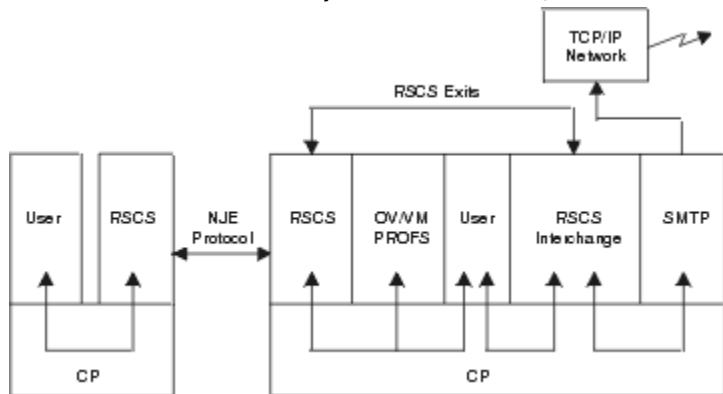


Figure 16. RSCS Interchange in the NJE and IP Network Environments

How Your Mail Is Sent to SMTP Users

As [Figure 17 on page 27](#) shows, you may take the following steps to send a note to someone in the SMTP network:

1. You write a CMS note to a friend and specify his or her SMTP address; your note goes to RSCS.
2. RSCS forwards the note to the RSCS Interchange server.
3. The RSCS Interchange server converts the note into SMTP recognized format, and sends the note to SMTP.
4. SMTP forwards the note to the SMTP user, your friend.

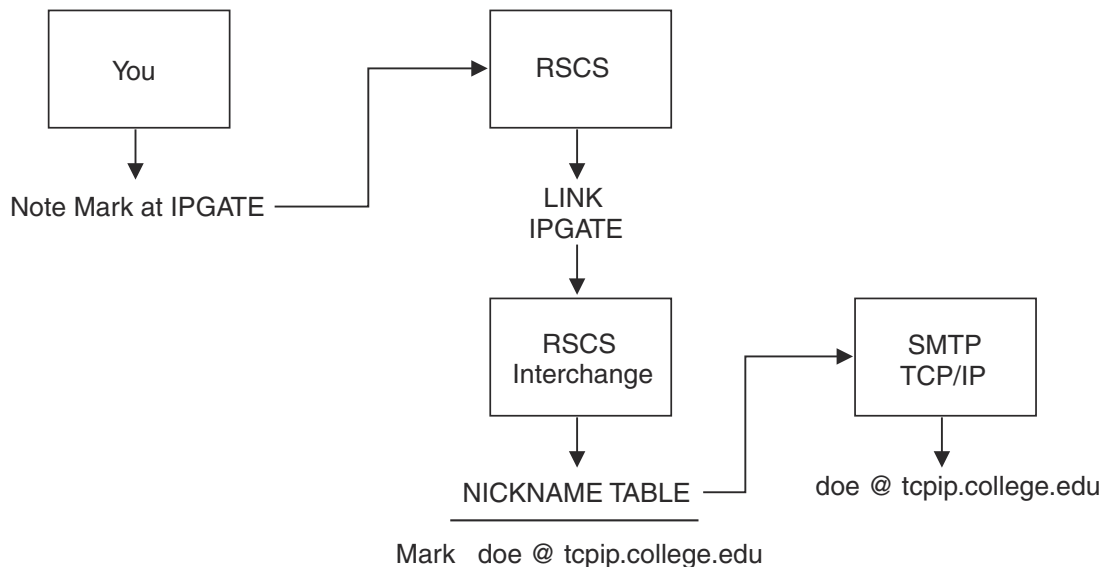


Figure 17. Sending a z/VM Note to an SMTP User

Some examples of mail that has travelled through RSCS Interchange follow.

Looking at z/VM-to-Network (SMTP) Mail

When a z/VM user writes a CMS note to someone, the CMS note has the following format:

```
NOTE userid at nodeid
```

A note that is sent from a z/VM user to an SMTP user through RSCS Interchange might look like the example in [Figure 18 on page 27](#).

```

Received: from RSCSNODE.tcpip.college.edu by RSCS2.tcpip.college.edu
  (IBM VM SMTP V2R2) with TCP; Sun, 20 Mar 22 10:59:41 EDT
Received: from RSCSNODE by RSCSNODE.tcpip.college.edu (IBM VM SMTP V2R2)
  with BSMTP id 8722; Sun, 20 Mar 22 10:59:30 EDT
Received: by RSCSNODE RSCS Interchange FL 7.3.0000 id 8721 (NOTE)
  for <joe@smpt10>; Sun, 20 Mar 22 10:59:28 -0400
Date: Sun, 20 Mar 22 10:59:50 -0400
From: <matthew@vmnode.tcpip.college.edu>
To: <francis@hurrah.tcpip.college.edu>
Subject: A test of RSCS Interchange
  
```

Here's an example of the SMTP results from RSCS Interchange.
Have a good day!

Figure 18. Example of z/VM Mail Received by SMTP

Looking at SMTP-to-z/VM Mail

A note from an SMTP user that is sent through RSCS Interchange to a z/VM user might look like the example in [Figure 19 on page 28](#).

```
Date: 10 March 2008, 10:59:00 EDT
From: Doug Jones ext.1234
To: John Smith ext.2345
cc: DRB testid
Subject: SMTP note to John
Return-Path: <doug@barney.tcpip.college.edu>
```

```
DOUG at SMTP10
SMITH at VMNODE
DRB at VMNODE
```

Example of a note to a z/VM user through RSCS Interchange.
Have a good day!

Figure 19. Example of SMTP Mail Sent to z/VM through RSCS Interchange

By comparison, [Figure 20 on page 28](#) shows an example of a note from an SMTP user that was not sent through RSCS Interchange; note the large logical record length.

```
Received: from COLLEGE.college.edu by vnet.IBM.COM (IBM VM SMTP V2R2) with TCP;
Mon, 10 Mar 08 13:18:47 EDT
Received: from COLLEGE.COLLEGE.EDU by COLLEGE.college.edu (IBM VM SMTP V2R2)
with BSMTP id 6041; Mon, 10 Mar 08 13:24:48 EDT
Received: from COLLEGE.COLLEGE.EDU (NJE origin BARNEY@COLLEGE) by COLLEGE.COLLEGE.1 V1.2a/1.8a) with
RFC822 id 3889;
Mon 10 Mar 2008 13:23:50 -0400
Subject: Re: SMTP note received without using Interchange
Message-Id: <BARNEY.940710124037@college.college.EDU>
From: Barney <Barney@COLLEGE.college.edu>
To: doug <doug@VNET.IBM.COM>
Date: Mon, 10 Mar 08 12:40:37 EDT
```

Example of a note that was not received through RSCS Interchange.
And, have a good day!

Figure 20. Example of SMTP Mail Sent to z/VM Without Using RSCS Interchange

Part 2. Planning and Configuration Guide

Chapter 5. System Planning for RSCS

This section contains topics that explain what you need to do to identify RSCS to z/VM and other systems in the network. RSCS works for and with other programs and products that have certain requirements about the way you define and configure RSCS. Planning for those requirements is essential to integrating RSCS.

Planning Checklist for System Configuration

- **Coordinate** dependencies with the system programmers responsible for defining VTAM or TCP/IP.
- **Ensure** that GCS is set up and that VTAM, if required, is installed. Be aware that RSCS requires the VTAM macro library when assembling some modules. VTAM might also be needed when assembling RSCS modules because of user modifications.
- **Define** and discuss configuration, customization, and administration needs with the people who will institute the requirements.
- **Establish** a schedule to complete the configuration of your RSCS network.

VTAM Requirement

If you are planning to use SNA connections, VTAM V4.2 for VM/ESA (5654-010) or later must be installed on your z/VM system.

If you do not have VTAM, you cannot assemble the following RSCS modules:

- DMTSCT
- DMTSJE
- DMTSNE
- DMTSPT
- DMTVXT

You can assemble the DMTLOG module, but errors will occur. Although DMTLOG does not assemble cleanly, you can link-edit the resulting text deck into the RSCS load library without any adverse effects on tracing non-SNA links.

Note: If you do not use SNA connections or have VTAM installed on your system, you might receive an error message when you assemble RSCS modules. For example, you might receive the message, FILE VTAMAC MACLIB NOT FOUND. This occurs because the VTAM macro library is included on a GLOBAL statement in the RSCS control file. You can ignore this message if you are not using SNA connections.

Preparing z/VM for RSCS

The following topics describe the tasks you must perform to prepare to run RSCS on a z/VM system.

RSCS Virtual Machine

The RSCS virtual machine is predefined in the z/VM user directory. The user ID is RSCS. [Figure 21 on page 32](#) shows the sample RSCS definition for a z/VM system that is not a member of a single system image (SSI) cluster. For simplicity, commented-out statements included in the sample definition are not shown here. For more information about the z/VM user directory and explanations of directory statements, see [z/VM: CP Planning and Administration](#).

```

IDENTITY1 RSCS2 password 32M 32M BG3
BUILD ON *4 USING SUBCONFIG RSCS-1
ACCOUNT 1 RSCS
OPTION ACCT5 MAXCONN 20486 SVMSTAT7 LANG AMENG8 SETORIG9
MACHINE ESA
XAUTOLOG GCS10
IUCV ANY
*-----*
* THE FOLLOWING STATEMENT IS OPTIONAL *
*-----*
*NOPDATA11
*-----*
NAMESAVE GCS12
IPL GCS PARM AUTOGR13
CONSOLE 01F 3215 T OPERATOR14
SPOOL 00C 2540 READER A
SPOOL 00D 2540 PUNCH A
SPOOL 00E 1403 A

SUBCONFIG RSCS-1
LINK RSCS 401 191 RR
LINK TCPMAINT 198 198 RR
LINK MAINT 190 190 RR
LINK MAINT 193 193 RR
LINK MAINT 19E 19E RR
MDISK 401 3390 2098 018 MH1W01 RR READ WRITE MULTIPLE15
MDISK 403 3390 2116 006 MH1W01 RR READ WRITE MULTIPLE15

```

Figure 21. Sample RSCS Virtual Machine Definition

Note:

1. The IDENTITY statement identifies this as a multiconfiguration virtual machine definition. When included in the directory for an SSI cluster, this type of definition can be configured to enable the user ID to be logged on concurrently to multiple members of the cluster. The version of the definition shown here is for a non-SSI system, so the definition includes only one active BUILD statement and its associated SUBCONFIG entry and is equivalent to a single-configuration virtual machine definition (traditional USER definition).
2. If you change the name of this virtual machine, you might need to update other files that refer to the RSCS user ID.
3. Specifies the CP privilege classes (based on the default classes provided with z/VM) for which this virtual machine is authorized:

G

Allows RSCS to use general user CP commands.

B

Allows RSCS to use the privileged CP MSGNOH command to suppress CP message headers.

Note: This is the *only* reason class B is specified here. Your installation might consider redefining privilege classes, which is described in [z/VM: CP Planning and Administration](#). You could define one class that includes the CP MSGNOH command only. Then, that class would be specified here instead of B. You could also specify other classes if you wish to use additional privileged CP commands at the RSCS virtual machine console. However, **do not** specify class D.

4. The BUILD statement identifies the SUBCONFIG entry for this system, which contains system-specific statements. An asterisk (*) in the system name field indicates this definition is included in an SSI-ready (not SSI-enabled) directory, and the system name defaults to the IPLed system. In an SSI-enabled directory, this definition will contain a BUILD statement and its associated SUBCONFIG entry for each member of the cluster.
5. Allows RSCS to issue DIAGNOSE code X'4C' to create accounting records. This is optional. If enabled, RSCS can produce accounting records for each file received or transmitted and pass them to CP to include in the accounting log. You must write RSCS exits to gather the accounting information and issue DIAGNOSE code X'4C'. For more information, see [z/VM: RSCS Networking Exit Customization](#).

You must also define the exit routines on the EXIT configuration statement. See [“EXIT Statement” on page 137](#).

6. Specifies the maximum number of IUCV connections allowed for the RSCS virtual machine. The value can be 1 - 65535. If TCP/IP connections are defined, a minimum value of 1000 is recommended.
7. Specifies that RSCS is a service virtual machine.
8. Specifies that American English is the language that will be set for RSCS at logon.
9. Specifies that RSCS can use DIAGNOSE code X'F8', subcode X'00' for the Secure Origin ID support.
10. Specifies that the GCS virtual machine can issue the XAUTOLOG command for the RSCS virtual machine. (This does *not* authorize GCS to issue the AUTOLOG command.) If your GCS virtual machine has class A or B privileges, this statement is not required.
11. Authorizes the RSCS virtual machine to use NOP CCWs to transfer data to CP spool files. Because RSCS uses NOP CCWs to transfer, store, and send files, RSCS automatically issues a SET NOPDATA ON command during initialization. Therefore, this statement in the RSCS definition is optional; it is shown in this sample as a reminder. For more information about the NOPDATA statement, see [z/VM: CP Planning and Administration](#).
12. Authorizes the RSCS virtual machine to access the restricted GCS named saved system. In most cases, the GCS named saved segment is restricted; however, if it is not restricted on your system, you should not include this statement in the RSCS definition. For more information about the NAMESAVE statement, see [z/VM: CP Planning and Administration](#).
13. Required if GCS is autologged.
14. Specifies the secondary user ID to control operations when the RSCS virtual machine is disconnected; this is optional.

Note: If your RSCS system is very active, RSCS will send many messages to this user ID.
15. Defines the RSCS production disks. In an SSI cluster, each member system will have its own set of RSCS production disks.

Identifying RSCS to z/VM

The SYSTEM NETID file identifies networking programs to z/VM. An entry should be added to SYSTEM NETID for RSCS, to allow users to communicate with other systems in the network, using the CMS TELL, NOTE, and SENDFILE commands. Add an entry to the file as follows:

```

:
:  serial1 nodeid2 rscsid3
:

```

Figure 22. RSCS Entry in the SYSTEM NETID File

Note:

1. The processor's serial number.
2. RSCS local node identifier, as defined on the LOCAL statement of the RSCS configuration file. For more information, see [“LOCAL Statement” on page 64](#).
3. The user ID of your RSCS virtual machine, defined on the IDENTITY statement in the virtual machine definition (see [“RSCS Virtual Machine” on page 31](#)).

RACF Toleration

If you have RACF® Node Protection implemented, you must bypass TAG Control for RSCS because there can be conflicts between the way RSCS utilizes the TAG command and the way it handles Node protection (based on the CP TAG command).

To do this, define a separate RACF VMEvent USERSEL profile for the RSCS machine and specify No Control for the TAG command. For example:

```
rac setropts classact (vmxevent)
rac rdefine vmxevent userse1.rscs
rac ralter vmxevent userse1.rscs addmem(tag/noctl)
```

You must then activate the profile:

```
rac setevent refresh userse1.rscs
```

In the preceding examples, RSCS refers to the name of the RSCS machine.

This will exclude RSCS from RACF authorization of TAG commands and allow the RSCS user ID to TAG any file with any value needed to continue its processing. This does not compromise any RACF VM node authorization checking done on TAG commands for any other user, therefore Node protection integrity for the final destination is maintained. For more information, see [z/VM: RACF Security Server Security Administrator's Guide](#).

Joining the GCS Virtual Machine Group

RSCS runs in the GCS operating environment, sharing certain GCS services with other members of a virtual machine group. Besides the RSCS machine, this virtual machine group includes a *recovery machine*, which is the first machine to join the group. Other members that you might define (especially if you plan to use SNA links) include VTAM and NetView®. [Figure 23 on page 35](#) shows what your virtual machine group might look like.

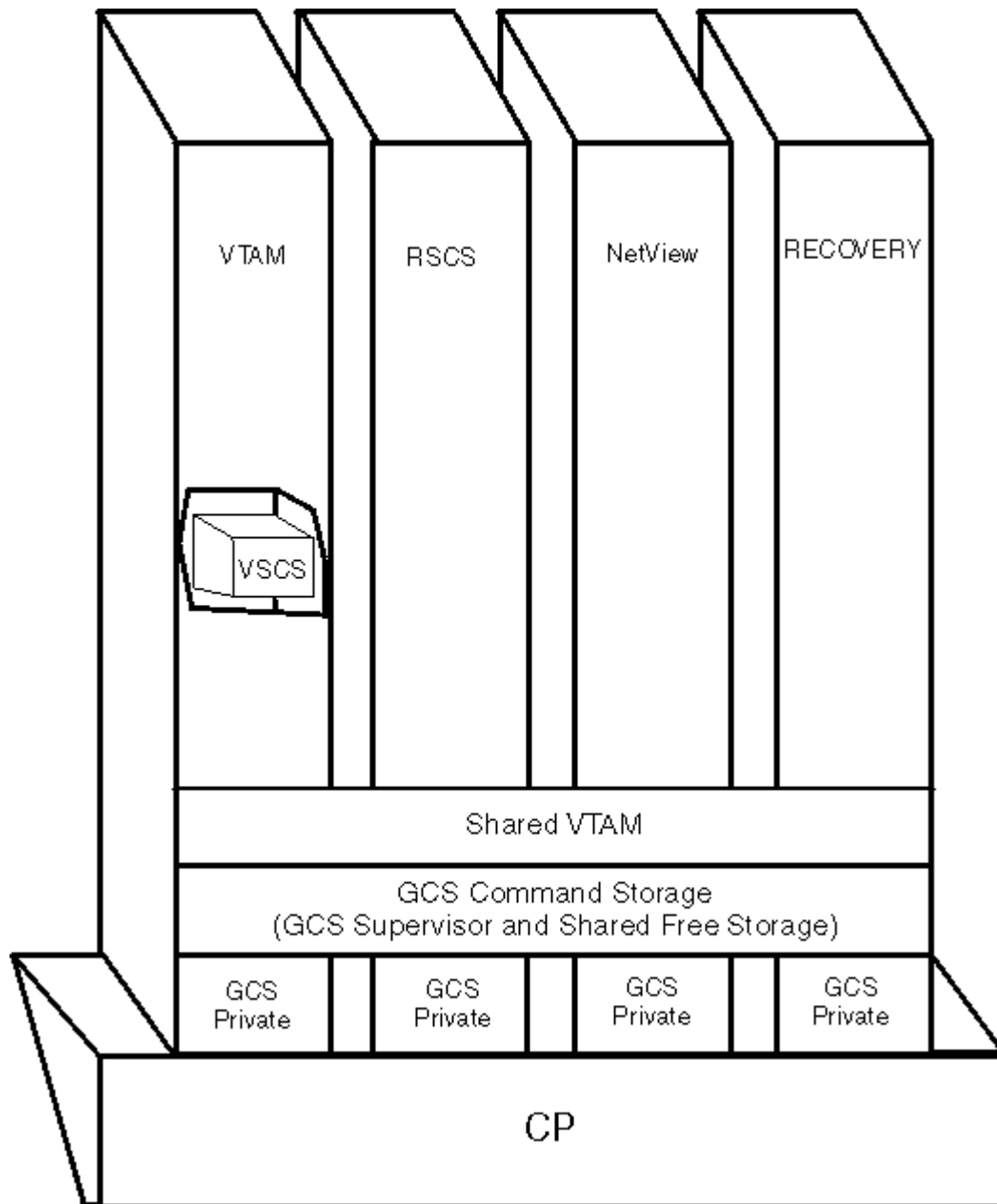


Figure 23. A Typical Virtual Machine Group

To join a virtual machine group, RSCS must be initialized from within the GCS environment. That means GCS must be IPLed first. To accomplish this, you must have already created a GCS shared segment and added the RSCS virtual machine's user ID to the GCS list of *authorized* users. For information about how to create the GCS shared segment and how to authorize user IDs, see [z/VM: Group Control System](#).

Logging on the RSCS Virtual Machine Automatically

The RSCS virtual machine can be logged on automatically. Because the recovery machine must be the first to join the GCS group, the PROFILE GCS exec of the recovery machine has been predefined to include an XAUTOLOG command for RSCS. (XAUTOLOG commands are also included for the RSCSDNS and RSCSAUTH virtual machines.) [Figure 24 on page 36](#) shows an example of the entry.

```

/* Invoked automatically when the recovery machine
   is IPLed. It autologs other machines in its
   GCS virtual machine group. */
:
'XAUTOLOG RSCS password'
'XAUTOLOG RSCSDNS password'
'XAUTOLOG RSCSAUTH password'
'XAUTOLOG ... '
:

```

Figure 24. RSCS Entry in the PROFILE GCS Exec for the Recovery Machine

Preparing for Non-SNA Connections

The following topics describe the tasks you need to perform to prepare RSCS to use non-SNA connections.

Defining Non-SNA Connections

You must define non-SNA line connections in the z/VM system configuration file (SYSTEM CONFIG). RDEVICE statements in this file describes I/O devices, control units, and channels that are attached to your real processor. For more information about the system configuration file, see [z/VM: CP Planning and Administration](#).

Figure 25 on page 36 shows a sample SYSTEM CONFIG file that defines a remote 3270 control unit to the system.

```

/*-----*
 * Sample SYSTEM CONFIG file definitions *
 *-----*/
/*-----*
 * This starts the RDEVICE statements for logical channel 0 *
 *-----*/
:
/*-----*
 * LOGICAL CHANNEL 0 TYPE (BY) LOCAL OPS CONSOLES AND PRINT *
 *-----*/
RDEVICE 0021-0030 TYPE 3279 MODEL 3 1
/*-----*
 * 3705 #1 NSC #1 00F0 ESC 0040-004F *
 *-----*/
RDEVICE 0040-004F TYPE BSC_ADAPTER 2
:
RDEVICE 00F0 TYPE 3705
/*-----*
 * THIS ENDS THE RDEVICE statements FOR LOGICAL CHANNEL 0 *
 *-----*/
:
/*-----*
 * Define primary and alternate system consoles *
 *-----*/
OPERATOR_CONSOLES 0021 0022 0023 0024 0027 0030 3

```

Figure 25. Defining Devices and Connections — Sample SYSTEM CONFIG

Note:

1. Defines the device addresses of 16 consecutive 3279 Model 3 displays with addresses starting at 0021.
2. Defines the device addresses of 16 consecutive 3705 line adapters. The TYPE operand specifies the line protocol to be used (BSC).
3. Completes the real I/O definition by defining the primary and alternate system consoles. The primary system console is defined at address 0021; this is the first 3279 Model 3 display defined in note “1” on page 36. The alternate system consoles are defined at addresses 0022, 0023, 0024, 0027, and 0030.

Defining a 37xx TCU

The following excerpts from a 37xx emulation program (EP) generation show examples of statements defining devices and connections that RSCS will use. Refer to the appropriate documentation for the particular 37xx telecommunication control units (TCU) that are in use for complete information on using and formatting statements.

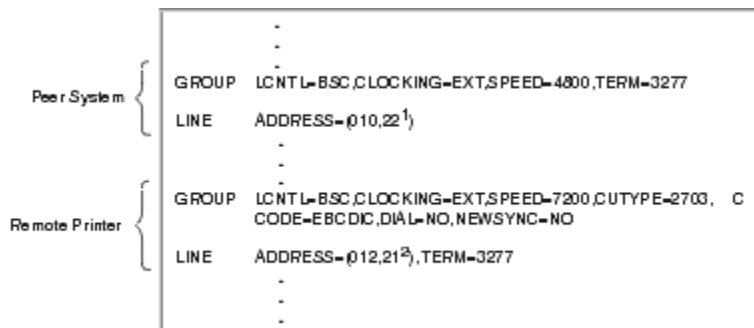


Figure 26. Examples of 37xx TCU Statements for Devices and Connections Used by RSCS

Note:

1. Corresponds to the address (in the TCU) of the line to the remote peer system as defined to z/VM.
2. Corresponds to the address (in the TCU) of the line to the remote printer's cluster control unit.

Requirements for Non-SNA Connections

For non-SNA connections, RSCS handles its own communications. It has specific requirements for communicating to and with the programs and devices it supports.

3270 Printers

RSCS requires a 327x Control Unit or integrated adapter for sending output to its printers. RSCS does not distinguish between local (channel-attached) and remote connections. z/VM handles this distinction. Connections to SNA 3270 printers should follow z/VM requirements.

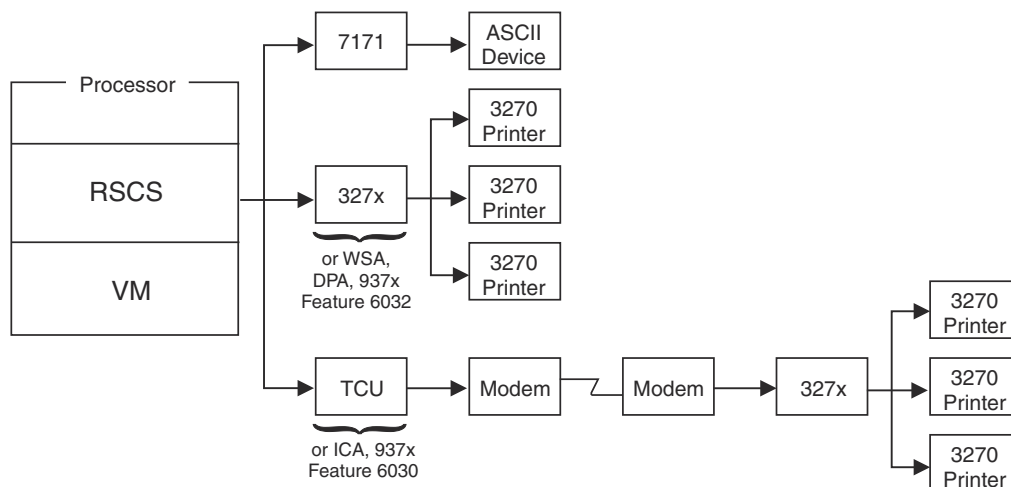


Figure 27. Requirements for Non-SNA Printers

RSCS supports the 3270 data stream for 3270 Information Display System printers. The RSCS non-SNA printer link driver (3270P) can communicate with any printer capable of receiving this form of data.

Depending on the capabilities of the printer, RSCS can support:

- Alphanumeric output files (base color only)

System Planning

- Graphics (GDDM) output files (monochrome or color)
- Line lengths greater than 132 characters
- Early Print Complete (EPC) feature
- Double byte character set (DBCS) output files
- Intelligent printer data stream (IPDS) output files for printers operating in GDDM mode

Before you connect a non-SNA printer to RSCS, check the hardware description or specifications of the 3270 printer (or printer operating in a 3270 mode) to discover which features that printer supports.

ASCII Devices

It is possible to drive ASCII printers and plotters from RSCS. An IBM 7171 ASCII Device Attachment Control Unit or an IBM 9370 ASCII Subsystem Controller can make an attached ASCII device appear as a 3270 printer to RSCS through emulation. The 7171 provides a means to allow a program to send ASCII data streams rather than 3270 (EBCDIC) data streams to an attached printer.

Remote Workstations

RSCS requires a real or emulated 2701 Transmission Control Unit (TCU) for communicating with remote workstations (half-duplex) to receive I/O. For example, a 37xx can be generated with an emulator program for this function.

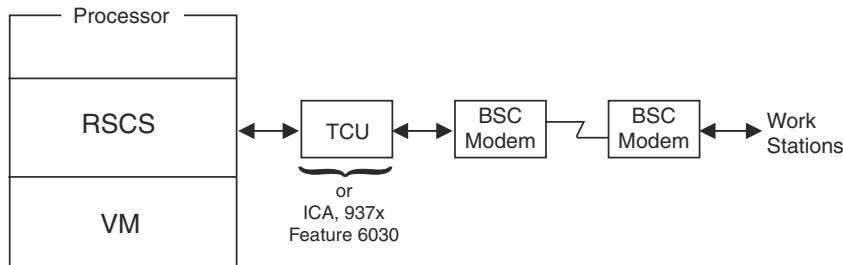


Figure 28. Requirements for Non-SNA Remote Workstations

RSCS supports non-multi-leaving workstations in secondary mode only. These devices must operate within the DATA-LINK Concept using the DATA-LINK control sequences and characters as described in the binary synchronous communications (BSC) documentation.

RSCS supports the following devices and any device that emulates one of these supported devices:

- 2770 Data Communication System
- 2780 Data Transmission Terminal
- 3780 Data Communications Terminal

RSCS supports the level of multi-leaving protocols as preprocessed by the workstation package H RTPB360. RSCS operates as either host or remote to this protocol.

Preparing for TCP/IP Connections

The following topics describe the tasks you need to perform to enable RSCS to use TCP/IP connections (LPD-type, LPR-type, TCPASCII-type, TCPNJE-type, TN3270E-type, UFT-type, and UFTD-type links).

Defining TCP/IP Connections

As [Figure 29 on page 39](#) shows, a TCP/IP network connection is required for RSCS to use any of the TCP connections.

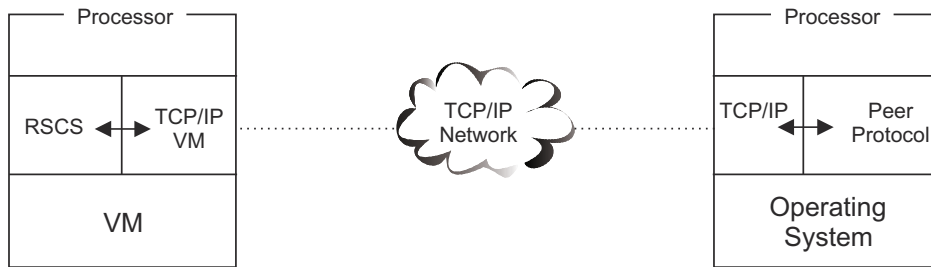


Figure 29. Requirements for TCP/IP Connections

For each TCP/IP-type link defined, RSCS uses one IUCV connection to communicate with the TCP/IP virtual machine. The port redirector task also uses one IUCV connection to each TCP/IP virtual machine for which it is listening for an incoming connect request to the local node for TCPNJE-type links. To ensure that a sufficient number of IUCV paths can be established, you should ensure an appropriate MAXCONN value is specified on the OPTION statement in the RSCS virtual machine definition. A minimum MAXCONN value of 1000 is recommended; this value might vary, depending on the needs of your installation.

For each LPD-type, LPR-type, TCPASCII-type, TCPNJE-type, UFT-type, or UFTD-type link defined, RSCS must know the following information:

- The host IP address (dotted decimal format) or fully qualified host (domain) name of the peer system
- The remote port number to which the link will connect
- Name of the TCP/IP virtual machine on the peer system

For LPR-type links, RSCS must also know the name of the default print queue to which files will be sent in the TCP/IP network. For TCPASCII-type links, a terminal server must be defined on the remote node. As [Figure 29 on page 39](#) shows, a peer NJE networking program also must be available on the remote node to establish a TCPNJE-type link.

In a TCP/IP network, each node or host system is assigned an address, called an IP address. The IP address uniquely identifies the host within the network and enables hosts to communicate with each other. For RSCS, the IP address is defined in dotted decimal format and must be a valid decimal number:

```
nnn.nnn.nnn.nnn
```

If the RSCS domain name server (RSCSDNS) is running, fully qualified host name addresses are also supported:

```
name.name.name.name
```

Attention

RSCS supports only IPv4 addressing; it does not support IPv6.

For more information about the RSCS domain name server, see [Chapter 12, “Setting Up the RSCS Domain Name Server,”](#) on page 113.

For more information about preparing TCP/IP connections, see [z/VM: TCP/IP Planning and Customization](#) and [z/VM: TCP/IP User's Guide](#).

Using TCP/IP Well-Known Ports

By default, TCP/IP restricts the use of well-known ports (1 - 1023) with the RESTRICTLOWPORTS operand on the ASSORTEDPARMS statement in the TCP/IP configuration file (PROFILE TCPIP). RSCS usage of well-known ports includes the following defaults:

175

TCPNJE-type links

515

LPD-type links

608

UFTD-type links

721 - 731

LPR-type links

900

RSCSDNS virtual machine

To use a well-known port when usage is restricted, the appropriate RSCS virtual machine (such as RSCS or RSCSDNS) must be authorized on a PORT statement in the TCP/IP configuration file. For more information, see [z/VM: TCP/IP Planning and Customization](#).

An alternative to using well-known ports for LPR-type links is to specify SECURE=NO on the PARM statement for the link or on the DEFINE or START command.

LPR Connections

LPR-type links are used to send data streams to a line printer daemon in a TCP/IP network. When defining an LPR-type link, you must specify the name of a default printer queue to which files will be sent.

You must also specify the name of an exit routine that is associated with the LPR-type link; the link will not start if an exit routine is not provided. LPR exit routines enable RSCS to customize the data, as needed.

LPD Connections

To receive print data streams from a TCP/IP LPR client for delivery to a network destination or printer, an LPD-type link acts as a gateway for processing files from the TCP/IP world to the NJE world through RSCS. The LPD link driver has been designed to comply with RFC 1179.

The LPD link driver will handle only one print data stream at a time. Because the driver has been designed to allow multiple *listens* on the same TCP/IP port number, more than one LPD link driver might be started to handle multiple concurrent incoming streams.

The LPD link driver will bind to well-known port 515 and listen for a TCP/IP LPR client to connect and send a print data stream. The port number on which the LPD driver listens might be overridden by defining a different port number on the PARM statement for the LPD-type link or on the DEFINE or START command. This allows for site installation testing and more flexibility.

The maximum data length supported when receiving any individual print file by the LPD link driver is 2 GB (2,147,483,647 bytes).

The only LPR print commands the LPD link driver supports are **Print any waiting jobs** and **Receive a printer job**.

This link driver is an input-only driver. Only data is transferred in. No data or messages will be transferred out over this link. Any files or messages queued to this link will remain queued until explicitly transferred or purged. Use RSCS routing capabilities to avoid having files or messages queued to this link. For example, add the following route to the RSCS configuration file to cause files or messages bound to an LPD-type link to instead be queued on a NOTIFY-type link:

```
ROUTE LPD TO *UNKNOWN
```


TCP/IP Machine Configuration file

If you currently are using VM TCP/IP LPSERVE as your VM daemon and wish to replace this with the RSCS LPD link driver, then you will need to make the following changes in the TCP/IP machine's configuration file:

1. Remove LPSERVE from the autolog list.
2. Change LPSERVE to RSCS for the port 515 reserve list.
3. Define the LPD-type link in RSCS with the appropriate exit (and customize if appropriate to match the LPSERVE configuration).

Note: In order to test the RSCS LPD-type link, you will first need to shut down LPSERVE or define a port other than 515 for RSCS to listen on. Most LPR commands do not allow you to connect to a port other than 515.

Configuring an RSCS TN3270E Printer Link

The TN3270E protocol supported by TCP/IP provides for the creation of 3270 printer sessions in addition to traditional display sessions. This link provides the same capabilities as the 3270P-type link (as described in “3270 Printers” on page 37), except the printer is attached via Telnet to the z/VM system. To make this capability available, perform the following steps:

1. Define an arbitrary name, called the LU name, in the TN3270E control statement in the PROFILE TCPIP file. This statement assigns a virtual line address to the printer session.
2. Enable the TN3270E Printer Management exit by using the TN3270EEXIT parameter of the INTERNALCLIENTPARMS statement in the PROFILE TCPIP file.
3. Define an RSCS TN3270E-type link that has a line address which matches the line address defined in Step 1. While not required, you might find administration and problem determination easier if the LU name is the same as the associated RSCS link name.
4. Make sure the user has the LU name configured in their TN3270E-capable emulator. This must be the same LU name defined in Step 1.

TCPASCII Connections

The TCPASCII-type link is similar to the ASCII-type link. Both links enable RSCS to send data streams to ASCII printers and plotters. However, unlike an ASCII-type link, a TCPASCII-type link can send data to an ASCII device that is attached to a terminal server in a TCP/IP network. A hardware terminal control unit, such as a 7171 controller, is not required.

TCPASCII-type links also use the ASCII printer and plotter exit routines to customize data streams to specific devices. The attention interrupt processing exit, however, has been modified. You must specify the name of an exit routine that is associated with the TCPASCII-type link; the link will not start if an exit routine is not provided.

TCPNJE Connections

TCPNJE-type links are defined by a unique local/remote IP address and port number pair. The TCPNJE-type link must know the IP address of the remote NJE node. Also, port numbers must be defined by the RMTPORT and LCLPORT operands on the PARM statement for the TCPNJE-type link. The link connects to the port defined by the RMTPORT operand. The LCLPORT operand identifies the port on which the port redirector task (on behalf of the link) will listen for connection requests from the remote node. Well-known port 175 is the default value for the RMTPORT and LCLPORT operands.

If the local and remote nodes are on different systems, the default port values for these operands can be used for these links. However, if a TCPNJE-type link is established between two RSCS virtual machines on the same z/VM system, you must specify different port values on the RMTPORT and LCLPORT operands.

If you anticipate that one or both of the nodes will have multiple IP interfaces defined to the TCP/IP stack, you might need to specify the IP address on which socket connect requests should be sent for the TCPNJE-type link. To do this, use the LCLHOST operand on the PARM statement for the link. The IP

address specified on the LCLHOST operand must match the IP address provided with the HOST operand on the remote side.

TCPNJE links can be encrypted through the TLS protocol. To secure your link, you will specify the label of an x.509 digital certificate on the PARM statement for your TCPNJE link. The certificate and its corresponding TLSLABEL must exist in the TLS/SSL Server certificate database. For additional information on the TLS/SSL Server and managing its certificate database, see [z/VM: TCP/IP User's Guide](#) and [z/VM: TCP/IP Planning and Customization](#).

Note: Before implementing a TCPNJE-type link to JES2, you should check the RSCS and JES2 PSP buckets for required service.

UFT Connections

UFT-type links send data streams to a remote TCP/IP UFT server for distribution to a user, acting as a sendfile gateway between the RSCS NJE world and the TCP/IP UFT world. The UFT-type link is single threaded, handling one file at a time, although successive files might be delivered to different hosts and user IDs. Multiple outgoing UFT data streams can be provided by starting multiple UFT-type links and having them be members of an RSCS GROUP.

The UFT-type link provides I/O, protocol, and error handling. All policy decisions (for example character set translation from EBCDIC to ASCII or binary) are handled by exits. In addition, the exit is responsible for extracting the data from the NETDATA file to send to the remote UFT daemon. The TCPIP DATA file must be updated to include the following statement:

```
UFTserverID RSCS
```

UFTD Connections

The UFTD-type link is a UFT daemon which receives data streams from a TCP/IP UFT client for delivery to a network destination or printer, acting as a gateway for processing files from the TCP/IP world to the NJE world through RSCS.

The UFTD link driver will handle only one data stream at a time. Because the driver has been designed to allow multiple *listens* on the same TCP/IP port number, more than one UFTD link driver might be started to handle multiple concurrent incoming streams.

The UFTD link driver will bind to well-known port 608 and listen for TCP/IP UFT clients to connect and send a data stream. The port number on which the UFTD link driver listens can be overridden by defining a different port number on the PARM statement for the UFTD-type link or on the DEFINE or START command. This allows for site installation testing and more flexibility.

The maximum data length supported when receiving any individual file by the UFTD link driver is 2 GB (2,147,483,647 bytes).

This link driver is an input-only driver. No data or messages will be transferred out over this link. Only data is transferred in. Any files or messages queued to this link will remain queued until explicitly transferred or purged. Use RSCS routing capabilities to avoid having files or messages queued to this link. For example, add the following route to the RSCS configuration file to cause files or messages bound to a UFTD-type link to instead be queued on a NOTIFY-type link:

```
ROUTE UFTD TO *UNKNOWN
```

TCP/IP Machine Configuration file

If you currently are using VM TCP/IP UFTD as your VM UFT daemon and wish to replace this with the RSCS UFTD link driver, then you will need to make the following changes in the TCP/IP machine's configuration file:

1. Remove UFTD from the autolog list.
2. Change UFTD to RSCS for the port 608 reserve list.

- Define the UFTD-type link in RSCS with the appropriate exit (and customize if appropriate to match the UFTD configuration). The following statement must be removed from the TCPIP DATA file:

```
UFTserverID UFTD
```

Note: In order to test the RSCS UFTD-type link, you will first need to shut down UFTD or define a port other than 608 for RSCS to listen on.

Preparing for Networking Links

The networking links provide for communications between RSCS and other NJE-compatible subsystems that follow the NJE transmission protocol (GATEWAY-type, NJE-type, SNANJE-type, and TCPNJE-type) or provide special services (LISTPROC-type).

Any NJE-type, SNANJE-type, or TCPNJE-type link that is connecting to a JES2 subsystem must use a PATHMGR=NO or CONNECT statement in the JES2 subsystem's initialization deck. When you are preparing to establish NJE-type or SNANJE-type links, contact the system administrators of the remote JES2 subsystems to remind them to add a CONNECT statement for your node. They must also add CONNECT statements for all nodes to which files will be sent *through your node*.

Attention

- For each defined RSCS stream, JES2 must have one SYSIN receiver and one SYSOUT receiver. If you do not have this set up, RSCS will do one or both of these:
 - Issue message DMT930E
 - Hang one file without transmitting any records
- Only one SNANJE-type link can be defined between one RSCS node and the same remote NJE node.

RSCS has no way of knowing the specific requirements for GATEWAY-type links. Before you can start a GATEWAY-type link, your installation must write the exit that uses the Gateway Programming Interface (GPI). The exit, not RSCS, defines the requirements. For information about writing the exit, see [z/VM: RSCS Networking Exit Customization](#).

LISTPROC-type links require no preparation, other than starting the link driver.

For information about TCPNJE-requirements, see [“TCPNJE Connections” on page 41](#).

Requirements for Non-SNA Networking Links

RSCS can communicate with peer NJE systems (for example, MVS™) that support the following link protocols:

- Binary Synchronous Communication (BSC)
- Channel-to-channel (CTC)
- Transmission Control Protocol (TCP); see [Figure 29 on page 39](#)

The requirements for establishing non-SNA networking links to these systems vary, depending on the specific type of connection you are using.

For example, when using BSC links, RSCS requires a real or emulated 270X Transmission Control Unit (half-duplex) to communicate with other systems (see [Figure 30 on page 44](#)).

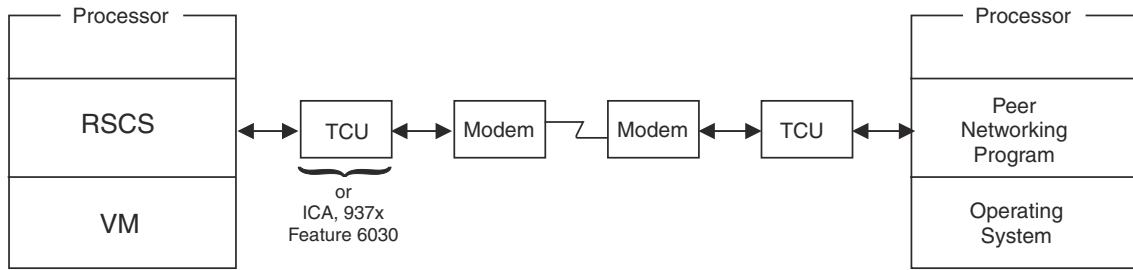


Figure 30. Requirements for Non-SNA Networking Links (BSC)

However, as [Figure 31 on page 44](#) shows, when using the CTC protocol, a CTCA, ESCON CTCA, FICON CTCA, or 3088 attachment is required between the systems.

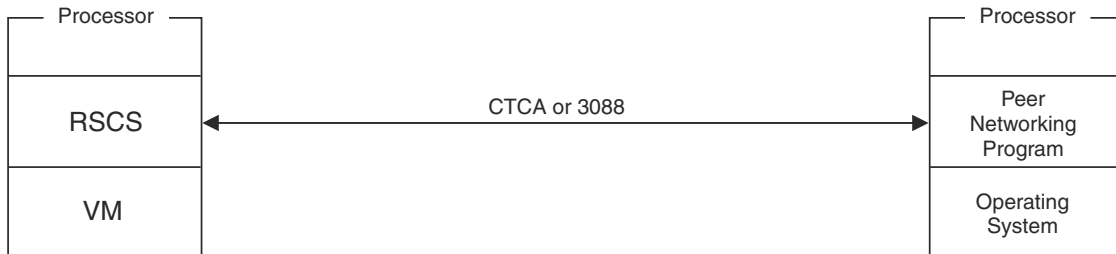


Figure 31. Requirements for Non-SNA Networking Links (CTCA or 3800)

Requirements for SNA Networking Links

RSCS also supports communications with peer NJE systems that support the Systems Network Architecture (SNA) LU Type 0 protocol. An SNA connection, such as a VTAM link, must be established between each system. Communication connections to SNA systems should follow VTAM requirements (see [Figure 32 on page 44](#)).

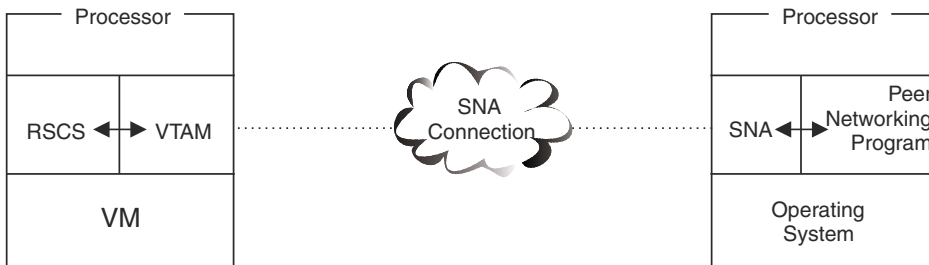


Figure 32. Requirements for SNA Networking Links (SNANJE-Type)

Preparing VTAM for SNA Connections

Attention

The examples in this topic are intended for illustrative purposes only. See the *VTAM Resource Definition Reference* for specific information on the content and format of required files. See the [“Bibliography” on page 319](#) for other titles that can be helpful.

To use VTAM for communicating over SNA connections, RSCS runs as a VTAM application. RSCS, therefore, must be defined to VTAM as one of its applications. This is done by creating a new major node definition or adding to an existing one. For an example, see [Table 4 on page 45](#).

Table 4. Defining RSCS as a VTAM Application

RSCSNET VTAMLST			
RSCSNET	VBUILD	TYPE=APPL	
RSCS ¹	APPL	ACBNAME=RSCS, ¹	C
		MODETAB=RSCSTAB, ²	C
		DLOGMOD=RSCSNJE0, ³	C
		AUTHEXIT=YES, ⁴	C
		AUTH=(ACQ), ⁵	C
		VPACING=3 ⁶	
		:	

Note:

1. The ACBNAME will be used on the RSCS NETWORK command that starts RSCS/VTAM communications. (If not specified, ACBNAME defaults to the label on the APPL statement.)

```
NETWORK START APPLID RSCS
```

If not specified in the command, APPLID defaults to the local node ID. In RSCS, the term *local node ID* refers to the node ID of its own local system. It is defined on the LOCAL statement in the configuration file. For more information on the use of common names for these parameters, see [“LOCAL Statement”](#) on page 64.

2. Identifies the table that includes RSCS logon mode entries in text format. For more information, see [Appendix B, “Logon Mode Table Entries,”](#) on page 307. If omitted, the default table supplied with the VTAM product is used.
3. Specifies the default logon mode entry within the table to be used for sessions. (If omitted, the first entry in the table is used.) To override this, use the LOGMODE operand on the LINKDEFINE statement, or use the DEFINE or START command.
4. This parameter is required. It authorizes RSCS to use expedited processing in its exit routines. For RSCS, they will run in supervisor state, key 0.
5. This parameter is required. It authorizes RSCS, as a primary logical unit, to start a session by simulating a logon request (SIMLOGON macro).
6. Specifying this parameter uses the VTAM pacing facility, which controls the number of buffers that VTAM allows to flow at one time during a session between the sending and receiving nodes. Specifying a pacing value reduces the chance that a large file will overload the transmission capacity of a node. Select a valid pacing value in the range 0 - 63.

A pacing value is also supplied in the logon mode table. For more information, see the pacing value entries for RSCSNJE0 in [Appendix B, “Logon Mode Table Entries,”](#) on page 307.

SNA 3270 Printer Support

For SNA 3270 printer support, your SNA printers must be defined to VTAM in LU_LU type sessions. An LU (logical unit) is a port through which your printers access the SNA network to communicate with VTAM.

All 3270 printers can be supported by LU_T0 or LU_T3 type sessions. Some LU_T0 and LU_T3 printers can support the GDDM Extended 3270 (non-IPDS) graphics capability. Therefore, if graphics are being supported, the session type must be either an LU_T0 or LU_T3.

Some LU_T0 and LU_T1 printers (for example, 3812 and 4224) can support GDDM IPDS graphics capability. If IPDS graphics are being supported, the session type must be either LU_T0 or LU_T1.

Some 3270 printers can also be supported by LU_T1 type sessions. The LU_T1 sessions use SCS (SNA Character String) data stream and SNA protocols which help drive printers more efficiently. All LU_T1 type sessions have the following characteristics:

- Better printing performance (closer to rated speeds) than LU_T0 or LU_T3 sessions.

5. Identifies the table that includes RSCS logon mode entries in text format. For more information, see [Appendix B, “Logon Mode Table Entries,” on page 307](#). If omitted, the default table supplied with the VTAM product is used.

If you define your own table, or modify one of the supplied tables, see [Appendix B, “Logon Mode Table Entries,” on page 307](#) for any specific values. These values must be present for a logon mode table to be acceptable for use with RSCS.

If you are sharing printers with VSCS or other VTAM applications, their logon mode entries should be in this table.

6. Specifies the default logon mode entry within the table to be used for sessions. (If omitted, the first entry in the table is used.) To override this, use the LOGMODE operand on the LINKDEFINE statement or the DEFINE or START command.

If you are sharing printers with VSCS or other VTAM applications, do not specify a default here.

7. Specifies the name of the application program to which the LU will be automatically logged on when the LU is started. This keyword is optional. If this feature is used, the application name specified should match the ACBNAME value on the APPL statement.

Requirements for SNA 3270 Printers

RSCS does not distinguish between local (channel-attached) and remote connections. VTAM handles this distinction; connections to SNA 3270 printers should follow VTAM requirements.

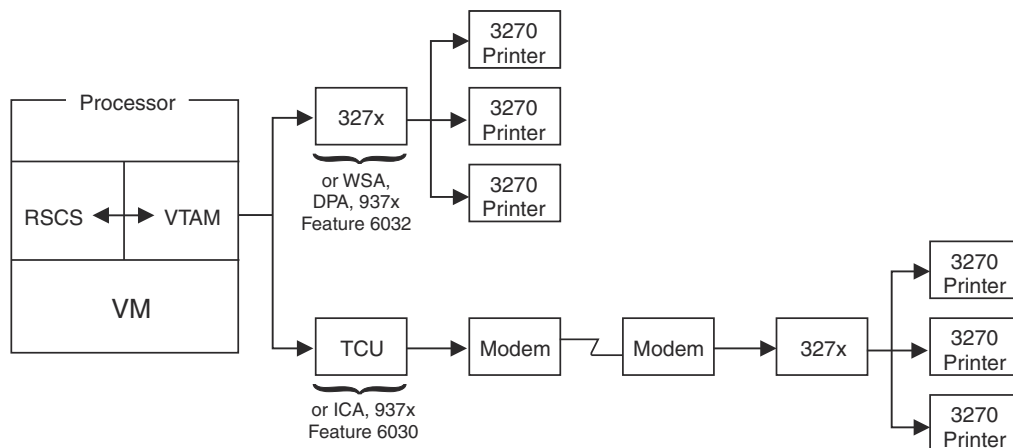


Figure 33. Requirements for SNA Printers

RSCS supports SNA 3270 printers in LU_T0, LU_T1, and LU_T3 mode. The RSCS SNA printer link driver (SNA3270P) can communicate with any printer capable of receiving this form of data.

Depending on the capabilities of the printer, RSCS can support:

- Alphanumeric output files (base color only)
- Graphics (GDDM) output files (monochrome or color)
- Line lengths greater than 132 characters
- Double byte character set (DBCS) output files
- Intelligent printer data stream (IPDS) output files for printers operating in GDDM mode

Before you connect a SNA printer to VTAM and establish a session with RSCS, check the hardware description or specifications of the SNA 3270 printer (or printer operating in an SNA 3270 mode) to discover which features that printer supports.

Requirements for SNA RJE Workstations

RSCS requires that a System/36 RJE workstation be connected to VTAM. VTAM establishes the requirements for this connection.

To establish a connection with a System/36 multiple session RJE (MSRJE) workstation in an LU_T1 session, you must define the System/36 as a physical unit, with each MSRJE subsystem defined as an LU to VTAM. RSCS supports a subset of the LU_T1 protocol under the System/36 implementation.

Table 7 on page 48 shows an example of how to define System/36 RJE workstations to VTAM.

<i>Table 7. Defining a Local SNA RJE Workstation to VTAM</i>			
RSCSRJE VTAMLST			
RSCSRJE	VBUILD	TYPE=CA	
RJEGROUP	GROUP	LNCTL=SDLC, DIAL=NO, VPACING=2, PACING=2	C
RJELINE	LINE	ADDRESS=030, REPLYTO=24	
RJEPU	PU	ADDR=C1, MAXDATA=256, PUTYPE=2, SSCPFM=USSCS	C
RJELU02	LU	LOCADDR=02, MODETAB=RSCSTAB, DLOGMOD=RSCSRJE1	C
RJELU03	LU	LOCADDR=03, MODETAB=RSCSTAB, DLOGMOD=RSCSRJE1	C
RJELU04	LU	LOCADDR=04, MODETAB=RSCSTAB, DLOGMOD=RSCSRJE1	C
:			

Cross-Domain Support

RSCS communicates with other SNA systems through VTAM. On those other SNA systems, RSCS communicates with printers and peer networking programs. The peer networking programs (like RSCS) must be defined as VTAM applications.

To allow for this *cross-domain* communication, you must define the following to VTAM (on your system):

1. The systems (VTAMs) through which RSCS will communicate. (For example, note the CTC adapter connection.)

NETCTC VTAMLST			
NETCTC	VBUILD	TYPE=CA	
NETGRP	GROUP	LNCTL=CTCA	
LINE0102	LINE	ADDRESS=705	
PU0102	PU		
:			

2. The VTAM applications (peer networking programs) and printers on the other systems that RSCS will communicate with.

The following is an example of a file containing CDRSC (Cross Domain Resource) statements to identify the major node name (application or printer) and name of the cross-domain resource manager for that node. The next example shows a file in the other system that defines the peer networking program as a VTAM application.

NETCDRSC VTAMLST (your system)			
:			
NETCDRSC	VBUILD	TYPE=CDRSC	
PEER	CDRSC	CDRM=PERCDRM	
PRINTER	CDRSC	CDRM=PERCDRM	
:			

OTHERSYS VTAMLST

```
OTHERSYS    VBUILD    TYPE=APPL
PEER        APPL      ACBNAME=PEER,AUTH=(ACQ)
:
```

See [Table 5 on page 46](#) and [Table 6 on page 46](#) for examples of defining printers to VTAM.

- The subarea of the application. In the following example, the file contains CDRM (Cross Domain Resource Manager) statements to define subareas managed by various cross-domain resource managers.

NETCDRM VTAMLST

```
NETCDRM     VBUILD    TYPE=CDRM
RSCSNET     CDRM      SUBAREA=1
PERCDRM     CDRM      SUBAREA=2
:
```

- The path to the other domain. The following example shows a PATH statement defining a path to the other domain.

PATH0102 VTAMLST

```
PATH0102    PATH      DESTSA=2,...
```

Supplying Logon Mode Table Entries

A logon mode table is used by VTAM in communicating with other logical units (LUs). Samples of logon mode table entries (to be defined by the VTAM which owns the LU) are supplied with RSCS in the RSCSTAB ASSEMBLE file. [Appendix B, “Logon Mode Table Entries,” on page 307](#) shows the contents of this file.

When you are preparing to establish SNANJE-type, SNARJE-type, or SNA3270P-type links, contact the system administrators of remote VTAM subsystems to ensure that a logon mode table entry is defined for your node.

See the samples in [Appendix B, “Logon Mode Table Entries,” on page 307](#) for specific values to be used in defining these table entries to be used by VTAM. These values must be present for a logon mode table to be acceptable for use with RSCS. If any of the table entries are changed, RSCSTAB ASSEMBLE must be re-assembled, under VTAM, before executing VMFLKED.

The name of this file containing the RSCS entries can be added to the others listed in the RSCSTAB LKEDCTRL file (see [Table 8 on page 49](#)).

Table 8. TEXT File Containing Logon Mode Table for RSCS

RSCSTAB LKEDCTRL

```
:
```

```
INCLUDE RSCSTAB
NAME RSCSTAB(R)
```

The members in this file must be link-edited (VMFLKED command) into a LOADLIB file (identified by a GLOBAL statement in VTAM's PROFILE GCS exec).

Definitions for SNA Networking Links

To connect a VM system and a peer NJE system through VTAM, RSCS and the peer NJE system must be defined as VTAM applications. The following examples show how to define RSCS and a peer NJE system to VTAM.

RSCS CONFIG (your system)

```
LOCAL YOURSYS * VMRSCS
LINKDEFINE OTHERSYS TYPE SNANJE LUNAME PEER
:
```

RSCSNET VTAMLST (your system)

```
RSCSNET    VBUILD    TYPE=APPL
RSCS       APPL      ACBNAME=VMRSCS, ...,
:
```

RSCS CONFIG (other system, if peer system, is a VM system)

```
LOCAL OTHERSYS * PEERNET
LINKDEFINE YOURSYS TYPE SNANJE LUNAME RSCS
:
```

OTHERNET VTAMLST (other system)

```
OTHERNET    VBUILD    TYPE=APPL
PEER        APPL      ACBNAME=PEERNET, ...,
:
```

Note: The link ID must be the name of the node (local ID specified on the LOCAL statement) to which the link connects, unless you specify the NODE operand on the LINKDEFINE statement. If you specify NODE, you can use a nickname as the link ID. For more information about using NODE, see Usage Notes [“1”](#) on page 156, [“2”](#) on page 157, and [“3”](#) on page 157.

Virtual Storage Considerations

The amount of virtual storage that RSCS needs varies based on your network configuration. 32 MB is predefined in the sample definition for the RSCS virtual machine. IBM recommends that you define 32 MB or more because:

- No matter how much storage you define, RSCS uses only what it needs. CP can page out whatever storage is left.
- You are less likely to run out of virtual storage. RSCS stops if it runs out of virtual storage during initialization. RSCS might stop links or file transmission if it runs out of virtual storage after initialization.

Calculating Virtual Storage

If you need to conserve virtual storage space and cannot afford to define a large virtual storage size, you must consider the size and complexity of your network during its peak activity. The larger and more complex your network, the more storage RSCS needs.

As [Figure 34 on page 51](#) shows, you should start with a base of 7 MB and ask yourself the following questions:

- How many links did you define? (You need approximately 450 bytes of storage per link.)

- How many links are active? On each active link:
 - How many buffers are there?
 - How large is the buffer size?
 - How many streams are there?

To calculate the amount of storage for an active link, multiply the number of buffers by the buffer size. (Printers need one buffer, workstations need two buffers, and networking links need three buffers.) Then add 6 KB for general link overhead and add 1 KB for each stream.

- How many tag shadow elements did you reserve? (You need approximately 40 bytes of storage for every tag shadow element over the 20,000 minimum.)
- Are you storing GCS or VTAM within or above the limits of RSCS virtual storage? (If you store GCS or VTAM in the limits of RSCS virtual storage, add the size of the GCS or VTAM segment.)

```

7 + (Number of defined links x 450)
+ ((Number of buffers x buffer size)
  + (Number of streams x 1 KB) + 7 KB)
  -- For all active links
+ ((Number of tag shadows over 20,000) x 40)
+ Size of GCS or VTAM segment
  -- If within RSCS virtual storage

```

Figure 34. Formula for Calculating Storage

Defining the RSCS Virtual Storage Size

After deciding the virtual storage size, you must enter it on the IDENTITY statement in the RSCS virtual machine definition. As the following example shows, the virtual storage size is specified as the *stor* value.

```
IDENTITY rscsid password stor mstor BG
```

For more information about the RSCS virtual machine definition, see [Figure 21 on page 32](#).

Considerations When Defining or Starting RSCS Links

Virtual storage might become a critical resource for the number of links that are being defined at your installation. If this occurs, you should specify the ASTART and ITO operands on the LINKDEFINE statement or DEFINE command for the applicable links.

This approach ensures optimal storage management because activated links that are idle for a lengthy period of time can use much needed storage. Auto-start links will not be activated until a file is enqueued to it. When the ITO operand is specified and there is no activity on that link for the period of time equal to the ITO value, RSCS automatically deactivates the link. When RSCS deactivates a link because of an ITO "time out", the link retains its auto-start capability. RSCS will disable the auto-start capability of a link if the link was deactivated for these reasons:

- The link abends.
- An unrecoverable line error occurs.
- A remote workstation node sends a signoff record.
- A operator issues an RSCS DRAIN, FORCE, or STOP command for the link.

Note: This does not work for LPD-type and UFTD-type links. These links must remain active all the time, as they are waiting on connect requests from TCP/IP clients.

In addition, it is important to determine the appropriate ITO value for your installation. This value determines how many minutes an auto-start link can have no activity on it before it's deactivated.

Based on the activity of your network, the ITO value significantly contributes to the efficiency of storage utilization. For more information on the auto-start links, see [z/VM: RSCS Networking Operation and Use](#).

Performance Considerations

Because RSCS runs in a virtual machine on z/VM, it must compete for system resources, such as processor time or storage, with other virtual machines. The topics in this section describe areas in which performance problems might occur; it also contains suggestions for enabling RSCS to operate at maximum capability.

Common Problems

The following situations might occur in your RSCS network if RSCS does not receive sufficient z/VM resources.

- BSC connections time-out

If RSCS performance is degraded on the local system, the remote node of a BSC connection (RJE-type, MRJE-type, or NJE-type link) might time out. The resulting line errors might cause the remote node to end the session.

- Many files await processing

When RSCS is not receiving sufficient z/VM resources, a backlog of files might build in CP spool for RSCS or at the remote peer systems. When the number of files in the RSCS virtual machine's reader reaches the value specified on the SLOWDOWN command or the limit imposed by z/VM, RSCS will not receive additional files from remote peer systems. This can create a backlog of files on the remote systems that can, in turn, slow file transmission throughout the RSCS network.

- File processing speed

When RSCS is not receiving sufficient z/VM resources, it might not be able to maximize its use of certain device features. For example, RSCS printer links might not be able to take full advantage of high-speed printers in the network. Also, networking links might not benefit from the bandwidth of the communication lines to the remote peer systems.

The transfer rate of files across TCPNJE-type links is dependent on the buffer sizes defined for the link and within the TCP/IP stack. The available buffer sizes within the TCP/IP stack should match or be larger than the buffer size defined on the TCPNJE-type link PARM statement.

z/VM Resource Considerations

The RSCS virtual machine primarily competes for three z/VM resources:

- Instruction processing
- Processor Storage
- Spooling

For more information about z/VM system performance considerations, see [z/VM: Performance](#). Performance considerations for guest virtual machines are also helpful when determining performance requirements for RSCS.

Instruction Processing

As a server, when RSCS receives work, it requires access to the instruction processing resources on the system processor.

To ensure access to these resources, the quick dispatch option should be assigned to the RSCS virtual machine. This option is enabled by specifying the QUICKDSP operand on the OPTION statement in the RSCS virtual machine definition. The z/VM system administrator can also issue the CP QUICKDSP command to enable this option, if the RSCS virtual machine is logged on. The quick dispatch option places

the RSCS virtual machine in CP queue 0; this ensures that RSCS will have access to system resources when they are needed.

The system-resource access priority for RSCS is determined by the value specified on the SHARE directory statement or by the CP SHARE command. That is, the z/VM system administrator can specify the percentage of system resources that the RSCS virtual machine will receive. This value can be adjusted according to the needs of your installation.

For more information about these directory statements, see *z/VM: CP Planning and Administration*. For more information about the QUICKDSP and SHARE commands, see *z/VM: CP Commands and Utilities Reference*.

Processor Storage

When a RSCS task attempts to access a page of storage that resides in paging space, z/VM causes the entire RSCS virtual machine to enter a wait state. While z/VM is satisfying this storage access, RSCS cannot perform any other work.

To reduce this problem, the z/VM system administrator can issue the CP RESERVED command for the RSCS virtual machine. Using the RESERVED command, a number of real storage pages can be reserved for exclusive use by RSCS. When sufficient storage is reserved, RSCS can process work with little or no storage paging.

RSCS can also use the pseudo page fault support in GCS. To enable this support, the SET PAGEX ON command should be added to the PROFILE GCS file for the RSCS virtual machine. With this support, GCS will place the individual link driver in a page wait while CP processes the paging request; however, GCS will continue to dispatch other RSCS link driver tasks.

For more information about the RESERVED and SET PAGEX commands, see *z/VM: CP Commands and Utilities Reference*.

Spooling

The CP spool is the DASD storage that contains all of the files for RSCS processing. Files created by local users are sent to the reader of the local RSCS virtual machine in spool. Files that are received from remote peer systems are written to CP spool. In addition, RSCS temporarily saves store-and-forward files into CP spool.

If performance problems are occurring in the CP spool, RSCS performance will also be degraded. The following items might affect spooling performance:

- DASD contention, involving:
 - Spool space volumes
 - Check point area
 - Warm start area
- Master processor contention on multiple processor systems
- Security access control system (for example, RACF) performance

Chapter 6. Planning for RSCS Configuration

Planning for RSCS configuration involves collecting data and making informed decisions about the needs of your site. This section contains topics that describe how to provide RSCS with information about your network and how RSCS uses that information. It also describes options, alternatives, and other considerations that should be evaluated as you develop a plan for configuration. The following checklist contains items to consider when planning for configuration.

Note: In addition to the configuration information described in this section, *z/VM: RSCS Networking Exit Customization* provides information about customer interfaces that allow you to customize the operation of RSCS.

Planning Checklist for RSCS Configuration

- **Coordinate** destination identifiers and passwords with system programmers responsible for configuring RSCS at remote locations.
- **Gather** information for creating the RSCS events file. The person who creates the events file will need the following information:
 - Which events are to be scheduled?
 - How often should the event be scheduled?
 - Will there be any special events for nights, weekends, or holidays?
- **Gather** information for creating the RSCS configuration file. The person who creates the configuration file will need the following information to define your network to RSCS:
 - Node IDs for each node
 - Host IP address of the remote node, if applicable
 - What each node is:
 - Non-SNA 3270 printer (model, features)
 - SNA 3270 printer (model, features)
 - LPR printer or LPD daemon
 - RJE non-multi-leaving workstation
 - RJE multi-leaving workstation
 - RJE workstation using SNA
 - NJE peer (SNA, non-SNA, or TCP/IP)
 - ASCII printer or plotter that is attached to a control unit, daemon, or terminal server
 - A Unsolicited File Transfer (UFT) client or daemon
 - Whether RSCS is to communicate over a direct connection or indirectly through another peer system
 - Whether there will be more than one link to this node
 - Whether alternate routes need to be defined
 - Whether to check for routing loops
 - Whether a link is to be started automatically
 - Addresses of virtual channels to be reserved
 - Any reserved addresses for dynamic allocation of BSC communications lines
 - Type of dump to request for RSCS failures and who will receive the dump
 - User IDs (and node IDs, if not local) of users who will receive line traces
 - Devices (ports) available for auto-dial links

Configuration Planning

- Phone numbers for auto-dial links
- Devices (ports) available for auto-answer links
- Passwords for auto-answer links
- ITO settings, especially for long distance links
- Whether a fanout link should be specified
- Which default FCB images will the link handle
- Whether NOTIFY-type links will be defined and their corresponding message templates
- Whether a list processor will be defined
- User IDs (and node IDs, if not local) of any alternate operators
- Link IDs that alternate link operators will operate
- How often message traffic preempts file traffic
- Whether to suppress CP message headers
- Whether to suppress file transmission messages
- Type of operator form (to match CP usage)
- Whether any sections of your configuration file are critical to the operation of your local node
- What print queue names, if any, are required
- Whether RSCS is communicating with a line printer or UFT daemon
- Remote TCP/IP fully qualified host name or dotted decimal address and IP port
- Whether RSCS dynamic authorization (RSCSAUTH) is required and enabled
- **Identify** whether exit routines will be used.
 - If a sample exit routine is available, does it meet your needs, or will additional function be required for your site?
 - Who will write any new exit routines or modify existing samples?
- **Determine** whether the language to be used for local and remote messages will be American English or uppercase English.

RSCS Configuration File

Configuration of RSCS starts with the RSCS configuration file. Statements in the configuration file define the network in which RSCS will operate. They also define operational parameters for use within that network. RSCS comes with a sample configuration file that you can either use as a model or edit for actual use (see “[RSCSSMPL CONFIG](#)” on page 298)

The configuration file statements fall into three categories:

Network definition statements

Identify the components that make up your network by their node names (node IDs).

RSCS functions

Define how RSCS will operate within your network.

CP functions

Direct how RSCS uses certain CP functions.

[Table 9 on page 57](#) lists the configuration statements that pertain to each of these categories.

Table 9. Categories of Configuration File Statements

Category	Configuration File Statements		
Network Definition Statements	DEST	LOCAL	
	LINKDEFINE	ROUTE	
RSCS Functions	AUTH	LANGUAGE	SAFCLAS
	CHANNELS	LINKDEFINE	SETMSG
	DEST	LINKTYPE	SHADOWS
	EXIT	OPTION	SLOWDOWN
	FCB	PARM	TOLERANCE
	FORM	PORT	TRACEDEST
	HIDECHARACTER	RECOVERY	UPARM
	IMBED	REROUTE	
	ITRACE	RETRY	
CP Functions	DUMP	MSGNOH	OPFORM

Note: Because DEST and LINKDEFINE provide RSCS information and network definition information, they belong in both categories.

This section provides a brief description of each of these configuration file statements. If you want more information about a specific statement, see [Chapter 15, “RSCS Configuration File Statements,”](#) on page 129. If you want to see an example of how to use the statement, see [“RSCSSMPL CONFIG”](#) on page 298.

Understanding Identifier Naming Conventions

Most RSCS configuration statements (and commands) require you to specify some sort of identifier (ID). For example, when you define a link, you must specify the name (link ID) of the link you are defining.

RSCS commands and statements use the following types of identifiers:

user ID

Name by which z/VM (and RSCS) knows a user

node ID

Name of a system, device, or workstation

link ID

Name of the link from your system to another node

group ID

Name of a collection of nodes

Within each identifier type, the identifiers should be unique. You cannot have two users with the same name on the same node, two links with the same name, and so forth.

Between identifier types, the identifiers should be unique to avoid confusion. If you create node IDs, group IDs, user IDs, and so forth with the same name, you may send something across the network and discover that it did not arrive at the destination you thought it would. The only exception to this rule is that link IDs and node IDs usually have the same identifier.

Node IDs are the names by which systems in a network identify one another. Therefore, each node within a given network should have a unique node ID. The only exception to this is for nodes that are not known throughout the network. For example, there can be two printer links with the same name in your network if they are not shared by the same group of nodes.

These node IDs usually are not chosen randomly. Instead, installations often define conventions for assigning names, based on geographic location or type of operating system. Examples of this are node IDs like NEWYORK and RESVM1.

If you use RSCS to communicate with other systems, the node ID you choose to define your own (local) system must be acceptable to the other systems. Some communications programs impose restrictions on the characters that make up the node ID. Check this with personnel at other sites. Beyond this, you are not required to follow any other naming conventions within your network or at a given site. However, it is recommended that you set up conventions to help maintain order and to avoid confusion.

When creating node IDs, try to use simple and identifiable names. Use common names when referring to the same things across functions or components, whenever possible. For example, if you are using RSCS in a VM/DDP (Distributed Data Processing) Network with the VM/Pass-Through Facility product, it will be helpful to coordinate use of names between the two products.

For 3270 printer and remote workstation nodes at your site, consider node IDs that describe the type of printer at that node or the building where the device is located.

SNA connections defined to RSCS will include LU names that have also been defined to VTAM. VTAM has its own conventions for LU names, which may be different from the RSCS conventions. Because VTAM networks are often larger and consist of multiple domains, you might have LU names that conform to the recommended conventions and have meaning to network administrators, but mean nothing to operators and users.

The LINKDEFINE statement or DEFINE command used to define SNA links to RSCS correlates the VTAM LUNAME with a RSCS link ID. RSCS, in turn, uses that link ID to refer to any of its links, SNA and non-SNA. An LU name that has already been specified by some other link (whether from the configuration file or on DEFINE or START) cannot be used again by any other link. If the LU name of a given link is necessarily cryptic, its link ID need not be. You can devise a link ID that is more significant and easier for operators and users to remember. The link ID is what they will specify to operate and use SNA systems and printers through RSCS.

An auto-answer link is automatically started when it receives a call over switched telecommunication facilities from a remote node. For links that will be auto-answered, there are certain naming conventions. For NJE-type links, RSCS checks the node ID that the remote system transmits in its sign-on record against the link ID (or node ID, if you specified the NODE operand when defining the link) of the desired link. For RJE-type links, the workstation name that the remote workstation transmits in its sign-on record serves the same purpose. In both cases, you should follow the conventions required by the current version of RSCS.

For MRJE-type links other than auto-answer, RSCS requires a different convention. The sign-on record transmitted by the remote workstation must have a workstation name of the form:

```
REMOTEnn
```

where *nn* is a number in the range 1 - 99. This number is specified on the PARM RMT operand and is not related to the link ID used to start the link.

Because the auto-answer task does not have access to the PARM operands for a link, the requirement that a workstation name be of the form REMOTEnn is relaxed. In this situation, you can specify the actual link ID in the sign-on record. If the operating system in the remote workstation is the type that only names of the form REMOTEnn can be transmitted, then you must specify a LINKDEFINE statement that has the same name in the RSCS configuration file. Otherwise, calls from that workstation will not be accepted.

Note

RSCS also permits the specification of a link ID in the sign-on record for MRJE-type links that are not auto-answer. In such a case, remember to omit the RMT operand, because no remote number can be specified by a PARM option. For an auto-answer MRJE-type link, the RMT operand is always ignored.

For more information about auto-answer links, see the discussions of [“Automatic Link Management”](#) on page 78, and the [“PORT Statement”](#) on page 67.

Other Conventions

TCP/IP connections (LPR-type, TCPASCII-type, TCPNJE-type, and UFT-type links) use IP (dotted decimal) addresses, in which the address consists of four numbers (0 - 255) separated by periods:

```
nnn.nnn.nnn.nnn
```

If the RSCS domain name server (RSCSDNS) is running, you can use fully qualified host name (domain name) addresses instead, in which the address consists of multiple names separated by periods:

```
name.name.name.name
```

By default, RSCS uses the following TCP/IP well-known ports:

- LPR-type links use local ports 721 - 731 (unless SECURE=NO is specified). These links connect to remote port 515.
- LPD-type links use local port 515 to listen for connect requests.
- TCPNJE-type links connect to remote port 175 and use local well known port 175 to listen for connect requests.
- UFT-type links connect to remote port 608.
- UFTD-type links listen for connect requests on local port 608.

The port setting can be changed on the PARM statement for the link or on the DEFINE or START command.

Using TCP/IP Well-Known Ports

By default, TCP/IP restricts the use of well-known ports (1 - 1023). To use a well-known port when usage is restricted, the RSCS server must be authorized on a PORT statement in the TCP/IP configuration file (PROFILE TCPIP). For more information, see *z/VM: TCP/IP Planning and Customization*.

Comments

RSCS does not process comments in the configuration file. They let you document the contents of your configuration file so that others can understand it more easily. There are two ways to place comments in your configuration file:

1. Use the REXX delimiters (/ * and */). You can use the REXX delimiters anywhere in the configuration file, but they must begin and end on the same line.
2. Place an asterisk in column one. When RSCS finds an asterisk in the first column, it ignores the rest of the line.

For example, if you use the IMBED statement to divide your configuration file into manageable pieces, you might want to insert comments describing the contents of the imbedded file:

```
/*-----*/
/* Imbed the file containing the printer links */
/*-----*/
  IMBED PRINTRS

/*-----*/
/* Imbed the file containing the system links */
/*-----*/
  IMBED SYSLINK

/*-----*/
/* Imbed the file containing the workstation links */
/*-----*/
  IMBED WSLINK
```

For more information, see [“Comment” on page 131](#).

AUTH Statement

AUTH statements identify user IDs that are authorized to enter privileged RSCS commands. User IDs that are authorized to enter these commands can affect RSCS operations. User IDs can be authorized as:

- Alternate link operators, who can enter commands that affect a specified link
- Alternate RSCS operators, who can enter commands affecting RSCS itself

For more information on the types of operator roles, see [“Operating the RSCS Virtual Machine” on page 75](#).

The number of AUTH statements in your configuration file depends on how many other operators you want in your network.

For more information, see [“AUTH Statement” on page 132](#).

Note: The AUTH statement is not the only way you can authorize users for privileged RSCS commands. The sample RSCS configuration file contains an AUTH statement for the RSCS dynamic authorization virtual machine (RSCSAUTH). If RSCSAUTH is enabled, users can be authorized for RSCS commands through RSCSAUTH without defining additional AUTH statements. For more information about RSCSAUTH, see [z/VM: RSCS Networking Operation and Use](#).

CHANNELS Statement

The CHANNELS statement reserves one or more specified virtual channels for exclusive use by RSCS. RSCS needs the exclusive use of at least one channel to handle the spool files it is receiving or sending. This is because every time it receives or sends a spool file, RSCS has to allocate a virtual printer, reader, or punch to handle it. If there are many files to handle, RSCS needs a pool of devices to draw from, and one channel provides an equivalent of 256 virtual devices for RSCS to use.

Also, virtual addresses for RSCS links and other devices at your installation must be assigned on channels other than the RSCS device pool. If you do not include a CHANNELS statement in your configuration file, RSCS uses channel X'0F'. If you want RSCS to use a different channel for its device pool or more than one channel, you will need a CHANNELS statement in your configuration file.

For more information, see [“CHANNELS Statement” on page 134](#).

DEST Statement

DEST statements identify page-image printers under the control of Print Services Facility/VM (PSF) on the local system. These statements contain a list of destination addresses that, besides being second-level addresses (workstation or printer links) and user IDs, are valid for the local system.

For more information, see [“DEST Statement” on page 135](#).

DUMP Statement

The DUMP statement specifies the format for any dumps the system generates. If you use the Dump Viewing Facility to record RSCS problems, a VM-type dump is required; this is the default. The DUMP statement also has an optional parameter you can use to have dumps sent automatically to a specified user.

For more information, see [“DUMP Statement” on page 136](#).

EXIT Statement

The EXIT statements specify which RSCS exit points will have corresponding exit routine modules loaded during RSCS initialization. If you use the ON operand, RSCS loads and enables the exit routine during initialization. If you use the OFF operand, RSCS loads the exit routine but does not enable the exit point. This lets you enable the exit point at a later time using the EXIT command, which is described in [z/VM: RSCS Networking Operation and Use](#).

Before initializing RSCS, you must:

1. Correctly link-edit all necessary exit routine modules into a CMS LOADLIB. For example, you can use the RSCSEXIT LOADLIB provided with RSCS.
2. Issue the (GCS) GLOBAL command to allow GCS access to that LOADLIB.

Entry point names that do not correspond to module names must be identified to the linkage editor using the linkage editor ALIAS statement or using the GCS IDENTIFY macro.

For more information, see [“EXIT Statement” on page 137](#).

FCB Statement

The FCB statement lets you define the characteristics of an FCB image. RSCS simulates this FCB image on SNA3270P-type and 3270P-type links. RSCS supports all twelve channel skips.

For more information, see [“FCB Statement” on page 139](#).

FORM Statement

The FORM statement lets you dynamically define or modify a form that will be processed on a link. With this statement you can specify the type of separator page, the width, the length, and the lines per inch associated with the form.

RSCS creates a form table during initialization based on the FORM statements in the configuration file. When you send a file on a link, RSCS checks the form table to see if there is an entry associated with the form name specified on the SPOOL command. If there is an entry, RSCS temporarily resets the default parameter settings to those specified on the FORM statement for use in processing this file.

If there is no entry in the form table associated with that form name or if the entry specifies that you want default values, RSCS passes the values originally specified at link driver initialization time to that link driver for use in processing the file.

For more information, see [“FORM Statement” on page 140](#).

HIDECHARACTER Statement

The HIDECHARACTER statement lets you specify the hide character. When used within a link's parameter definition or within the RSCS events file, the hide character masks (or hides) information from users issuing RSCS QUERY commands. The hide character does not affect how RSCS processes the *hidden* information in the configuration and events files. RSCS uses the hide character only when issuing responses to QUERY commands.

By default, the hide character is a back slash (\). To mark information as hidden, place a hide character immediately before and after the sensitive information. (When marking a section of a link's parameter definition, be careful not to include PARM or the link ID between the two hide characters.) When generating QUERY command responses, RSCS replaces each character it finds between two hide characters with the letter X.

For example, suppose your configuration file had the following statements:

```

:
LINKDEFINE NYWORK01 TYPE MRJE
  PARM NYWORK01 BUFF=3200 PASS=NYW01AAA PHONE=5551111

LINKDEFINE NYWORK02 TYPE MRJE
  PARM NYWORK02 BUFF=3200 PASS=\NYW01AAA\ PHONE=\55 51111\

LINKDEFINE NYWORK03 TYPE MRJE
  PARM NYWORK03 BUFF=3200 \PASS=NYW01AAA\ PHONE=555\ .1111\
:

```

If a user issued a QUERY command, this would be the result:

```

msg rscs query system links type mrje show id + parm
Ready;
Link
Name      Parm Text

```

```
NYWORK01 BUFF=3200 PASS=NYW01AAA PHONE=5551111
NYWORK02 BUFF=3200 PASS=XXXXXXXX PHONE=XXXXXXX
NYWORK03 BUFF=3200 XXXXXXXXXXXXXXXX PHONE=555XXXX
3 links found
```

Note: You also may be able to use the hide character in exit routines.

For more information, see [“HIDECHARACTER Statement”](#) on page 142.

IMBED Statement

The IMBED statement lets you include another file into the RSCS configuration file. RSCS processes the records in the imbedded file at the point where you placed the IMBED statement as if they were a part of the main (or master) configuration file. The imbedded files must follow the same rules that apply to the master configuration file. (See [“Rules for Coding the RSCS Configuration File”](#) on page 129 for more information about the rules for coding your configuration file.)

RSCS lets you include (nest) IMBED statements within imbedded files, if the nesting does not go beyond 10 levels deep. This means you can have one master configuration file and nine levels of imbedded files.

For more information, see [“IMBED Statement”](#) on page 143.

ITRACE Statement

The ITRACE statement lets you:

- Define the initial state of the RSCS internal trace table
- Define the events RSCS will trace
- Invoke tracing when RSCS initializes

For more information, see [“ITRACE Statement”](#) on page 145.

LANGUAGE Statement

The LANGUAGE statement lets you specify the language you want to use for local and remote messages. With this statement, you specify the name of the load module that contains the RSCS message file.

The default RSCS message file is American English, which is supplied on the RSCS base tape. RSCS also supplies a version of the message file that has been converted into uppercase English. The uppercase English file is supplied on the RSCS base tape within the RSCS load library.

For more information, see [“LANGUAGE Statement”](#) on page 150.

LINKDEFINE Statement

The LINKDEFINE statement defines the default attributes of a single RSCS link in your network. The link may be to a workstation, printer, or other z/VM or non-z/VM system. There will be as many LINKDEFINE statements in your configuration file as necessary to define links to all the adjacent nodes in your network.

LINKDEFINE statements identify nodes that RSCS will communicate with directly, either through a physical connection or through VTAM. Some nodes that RSCS communicates with directly are physically connected to your system by BSC (Binary Synchronous Communication) lines or CTCAs (channel-to-channel adapters). Other nodes that RSCS communicates with directly through VTAM (regardless of their physical connections to the system) are connected with SNA links or through TCP/IP networks (for TCPNJE-type and other TCP/IP links). Also, RSCS views nodes defined with LINKDEFINE statements as adjacent nodes, whether they are physically adjacent or not. Figure 35 on page 64 shows adjacent nodes identified to RSCS by LINKDEFINE statements and connected to a z/VM system by various SNA and non-SNA links.

Note: Do not connect multiple CTCA links to the same real channel. If you do, unrecoverable I/O interlocks may occur on the CTCAs. Other CTCA compatible channel control units, such as the IBM 3088, may be used without this problem occurring.

LINKDEFINE statements not only define adjacent nodes, but also include information that RSCS needs to communicate with those nodes. Some of this information might include:

- Whether the link ID is a nickname
- What you are connecting to: printer, workstation, peer system, or service machine
- Whether it is an auto-start link
- Whether it is a retry link
- What class of files to send to the node
- What FCB processing mode the link will use
- What method of queuing the link will use
- What slowdown values are defined for the link
- Whether SNA or non-SNA link
- Where to find session parameters for SNA nodes

You could specify this information on the DEFINE or START command after initializing RSCS. However, doing that could make those commands cumbersome to enter. Instead, use the LINKDEFINE statement to define permanent or default information. Use the commands to make temporary, short-term changes in the defaults or to dynamically add or delete nodes, as needed.

For more information, see [“LINKDEFINE Statement” on page 151](#).

LINKTYPE Statement

The LINKTYPE statement lets you dynamically define additional link drivers to those provided with the base RSCS system. With the LINKTYPE statement, you can specify a new link driver type, the entry point to the link driver code, and some basic characteristics of the link so that RSCS knows how to interact with the other node.

Attention

Writing a link driver requires a thorough knowledge of system programming, RSCS programming conventions, and RSCS internals. Without having this knowledge, if you write a link driver and use the LINKTYPE command to define it to RSCS, you can seriously degrade system performance and cause a system problem.

For more information, see [“LINKTYPE Statement” on page 158](#).

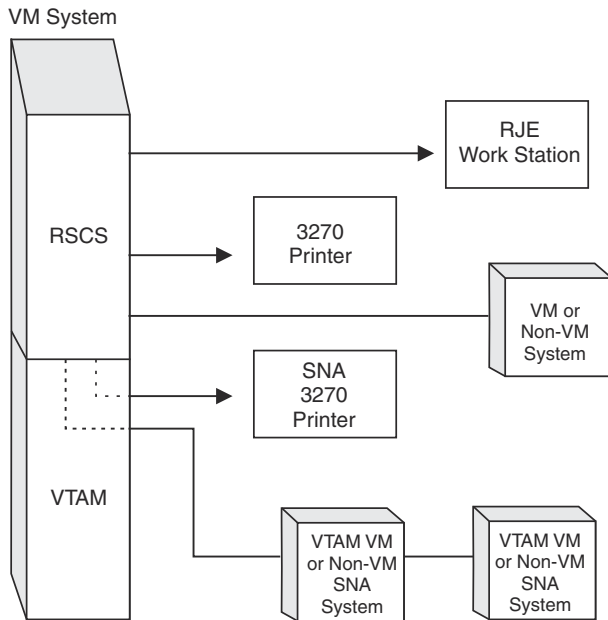


Figure 35. Links for Direct Communication

LOCAL Statement

The LOCAL statement defines the node ID of RSCS's local system (local node ID). If included, there must be only one LOCAL statement in your configuration file. If omitted, RSCS uses the system ID specified on the SYSTEM_IDENTIFIER or SYSTEM_IDENTIFIER_DEFAULTS statement in the z/VM system configuration file.

RSCS uses the local node ID to:

- Identify files and messages it sends out
- Recognize files and messages it receives

Networks with SNA connections use the LOCAL statement to define the RSCS application name to VTAM. You can do this in one of two ways:

Method 1:

Specify the same name for the local node ID and for the RSCS application name defined to VTAM. The following figures show what the statements would look like in the VTAM and RSCS files, respectively, if you were defining VMSYS as the RSCS application name.

RSCSNET VTAMLST			
RSCSNET	VBUILD	TYPE=APPL	
RSCS	APPL	ACBNAME=VMSYS, . . . ,	C
⋮			

RSCS CONFIG
LOCAL VMSYS
⋮

Method 2:

Specify different names for the local node ID and for the RSCS application name defined to VTAM by using the APPLID option of the LOCAL statement. The following figures show what the statements would look like in the VTAM and RSCS files, respectively, if you were defining VMSYS as the local node ID and VMRSCS as the RSCS application name.

RSCSNET VTAMLST			
RSCSNET	VBUILD	TYPE=APPL	
RSCS	APPL	ACBNAME=VMRSCS, . . . ,	C
:			

RSCS CONFIG
LOCAL VMSYS * VMRSCS
:

Although it is not required, matching these two names can cause less confusion and simplify the command used to start RSCS/VTAM communications. When using that command (RSCS NETWORK START), you are supposed to specify the VTAM application name (ACBNAME) on the APPLID parameter as follows:

```
network start applid acbname
```

If you do not specify APPLID on the NETWORK START command, RSCS checks to see if you specified one on the LOCAL statement in your configuration file. If you did not specify APPLID on the LOCAL statement, APPLID defaults to the local node ID on the LOCAL statement. Therefore, if you match the two names in either of the two methods, you can simplify the NETWORK START command and still be certain RSCS will use the default you have chosen. If you enter:

```
network start
```

the APPLID will default to:

Method 1:

VMSYS (the local node ID).

Method 2:

VMRSCS (the value of the APPLID option of the LOCAL statement).

In both cases, APPLID defaults to the same name as the desired RSCS application name (ACBNAME).

For more information, see [“LOCAL Statement”](#) on page 160.

MSGNOH Statement

The MSGNOH statement requests that RSCS use the CP command MSGNOH (rather than MSG) when sending messages to users on the local system. With this command, CP does not add its own header when handling the message. This reduces the size of the message when it is displayed, making it easier to read.

Special Note about Columnar Messages
Many of the responses to QUERY commands are issued as columnar messages. If the message is too long, the columns will wrap to the next line making the message difficult to read. The messages were designed with the idea that you would use MSGNOH to send messages to local users. If you do not use MSGNOH, be aware that QUERY command responses may be affected.

If you want RSCS to use MSGNOH instead of MSG to send messages, you will need a MSGNOH statement in your configuration file. RSCS must also be authorized to use the MSGNOH command. For more information, see [“RSCS Virtual Machine” on page 31](#).

For more information, see [“MSGNOH Statement” on page 161](#).

OPFORM Statement

The OPFORM statement informs RSCS of the default operator form name to be used for print files. If you do not specify a form name when the link is started, RSCS uses the default operator form name specified on the OPFORM statement to determine what type of file form to select for processing.

It is similar to the PRINTER operand of the FORM_DEFAULT system configuration file statement. However, OPFORM does not change or control the form of the file. For consistency between z/VM and RSCS operations, this name should match the operator form name corresponding to the user form name specified on the PRINTER operand.

If you do not include an OPFORM statement, RSCS will use STANDARD for the operator form name. This is the same name that CP uses if no operator form name was specified on the FORM_DEFAULT statement. If this is not what your installation requires or uses, you will need to include the OPFORM statement in your configuration file.

For more information, see [“OPFORM Statement” on page 162](#).

OPTION Statement

The OPTION statement allows your installation to define defaults for certain RSCS system and link operands. With these operands, your installation can:

- Suppress file transmission messages (accepted, queued, intermediate, and final)
- Dictate what type of job name to use for networking links
- Decide whether links will process limited or unlimited numbers of data set headers
- Decide whether links will check for routing loops
- Decide the degree to which message traffic preempts file traffic
- Decide whether origin user ID information should be read from and written to the CP spool.
- Decide whether to implement a limit on the number of response messages to QUERY commands.

You can override most of the operands specified on the OPTION statement.

For more information, see [“OPTION Statement” on page 163](#).

PARM Statement

The PARM statement corresponds to and must follow its respective LINKDEFINE statement in your configuration file. It defines specific details for communicating with the associated nodes. A PARM statement defines some of the following details:

- Communication buffer sizes
- Passwords for remote workstation and system nodes
- Formatting information for printer nodes
- Number of streams available for multistream transmission.

Because these details vary among the different types of links, the content and format of information on PARM statements also varies by link type.

Like LINKDEFINE statements, you can specify PARM information on the DEFINE or START command after RSCS initialization. PARM information can also make the DEFINE and START commands cumbersome to enter. Therefore, you should use the PARM statement to define permanent or default information. Use the commands to make temporary changes to the defaults or to dynamically add information on new nodes.

Depending on operations in your network, PARM statements may be *required* for SNA or TCP/IP links. When starting a session with an SNA or TCP/IP system or printer from RSCS, PARM options for the session can be specified on the RSCS DEFINE or START command. However, these sessions can also be started from VTAM or the other system, but the PARM options *cannot* be specified on the VTAM or other system's commands to start the session. Here, the only source for the needed information is the PARM statement in the configuration file.

If a link driver is to be started by an auto-answer task, certain key parameter options are required.

For more information, see [“PARM Statement” on page 168](#).

PORT Statement

The PORT statement specifies that a port is connected to the public switched telephone network and that the port may be used for an auto-dial or auto-answer link.

It also reserves a virtual address for a BSC telecommunication line to be dynamically allocated to a link that is started without explicit port specification. When RSCS processes the configuration file it saves these addresses in a port table. Usually you specify a port address for a link on its LINKDEFINE statement or on the DEFINE or START command, but when you start a link without specifying an address by the usual means (such as when a call is received over an auto-answer link), RSCS gets one from its reserve in the port table.

PORT statements are optional, depending on how you usually operate. As many PORT statements as necessary can be included in the RSCS configuration file. They can be positioned anywhere in the configuration file following the optional LOCAL and CHANNEL statements.

For more information, see [“PORT Statement” on page 251](#).

RECOVERY Statement

The RECOVERY statement lets you specify which CP command (or commands) should be issued when a RSCS supervisor task abends causing RSCS to become disabled. If you specify more than one CP command, you must separate them using the logical line end symbol (#).

For example, you can use the RECOVERY statement to tell RSCS to send a message to an operator or special service machine and then re-IPL GCS:

```
recovery msg operator rscs abend; re-ipling#ipl gcs
```

Note: If you specify a CP command that resets the RSCS virtual machine (such as IPL), be aware that any CP commands you specify after that point will be erased. In the example above, the operator would never have received the message if you had reversed the order of the commands. RSCS would pass the IPL command to CP, which would re-IPL GCS thereby wiping out the RSCS virtual machine and the MSG command. To avoid this problem, always place commands that reset RSCS at the end of the RECOVERY command.

For more information, see [“RECOVERY Statement” on page 252](#).

REROUTE Statement

The REROUTE statement lets you modify how commands, files, and messages are routed. Using this statement you can choose to reroute:

- Commands
- Files only
- Messages only
- Files and messages

For:

- A specific user ID

Configuration Planning

- A specific user ID when that user is not receiving (messages only)
- A generic group of user IDs
- All node IDs and user IDs when they are not receiving (messages only)
- The entire system

To:

- A specific user ID at a specific node ID
- The system unit record device

Also, you have the option of suppressing the reroute messages (DMT120I, DMT333I, and DMT336I) for specific reroutes. This means you can reroute commands, files, and messages without telling the users who sent them that they are being rerouted.

For more information, see [“REROUTE Statement”](#) on page 253.

RETRY Statement

If you specify the RETRY operand when defining an SNA or auto-dial link, RSCS will automatically try to restart the link at specific intervals. The RETRY statement lets you specify the intervals RSCS should use when trying to restart a deactivated auto-dial or SNA link.

If you specify the RETRY operand and you specify the following RETRY statement:

```
retry 1 9 10
```

RSCS tries to restart the deactivated link after 1 minute. If that fails, RSCS waits 9 minutes (a total of 10 minutes since the link deactivated) and tries to restart the link. If that fails, RSCS tries to restart the link after 10 minutes (a total of 20 minutes since the link deactivated). If that fails, RSCS continues to retry the link every 10 minutes, indefinitely.

If you specify the RETRY operand and omit the RETRY statement, RSCS tries to restart the link after 1 minute. If that fails, RSCS waits 10 minutes and tries to restart the link again. If that fails, RSCS tries after 19, 27, 34, 40, 45, 49, 52, 54, and 55 minute intervals. If the link still does not restart, RSCS continues to try to restart the link at 55 minute intervals, indefinitely.

For more information, see [“RETRY Statement”](#) on page 257.

ROUTE Statement

The ROUTE statement defines a path to a remote device or system with which RSCS will communicate indirectly through one or more links. Files destined for the remote node may, in fact, be stored and sent on one or more intermediate systems before reaching their destination. RSCS uses routing information to determine the next link (or links) in the route to the nonadjacent node. RSCS then sends files, commands, and messages through that link (or links). [Figure 36 on page 69](#) shows the physical connections between the system, the intermediate nodes, and the nonadjacent nodes.

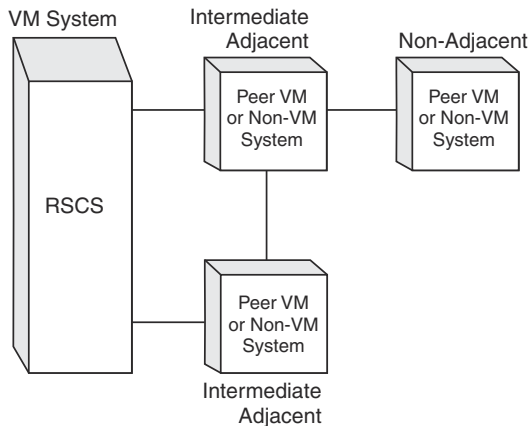


Figure 36. Routes for Indirect Communication

You can also use ROUTE statements to define multiple primary links, alternate links, and routing groups. Multiple primary links are sets (pools) of links over which RSCS can send traffic to reach a common destination. For example, suppose you have three 3812 printers located in one room and you want to create a printer pool for your users. You could specify the following statement in your configuration file:

```
route group prt3812 to link prt3812a prt3812b prt3812c
```

Instead of routing their files to a specific printer, users could route their files to the PRT3812 pool and RSCS would print the file on the first available printer.

An alternate link is a link RSCS can use to send data when none of the primary links are connected. For example, suppose you have a 3820 printer in the same room as the three 3812 printers in the previous example. If you specify the following statement in your configuration file:

```
route group prt3812 to link prt3812a prt3812b prt3812c alternate prt3820
```

RSCS would route the files to the 3820 printer when the links to all three 3812 printers were down.

A routing group is a collection of nodes that are grouped together for ease of routing. For more information about creating routing groups, see [Chapter 11, “Defining the RSCS Network Structure,” on page 105](#).

Once your RSCS system has initialized, you can use the ROUTE command to dynamically add, change, or delete routes. For more information about the ROUTE command, see [z/VM: RSCS Networking Operation and Use](#).

For more information, see [“REROUTE Statement” on page 253](#).

SAFCLASS Statement

z/VM can add a classification title to each page of output for a specified output class. This classification is specified on the PRINTER_TITLE system configuration file statement (see [z/VM: CP Planning and Administration](#) for more information). This can result in multiple classification titles being added to pages in an output file that is being transmitted across several intermediate nodes *in route* to its destination.

The SAFCLASS statement specifies the output class that is to be used by RSCS while creating store-and-forward spool files, to avoid spooling output to a class that has been defined to the local z/VM system as having classification titles. (When the file is transmitted, it is transmitted with the *original* class, not the class specified on SAFCLASS.)

If you do not include a SAFCLASS statement, or include one as:

```
SAFCLASS class *
```

RSCS will use the class specified when the spool file was created.

For more information, see [“SAFCLASS Statement” on page 261.](#)

SETMSG Statement

The SETMSG statement lets you route one or more RSCS messages to other virtual machines, based on message numbers. (Only messages marked as private cannot be shared; for more information, see [z/VM: RSCS Networking Messages and Codes.](#)) Using SETMSG statements, you can direct messages to a specific virtual machine and thereby use that virtual machine to monitor problems. For example, you could direct all error messages involving VTAM to the user responsible for fixing those problems.

For more information, see [“SETMSG Statement” on page 262.](#)

SHADOWS Statement

The SHADOWS statement lets you specify the number of tag shadow elements (TASHADOWs) you want RSCS to reserve for multiple primary links. When you use the ROUTE statement (or command) to define multiple primary links (pools of links), a file may be queued on several links. RSCS transmits the file over the first available link, but each link may already have files in the queue. This means RSCS must represent your file in several different file queues.

Rather than placing a copy of your file's tag slot in each queue, RSCS creates a tag shadow element for the file and places a copy of the tag shadow element in each queue. These tag shadow elements contain queuing-related information. If the file is queued on a networking link (GATEWAY-type, LISTPROC-type, NJE-type, SNANJE-type, or TCPNJE-type), the tag shadow elements also contain a stream mask. The queuing priority value and stream mask allow each link on which the file is eligible to be queued to have its own queuing method and transmission algorithm.

If you do not specify the SHADOWS statement in your configuration file, RSCS reserves 20000 tag shadow elements. If you plan to take advantage of the multiple primary link support, you should increase this number. We recommend you multiply the maximum number of links for any route by 10,000. For example, if you have a maximum of 4 links per route, you would reserve 40,000 tag shadow elements.

If you take advantage of the multiple primary link support and do not increase the number of tag shadow elements, RSCS may run out of tag shadow elements and be forced to enter a special *degraded* mode. When in degraded mode, RSCS tries to balance the use of tag shadow elements against the needs of each file. This means that RSCS cannot queue all files on all links for which they are eligible. Running in degraded mode will adversely affect system performance.

For more information, see [“SHADOWS Statement” on page 264.](#)

SLOWDOWN Statement

The SLOWDOWN statement lets you specify the absolute (base) system slowdown values. These values tell RSCS when to automatically start or stop slowdown mode for all active GATEWAY-type, NJE-type, SNANJE-type, and TCPNJE-type links.

z/VM imposes a limit (9,999) on the number of spool files that a single z/VM user ID can own. If your RSCS virtual machine is part of a busy network, RSCS may already have met this limit because of link or node outages.

Using the SLOWDOWN statement (or command), you can specify your own limits to tell RSCS when to prevent and resume inbound traffic. For example, suppose you specified the following SLOWDOWN statement in your configuration file:

```
slowdown 8000 6500
```

This statement tells RSCS to issue HOLD INPUT commands for every networking link when the RSCS file inventory reaches 8,000 or more files and to issue FREE commands for every networking link when the file inventory drops below 6,500 files.

You can also use the SLOWDOWN operand on the LINKDEFINE statement or the DEFINE command to specify slowdown values for a specific link.

For more information, see [“SLOWDOWN Statement” on page 265](#).

TOLERANCE Statement

The TOLERANCE statement lets you mark sections of the configuration file as tolerant or intolerant of errors. If omitted, the default is TOLERANCE ON, which means RSCS tolerates errors. If RSCS finds an error, it flags the statement and continues initializing.

But, suppose you have a section of your configuration file that is critical to the definition of your network. For example:

- Heavily used links
- Operator authorizations
- An exit package that defines security exit routines for your local node

If you do not want RSCS to start when there is an error in one of these critical sections, surround the section with TOLERANCE statements:

```

:
tolerance off
linkdefine ...
tolerance on
:

```

For more information, see [“TOLERANCE Statement” on page 266](#).

TRACEDEST Statement

The TRACEDEST statement lets you specify where you want RSCS to send trace files for all links. If you specify a local user, you can place the TRACEDEST statement anywhere in the configuration file after the LOCAL statement (if one exists). If you specify a remote user, you must place the TRACEDEST statement after the LINKDEFINE or ROUTE statement that identifies the remote node.

To temporarily override the destination for one trace file, use the TO operand of the RSCS TRACE command.

If you omit the TRACEDEST statement and you do not override the destination using the TRACE command, RSCS sends the trace file to the virtual printer associated with the local RSCS virtual machine.

For more information, see [“TRACEDEST Statement” on page 267](#).

UPARM Statement

The UPARM statement lets you specify additional, installation-defined parameters for your links. These UPARM parameters have no effect on PARM (or OPARM) parameters.

For more information, see [“UPARM Statement” on page 268](#).

RSCS Message File

Most RSCS messages reside in the RSCS message file. This file allows most messages to be easily translated (the exceptions are the message compiler messages). All messages are described in [z/VM: RSCS Networking Messages and Codes](#).

When you install RSCS, the default language for the message file is American English. This message file is supplied on the base tape and can be changed to uppercase English during installation.

If you want your RSCS messages to appear in a language other than American English or uppercase English, RSCS makes it possible for you to translate the message file into a language of your choosing.

For more information about the RSCS message file, see the [“RSCS Message Repositories” on page 101](#) or [z/VM: RSCS Networking Exit Customization](#).

Customizing RSCS

RSCS provides various types of exits and other facilities that you can use to customize RSCS operations.

RSCS Exit Facility

The RSCS exit facility provides interfaces, called *exit points*, between RSCS and user-written code. Because RSCS is a general-purpose networking product, the exit points let you perform processing that is required by local circumstances.

For more information on using the RSCS exit facility, see [“Using Exit Routines to Customize RSCS” on page 101](#) and [z/VM: RSCS Networking Exit Customization](#).

Transmission Algorithms

The networking links (GATEWAY-type, LISTPROC-type, NJE-type, SNANJE-type, and TCPNJE-type) use *multistreaming*, which allows more than one file to be transmitted and received concurrently over the same link. On LISTPROC-type, NJE-type, SNANJE-type, and TCPNJE-type links, multistreaming uses up to seven streams over which files can be transferred. On GATEWAY-type links, multistreaming uses up to 32 streams over which files can be transferred. RSCS, in turn, uses exit routines called *transmission algorithms* to assign files to the various streams.

For more information on using transmission algorithms, see [“Using Exit Routines to Customize RSCS” on page 101](#) and [z/VM: RSCS Networking Exit Customization](#).

ASCII Exits

The ASCII-type link uses an ASCII printer and plotter exit routine to build a device-specific data stream for transmission to printers and plotters connected to RSCS through a controller such as the IBM 7171 ASCII Device Attachment Control Unit or the 9370 ASCII Subsystem Controller.

TCPASCII-type links also use ASCII printer and plotter exit routines to build device-specific data streams. However, in this case, the ASCII printer or plotter is attached to a terminal server in a TCP/IP network.

Like the RSCS exit facility, ASCII printer and plotter exit routines remain independent of RSCS code. Using printer and plotter exit routines, you can modify initialization, process spool files, translate output records, and perform end-of-file processing.

For more information on using ASCII exits, see [“Using Exit Routines to Customize RSCS” on page 101](#) and [z/VM: RSCS Networking Exit Customization](#).

Gateway Programming Interface

The gateway programming interface (GPI) lets you create routines that let RSCS exchange data between systems using NJE or other networking protocols. In RSCS, GPI support consists of the GATEWAY link driver and *gateway service macros* that provide networking functions for the *gateway program*, which you supply. The system to which the gateway program is communicating must also provide a corresponding routine to communicate with RSCS.

For more information on using the gateway programming interface, see [“Using Exit Routines to Customize RSCS” on page 101](#) and [z/VM: RSCS Networking Exit Customization](#).

LPR Exits

The TCP/IP LPR-type links use exit routines to build device-specific data streams for transmission to a remote printer in a TCP/IP network. These exit routines can also be used to control where the data is sent (that is, the remote host and port that will receive the data).

LPR exit routines are similar to ASCII exit routines. You can use these exit routines to modify initialization and various points at file processing.

For more information on using LPR exits, see [“Using Exit Routines to Customize RSCS” on page 101](#) and [z/VM: RSCS Networking Exit Customization](#).

LPD Exits

The TCP/IP LPD-type links use exit routines when receiving print streams from LPR clients in a TCP/IP network. These exit routines can also be used to control where the data is sent (that is, the remote node and user ID that will receive the data).

LPD exit routines are similar to LPR exit routines. You can use these exit routines to modify initialization and various points at file processing.

For more information on using LPD exits, see [“Using Exit Routines to Customize RSCS” on page 101](#) and [z/VM: RSCS Networking Exit Customization](#).

UFT Exits

The TCP/IP UFT-type links use exit routines to build data streams for transmission to remote users in a TCP/IP network. These exit routines can also be used to control where the data is sent (that is, the remote host and port that will receive the data).

UFT exit routines are similar to LPR exit routines. You can use these exit routines to modify initialization and various points at file processing.

For more information on using UFT exits, see [“Using Exit Routines to Customize RSCS” on page 101](#) and [z/VM: RSCS Networking Exit Customization](#).

UFTD Exits

The TCP/IP UFTD-type links use exit routines when receiving data streams from remote UFT clients in a TCP/IP network. These exit routines can also be used to control where the data is sent (that is, the node and user ID that will receive the data).

UFTD exit routines are similar to LPD exit routines. You can use these exit routines to modify initialization and various points at file processing.

For more information on using UFTD exits, see [“Using Exit Routines to Customize RSCS” on page 101](#) and [z/VM: RSCS Networking Exit Customization](#).

Chapter 7. Planning for RSCS Operation

z/VM: RSCS Networking Operation and Use describes the operation tasks associated with RSCS. Some of these tasks deal with the RSCS virtual machine itself, while others deal with its links and supported devices (like printers and workstations). Some tasks are performed locally and some are done at remote locations. All tasks can be done manually but many daily or often repeated tasks can be automated using execs or the event scheduler. The following checklist contains items to consider when planning for operation.

Planning Checklist for RSCS Operation

- **Identify** which tasks need to be performed and when they need to be performed.
- **Decide** whether each task needs to be performed manually, can be performed by an exec, or can be scheduled using the event scheduler.

For **manual** tasks:

- **Determine** who will perform the task.
- **Ensure** that operators are authorized to perform tasks.
- **Arrange** for training operations people to do manual tasks.

For **automated** tasks:

- **Identify** someone to write the automated procedures.
- **Establish** a schedule for completion so they can be added to the system.
- **Arrange** for training operations people to use the automated procedures.
- **Ensure** that operators are authorized to use the automated procedures.

For **scheduled** tasks:

- **Identify** someone to create and customize the RSCS events file.
- **Establish** a schedule for completion so the events file can be added to the system.
- **Ensure** that operators are authorized to dynamically add, delete, resume, and suspend events using the RSCS SCHEDULE command.

Operating the RSCS Virtual Machine

RSCS runs without regular operator intervention. Once initialized, the RSCS virtual machine can be disconnected to continue running on its own. (Service machines, like RSCS, are often designed this way so that one operator can oversee several different machines.)

When RSCS is running disconnected, messages to its virtual machine console can also be directed to another user or operator. This can be done in one of the following ways:

- Through the z/VM Single Console Image Facility (SCIF), by specifying a secondary user in the RSCS virtual machine definition. The secondary user acts in place of the RSCS operator with full operator privileges.
- Through a parameter on the RSCS DISCONNECT command. Aside from receiving messages, other operator privileges are not automatically extended to this user unless otherwise authorized.
- Through the RSCS SET command, authorized alternate operators can receive messages about the link (or system) for which they are authorized.
- Through the RSCS SETMSG statement or command, authorized alternate operators can subscribe users to receive any RSCS message except a private message. (See *z/VM: RSCS Networking Messages and Codes* for more information about private messages.)

Commands affecting operation of the RSCS virtual machine are restricted to the RSCS node operator (or secondary user). However, it is possible to authorize other users for these privileged commands. This can be done in two ways:

- By defining AUTH statements in the configuration file to designate one or more authorized alternate operators for RSCS. A system-authorized alternate RSCS operator:
 - Has the same command privileges as the RSCS node operator.
 - Can be logged on at the same time as the RSCS node operator (that is, the RSCS operator need not be disconnected).
 - Can be on a different system (node ID), unlike a secondary user, who must be on the same system.
- By using RSCS dynamic authorization (RSCSAUTH) to give users authority for sets of commands. A user authorized for commands through RSCSAUTH:
 - Does not need to be authorized in the RSCS configuration file. (Only RSCSAUTH needs to be authorized on an AUTH statement in the RSCS configuration file.)
 - Has authority equivalent to an alternate operator for the specified commands.
 - Can be given greater or lesser authority (to more, fewer, or different commands), without updating the RSCS configuration file and restarting RSCS.
 - Can be on a different system (node ID). The user does not need to be authorized through RSCSAUTH on the user's system, only on the system which is the target of the command.

Operating Links

Commands affecting operations on a particular RSCS link are available not only to the RSCS node operator but also to link-authorized alternate operators. (*z/VM: RSCS Networking Operation and Use* identifies this subset of RSCS commands.) Link operators are users or operators who are usually at the same remote location as the printer or workstation that they are operating. However, it is possible to authorize a link operator who is at some other location, for a particular link. A link-authorized alternate operator may be designated to:

- Exercise more control over the link, and the information that passes over it, than a workstation or printer operator can.
- Provide backup coverage for the on-site workstation or printer operator.

Like system-authorized alternate operators, link operators are authorized through the AUTH statement in the configuration file. The AUTH statement also identifies the link that the specified user is authorized to operate. Link-authorized alternate operators can enter any RSCS command in the subset available for link operations, but only for their authorized links.

Users can also be authorized for links through RSCSAUTH. A user authorized for links through RSCSAUTH:

- Does not need to be authorized in the RSCS configuration file. (Only RSCSAUTH needs to be authorized on an AUTH statement in the RSCS configuration file.)
- Has authority equivalent to an alternate operator for the specified links and can enter any RSCS link control commands for which they have been authorized through RSCSAUTH.
- Can be given greater or lesser authority (to more, fewer, or different links), without updating the RSCS configuration file and restarting RSCS.
- Can be on a different system (node ID). The user does not need to be authorized through RSCSAUTH on the user's system, only on the system which is the target of the command.

Initiating SNA Sessions

SNA sessions do not always have to be initiated from the local RSCS side. You can begin a session with a SNA system:

- Explicitly, from RSCS (operator enters a START command)

- From the remote system by an equivalent of the START command (for example, \$S N, LNE n A=applid from JES2).

You can begin a session with a SNA printer:

- Explicitly, from RSCS (operator issues a START command)
- From VTAM, (operator issues a VARY NET, LOGON... command or through automatic logon by VTAM).

One way of initiating may be more appropriate for operations within your network environment than another. This will affect what you put in your configuration file.

Preparing to Automate Operations with Execs

Automated procedures (execs) make an operator's job simpler. What you choose to automate through execs will depend on your installation's particular requirements. Remember, however, that whatever procedures you do not automate, your operators must be trained to do manually.

Operators perform three major tasks: (1) initialization, (2) ongoing operation, and (3) shutdown. See *z/VM: RSCS Networking Operation and Use* for examples of execs that will automate each major task. Manual procedures are not discussed in this document. See *z/VM: RSCS Networking Operation and Use* for a complete list of operational tasks and procedures. From that list, you can choose what step-by-step procedures your operators will follow.

Automating Initialization

RSCS must be initialized from within the GCS environment. RSCS can be logged on automatically by the GCS recovery machine when the GCS saved system is itself initialized. Initialization of the RSCS virtual machine can be automated by including an XAUTOLOG command in the PROFILE GCS exec for the recovery machine. In fact, XAUTOLOG commands have been predefined in the PROFILE GCS exec to autolog the RSCS, RSCSDNS, and RSCSAUTH virtual machines.

RSCS links can be started automatically through execs. These execs can be designed as separate procedures that can restart links as necessary (see *z/VM: RSCS Networking Operation and Use* for examples), or grouped together in a single startup procedure that is called by the RSCS virtual machine's PROFILE GCS exec.

Automating Ongoing Operation

Once RSCS and its links have been initialized, little is required for ongoing operation. An operator's primary activities are to:

- React to special conditions that arise
- Respond to the needs of users
- Make dynamic adjustments

Most of these are not routine activities and predicting when they will occur would be difficult. Operations personnel should be prepared for this role.

Using the SET command or the SETMSG command or statement, you can help operators react quickly to special conditions. RSCS sends all sorts of messages to the RSCS console and to other virtual machines. The operator monitoring the RSCS console has to sort out the informational messages from the error messages. Users monitoring other virtual machines also have to sort through messages. Using the SET command, you can select another operator to receive specific RSCS console messages at the same time as the RSCS console receives them. Using the SETMSG command or statement, you can select another virtual machine to receive any nonprivate RSCS message. Because the specified user will not have to sort through all the messages, that user can react more quickly to problem situations.

Your installation can use exec procedures to simplify command entry. An exec that would drop the need for prefixing commands with the prefix SMSG *rscsid* would be an example. Similarly, execs can help operators respond to users' needs more quickly and to speed problem recovery. The extent and nature of ongoing operations at your installation will best determine the need for execs.

Your installation can use exec procedures to monitor the network. For example, if you have operators authorized to monitor the file queues of several links, your installation may choose to use an exec to monitor those links for the operator. (*z/VM: RSCS Networking Operation and Use* shows an example of such an exec.) Also, if your operators always perform certain actions in response to certain situations, your installation may choose to use exec procedures to monitor the network and process the appropriate actions.

Automatic Link Management

Automatic link management provides the capability to reduce or eliminate the need for an operator to manually start and stop links, or to dial (or answer) telephone connections for links that use them.

For example, you can define a link as an **auto-start** link by including the ASTART operand on the LINKDEFINE statement. This causes it to be automatically started when a file is queued for transmission over it.

You can define an NJE-type, RJE-type, or MRJE-type link (communicating over switched telecommunication facilities) as either an **auto-dial** link or an **auto-answer** link.

An auto-dial link automatically dials the remote node when the link is started, if the telecommunication facilities include an Automatic Calling Unit (ACU). Do this by specifying the PHONE parameter either on its PARM configuration file statement or in a DEFINE command.

An auto-answer link is automatically started when it receives a call from a remote node. Do this by enabling a port.

You can define an RJE-type link as an **auto-sign-on** link by including one or more of the auto-sign-on operands on the PARM statement for the link. The auto-sign-on operands are BUFF, CMPR, HDR, LPRT, PCH, PRINT, SYSTEM, TEXT, TRS, or TYPE. Specifying these operands before you start the link relieves operators of having to manually submit sign-on cards from remote work stations.

Note: An auto-sign-on link can also be an auto-dial or auto-start link, but not an auto-answer link.

Shutting Down RSCS

There is no need to automate this phase of RSCS operations. An orderly shutdown of RSCS virtual machine activity can be requested through the RSCS SHUTDOWN command. Issuing this command is as simple as calling an exec. (See *z/VM: RSCS Networking Operation and Use* for details.)

Preparing to Automate Operations Using the RSCS Event Scheduler

The RSCS event scheduler lets you automate events that must take place on particular days of the week or at particular times of the day. For example, you can automate these types of tasks:

- Closing the RSCS console
- Changing operational shifts
- Starting and stopping diagnostic procedures (such as traces)

You can use the event scheduler to ask RSCS to:

- Process commands at a particular time of day
- Repetitively run commands:
 - Every *m* minutes
 - Every *n* minutes past the hour
- Process commands based on:
 - A particular date
 - A certain day of the week
 - A certain day of the month

- Certain specially defined days
- Periodically change held files queued on LPR-type or UFT-type links to nohold for retrying the print or data file delivery

By using the RSCS event scheduler to take care of issuing commands at specified times, you can lessen the work load of operators and reduce the chance of the task being overlooked.

Chapter 8. Planning for RSCS Administration

RSCS administration includes managing the use of RSCS and the facilities made available to its users. Who will use RSCS resources and how often? What will the facilities require from users? What will users require? The following checklist contains items to consider when planning for administration.

Planning Checklist for RSCS Administration

- **Determine** if users will be operating remote devices.
- **Determine** if users will be establishing communications with RSCS.
- **Identify** users responsible for operational tasks and arrange for required training.
- **Ensure** that users have access to information on the following:
 - Other user IDs and system node IDs in your network
 - How to:
 - Submit jobs from remote workstations
 - Submit jobs to remote systems
 - Direct output where desired
 - RSCS commands for manipulating their files.

(*z/VM: RSCS Networking Operation and Use* contains information that may be helpful to your users.)
- **Check** with administrators at other sites to determine the following:
 - How and when to define your users on their systems
 - If you have to schedule or account for the time you use on their systems
- **Determine** whether any automated procedures are required to assist users.
- **Assign** someone to write the automated procedures.
- **Establish** a schedule for completion of the automated procedures so they can be added and verified during the RSCS configuration process.

Tending to Remote Workstations and Printers

Remote workstations and printers will require some attention. You may choose to make individual device users responsible for basic operations like powering a workstation on and off or refilling a printer's paper supply. Some minimal training may be required.

Note

z/VM: RSCS Networking Operation and Use describes operational tasks associated with establishing communications between such devices and RSCS. You may decide to let individual device users perform some or all of these tasks. Some operations, such as starting and stopping links, require special authorization for the users.

Besides manually performed operations, remote workstations require programming-type preparations. Input sent from remote workstations must start with an ID statement. The ID statement tells RSCS where to send the input. Also, the input must contain statements, commands, or whatever is necessary to tell the destination system where to direct output after processing.

Output files for printers are first directed to RSCS by the CP SPOOL command. They also must be *tagged* with a destination address (using the CP TAG command), so that RSCS knows which printer to send them to.

Preparing Network Information

Files to be transmitted to another system are first directed (by the CP SPOOL command) to RSCS for handling. The file tag contains the destination address to which RSCS will send the file. This is supplied on the CP TAG command.

Certain facilities of z/VM, such as the CMS TELL, NOTE, and SENDFILE commands, and licensed programs such as PROFS or OfficeVision, simplify this requirement to use RSCS. Their users need not be aware that whenever they send messages or files to users on other systems, they are, in fact, using RSCS. These CMS commands, the PROFS program, and the OfficeVision program issue RSCS SMSG commands to tell RSCS the required SPOOL and TAG commands for the users. Users only need to identify the system (node ID) and user (user ID) to which they want the data sent. They can even specify the user ID and node ID as a familiar name or nickname.

When sending data to other systems, users need to know the node IDs of the systems, user IDs of other users, names of application programs at the destination, and other similar information. Their user IDs may have to be defined to other systems before they are able to log on those systems. You can check with administrators at other locations to find out what they require to accommodate your users.

Manipulating Files in Transit

Even after RSCS has control of files, users can manipulate any files they have sent to RSCS for handling and any files that are being sent to them through RSCS. RSCS commands let users CHANGE, FLUSH, PURGE, QUERY, or TRANSFER their files. *z/VM: RSCS Networking Operation and Use* describes how to use these commands to manipulate files.

Simplifying Jobs

Users determine the contents of their files, and RSCS does not change files when it receives them for handling. Jobs in different formats are all treated the same. However, jobs must be submitted in formats acceptable to the destination application or system. It is the responsibility of the user to provide the correct format. Depending on the needs of your users, these formats can be simplified for them through site-developed aids and exec procedures.

Candidates for simplification, include:

- Complex JCL (Job Control Language) can be built for the user based on a particular destination address or through user prompting before the file is sent to RSCS.
- To automate LPR-type links or make them easier to use, use the LPRXFORMS exit. For more information, see *z/VM: RSCS Networking Operation and Use*.
- Required printer or print location can be determined through user prompting, then appropriate commands issued for the user.
- An address that has no identifiable meaning to a user can be defined in more familiar terms (giving it a nickname) that the user can specify.

Simplifying and automating complex or often-repeated procedures can be helpful to all users. For those who use an application infrequently, automated procedures (site-developed execs or menus) ease the burden of recalling special parameters and options. Frequent users can eliminate repeating the information. Users can devise their own method to circumvent this, but all would benefit from a common procedure supplied by your administration. Then, if a change occurs, only the common procedure would have to change, not each user's own procedure.

z/VM: RSCS Networking Operation and Use discusses the various tasks that operators and users can perform.

Chapter 9. Planning for RSCS Diagnosis

As a networking product, RSCS relies on communication lines to complete its work. Problems on communication lines can affect the ability of RSCS to transfer data. Although problems may occur within RSCS itself, often the problem is with the facilities (hardware, transmission lines, or connections) that RSCS uses.

To isolate and solve a problem, different people and different courses of action may be needed. RSCS may be able to recover from, or circumvent, a problem automatically or with the help of an operator. However, if a problem recurs frequently, it should not be left unchecked.

The topics in this section describe some considerations for diagnosing RSCS problems. The following checklist describes some things to consider when planning for diagnosis.

Planning Checklist for RSCS Diagnosis

- **Ensure** that operators are trained to respond to problem situations — either to take action themselves or to call for help.
- **Identify** support personnel who will be on call when operators need help.
- **Determine** procedures that users should follow if they run into problems.
- **Identify** off-site contacts for problems that involve communication lines or other systems.

Diagnosing Problems Using RSCS Facilities

RSCS provides facilities to assist in diagnosing problems. These facilities include messages, commands, tracing facilities, and dump formatting routines. RSCS also provides facilities to automatically attempt to recover from some error situations.

RSCS produces messages to document its actions. It also produces diagnostic messages if errors occur. These messages and their explanations often suggest follow-up actions to resolve or diagnose the problem. For more information about diagnostic messages, see [z/VM: RSCS Networking Messages and Codes](#).

For example, when RSCS initializes, it issues message DMTMAN010I. This message identifies the RSCS load address and its address range. The information is also displayed at the RSCS console if an abend occurs. You can also display this information at any time by issuing a QUERY SYSTEM LOADADDRESS command, described in [z/VM: RSCS Networking Operation and Use](#).

To record diagnostic and error messages for later use, you should spool the RSCS console. You should then ensure that this spool file is closed at regular intervals to prevent the console file from becoming too large. To do so, you can use the RSCS event scheduler to automatically spool the console and close the file at predetermined times. See [“Automating Simple Events” on page 95](#) for more information about using the event scheduler to close the RSCS console.

RSCS link drivers, which handle I/O activity on their associated links, track their own activities. Each link driver records the number of successful I/O transactions and the number of errors that occur. You can display these statistics for a link by issuing a QUERY SYSTEM COUNTS command, described in [z/VM: RSCS Networking Operation and Use](#). You can then determine the amount of data being transferred on a link and the link's performance while processing this data.

Using Tracing Facilities

If you need more information about link-related problems, you can start a trace with the RSCS TRACE command. The trace records additional details about the data and its flow across a link.

While a trace is active, data is logged in a spool file. The trace content differs for SNA, non-SNA, and TCP/IP links. For SNA-type links, the trace record can identify specific SNA requests that occur on the

Diagnosis Planning

link. For networking links, the RECORDS option identifies the NJE record segments transferred on the link. For TCP/IP links, socket information and IUCV protocol connection information is also traced. For more information, see the line trace examples in [z/VM: RSCS Networking Diagnosis](#).

When you deactivate the trace, RSCS can send this file to the user ID specified on the TRACEDEST statement or the TRACE command. See [z/VM: RSCS Networking Operation and Use](#) for more information about the TRACE command format.

You can also use the RSCS ITRACE facilities to record specific RSCS events. Using the ITRACE command and statement, you can select the events you want to record and determine how the trace table will be processed. You can also use the ITRACE macro to record events in exit routines. For more information, see [z/VM: RSCS Networking Operation and Use](#) and [z/VM: RSCS Networking Exit Customization](#).

For some error conditions, however, you may need to use instruction traces. For example, if you suspect a looping condition is causing an error, the TRACE command can help you identify the problem. See [z/VM: CP Commands and Utilities Reference](#) for more information about this command.

Producing Dumps

RSCS can automatically produce dumps when it abends. You can also request a dump by issuing a VMDUMP command. If you specify a user ID on the DUMP statement or the VMDUMP command, that user will receive the dump file automatically. For more information, see the description of the VMDUMP format for RSCS in [z/VM: RSCS Networking Diagnosis](#).

Displaying Dump Information

If RSCS creates a dump or you enter the VMDUMP command, you can use z/VM facilities to process and display the dump's contents. However, to use these facilities, RSCS must produce VM-type dumps. To do so, you should specify the VM option on the DUMP statement. For the VMDUMP command, specify the RSCSV3 dump format.

RSCS provides support to let you use the Dump Viewing Facility. See [z/VM: Dump Viewing Facility](#) for more information about creating and analyzing RSCS dumps.

Establishing Information-Collecting Procedures

Because operators may receive requests to collect diagnostic information, they should be instructed on what to do in various situations and when to get outside help. Some of this work can be simplified using execs, RSCS commands or statements.

For example, you can create execs to issue TRACE commands for a link and send the trace log to a specific user. You can also use the TRACEDEST statement to identify the person to receive all trace files. You can use the RETRY operand and statement to tell RSCS to restart an auto-dial or SNA-type link. To subscribe a user to receive specific RSCS messages, you can use the SET command or the SETMSG command and statement. You can use the RSCS event scheduler to issue commands automatically at defined times (for example, closing console spool files). You can also use the RECOVERY statement to issue specific CP commands, including re-IPLing GCS, if RSCS abends.

Chapter 10. Configuring RSCS

RSCS comes with various sample files that you can use for reference or edit for actual use. To customize these sample files, log on to the RSCS user ID and access the 401 disk as an available file mode. You can then use a CMS editor to change the files.

Note: For additional information about customizing your RSCS network, see [Chapter 11, “Defining the RSCS Network Structure,”](#) on page 105 and [z/VM: RSCS Networking Exit Customization](#).

Sample Files for Configuring RSCS

The sample files supplied with RSCS were copied to their usable form (file name and file type) as part of RSCS installation. For more information, see the RSCS program directory.

The following RSCS sample files are on the RSCS 401 disk:

Note: This is not a complete list of the sample files shipped with RSCS. For example, the sample exit packages are not listed. See the RSCS program directory for a complete list of sample files.

File Name

Purpose

RSCS CONFIG

RSCS configuration file.

Note: Two sample configuration files are shipped with RSCS, either of which could have been copied to the RSCS 401 disk as RSCS CONFIG:

- RSCS CONFSAMP is self-contained.
- RSCSSMPL CONFSAMP is a master file that imbeds the following configuration files, each containing a specific group of statements:
 - AUTHS
 - DEST
 - EXITS
 - FRONT
 - ITRACE
 - NETLINKS
 - PRINTERS
 - ROUTES
 - TRAILER

[Appendix A, “Sample RSCS Configuration File,”](#) on page 297 shows what RSCSSMPL and its imbeds look like.

PROFILE GCS

Exec that contains the commands to be run when you IPL GCS.

NOTHERE TEMPLATE

Unknown user ID template.

UNKNOWN TEMPLATE

Unknown node ID template.

LPRHOLD TEMPLATE

Template for undeliverable files queued to LPR-type links.

UFTHOLD TEMPLATE

Template for undeliverable files queued to UFT-type links.

Setting Up the RSCS Configuration File

Chapter 6, “Planning for RSCS Configuration,” on page 55 describes various statements in the RSCS configuration file and provides information for planning purposes. This section provides more detail and background to help with your implementation.

You must define the name of your RSCS configuration file to GCS using a GCS FILEDEF command. (You can either enter this command or place it in the PROFILE GCS exec for the RSCS virtual machine.) RSCS supplies a sample GCS profile that uses RSCS CONFIG as the name of the configuration file. (See *z/VM: RSCS Networking Operation and Use* for an example of the GCS profile.) The configuration file can reside on any disk that is accessible when RSCS is initialized. Note, however, that the disks access by your PROFILE EXEC may not be the same as those accessed by the PROFILE GCS file. When RSCS is initialized, the disks listed in the PROFILE GCS will be accessible.

RSCS supplies a sample configuration file called RSCS CONFSAMP. If you are migrating from a previous version of RSCS, configuration files from RSCS V2 and RSCS V3 may still be used. However, if you are migrating from RSCS V2, make sure that you note the differences in the “ROUTE Statement” on page 258.

During the installation RSCS, your previous CONFIG file (if you were migrating from a previous release of RSCS) or the RSCS CONFSAMP file was copied to the RSCS 401 disk. (This disk is also the RSCS 191 disk.) When you copy the RSCS CONFSAMP file, it is renamed to * CONFIG (* is a file name).

Note: “RSCSSMPL CONFIG” on page 298 shows another sample configuration file, which implements the IMBED statement. This sample file and its associated imbed files are also supplied with RSCS.

During initialization, RSCS processes your configuration file. This processing includes:

- Allocating storage for:
 - Tag slots
 - Tag shadow elements (using the number you specified on the SHADOWS statement or using the default)
- Creating tables:
 - An authorization table with entries for each AUTH statement
 - A channels table with entries for each address specified on the CHANNELS statement
 - A destination identifier table with entries specified on DEST statements
 - A FCB table with entries for each FCB statement
 - A form table with entries for each FORM statement
 - An internal trace table with entries for each ITRACE statement
 - A link table with entries for each link definition
 - A port table with entries for each PORT statement
 - A reroute table with entries for each REROUTE statement
 - A route table with entries for each ROUTE statement.

If RSCS encounters an error while processing a configuration file statement, it displays the statement and a message describing the error. RSCS will continue to process the configuration file unless the error:

- Has a T (terminal) severity
- Occurs in a section of your configuration file that you marked as intolerant of errors (using the TOLERANCE statement).

Defining Links and Routes

Table 10 on page 87 shows which TYPE operand to specify on a LINKDEFINE statement when communicating over TCP/IP, other non-SNA, or SNA links to remote nodes.

Table 10. TYPE Operand Specifications

If node is:	(Non-SNA) TYPE	(SNA) TYPE
A client in a TCP/IP network	LPD, UFTD	n/a
A daemon in a TCP/IP network	LPR, UFT	n/a
Another (peer) system	NJE	SNANJE
Another (peer) system in a TCP/IP network	TCPNJE	SNANJE
ASCII printer or plotter attached to a protocol converter	ASCII	n/a
ASCII printer or plotter attached through a terminal server	TCPASCII	n/a
Remote multi-leaving workstation	MRJE	n/a
Remote or local 3270 printer	3270P, TN3270E	SNA3270P
Remote workstation	RJE	SNARJE

Note: Non-SNA printers attached to VTAM require a link type of SNA3270P.

The following example shows how LINKDEFINE and LOCAL statements in different systems' configuration files relate to one another. For example, [Figure 37 on page 87](#) shows two connected z/VM systems, one in Chicago and one in New York. The LOCAL and LINKDEFINE statements define a non-SNA BSC link between the systems.

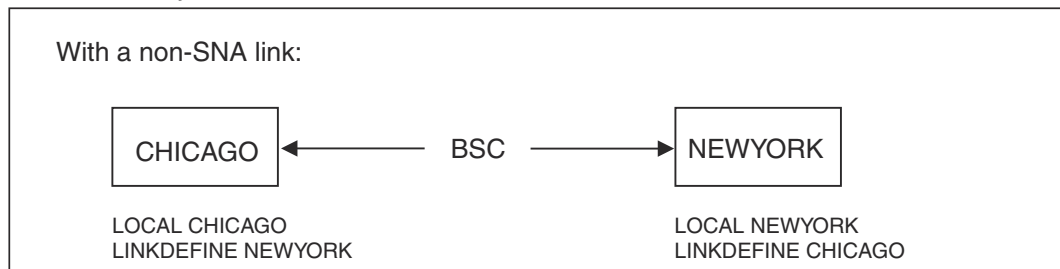


Figure 37. A Simple Two-System Network

The LOCAL statement contains a local node ID (CHICAGO or NEWYORK), which defines the name by which the local node is known to other nodes in the network. The LINKDEFINE statements identify the adjacent nodes (which, in this case, are the target nodes) and their link types (NJE-type in the non-SNA network, SNANJE-type in the SNA network, or TCPNJE-type in a TCP/IP network). These statements also identify the virtual device addresses or lines in the non-SNA network or logical unit names in the SNA network.

[Figure 38 on page 88](#) shows three connected z/VM systems: one in New York, one in Miami, and one in Dallas. The top half of the figure shows the systems connected by BSC lines. The bottom half of the figure shows the systems connected by SNA links. Note that in the bottom figure, VTAM must be installed and running on each system. Also, a VTAM link must be defined between each system in order to establish the SNANJE-type links.

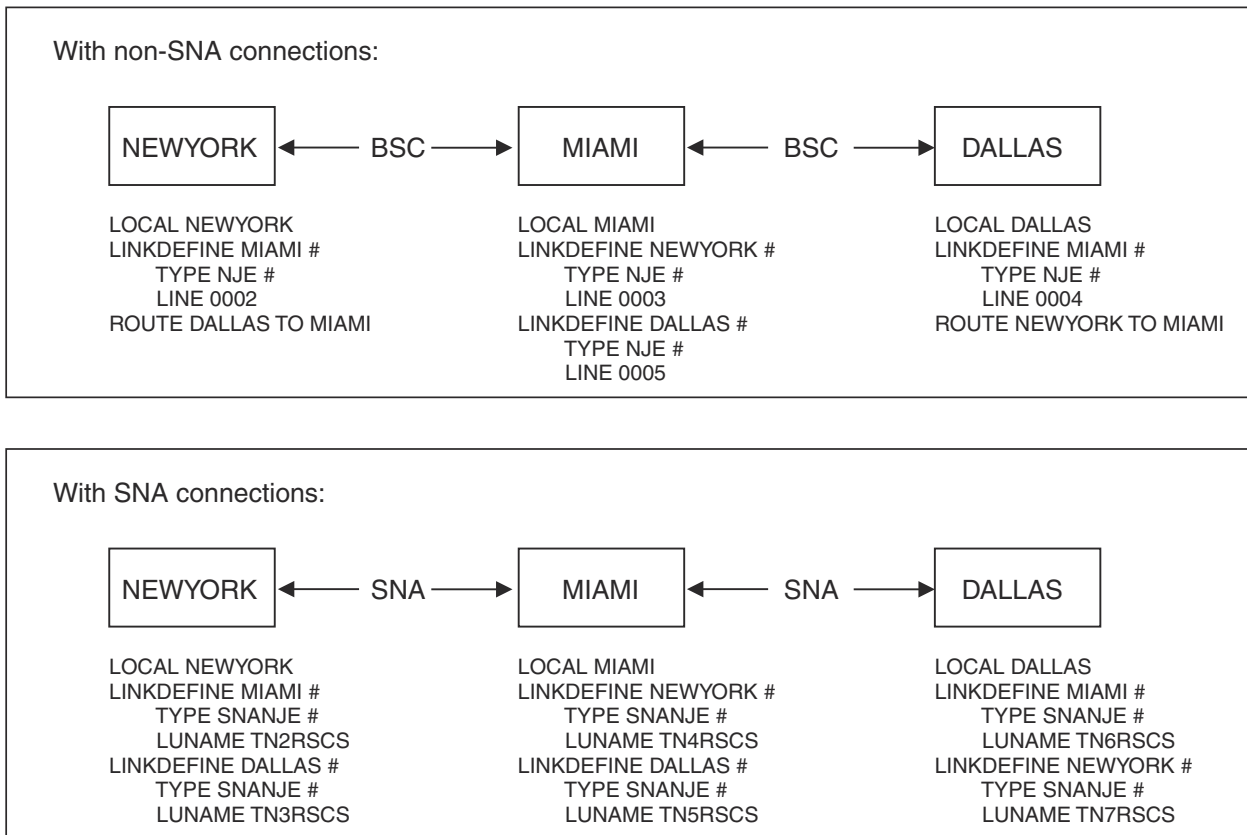


Figure 38. A Three-System Network as Defined to RSCS

Files transferred between NEWYORK and DALLAS are handled differently, depending upon whether the connections are defined as SNA or non-SNA. Notice how you use LINKDEFINE and ROUTE statements differently, depending upon the type of connections.

In Figure 39 on page 88, the three peer systems are located in a TCP/IP network. Each system is running TCP/IP and has defined TCPNJE-type links to each other.

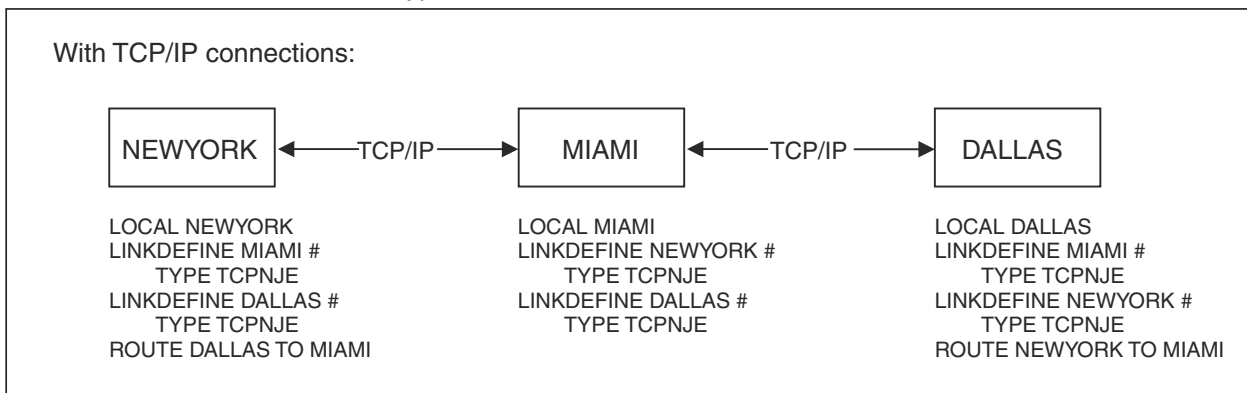


Figure 39. A Three-System Network Defined to RSCS with TCP/IP Drivers

In Figure 40 on page 89, four z/VM systems are connected in an NJE network.

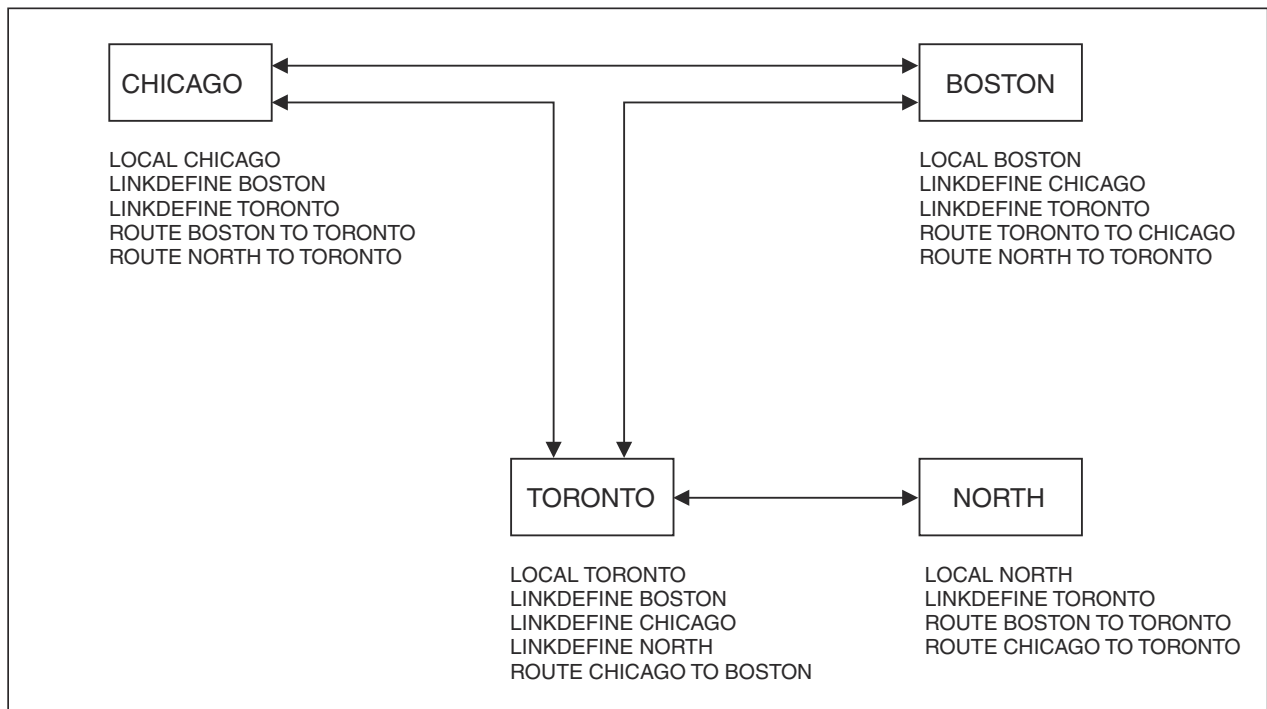


Figure 40. Alternative Paths in a NJE Network as Defined to RSCS

RSCS lets you to define direct paths and indirect paths to a node through an intermediate system. The configuration files for each system define alternate paths between two nodes in the network. Use caution, however, in defining alternate paths when multiple paths exist. Two systems should not define alternate paths through each other. This could cause a routing loop.

For example, suppose CHICAGO and TORONTO both defined alternate paths to BOSTON through each other. If their respective links to BOSTON became inactive, a file destined for BOSTON would be transmitted back and forth between CHICAGO and TORONTO continually (in a loop) until one of their BOSTON links became active again.

Similar examples could be cited for other paths in the sample network. Close coordination is necessary among personnel at different network nodes when defining a multisystem network. To avoid possible routing loops, define alternate paths only on one of any two systems, as shown in [Figure 40 on page 89](#). (Only CHICAGO defines an alternate path to BOSTON through TORONTO.)

Using Second-Level Addressing to Avoid Routes

Networks can take advantage of a technique called *second-level addressing* to reduce the number of entries in their route tables by including only system nodes. Because of the way RSCS handles destination addresses on files, remote systems can direct files to printers or workstations defined on the local system with LINKDEFINE statements or DEFINE commands, but not defined on the remote system with ROUTE statements or commands.

If you specify a valid value for one of the following:

- Printer link ID
- Workstation link ID (defined on the local system)
- List processor link ID
- Group ID (routed to printer and workstation links only)

in the user ID field of the TAG data accompanying the file, RSCS queues the file on the appropriate link or links. If the specified ID is not a printer link ID, workstation link ID, list processor link ID, or group ID,

RSCS spools the file to the specified user. (User IDs, local link IDs, workstation IDs, list processor IDs, and group IDs should be unique.)

RSCS does not support second-level addressing to system nodes or route groups containing a mixture of routes to networking and non-networking links. If the user ID field of the TAG data accompanying a file specifies a system link ID, RSCS tries to send the file to a z/VM user identified by the USERID field. If this fails, RSCS will do the following:

1. Send an error message to the file's originator.
2. Spool the file to SYSTEM (the local node).
3. Send the file to a NOTIFY link, if one is defined.

If your routing group contains a mixture of routes to networking and non-networking links and you specify this route group on the user ID field of the TAG data accompanying a file, the results are unpredictable.

Coding Statements in the Configuration File

Table 11 on page 90 shows an alphabetic listing of the possible statements and a brief description of each. For more information on individual statement formats and parameters, see [Chapter 15, "RSCS Configuration File Statements,"](#) on page 129.

Table 11. Summary of Configuration File Statements

Statement	Description
*	Indicates a comment.
/* ... */	Indicates a comment.
AUTH	Identifies an authorized alternate operator for RSCS (such as RSCSAUTH) or for a specified link.
CHANNELS	Reserves channels for exclusive use by RSCS for its virtual device pool.
DEST	Specifies local PSF printer destinations.
DUMP	Tells RSCS what type of dump to take when a RSCS problem (abend) occurs and the user (virtual machine) to which RSCS should send the dump.
EXIT	Defines and loads RSCS exit routine modules during RSCS initialization.
FCB	Defines an FCB image which RSCS will simulate on SNA3270P-type, 3270P-type, and TN3270E-type links.
FORM	Specifies the characteristics of a form name.
HIDECHARACTER	Specifies the default hide character. Use the hide character to mark sections of link parameter definitions in the RSCS configuration file or the RSCS events file that you do not want displayed in response to a QUERY command.
IMBED	Includes and processes a file as if it were a part of the main configuration file.
ITRACE	Defines the initial state of the RSCS internal trace table, defines the events RSCS will trace, and optionally start tracing when RSCS initializes.
LANGUAGE	Specifies the name of a load module containing a compiled RSCS message file.
LINKDEFINE	Defines the default attributes of an RSCS link.
LINKTYPE	Dynamically defines additional link drivers to the base RSCS system.
LOCAL	Contains the node ID of the local system and (optionally) the application ID by which VTAM knows RSCS.

Table 11. Summary of Configuration File Statements (continued)

Statement	Description
MSGNOH	Tells RSCS to use the CP MSGNOH command when issuing messages to local users.
OPFORM	Contains the operator form name.
OPTION	Lets installations define various RSCS system-oriented and link-oriented default parameters.
PARM	Defines operational characteristics that a link driver is to use for a specified link.
PORT	Specifies a virtual device number for a BSC line to be used by an auto-dial or auto-answer link, or for dynamic allocation of a BSC link.
RECOVERY	Specifies which CP command (or commands) should be issued when an RSCS task abends causing RSCS to become disabled.
REROUTE	Modifies the original destination of commands, files, and messages for specific systems and user IDs.
RETRY	Tells RSCS when to restart deactivated auto-dial and SNA links.
ROUTE	Describes the path to a nonadjacent node.
SAFCLASS	Specifies the output class to be used when RSCS creates store-and-forward files.
SETMSG	Starts the sharing of RSCS messages based on message number.
SHADOWS	Specifies how many tag shadow elements RSCS should reserve.
SLOWDOWN	Tells RSCS when to enable or disable slowdown mode.
TOLERANCE	Marks areas of the configuration file as tolerant or intolerant of errors.
TRACEDEST	Tells RSCS where to send trace files.
UPARM	Specifies additional, installation-defined parameters.

The configuration file may be of variable or fixed format. Statements in the file may be up to 256 characters long, but alphabetic characters must be in upper case (exceptions are comments and printer queue names specified for LPR-type links). Any information appearing after the last positional parameter on a statement is treated as a comment and ignored. Serial identification must not be present in the configuration file. If it is present, it will be taken as configuration file input. Blank lines are allowed in the file.

Table 12 on page 91 defines the required sequence and number of statements that you should have in your configuration file. All RSCS configuration file statements are optional.

Table 12. Number and Sequence of Statements

Statement	Position in file	Number
*	Anywhere (see note “1” on page 92)	No limit
/* ... */	Anywhere (see note “1” on page 92)	No limit
AUTH	After corresponding LINKDEFINE	No limit
CHANNELS	Before all LINKDEFINE, PORT	1
DEST	After LOCAL (if specified)	No limit
DUMP	After LOCAL (if specified)	1

Table 12. Number and Sequence of Statements (continued)

Statement	Position in file	Number
EXIT	After LOCAL (if specified)	No limit
FCB	Before all LINKDEFINE	No limit
FORM	After LOCAL (if specified)	No limit
HIDECHARACTER	After LOCAL (if specified)	1
IMBED	Anywhere (see note “2” on page 92)	No limit
ITRACE	After LOCAL (if specified)	No limit
LANGUAGE	After LOCAL (if specified)	1
LINKDEFINE	After LOCAL, CHANNELS, OPFORM, corresponding FCB	No limit
LINKTYPE	Before corresponding LINKDEFINE	No limit
LOCAL	First	1
MSGNOH	After LOCAL (if specified)	1
OPFORM	Before all LINKDEFINE	1
OPTION	After LOCAL (if specified)	No limit
PARM	After OPTION (if specified), corresponding LINKDEFINE	No limit
PORT	After LOCAL, CHANNELS	No limit
RECOVERY	After LOCAL (if specified)	1
REROUTE	After LOCAL (if specified)	No limit
RETRY	After LOCAL (if specified)	1
ROUTE	After corresponding LINKDEFINE	No limit
SAFCLASS	After LOCAL (if specified)	1
SETMSG	After LOCAL (if specified)	No limit
SHADOWS	After LOCAL (if specified)	1
SLOWDOWN	Before all LINKDEFINE	1
TOLERANCE	Anywhere (see note “2” on page 92)	No limit
TRACEDEST	Local destination: After LOCAL, if specified Remote destination: After corresponding LINKDEFINE or ROUTE	1
UPARM	After corresponding LINKDEFINE	No limit

Notes:

1. Because RSCS does not consider comments to be configuration file statements, you can specify comments before the LOCAL statement.
2. Configuration file statements define your network to RSCS. Two statements, IMBED and TOLERANCE, help RSCS process the configuration file, but do not affect your network definition. For this reason, IMBED and TOLERANCE are the only two statements that you can specify before the LOCAL statement.

- Although unlimited PARM statements are allowed, you can define only one PARM statement for each corresponding LINKDEFINE statement.

When coding statements in your configuration file, you should try to group related statements together. The sample configuration file (“RSCSSMPL CONFIG” on page 298) shows related LINKDEFINE, AUTH, PARM, and ROUTE statements grouped this way.

Setting Up the RSCS Events File

The RSCS event scheduler controls the scheduling and processing of all RSCS events. The following terms describe different types of events:

Daily

An event scheduled using the SCHEDULE command with the DAILY operand. The DAILY operand allows a temporary event to survive the midnight reloading of the events file.

External

Events scheduled from entries in the events file and using the SCHEDULE command.

Internal

Events scheduled by an exit routine. This lets you define events other than commands. For example, you could use the exit to schedule the posting of an ECB or to call a specific routine.

Midnight

The event scheduled by RSCS at midnight (24:00) that reloads the events file.

Permanent

Events scheduled from entries in the events file.

Repetitive

One event scheduled several times during a 24-hour period.

Single

One event scheduled once during a 24-hour period.

System

Events scheduled from entries in the events file.

Temporary

Events scheduled using the SCHEDULE command without the DAILY operand. These events will not survive the midnight reloading of the events file.

User

An event scheduled using the SCHEDULE command (with or without the DAILY operand).

Creating an Events File

Before RSCS can schedule and process events, you need to create an events file (or enter the SCHEDULE command as discussed in *z/VM: RSCS Networking Operation and Use*). Your events file must be a CMS file with a *ddname* of EVENTS. Like the RSCS configuration file, you must identify the events file to GCS using the GCS FILEDEF command. You can either enter the FILEDEF command manually or place it in RSCS PROFILE GCS exec. (*z/VM: RSCS Networking Operation and Use* shows examples of GCS FILEDEF commands for the RSCS configuration and events files.)

The event scheduler looks for a defined *ddname* of EVENTS. If it does not find a file by this name, the event scheduler initializes without reading an events file. If the event scheduler finds a file with a *ddname* of EVENTS, it reads the file and schedules the appropriate events.

Format of the Events File

The events file can be in either fixed format with a width of up to 256 characters or in variable format. All entries must be in uppercase, except for comments. There are three types of valid records: comments, special days, and events.

Comment Records

To define a comment record in the events file, place an asterisk (*) in the first column. The event scheduler ignores the rest of the line when it encounters an asterisk in column 1.

Special Day Records

To define a special day record in the events file, use the following format:



SPECIAL

Identifies a special day.

letter

Alphabetic character associated with the special date (or dates) you are defining.

date

Date in the *yy/mm/dd* format. If you want to define the special date that occurs every year, month, or day, specify one or two asterisks (*) instead of *yy*, *mm*, or *dd*. (See “Defining Holidays and Other Special Days” on page 97 for an example.)

Event Records

To define an event record in the events file, you need the following items in the following order:

Column Number	Column Contents	Description
1	Date	Tells RSCS the date that the event should take place. For daily events, use an asterisk. For specific dates, specify the date in the <i>yy/mm/dd</i> format. For generic dates, specify the date format (<i>yy/mm/dd</i>) and substitute asterisks for the generic parts. (For example, 07/**/01 indicates the first day of every month in the year 2007.)
2	Time	Tells RSCS the time that the event should take place. Specify the time in one of four forms: INIT Schedules the event every time RSCS is initialized. ++:mm Schedules the event every <i>mm</i> minutes past the hour. M:mmm Schedules the event every <i>mmm</i> minutes. hh:mm Schedules the event at a specific time.

Column Number	Column Contents	Description
3	Days of the week	<p>Tells RSCS on which days of the week to schedule the event. This column is eight characters long and can be divided into two logical sections: the first seven characters and the eighth character.</p> <p>The first seven characters represent the seven days of the week in the following order: Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday. If you do not want to schedule the day, use a dash (-) in the appropriate position. Any other character indicates to RSCS that you want the event scheduled on that day. For example, the following four groups of seven characters each tell RSCS to schedule the event on Mondays, Wednesdays, and Fridays:</p> <pre>M-W-F-- *-*-*-- X-X-X-- 1-3-5--</pre> <p>The eighth character represents the special day position. (See the description of the special day record which precedes this table.) Again, using a dash (-) tells RSCS that you do not want the event scheduled on any defined special day. To schedule the event on all defined special days, use an asterisk (*). To schedule the event on a class of special days, use the letter associated with the special days.</p> <p>If you want to schedule every day of the week and all defined special days, use one asterisk or mark all eight spaces.</p>
4, 5	Time range (low, high)	<p>Tells RSCS when to start and stop scheduling a repetitive task. For events scheduled all day, place an asterisk (*) in both columns. For a specific time range, specify a start time and a stop time, each in the <i>hh:mm</i> form. Specify the time using the 24-hour clock (that is, <i>hh</i> can be 00 - 23 and <i>mm</i> can be 00 - 59). You can either specify all day (an asterisk in each column) or a specific time range (<i>hh:mm</i> in each column). You cannot specify an asterisk in one column and a specific time in the other. Nor can you specify an asterisk for <i>hh</i> or <i>mm</i>.</p>
6	Command	<p>Tells RSCS what the event is. The column can be up to 132 characters long in the following form:</p> <pre>[taskname:]RSCS_command</pre> <p>taskname: is the 1 - 8 character name you are assigning to the task, followed by a colon (:), and a blank. Once RSCS initializes, you can use the task name to delete, resume, or suspend an event (or a group of events with the same task name) using the SCHEDULE command.</p> <p>RSCS_command is any of the RSCS commands described in z/VM: RSCS Networking Operation and Use. Remember: You can use the RSCS CP command to pass commands to CP and you can use the RSCS EXEC command to process an exec containing non-RSCS commands (such as GCS or CMS commands).</p>

Automating Simple Events

Suppose you wanted to close the RSCS console log every 30 minutes. You could automate this using the following entry in your events file:

```
*-----*
*                RSCS Events File                *
*-----*
*                Days-of-      Range              *
*-----*
```

Configuring RSCS

```
* Date      Time      the-week      Low  High  Command      *
* -----*
*          *      M:030      *          *          *      CP CLOSE CONS      *
```

Suppose you wanted to close the console every half hour, but you wanted it closed at a specific time each hour. The following events file would close the RSCS console log twice every hour: on the hour and on the half hour.

```
*-----*
*                               RSCS Events File                               *
*-----*
*                               Range                                         *
* Date      Time      Days-of-   Low  High  Command      *
* -----*
*          *      ++:00      MTWRFSS*  *          *      CONS: CP CLOSE CONS
*          *      ++:30      MTWRFSS*  *          *      CONS: CP CLOSE CONS
```

In this events file, the commands are prefaced with a task name (CONS). By specifying a prefix, your operators can suspend, resume, or delete both events using a single SCHEDULE command.

Suppose your RSCS virtual machine was not very active (or you like large console files). If you wanted to close the console only twice a day (once at noon and once at midnight), you could use the following events file:

```
*-----*
*                               RSCS Events File                               *
*-----*
*                               Range                                         *
* Date      Time      Days-of-   Low  High  Command      *
* -----*
*          *      00:00      *          *          *      CONS: CP CLOSE CONS
*          *      12:00      *          *          *      CONS: CP CLOSE CONS
```

If you wanted to close the console only once a month (on the first day of the month), you could use the following events file:

```
*-----*
*                               RSCS Events File                               *
*-----*
*                               Range                                         *
* Date      Time      Days-of-   Low  High  Command      *
* -----*
*          **/**/01  00:00      *          *          *      CP CLOSE CONS
```

If you wanted to change all files from hold to nohold every 10 minutes on a link named LPR5, you could add the following entry to your events file:

```
*-----*
*                               RSCS Events File                               *
*-----*
*                               Range                                         *
* Date      Time      Days-of-   Low  High  Command      *
* -----*
*          *      M:010      *          *          *      CHANGE LPR5 HOLD NOHOLD
```

Automating More Complex Tasks

Suppose your installation wrote an exit routine that prevented RSCS from sending large files during peak business hours (first shift). You could use the following events file to define when shifts start and end:

```
*-----*
*                               RSCS Events File                               *
*-----*
*                               Range                                         *
* Date      Time      Days-of-   Low  High  Command      *
* -----*
*          *      INIT      MTWRF---  00:00  07:59  SHIFT: SHIFT 2
*          *      INIT      MTWRF---  08:00  16:59  SHIFT: SHIFT 1
*          *      INIT      MTWRF---  17:00  23:59  SHIFT: SHIFT 2
*          *      INIT      -----SS*  *          *      SHIFT: SHIFT 2
```


The rerouted file is then held on the appropriate link until the user takes action on the file or the purge period (specified by the PURGE option) expires. In this case, RSCS will automatically purge the file.

Defining a NOTIFY-Type Link

You can define NOTIFY-type links to perform specific functions. For example, you can define a NOTIFY-type link to handle files sent to unknown users, another to handle files sent to undefined nodes, and others to handle files that could not be delivered on LPR-type or UFT-type links. As [Figure 41 on page 98](#) shows, you could specify the following statements in your configuration file.

```
LINKDEFINE *UNKNOWN TYPE NOTIFY /* Define the *UNKNOWN link */
LINKDEFINE *NOTHERE TYPE NOTIFY /* Define the *NOTHERE link */
LINKDEFINE *LPRHOLD TYPE NOTIFY /* Define the *LPRHOLD link */
LINKDEFINE *UFTHOLD TYPE NOTIFY /* Define the *UFTHOLD link */
:
:
PARM *UNKNOWN MSGFILE=UNKNOWN /* Use UNKNOWN TEMPSAMP */ #
PURGE=5 /* Hold files for 5 days */ #
CLASS=U /* Send files class U */ #
:
PARM *NOTHERE MSGFILE=NOTHERE /* Use NOTHERE TEMPSAMP */ #
PURGE=7 /* Hold files for 7 days */ #
CLASS=N /* Send files class N */ #
:
PARM *LPRHOLD MSGFILE=LPRHOLD /* Use LPRHOLD TEMPSAMP */ #
PURGE=5 /* Hold files for 5 days */ #
CLASS=L /* Send files class L */ #
:
PARM *UFTHOLD MSGFILE=UFTHOLD /* Use UFTHOLD TEMPSAMP */ #
PURGE=7 /* Hold files for 7 days */ #
CLASS=F /* Send files class F */ #
:
ROUTE * TO *UNKNOWN /* Unknown nodes to *UNKNOWN */
ROUTE *USER* TO *NOTHERE /* Unknown users to *NOTHERE */
ROUTE *LPRH* TO *LPRHOLD /* Undeliverable file on LPR link */
ROUTE *UFTH* TO *UFTHOLD /* Undeliverable file on UFT link */
```

Figure 41. Sample Statements to Define a NOTIFY-Type Link

For more information, see the “LINKDEFINE Statement” on page 151. For more information, see the “PARM Statement for a NOTIFY-Type Link” on page 201. For more information, see the “ROUTE Statement” on page 258.

Defining TEMPLATE Files

You must define the contents of the template that a NOTIFY-type link uses to send notes. The default file name of the template is the link ID of its associated link (unless you specify the MSGFILE operand on the link's PARM information). The file type must be TEMPLATE.

Each TEMPLATE file must have a logical record length less than or equal to 256 characters. TEMPLATE files contain general text, which is used for all messages, and variables, which represent information that may change for each message. Each variable must begin with an ampersand (&) and end with a period (.). You can use a variable several times in the TEMPLATE file. However, the length of each line cannot exceed 251 characters after the substitution value for the variable is added to the text. [Table 13 on page 98](#) describes each variable and the maximum length of its substitution value.

Table 13. Variables Supported in NOTIFY Message Template

Variable	Length	Description
&LOCALNODE;	8	Local node name
&LINK;	8	Link ID of this NOTIFY-type link
&PURGEPERIOD;	3	Number of days a file stays on the link before being purged
&FROMUSER;	8	Origin user ID
&FROMNODE;	8	Origin node ID

Table 13. Variables Supported in NOTIFY Message Template (continued)

Variable	Length	Description
&TOUSER;	8	Destination user ID
&TONODE;	8	Destination node ID
&RECORDS;	8	Number of records in file
&SPID;	4	Local spool ID of file
&OSPID;	4	Origin spool ID of file
&DIST;	8	Distribution code of file
&CLASS;	1	Spool file class of file
©	3	Copy count of file
&FILENAME;	8	File name of file
&FILETYPE;	8	File type of file
&DSN;	24	Data set name of file
&FILESPEC;	24	File name and file type or data set name, depending on setting
&TIMEUS;	8	Time the file originated in US form (<i>hh:mm:ss</i>)
&DATEUS;	8	Date the file originated in US form (<i>mm/dd/yy</i>)
&TIMEINT;	8	Time the file originated in international form (<i>hh.mm.ss</i>)
&DATEINT;	8	Date the file originated in international form (<i>dd/mm/yy</i>)

Sample TEMPLATE Files

RSCS supplies two sample TEMPLATE files on the product tape. Figure 42 on page 100 shows the UNKNOWN TEMPSAMP file, which is specified for the *UNKNOWN link.

```
&FROMNODE; &FROMUSER; 00 W=&FROMUSER;

From:      RSCS System at &LOCALNODE;
Contact:   (support person) at &LOCALNODE;
Regarding: File sent to &TOUSER; at &TONODE; on &DATEUS; at &TIMEUS;
           File name: &FILESPEC;; size: &RECORDS; records

We cannot deliver your file because you sent it to a node we do not
recognize. We are holding the file until we hear from you. You have
three choices of action:

1. Do nothing; we will automatically erase the file after &PURGEPERIOD; days.

2. Transfer the file to the correct node ID, using the following command:

   CP SMSG netid CMD &LOCALNODE; TRANSFER &SPID; TO newnode newuser

3. Erase the file, using the following PURGE command:

   CP SMSG netid CMD &LOCALNODE; PURGE &SPID;

The "netid" is your local RSCS machine (if you do not know the
"netid", enter the IDENTIFY command on the command line), "newnode"
is the correct node ID, and "newuser" is the correct user ID.

CMS users can enter these commands from the command line of the PEEK screen
or use the STACK subcommand to send the command. To use the STACK subcommand:

1. Use the DOWN command to make the example TRANSFER or PURGE command
   line in this note the current line.

2. Type over "netid", "newnode", and "newuser" with the correct information.

3. Issue STACK 1 on the command line.
```

Figure 42. Sample UNKNOWN TEMPSAMP File

Specifying Destination Identifiers

In addition to second-level addresses (for workstation or printer links) and user IDs, you may need to identify other destinations that are valid for your local system, such as page-image printers under the control of Print Services Facility/VM (PSF). RSCS Version 3 lets you specify destinations using the DEST. The DEST configuration statement lets you specify these destinations during RSCS initialization. The DEST command lets you dynamically add or delete destinations while RSCS is running. (See [z/VM: RSCS Networking Operation and Use](#) for more information about the DEST command.)

The NJE header of a file being transmitted contains a node address and a destination address at that node. The destination address can be either:

- The user ID of a virtual machine at the destination node
- A second-level address, a workstation or printer link defined only at that node
- The destination identifier of a PSF-controlled printer

Note: A destination identifier must be unique for that node. That is, if a destination identifier is the same as a user ID or a second-level address, a conflict will arise.

A user ID or second-level address was supplied by tag text information on the TAG command when the file was queued for transmission. A destination identifier was supplied by the DEST parameter of the SPOOL command when the file was queued for transmission. For more information on the SPOOL and TAG commands, see [z/VM: RSCS Networking Operation and Use](#).

When a file arrives at its destination node, RSCS compares the destination address for that file with the second-level addresses (links) that are defined at that location. If a matching link ID is found for a workstation or printer link, the file is transmitted over that link. If no link ID match is found, the identifier is compared with entries in the destination identifier table (if one exists). If a match is found, the file is spooled to the PSF-controlled output device that has a DEST the same as the destination identifier. If no

match is found, the identifier is assumed to be a user ID, and the file is spooled to the virtual machine with that user ID. (If no virtual machine has that user ID, CP sends the file to the SYSTEM output device, and an error message is sent to the file originator.)

Notes for 3812, 3816, and 4224 Printers

For 3812, 3816, and 4224 printers, a destination may actually represent a printer connected to RSCS. In this case, the files RSCS receives from users in the network must be processed by PSF before RSCS sends them to the printer.

To specify RSCS-attached printers, you must:

1. Define the printer names in your primary RSCS virtual machine, using the “[DEST Statement](#)” on page [135](#) or the DEST command (see [z/VM: RSCS Networking Operation and Use](#)).
2. Set up a second RSCS virtual machine on your system that will drive the printers. The node ID for the secondary RSCS virtual machine, which is specified on the LOCAL configuration file statement, should be different than the node ID for the primary RSCS virtual machine. Remember the node ID you specify; you will need it in Step “[5](#)” on page [101](#).
3. Define the printer names you specified in Step “[1](#)” on page [101](#) as printer links in the secondary RSCS virtual machine.
4. Connect your secondary RSCS virtual machine to your primary RSCS virtual machine using virtual CTC devices that have been defined in each virtual machine and coupled to each other. You must also define NJE-type links in each virtual machine to drive the channel-to-channel connection.
5. In the PSF 3820 PDM options file of the primary RSCS virtual machine, specify the RSCSTAG option (*nodeid* is the node ID of the secondary RSCS virtual machine you set up in Step “[2](#)” on page [101](#)):
RSCSTAG *nodeid*

The RSCSTAG option tells PSF the name of the RSCS node at which you attached the printers.

RSCS Message Repositories

The RSCS translation repository (DMTAMENG) contains a list of all RSCS messages. The RSCS conversion repository (DMTMGC) contains a list of all the variables for the messages in the RSCS message repository.

If you customize either of these files, you must recompile the message repositories, rebuild the RSCS load library, and initialize RSCS again. For more information about customizing message repositories, see [z/VM: RSCS Networking Exit Customization](#). The RSCS program directory also contains information about how to make local modifications.

Using Exit Routines to Customize RSCS

An **exit** is a procedure, outside the main RSCS program, that can be performed under specific conditions. An **exit point** is a location within the main RSCS program code from which control passes, and immediately after which control returns to, when the conditions are met. An **exit routine** is the code that is run when the conditions are met.

Because RSCS is a general-purpose networking product, exits let you customize RSCS processing according to the needs of your installation. The RSCS exit facility, transmission algorithms, ASCII exits, gateway programming interface, and loadable link drivers eliminate the need to alter RSCS source code to get the required processing.

The exits are classified according to the following categories:

- An **IBM-defined exit** is called when both of the following conditions are met:
 - Its corresponding exit point is associated with (using an EXIT statement) an assembled, link-edited, and loaded exit routine.
 - The exit point is enabled (using the EXIT statement or command).

RSCS has 48 IBM-defined exit points. Exit points must be defined and associated with an exit routine entry point (by using the EXIT statement) before RSCS is initialized.

- **User-defined** exit points can be used in addition to the IBM-defined exit points supplied with RSCS. These exit points are defined by placing EXITCALL macros within your exit routines. Like IBM-defined exit points, user-defined exit points must also be defined and enabled (using the EXIT statement and command). The RSCS exit facility does not distinguish between IBM-defined and user-defined exit points.
- A **transmission algorithm** is called when all of the following conditions are met:
 - The TA operand is specified on a LINKDEFINE statement or in a DEFINE or START command for a networking (GATEWAY-type, LISTPROC-type, NJE-type, SNANJE-type, or TCPNJE-type) link.
 - The networking link is started.
 - A file is queued for transmission on the networking link.
 - The link tries to get a file for transmission.

IBM supplies two transmission algorithms (0 and 1) in the DMTAXA module at labels DMTAXAG0 and DMTAXAG1, respectively. You can supply additional, external transmission algorithms, which are not part of the RSCS load library, and specify their entry point name on the TA parameter. For compatibility, you can also supply additional transmission algorithms at labels DMTAXAG2 through DMTAXAGF (corresponding to transmission algorithms 2 through F). However, this is not recommended because you must then reassemble DMTAXA and link-edit the RSCS load library. For more information about transmission algorithms, see *z/VM: RSCS Networking Operation and Use*.

- An **ASCII exit** is used when a file is queued for transmission over an ASCII-type or TCPASCII-type link. IBM supplies sample exit routines for the ASCII printer and plotter processing modules, for PostScript printers and generic TCP/IP attached printers.
- An **LPD exit** is used when a file is received over an LPD-type link from a LPR client in a TCP/IP network. IBM supplies a sample LPD exit routine module.
- An **LPR exit** is used when a file is queued for transmission over an LPR-type link to a daemon in a TCP/IP network. IBM supplies two sample LPR exit routine modules, one for a PostScript printer and one for a generic LPR attached printer. These exits operate similarly to the TCPASCII exit routines.
- The **gateway programming interface** assists in creating gateway programs that enable GATEWAY-type links to communicate with other systems using various networking protocols.
- The **loadable link driver** facility (LINKTYPE statement) can be used to create additional types of link drivers at your facility. For more information, see “LINKTYPE Statement” on page 158.
- A **UFT exit** is used when a file is queued for transmission over an UFT-type link to a daemon in a TCP/IP network. IBM supplies a sample UFT exit routine module.
- A **UFTD exit** is used when a file is received over an UFTD-type link from a UFT client in a TCP/IP network. IBM supplies a sample UFTD exit routine module.

All modules containing exit routines, transmission algorithms, or gateway programs must be loaded before you enable an exit point, specify a transmission algorithm for a networking link, or start an ASCII-type, GATEWAY-type, LPD-type, LPR-type, TCPASCII-type, UFT-type, or UFTD-type link.

Because exits are optional and are not needed for proper RSCS functions, they are not discussed further in this document. (You can use the sample exit routines provided with RSCS by including the RSCSEXIT LOADLIB when you initialize RSCS. For more information, see *z/VM: RSCS Networking Operation and Use*.) For details about using the RSCS exit facility and writing exit routines, see *z/VM: RSCS Networking Exit Customization*.

Attention

Only experienced system programmers should attempt to use the RSCS exit facility and transmission algorithms. Writing exit routines, transmission algorithms, and gateway programs, or installing user-defined exit points requires a thorough knowledge of telecommunication protocol, systems programming, RSCS programming conventions, and RSCS internals.

If you attempt to write exit routines and install new exit points without this knowledge, you run the risk of both seriously degrading your system's performance and causing system problems.

Optional RSCS Servers

The following optional RSCS servers can be set up to do additional processing:

- **RSCS domain name server (RSCSDNS)** performs domain name system (DNS) searches on behalf of RSCS, translating fully qualified host names (domain name addresses) to the dotted decimal IP addresses supported in a TCP/IP network. If the RSCS domain name server is running, you can specify host names on LPR-type, TCPASCII-type, TCPNJE-type, and UFT-type links. For more information, see [Chapter 12, “Setting Up the RSCS Domain Name Server,”](#) on page 113.
- **RSCS dynamic authorization (RSCSAUTH)** is a facility that handles user authorizations for RSCS commands and links without updating the RSCS configuration file. Using RSCS dynamic authorization, you can authorize a user for any set of commands and links. You can add, change, or remove authorizations without restarting the RSCS server. For more information, see [Chapter 13, “Setting Up RSCS Dynamic Authorization,”](#) on page 115.
- **RSCS Data Interchange Manager (RSCS Interchange)** handles communications between NJE networks and networks that use Simple Mail Transfer Protocol (SMTP). Although NJE networks and SMTP networks have different mail formats, RSCS Interchange allows users in those networks to communicate with each other using their usual method of sending notes. For more information, see [Chapter 14, “Setting Up the RSCS Data Interchange Manager,”](#) on page 119.

Chapter 11. Defining the RSCS Network Structure

If you are adding RSCS to a network that has very few system nodes, designing the structure of your RSCS network is usually a simple task. Configuring RSCS on a system with connections to many system nodes or on a system that is part of a large network of system nodes is more difficult and requires careful consideration of the features available in RSCS.

Defining a Network with a Single System Node

Suppose you installed RSCS on the sole system node in the network. This node provides connections to assorted workstations and printers. If you have five printers (two SNA, two non-SNA, and one ASCII) and an RJE workstation, your system might look like this:

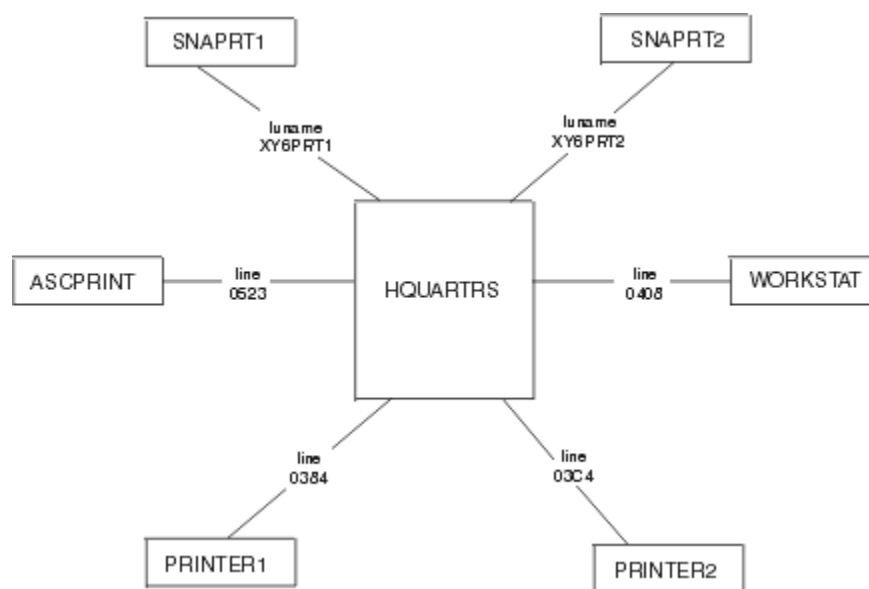


Figure 43. Sample Network with a Single System Node

The configuration file for this single system-node network would look like this:

```

LOCAL HQARTRS * RSCSHQ
/*-----*/
/* Link definitions                                     */
/*-----*/
LINKDEFINE SNAPRT1  TYPE SNA3270P  LUNAME XY6PRT1  QUEUE SIZE
LINKDEFINE SNAPRT2  TYPE SNA3270P  LUNAME XY6PRT2  QUEUE SIZE
LINKDEFINE ASCPRINT TYPE ASCII      LINE  0523      QUEUE FIFO
LINKDEFINE PRINTER1 TYPE 3270P      LINE  0384      QUEUE SIZE
LINKDEFINE PRINTER2 TYPE 3270P      LINE  03C4      QUEUE SIZE
LINKDEFINE WORKSTAT TYPE RJE        LINE  0408      QUEUE PRIORITY
:
  
```

The first two LINKDEFINE statements define the two SNA printers, SNAPRT1 and SNAPRT2. When defining SNA links, you must:

- Define RSCS as a VTAM application
- Define the printer to VTAM
- Supply a logon mode table entry.

For more information, see “Preparing VTAM for SNA Connections” on page 44. You can code the VTAM definitions for the SNAPRT1 and SNAPRT2 printers with the LOGAPPL parameter (see Table 5 on page 46

and Table 6 on page 46). The LOGAPPL parameter tells VTAM what system to log the printers onto when an operator powers them on. This means you would not have to enter a RSCS START command to start the link to these SNA printers.

The third LINKDEFINE statement defines the ASCII printer, ASCPRINT. When defining an ASCII printer, you must specify a line address that is controlled by the 7171 ASCII Device Attachment Control Unit or the 9370 ASCII Subsystem Controller.

The fourth and fifth LINKDEFINE statements define the two non-SNA printers, PRINTER1 and PRINTER2. When defining non-SNA printers, you must define the printer in the real I/O configuration file. For more information, see [“Preparing for Non-SNA Connections”](#) on page 36.

The last LINKDEFINE statement defines the RJE workstation. For more information about defining RJE workstations, see [“Preparing for Non-SNA Connections”](#) on page 36.

For more information, see the [“LINKDEFINE Statement”](#) on page 151. For more information, see the [“ROUTE Statement”](#) on page 258.

Creating a Printer Pool

Because all the nodes in this network are adjacent to the HQUARTRS node, you do not have to code ROUTE statements in the configuration file. Even with a simple configuration, however, there may be cases where ROUTE statements can improve your level of service to the users.

For example, if two or more of the 3270 printers are near each other and have the same printer features, you can create a printer *pool*. This RSCS feature lets users send their files to a pool of printers rather than to one individual printer. RSCS prints the file on the first available printer in the pool.

To create a printer pool using the two SNA printers in the previous example, you would add the following statement to the configuration file:

```
route group prtpool to link snaprt1 snaprt2
```

This statement places printers SNAPRT1 and SNAPRT2 into a common pool. Users can still send their files directly to SNAPRT1 and SNAPRT2. However, if a user sends a file to PRTPOOL, RSCS prints the file on the first available printer in the pool.

The printer pool support provides automatic load balancing across a set of printers and reduces the need for immediate operator intervention during a printer outage. You can place up to 255 printers in a single pool. Defining LPR printer pool groups can be especially helpful for print delivery throughput when many files are to be printed simultaneously.

Defining a Network with Multiple System Nodes

If you installed RSCS in a network that already has multiple system nodes, you must consider the structure of your network more carefully. For example, you may want to ask yourself:

- Does the configuration provide adequate bandwidth to the systems to which you expect to send the most traffic?
- Does the configuration provide sufficient redundancy so that a single machine outage does not cause a part of the network to be isolated from the rest of the network?

To a large extent, the answers to these questions will depend on how crucial the availability of your network is to your business and how expensive additional bandwidth and machines are.

Besides the printer and workstation links in [Figure 43 on page 105](#), suppose the HQUARTRS node also has connections to these system nodes:

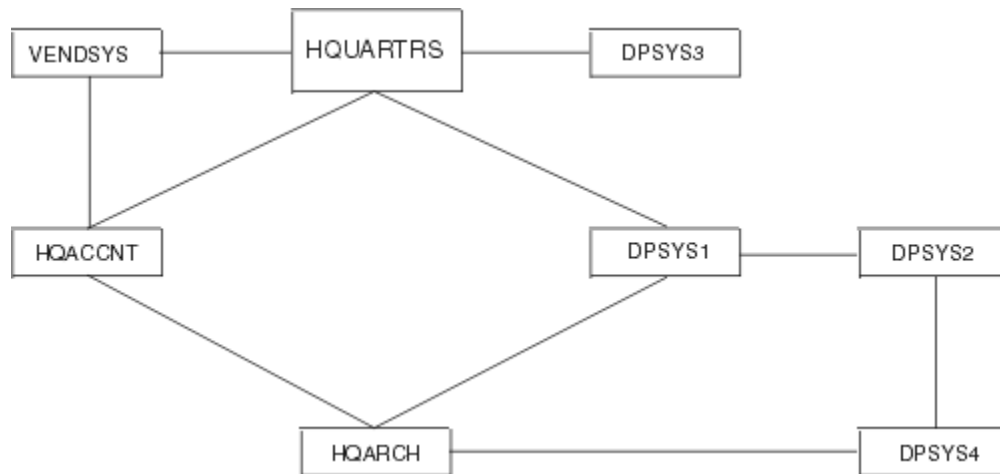


Figure 44. Sample Network with Multiple System Nodes

The configuration file for this multiple system-node network would look like this:

```

LOCAL HQARTRS * RSCSHQ
/*-----*/
/* Link definitions */
/*-----*/
LINKDEFINE HQACCNT TYPE SNANJE LUNAME XY6ACCNT QUEUE SIZE
LINKDEFINE DPSYS1 TYPE SNANJE LUNAME XY6PSYS1 QUEUE SIZE
LINKDEFINE DPSYS3 TYPE NJE LINE 0208 QUEUE SIZE
LINKDEFINE VENDSYS TYPE NJE LINE 0288 QUEUE SIZE
/*-----*/
/* Route definitions */
/*-----*/
ROUTE NODE DPSYS2 TO LINK DPSYS1
ROUTE NODE DPSYS4 TO LINK DPSYS1
ROUTE NODE HQARCH TO LINK HQACCNT
:

```

Adding Another Link to a Node

If the operators at the HQARTRS node discovered that the DPSYS3 link did not have sufficient capacity to handle the necessary traffic, they might consider upgrading the bandwidth of the link, or adding a second link to handle some of the traffic. Suppose the operators decided to add a second link at address 218. They replace the LINKDEFINE statement for DPSYS3 with the following LINKDEFINE and ROUTE statements:

```

linkdefine dpsys3a type nje line 208 queue size node dpsys3
linkdefine dpsys3b type nje line 218 queue size node dpsys3
:
route group dpsys3 to link dpsys3a dpsys3b

```

When you define a link, usually the name of the link matches the name of the remote node to which you are connecting. Because the operators defined two links to the same node, at least one of the links is going to have to use a nickname for the link ID. In this example, the operators chose nicknames for both links: DPSYS3A and DPSYS3B. The NODE operand on each LINKDEFINE statement tells RSCS the name of the node to which that link is attached, DPSYS3.

The ROUTE statement tells RSCS to send traffic destined for node DPSYS3 across either of the two links. RSCS sends traffic destined to DPSYS3 through whichever link is available first.

Making Transitions in Links

You could also create a second link for test purposes. Suppose you needed to change the link's link type from NJE to SNANJE. Instead of changing the existing link, you could create a second one with the new link type. This way, if there were any problems with the definition of the new link, there would be no

interruption in service between your local node and the remote node because the existing link would still be there to handle the traffic during an outage.

Creating Multiple Paths

Using the sample network in [Figure 44 on page 107](#) and the corresponding configuration file, you see that RSCS sends traffic destined to node HQARCH using node HQACCNT. Looking at the topology of the network, you can see that it is just as easy to send traffic using node DPSYS1. If the operators at node HQUARTRS considered the second path to be *just as good*, they could replace the ROUTE statement in the configuration file with the following ROUTE statement:

```
route node hqarch to link hqaccnt dpsys1
```

This ROUTE statement causes RSCS to send files destined to node HQARCH through whichever of the two links is available first.

Creating an Alternate Path

If the operators at node HQUARTRS decided that there had to be a path from node HQUARTRS to node HQARCH, they could change the ROUTE statement to include the ALTERNATE operand:

```
route node hqarch to link hqaccnt dpsys1 alternate vendsys
```

This ROUTE statement tells RSCS to use the links to nodes HQACCNT and DPSYS1 as the primary links for files destined to node HQARCH, but, if neither link is connected, RSCS should use the alternate link through node VENDSYS. There is no guarantee that this will get the files to node HQARCH, but no manual operator intervention in changing the routes could accomplish any more.

Special Note about Routing Loops

If you use alternate routes on several nodes in your network, it is easy to create loops in the routing of a node. RSCS has support to detect looping files, but it may sometimes require operator intervention to release the files after you resolve the routing problems.

Using Second-Level Routing to a Printer Pool

If users at node HQARCH want to print files on the PRTPOOL printer pool defined on node HQUARTRS ([Figure 43 on page 105](#)), they could send their files to PRTPOOL SYSTEM (assuming the operators routed PRTPOOL as a node on HQARCH and any intermediate nodes) or they could send their files to HQUARTRS PRTPOOL because the RSCS virtual machine at node HQUARTRS will perform second-level routing for printer groups. The latter implementation does not require any RSCS virtual machine except the one at node HQUARTRS to be aware of the PRTPOOL printer group.

Defining a Large Network

If the network described in [Figure 44 on page 107](#) represented just one location in a company's multi-location network, you must consider how the local network nodes should connect to the larger company network. Suppose you are an operator for a company with a network that has several local networks with slower links connected by a high-speed backbone network.

For example, if node HQUARTRS (as previously described) was a local network node connected to two other local network nodes using a high-speed backbone network, the entire network might look [Figure 45 on page 109](#).

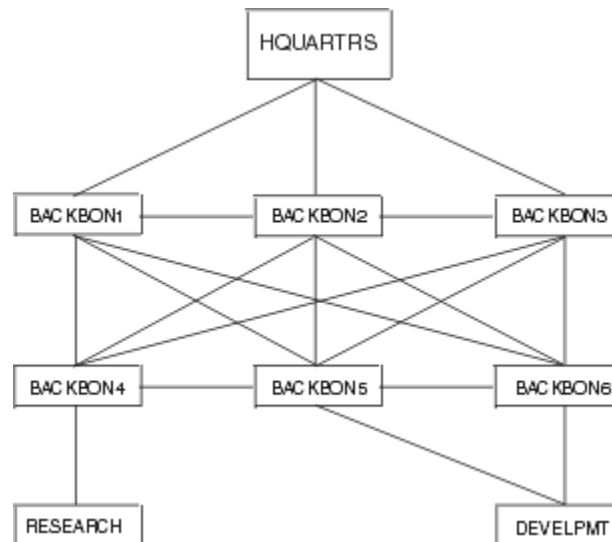


Figure 45. Sample Large Network

If node RESEARCH is representing 20 nodes (like node HQUARTRS is representing the seven nodes in [Figure 44 on page 107](#)), the operators could route each of those 20 nodes to one or more links on node HQUARTRS. However, there is a drawback to this approach. If the operators need to reroute the nodes in the RESEARCH network to avoid a network outage, they must enter 20 ROUTE commands at the HQUARTRS node.

Creating a Routing Group

To avoid this situation, the operators can place the 20 nodes in the RESEARCH network in a symbolic routing group at the HQUARTRS node. To create a routing group at node HQUARTRS, the operators could add the following ROUTE statements to their configuration file:

```
/*-----*/
/* Define the routing groups and their destinations */
/*-----*/
route group research to link backbon1 backbon2
/*-----*/
/* Define the nodes in group RESEARCH */
/*-----*/
route node rnode1 to group research
route node rnode2 to group research
:
route node rnode20 to group research
```

The first ROUTE statement defines group RESEARCH and tells RSCS to route all traffic destined to group RESEARCH through the high-speed links BACKBON1 and BACKBON2. The remaining ROUTE statements place each of the 20 nodes of the RESEARCH network into group RESEARCH.

If problems arise on BACKBON1 and BACKBON2, the operators at node HQUARTRS can redirect the traffic to BACKBON3 using one ROUTE command:

```
route group research to link backbon3
```

For more information about the RSCS ROUTE command, see [z/VM: RSCS Networking Operation and Use](#).

Creating a Routing Group Within a Routing Group

Nodes are not the only items that you can add to symbolic routing groups. The RSCS routing group support lets you add a routing group to another routing group.

Suppose the operators at node HQUARTRS defined routing groups for the RESEARCH network and the DEVELPMT network (a group called DEVGROUP with 15 development nodes). If both local networks

Defining the Network

were located on the west coast, the operators might decide to combine both groups into one group called WCOAST. They could accomplish this by adding the following ROUTE statements to the HQUARTRS configuration file:

```
/*-----*/
/* Define the routing groups and their destinations */
/*-----*/
route group wcoast to link backbon1 backbon2
route group devgroup to group wcoast
route group research to group wcoast
/*-----*/
/* Define the nodes in group RESEARCH */
/*-----*/
route node rnode1 to group research
route node rnode2 to group research
:
route node rnode20 to group research
/*-----*/
/* Define the nodes in group DEVGROUP */
/*-----*/
route node dnode1 to group devgroup
route node dnode2 to group devgroup
:
route node dnode15 to group devgroup
```

By combining the two groups into group WCOAST, the HQUARTRS operators can manipulate all 35 nodes in the RESEARCH and DEVELPMT networks using a single ROUTE command. They can also use a single ROUTE command to remove all the nodes in the RESEARCH group from the WCOAST group and route them through different links than those specified for WCOAST.

You can use any number of levels to create tree structures for your routing groups. And, you can use the RSCS QUERY command to get information about file queues on a group by group basis. This feature can help you find where the file traffic from your nodes is destined.

For more information about the RSCS QUERY command, see [z/VM: RSCS Networking Operation and Use](#).

Specifying Fanout Links

The company network in [Figure 45 on page 109](#) shows node HQUARTRS with links to three backbone nodes: BACKBON1, BACKBON2, and BACKBON3. Suppose that all three links are using virtual SNA circuits that traverse a single real link and that the real links in the network look like this:

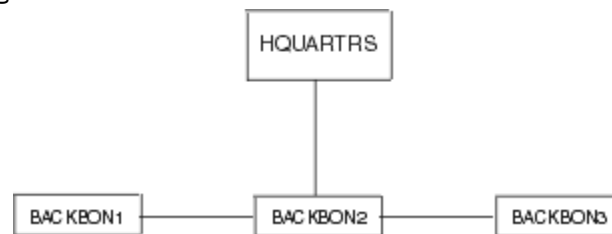


Figure 46. Physical Network Connection

[Figure 45 on page 109](#) shows the links to BACKBON1 and BACKBON3 sharing a common path through BACKBON2.

If someone sends a file to the list processor at node HQUARTRS which contains a distribution list with entries for each of the three backbone links, the list processor would split the file into three copies and send one copy through each backbone link. Because all three links share a portion of the physical path, you are sending three copies of the file over the physical link to node BACKBON2.

You can avoid this situation by telling RSCS more about your network. The LINKDEFINE statement has a FANOUT operand which you can use to tell RSCS about the commonality in the three paths. For more information about the FANOUT operand, see [“LINKDEFINE Statement” on page 151](#).

For example, if the operators at node HQUARTRS define the BACKBON1 and BACKBON3 links with a FANOUT operand of BACKBON2, they are telling RSCS that the BACKBON2 node is on the way to the other two nodes. The list processor uses this information when deciding how many copies of the file it needs to

create. Here, the list processor knows it only needs to send one copy to BACKBON2 because BACKBON2 can split out the other two copies destined for BACKBON1 and BACKBON3.

Chapter 12. Setting Up the RSCS Domain Name Server

Domain name system (DNS) support in RSCS allows you to specify a fully qualified host name (domain name address), such as `gd1vm7.endicott.ibm.com`, when defining LPR-type, TCPASCII-type, TCPNJE-type, and UFT-type links. The RSCS domain name server is an interface between the RSCS server and the DNS support defined in your z/VM TCP/IP configuration. It allows RSCS to resolve a host name into a dotted decimal IP address.

There is no configuration file for this server, but there is a PROFILE EXEC file on the RSCSDNS 191 disk.

RSCS Domain Name Server Virtual Machine

The z/VM user directory contains a sample definition for the RSCS domain name server virtual machine. The user ID is RSCSDNS. You might need to modify this sample definition for your installation. [Figure 47 on page 113](#) shows the sample RSCSDNS definition for a z/VM system that is not a member of a single system image (SSI) cluster. For simplicity, commented-out statements included in the sample definition are not shown here.

```
IDENTITY1 RSCSDNS password 32M 32M G
BUILD ON *2 USING SUBCONFIG RSCDNS-1
OPTION SVMSTAT3 LANG AMENG4
MACHINE ESA
IPL CMS PARM AUTOCD5
CONSOLE 01F 3215 T OPERATOR6
SPOOL 00C 2540 READER A
SPOOL 00D 2540 PUNCH A
SPOOL 00E 1403 A

SUBCONFIG RSCDNS-1
LINK RSCS 403 403 RR7
LINK MAINT 190 190 RR
LINK MAINT 19E 19E RR
LINK TCPMAINT 592 592 RR
MDISK 191 3390 2122 001 MH1W01 MR READ WRITE MULTIPLE
```

Figure 47. Sample RSCS Domain Name Server (RSCSDNS) Virtual Machine Definition

Note:

1. The IDENTITY statement identifies this as a multiconfiguration virtual machine definition. When included in the directory for an SSI cluster, this type of definition can be configured to enable the user ID to be logged on concurrently to multiple members of the cluster. The version of the definition shown here is for a non-SSI system, so the definition includes only one active BUILD statement and its associated SUBCONFIG entry and is equivalent to a single-configuration virtual machine definition (traditional USER definition).
2. The BUILD statement identifies the SUBCONFIG entry for this system, which contains system-specific statements. An asterisk (*) in the system name field indicates this definition is included in an SSI-ready (not SSI-enabled) directory, and the system name defaults to the IPLed system. In an SSI-enabled directory, this definition will contain a BUILD statement and its associated SUBCONFIG entry for each member of the cluster.
3. Specifies that RSCSDNS is a service virtual machine.
4. Specifies that American English is the language that will be set for RSCSDNS at logon.
5. Allows RSCSDNS to be autologged and to run its PROFILE EXEC.
6. Specifies the secondary user ID to control operations when RSCSDNS is disconnected; this is optional.

Note: If the RSCS domain name server is very active, it will send many messages to this user ID.

7. Links to the RSCS production disk containing the RSCS domain name server system code.

Running GETHOSTC

RSCSDNS has access to a copy of the GETHOSTC MODULE file on the RSCS production disk (RSCS 403). GETHOSTC runs on RSCSDNS and performs DNS searches on behalf of RSCS. GETHOSTC listens on a TCP/IP port for a connect request from the RSCS server. When a connect request is received, GETHOSTC reads the fully qualified host name sent by RSCS (up to 255 characters), issues the DNS search, and returns the 4-byte dotted decimal IP address to RSCS.

You need to define the port on which GETHOSTC will listen. This port is identified on the DNSPORT operand on the PARM statement for LPR-type, TCPASCII-type, TCPNJE-type, and UFT-type links. The default for GETHOSTC and for these PARM statements is port 900.

Using TCP/IP Well-Known Ports

By default, TCP/IP restricts the use of well-known ports (1–1023). To use port 900 or any other well-known port when usage is restricted, the RSCSDNS server must be authorized on a PORT statement in the TCP/IP configuration file (PROFILE TCPIP). For more information, see [z/VM: TCP/IP Planning and Customization](#).

Before invoking GETHOSTC, the RSCSDNS virtual machine must be set up as follows:

- Link and access the C/C++ runtime library (supplied with z/VM as part of Language Environment®).
- Global the SCEERUN LOADLIB:

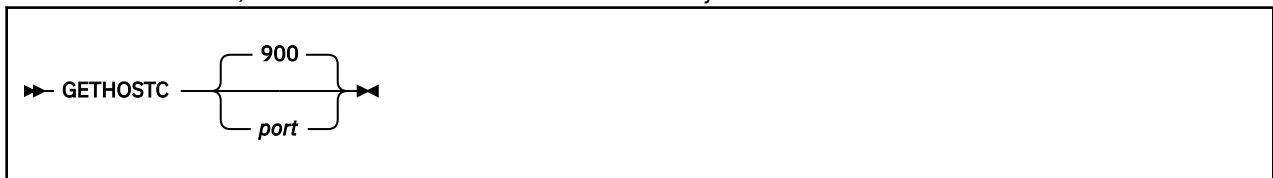
```
GLOBAL LOADLIB SCEERUN
```

SCEERUN is part of the C/C++ runtime library.

- Link and access the following minidisks:
 - TCPMAINT 592, or equivalent
 - RSCS 403

Note: The links are predefined in the sample definition for the RSCSDNS virtual machine.

To start GETHOSTC, issue the GETHOSTC command. The syntax is:



port

is the TCP/IP port on which GETHOSTC will listen for connect requests. The default is 900.

Multiple virtual machines can run GETHOSTC, all listening on the same port. If a well-known port is used, each of the user IDs running GETHOSTC must be authorized on a PORT statement in the TCP/IP configuration file (PROFILE TCPIP).

Chapter 13. Setting Up RSCS Dynamic Authorization

This section contains topics that explain what you need to do to set up RSCS dynamic authorization (RSCSAUTH) on your system. RSCS dynamic authorization allows you to authorize users for RSCS commands and links without updating the RSCS configuration file. Therefore a restart of the RSCS server is not required when you add, change, or delete authorizations. You can also use RSCS dynamic authorization to manage the authorizations for RSCS commands created by your installation.

Planning Checklist for RSCS Dynamic Authorization

- **Ensure** that you have a copy of the RSCS program directory.
- **Gather** information required for the RSCS dynamic authorization configuration file:
 - Person who will be fully authorized for the server.
 - Minidisk or SFS directory where the dynamic authorization tables and log files will be kept.

RSCS Dynamic Authorization Sample Files

The RSCS dynamic authorization sample files supplied with RSCS were copied to their usable form (file name and file type) on the indicated disks as part of RSCS installation. For more information, see the RSCS program directory.

The RSCS dynamic authorization sample files are:

- On the RSCS 403 disk:
 - RSCSAUTH CONFIG**
RSCS dynamic authorization configuration file
- On the RSCSAUTH 191 disk:
 - PROFILE EXEC**
RSCS dynamic authorization server profile

RSCS Dynamic Authorization Server Virtual Machine

The z/VM user directory contains a sample definition for the RSCS dynamic authorization server virtual machine. The user ID is RSCSAUTH. You might need to modify this sample definition for your installation. [Figure 48 on page 115](#) shows the sample RSCSAUTH definition for a z/VM system that is not a member of a single system image (SSI) cluster. For simplicity, commented-out statements included in the sample definition are not shown here.

```
IDENTITY1 RSCSAUTH2 password 32M 32M BG3
BUILD ON *4 USING SUBCONFIG RSCATH-1
OPTION SVMSTAT5 LANG AMENG6
MACHINE ESA
IPL CMS PARM AUTOCR7
CONSOLE 01F 3215 T
SPOOL 00C 2540 READER A
SPOOL 00D 2540 PUNCH A
SPOOL 00E 1403 A

SUBCONFIG RSCATH-1
LINK RSCS 403 403 RR8
LINK MAINT 190 190 RR
LINK MAINT 19E 19E RR
MDISK 191 3390 2123 010 MH1W01 MR READ WRITE MULTIPLE
```

Figure 48. Sample RSCS Dynamic Authorization Server (RSCSAUTH) Virtual Machine Definition

Note:

1. The IDENTITY statement identifies this as a multiconfiguration virtual machine definition. When included in the directory for an SSI cluster, this type of definition can be configured to enable the user ID to be logged on concurrently to multiple members of the cluster. The version of the definition shown here is for a non-SSI system, so the definition includes only one active BUILD statement and its associated SUBCONFIG entry and is equivalent to a single-configuration virtual machine definition (traditional USER definition).
2. If you change the name of this virtual machine, you will need to update other files that refer to the RSCSAUTH user ID, such as the AUTH statement in the RSCS CONFIG file and the XAUTOLOG command in the PROFILE GCS file.
3. Specifies the CP privilege classes (based on the default classes provided with z/VM) for which this virtual machine is authorized:

G

Allows RSCSAUTH to use general user CP commands.

B

Allows RSCSAUTH to use the privileged CP MSGNOH command to suppress CP message headers.

Note: This is the *only* reason class B is specified here. Your installation might consider redefining privilege classes, which is described in *z/VM: CP Planning and Administration*. You could define one class that includes the CP MSGNOH command only. Then, that class would be specified here instead of B. You could also specify other classes if you wish to use additional privileged CP commands at the RSCSAUTH console. However, **do not** specify class D.

4. The BUILD statement identifies the SUBCONFIG entry for this system, which contains system-specific statements. An asterisk (*) in the system name field indicates this definition is included in an SSI-ready (not SSI-enabled) directory, and the system name defaults to the IPLed system. In an SSI-enabled directory, this definition will contain a BUILD statement and its associated SUBCONFIG entry for each member of the cluster.
5. Specifies that RSCSAUTH is a service virtual machine.
6. Specifies that American English is the language that will be set for RSCSAUTH at logon.
7. Allows RSCSAUTH to be autologged and to run its PROFILE EXEC.
8. Links to the RSCS production disk containing the RSCS dynamic authorization system code.

Preparing RSCS for RSCS Dynamic Authorization

The RSCS configuration file contains a sample AUTH statement authorizing the RSCSAUTH user ID as a system-authorized alternate operator. This provides RSCSAUTH with authority for all RSCS commands (except INIT) and all links.

Note: The RSCSAUTH server also cannot issue the RSCS CMD command.

You do not need to make any changes to the AUTH statement unless you have changed the name of the RSCS dynamic authorization virtual machine.

The PROFILE GCS exec of the recovery machine includes an XAUTOLOG command for the RSCSAUTH user ID. You do not need to make any changes unless you have changed the name of the RSCS dynamic authorization virtual machine.

Configuring RSCS Dynamic Authorization

The name of the RSCS dynamic authorization configuration file is RSCSAUTH CONFIG. The following configuration statements are required. If either statement is omitted or is not valid, the server will not start.

- AUTHUSER identifies the user ID that is fully authorized for the RSCSAUTH server.
- FILEMODE identifies the location of the RSCSAUTH authorization tables and log files.

These statements are included in the sample configuration file shipped with RSCS. See [Appendix C, “Sample RSCS Dynamic Authorization Configuration File,”](#) on page 311. You might need to modify the sample file for your installation.

For descriptions of the RSCSAUTH configuration statements, see [Chapter 16, “RSCS Dynamic Authorization Configuration File Statements,”](#) on page 269.

Authorizing Locally Defined RSCS Commands

You can use RSCS dynamic authorization to manage the user authorizations for RSCS commands created by your installation. Create a file called RSCSCMDS CONFIG on the RSCSAUTH 191 disk and list each local command on a separate line in the file, using the following format for each entry:

```
command_name minimum_abbreviation
```

You can then use the COMMANDGROUP command to add these commands to command groups managed by RSCS dynamic authorization. For information about the COMMANDGROUP command, see [z/VM: RSCS Networking Operation and Use](#).

Chapter 14. Setting Up the RSCS Data Interchange Manager

This section contains topics that explain what you need to do to set up the RSCS Data Interchange Manager (RSCS Interchange) on your system. RSCS Interchange handles communications between NJE networks and SMTP networks.

Planning Checklist for RSCS Interchange

- **Ensure** that you have a copy of the RSCS program directory.
- **Gather** information required for the RSCS Interchange configuration file:
 - SMTP network domain name
 - Name of the link from RSCS to RSCS Interchange
- **Identify** whether exit routines will be used.
 - If a sample exit routine is available, does it meet your needs, or will additional function be required for your site?
 - Who will write any new exit routines or modify existing samples?
- **Prepare** for RSCS Interchange user authorization (privileged users) and setup:
 - Who will be authorized to issue RSCS Interchange commands?
 - Will general users be allowed to modify the nickname table?

RSCS Interchange Sample Files

The RSCS Interchange sample files provided with RSCS were copied to their usable form (file name and file type) on the indicated disks as part of RSCS installation. For more information, see the RSCS program directory.

The RSCS Interchange sample files are:

- On the RSCS 403 disk:
 - XCHANGE CONFIG**
RSCS Interchange configuration file
 - PROFILE EXEC**
RSCS Interchange server profile.
 - ACHAFAIL TEMPLATE**
RSCS Interchange failure template
- On the RSCS 401 disk:
 - ACHARSCS CONFIG**
Updates to the RSCS configuration file

Planning for RSCS Interchange Exits

If you choose to write exit routines for RSCS Interchange, REXX is the recommended programming language. You can also use other programming languages, as long as a MODULE file is created.

RSCS Interchange Server Virtual Machine

The z/VM user directory contains a sample definition for the RSCS Interchange server virtual machine. The user ID is XCHANGE. You might need to modify this sample definition for your installation. [Figure 49](#)

on page 120 shows the sample XCHANGE definition for a z/VM system that is not a member of a single system image (SSI) cluster. For simplicity, commented-out statements included in the sample definition are not shown here.

```
IDENTITY1 XCHANGE2 password 32M 32M BG3
BUILD ON *4 USING SUBCONFIG XCHNGE-1
ACCOUNT 1 RSCS
OPTION ACCT5
MACHINE ESA
IUCV ANY
IPL CMS PARM AUTO6
CONSOLE 01F 3215 T OPERATOR7
SPOOL 00C 2540 READER A
SPOOL 00D 2540 PUNCH A
SPOOL 00E 1403 A

SUBCONFIG XCHNGE-1
LINK MAINT 190 190 RR
LINK MAINT 193 193 RR
LINK MAINT 19E 19E RR
LINK RSCS 403 403 RR8
MDISK 191 3390 2085 010 MH1W01 MR READ WRITE MULTIPLE
MDISK 5BF 3390 2095 003 MH1W01 MR READ WRITE MULTIPLE
```

Figure 49. Sample RSCS Interchange Server (XCHANGE) Virtual Machine Definition

Note:

1. The IDENTITY statement identifies this as a multiconfiguration virtual machine definition. When included in the directory for an SSI cluster, this type of definition can be configured to enable the user ID to be logged on concurrently to multiple members of the cluster. The version of the definition shown here is for a non-SSI system, so the definition includes only one active BUILD statement and its associated SUBCONFIG entry and is equivalent to a single-configuration virtual machine definition (traditional USER definition).
2. If you change the name of this virtual machine, you will need to update any other files that refer to the XCHANGE user ID, such as the PARM statement in the ACHARSCS CONFIG file.



Attention: If RSCS has been installed to an SFS directory, IBM recommends that you do not change the name of this virtual machine. In addition to updating the ACHARSCS CONFIG file, other system changes are required. The name of the base SFS directory must be changed, and therefore links to that directory in the PROFILE EXEC files for RSCSDNS and RSCSAUTH must be updated. A PPF override is also required to ensure that any future service updates can be applied.

3. Specifies the CP privilege classes (based on the default classes provided with z/VM) for which this virtual machine is authorized:

G

Allows XCHANGE to use general user CP commands.

B

Allows XCHANGE to use the privileged CP MSGNOH command to suppress CP message headers.

Note: This is the *only* reason class B is specified here. Your installation might consider redefining privilege classes, which is described in *z/VM: CP Planning and Administration*. You could define one class that includes the CP MSGNOH command only. Then, that class would be specified here instead of B. You could also specify other classes if you wish to use additional privileged CP commands at the XCHANGE console. However, **do not** specify class D.

4. The BUILD statement identifies the SUBCONFIG entry for this system, which contains system-specific statements. An asterisk (*) in the system name field indicates this definition is included in an SSI-ready (not SSI-enabled) directory, and the system name defaults to the IPLed system. In an SSI-enabled directory, this definition will contain a BUILD statement and its associated SUBCONFIG entry for each member of the cluster.

5. The ACCT option allows XCHANGE to issue DIAGNOSE code X'4C' to create accounting records. This is optional. If enabled, RSCS Interchange can produce accounting records and pass them to CP to include in the accounting log. You must write RSCS Interchange accounting exits to gather the accounting information and issue DIAGNOSE code X'4C'. For more information, see *z/VM: RSCS Networking Exit Customization*. You must also identify the exit routines on the RSCS Interchange EXIT configuration statement. See “EXIT Statement” on page 283.
6. Allows XCHANGE to be autologged and to run its PROFILE EXEC.
7. Specifies the secondary user ID to control operations when XCHANGE is disconnected; this is optional.

Note: If RSCS Interchange is very active, it will send many messages to this user ID.
8. Links to the RSCS production disk containing the RSCS Interchange system code.

Preparing RSCS for RSCS Interchange

You need to prepare RSCS to communicate with the RSCS Interchange server. RSCS Interchange supplies a file called ACHARSCS CONFIG that contains a sample addition to the RSCS configuration file to identify the supplied exits and the statements required to set up the link from RSCS to RSCS Interchange. The RSCS configuration statements used are:

- EXIT
- IMBED
- LINKTYPE
- LINKDEFINE
- PARM

The following topics describe what these statements do. You can modify the statements, as appropriate.

RSCS EXIT Statements for RSCS Interchange

The EXIT statements specify which RSCS exit points will have corresponding exit routine modules loaded during RSCS initialization. If you use the ON operand, RSCS loads and enables the exit routine during initialization. If you use the OFF operand, RSCS loads the exit routine but does not enable the exit point. This allows you to enable the exit point at a later time using the EXIT command, which is described in *z/VM: RSCS Networking Operation and Use*.

Before initializing RSCS, you must:

1. Correctly link-edit all necessary exit routine modules into a CMS LOADLIB.
2. Enter the (GCS) GLOBAL command to allow GCS access to that LOADLIB.

Entry point names that do not correspond to module names must be identified to the linkage editor using the linkage editor ALIAS statement or using the GCS IDENTIFY macro.

The RSCSEXIT LOADLIB file, which is a sample file shipped with RSCS, contains RSCS Interchange exit routines that use Exits 0, 21, and 24.

EXIT Statements (in ACHARSCS CONFIG)

```
EXIT 0  ON FIRST ACHAEX0
EXIT 21 ON FIRST ACHAEX21
EXIT 24 ON FIRST ACHAEX24
```

RSCS IMBED Statement for RSCS Interchange

The IMBED statement lets you include another file into the RSCS configuration file. RSCS processes the records in the imbedded file at the point where you placed the IMBED statement as if they were a part of the main (or master) configuration file. The imbedded files must follow the same rules that apply to the master configuration file.

RSCS Interchange

You need to modify (XEDIT) the RSCS configuration file to include the statements contained in the ACHARSCS CONFIG file. You can imbed the ACHARSCS CONFIG file into the RSCS configuration file by placing an IMBED statement at the bottom of the RSCS configuration file, or you can enter the XEDIT GETFILE command to pull in the ACHARSCS CONFIG file.

IMBED Statement (in RSCS CONFIG)

```
IMBED ACHARSCS CONFIG
```

RSCS LINKTYPE Statement for RSCS Interchange

The LINKTYPE statement lets you dynamically define additional link drivers to those provided with the base RSCS system. With the LINKTYPE statement, you can specify a new link driver type, the entry point to the link driver code, and some basic characteristics of the link, so that RSCS knows how to interact with the other node. For RSCS Interchange, a new link type called XCHANGE is defined.

LINKTYPE Statement (in ACHARSCS CONFIG)

```
LINKTYPE XCHANGE ACHAGATE NOLINE
```

RSCS LINKDEFINE Statement for RSCS Interchange

The LINKDEFINE statement defines the default attributes of one RSCS link in your network. The link can be to a workstation, printer, another z/VM system, or a non-z/VM system.

The LINKDEFINE statement for RSCS Interchange defines the link where mail traffic will flow to the RSCS Interchange server.

LINKDEFINE Statement (in ACHARSCS CONFIG)

```
LINKDEFINE linkid TYPE XCHANGE
```

Note: This link is **never** started. It identifies to RSCS the path that files will take for RSCS Interchange. However, the exit routines supplied with RSCS Interchange will handle the actual file routing to the RSCS Interchange server.

RSCS PARM Statement for RSCS Interchange

The PARM statement must follow the corresponding LINKDEFINE statement in the configuration file. The PARM statement defines specific details for communicating with the associated nodes.

For RSCS Interchange, the only parameter accepted is the server user ID being used for the RSCS Interchange link.

PARM Statement (in ACHARSCS CONFIG)

```
PARM linkid SERVERID=XCHANGE
```

Modifying the NOTHERE TEMPLATE

Because RSCS Interchange introduces SMTP users to RSCS terminology, you might want to modify the NOTHERE TEMPLATE file (located on the RSCS 191 disk) to add a note that addresses those SMTP users when they send a note to a user ID that does not exist on this node. For example, you can add the following to your NOTHERE TEMPLATE:

If you are an SMTP user, the mail you sent cannot be delivered; you can take no further action on this file. Correct the destination of your mail before attempting to resend this mail again.

Preparing RACF for RSCS Interchange

If you have the RACF Security Server for z/VM installed on your system, you must issue two RACF PERMIT commands. These commands allow RSCS and RSCS Interchange to define an alternate user ID as the owner of a spool file. The commands also allow RSCS and RSCS Interchange to issue DIAGNOSE code X'D4'.

The commands you need to enter are:

```
PERMIT userid CLASS(VMBATCH) ID(userid) ACC(CONTROL)
PERMIT RSCS CLASS(VMBATCH) ID(userid) ACC(CONTROL)
```

Note: The *userid* in these commands is the user ID of the RSCS Interchange virtual machine. This is XCHANGE, unless you have defined a different name.

Preparing SMTP for RSCS Interchange

The system administrator for the SMTP virtual machine must update the SMTP configuration file and SYSTEM NETID to work with RSCS Interchange. The domain names in the RSCS Interchange configuration file and the SMTP configuration file must match. In addition, the domain name server supporting SMTP must have this domain name identified to it. For more information about SMTP preparation, see [“Defining Connections to SMTP and RSCS”](#) on page 123.

Configuring RSCS Interchange

The RSCS Interchange configuration file, XCHANGE CONFIG, defines the connections to RSCS and SMTP, privileged users, and other information. The following configuration statements are required:

- DOMAIN specifies the domain name address of the RSCS Interchange server in the TCP/IP network.
- RSCSLINK identifies the link between the RSCS Interchange server and the RSCS server.

For more information, see [“Defining Connections to SMTP and RSCS”](#) on page 123.

There are additional statements you might want to include in the configuration file, such as:

- AUTHORIZE identifies privileged users.
- EXIT identifies user-installed exit routines.

For more information, see Chapter 17, “RSCS Data Interchange Manager Configuration File Statements,” on page 279 and Appendix D, “Sample RSCS Data Interchange Manager Configuration File,” on page 313.

Defining Connections to SMTP and RSCS

You can establish communications with SMTP and RSCS through the transfer of spool files to these machines, and by receiving files from these machines through spooling. To accomplish this:

1. Identify the SMTP and RSCS servers to RSCS Interchange through the SMTPNAME and RSCSNAME configuration statements.
2. Identify the SMTP domain to RSCS Interchange through the DOMAIN configuration statement.
3. Identify the RSCS link for SMTP (RSCS Interchange) to RSCS Interchange through the RSCSLINK configuration statement.
4. Modify the SYSTEM NETID file on the SMTP virtual machine to send all mail traffic for the z/VM network to the RSCS Interchange server.
5. Add a mail exchange (MX) record to the TCP/IP domain name server.

See Table 14 on page 124 for additional information about these required changes. The changes must be completed before you initialize RSCS Interchange.

Table 14. File Changes Required for RSCS Interchange

Area	File	Required Change
RSCS Interchange server	configuration file	DOMAIN <i>domain_name</i> RSCSLINK <i>linkid</i> RSCSNAME <i>rscsid</i> SMTPNAME <i>smtpid</i>
SMTP server	configuration file	RSCSDOMAIN <i>domain_name</i> GATEWAY RSCSFORMAT NETDATA
	SYSTEM NETID A	cpuid <i>linkid</i> xchange
TCP/IP domain name server	DNS data file	<i>localid</i> A <i>ip_address</i> *. <i>domain_name</i> MX 0 <i>localid</i>
RSCS server	configuration file	LOCAL <i>localid</i> LINKDEFINE <i>linkid</i> TYPE XCHANGE PARM <i>linkid</i> SERVERID=XCHANGE

Notes:

1. The *domain_name* identifies the SMTP domain in which the RSCS Interchange server, SMTP, and the TCP/IP domain name server communicate; it must match. However, this name must not match the name in the TCP/IP data file. If it does, TCP/IP will spool the file directly to the user ID's reader. See "Selecting Domain Names" on page 124 for more information.
2. The *linkid* identifies the link defined in RSCS for communicating to RSCS Interchange. This table shows an example of z/VM implementation. The entries must be included in the domain name server to enable the gateway to work correctly.
3. If your RSCS user ID is not RSCS, you must include an RSCSNAME statement in the RSCS Interchange configuration file to reflect this user ID.
4. If your SMTP user ID is not SMTP, you must include an SMTPNAME statement in the RSCS Interchange configuration file to reflect this user ID.
5. Changes to the SYSTEM NETID on SMTP must be filed to the SMTP A-disk so as not to interfere with the system-wide SYSTEM NETID file.
6. The *localid* is the local node on which RSCS, SMTP, and RSCS Interchange all reside.
7. The LINKDEFINE and PARM statements for the RSCS configuration file are included in the sample ACHARSCS CONFIG file supplied with RSCS Interchange. See "Preparing RSCS for RSCS Interchange" on page 121.

Selecting Domain Names

The *domain_name* you specify on the DOMAIN statement for RSCS Interchange must be a *sub_domain_name* of the *domain_name* that is specified in the TCP/IP domain name server data file. When TCP/IP receives a file, this enables TCP/IP to distinguish its own host/domain name from the RSCS Interchange domain. TCP/IP can then determine when to send the file to RSCS Interchange.

The *.*domain_name* MX 0 *localid* entry specified in the TCP/IP domain name server data file is called a "wildcard entry". The asterisk (*) is the user ID and node ID to which the file is sent; the RSCS Interchange *domain_name* is appended to this value. When SMTP receives a file, it can then send it to the RSCS Interchange server. The RSCS Interchange server determines the correct file destination by removing the data to the right of the period (.). The RSCS Interchange server then sends the file to the correct reader on its system or forwards the file to the correct system.

The *domain_name* server does not have to be on the same system as the RSCS Interchange server. However, the nickname tables on your local system must be updated to include the user IDs and node IDs of all users that can send or receive files.

Part 3. Reference

Chapter 15. RSCS Configuration File Statements

This section contains topics that describe the configuration file statements you can use to define your network to RSCS.

Rules for Coding the RSCS Configuration File

1. All RSCS configuration file statements are optional.
2. Most information specified in configuration file statements must be coded in uppercase. The only exception is printer queue names that may be specified on some PARM statements. Comments may also be specified in lowercase or mixed case.
3. The configuration file cannot contain sequence numbers.
4. If specified, the LOCAL statement must be the first network definition statement. Comment, IMBED, and TOLERANCE statements are not considered network definition statements; they can be placed before the LOCAL statement.
5. Most configuration file statements are positional. This means that the placement of the variable information is important.
6. The configuration file can be in either fixed format with a width of up to 256 characters or variable format.
7. You can break a configuration file statement into more than one line by using the continuation character (#). When joined, this statement cannot exceed the 256 character limit for statement length.
8. You can break your configuration file into more than one file by using the IMBED statement.

For information about how to read the syntax diagrams used in this chapter, see [“Syntax, Message, and Response Conventions”](#) on page xvii.

Summary of RSCS Configuration File Statements

Table 15 on page 129 lists the RSCS configuration file statements and their functions.

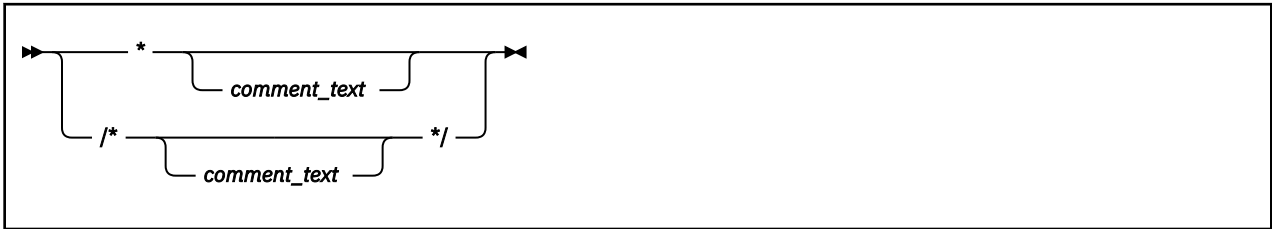
Table 15. Summary of Configuration File Statements

Statement	Description
comment	Specifies a comment.
AUTH	Identifies an authorized alternate operator for RSCS or for a specified link.
CHANNELS	Reserves channels for exclusive use by RSCS for its virtual device pool.
DEST	Specifies local PSF printer destinations.
DUMP	Tells RSCS what type of dump to take when an RSCS problem (abend) occurs and the user (virtual machine) to which RSCS should send the dump.
EXIT	Defines and loads RSCS exit routine modules during RSCS initialization.
FCB	Defines an FCB image which RSCS will simulate on SNA3270P-type, TN3270E-type, and 3270P-type links.
FORM	Specifies the characteristics of a form name.
HIDECHARACTER	Specifies the default hide character. Use the hide character to mark sections of link parameter definitions in the RSCS configuration file or the RSCS events file that you do not want displayed in response to a QUERY command.

Table 15. Summary of Configuration File Statements (continued)

Statement	Description
IMBED	Includes and processes a file as if it were a part of the main configuration file.
ITRACE	Defines the initial state of the RSCS internal trace table, defines the events RSCS will trace, and optionally starts tracing when RSCS initializes.
LANGUAGE	Specifies the name of a load module containing a compiled RSCS message file.
LINKDEFINE	Defines the default attributes of an RSCS link. Note: The LINKDEFINE statement replaces the LINK statement.
LINKTYPE	Defines the initial attributes of an installation-defined link type.
LOCAL	Contains the node ID of the local system and (optionally) the application ID by which VTAM knows RSCS.
MSGNOH	Tells RSCS to use the CP MSGNOH command when issuing messages to local users.
OPFORM	Contains the operator form name.
OPTION	Lets installations define various RSCS system-oriented and link-oriented parameters.
PARM	Defines operational characteristics that a link driver is to use for a specified link.
PORT	Specifies a virtual device number for a BSC line to be used by an auto-dial or auto-answer link, or for dynamic allocation of a BSC link.
RECOVERY	Specifies which CP commands should be issued when an RSCS supervisor task abends causing RSCS to become disabled.
REROUTE	Modifies the original destination of commands, files, and messages for specific systems and user IDs.
RETRY	Tells RSCS when to restart deactivated auto-dial and SNA links.
ROUTE	Describes the path to a nonadjacent node.
SAFCLASS	Specifies the output class to be used when RSCS creates store-and-forward files.
SETMSG	Starts the sharing of RSCS messages based on message number.
SHADOWS	Specifies how many tag shadow elements RSCS should reserve.
SLOWDOWN	Tells RSCS when to enable or disable slowdown mode.
TOLERANCE	Marks areas of the configuration file as tolerant or intolerant of errors.
TRACEDEST	Tells RSCS where to send trace files.
UPARM	Specifies additional, installation-defined parameters.

Comment



Purpose

Comments provide a way to annotate the RSCS configuration file. They are ignored by RSCS. The sample configuration file shows examples of how to use comments. See [“RSCSSMPL CONFIG”](#) on page 298.

Specification

Include as many comments as needed; they are optional. Place them anywhere in the configuration file.

Operands

is the comment identifier. If secure printing finds an asterisk in column one, it treats the entire line as a comment.

/* */

are REXX comment delimiters. You can place the comment anywhere in the configuration file. Whenever RSCS finds an occurrence of the REXX start comment delimiter (/*) it treats everything on that line as a comment until it finds an occurrence of the REXX end comment delimiter (*/). However, you must start and stop the comment on the same line. If the REXX end comment delimiter is not on the same line as the start comment delimiter, RSCS flags the line as containing an error. For example:

Acceptable Comment

```
/*-----*/
/* Here is */
/* my comment */
/*-----*/
```

Unacceptable Comment

```
/*-----*
* Here is *
* my comment *
*-----*/
```

comment_text

is any number of characters of text up to the maximum record width of the file.

AUTH Statement

Purpose

The AUTH statement defines an alternate RSCS operator. This statement has two formats:

- Format 1 defines an alternate operator for a specific RSCS link (a link-authorized alternate operator).
- Format 2 defines an alternate operator for the RSCS system (a system-authorized alternate operator).

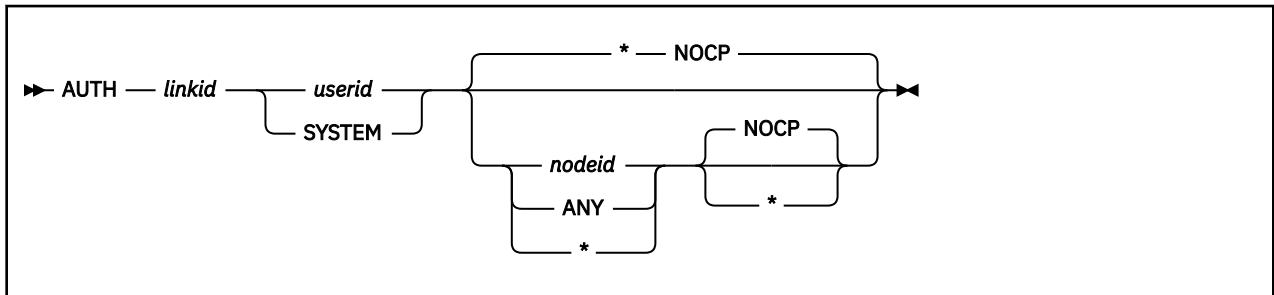
Note: Use format 2 to authorize RSCSAUTH.

The authorized operator can be a user on the local system or on another system.

Specification

Include as many statements as needed; they are optional. AUTH statements can be placed anywhere in the configuration file after the LOCAL statement (if one exists) and after the LINKDEFINE statement to which they refer.

Format 1



Operands 1

linkid

is the name of the link the specified user is authorized to operate. This value must be defined on a previous LINKDEFINE statement. Besides those RSCS commands available to general users, a link-authorized alternate operator is authorized to use the subset of operator commands that affect link operations. For information about this command subset, see [z/VM: RSCS Networking Operation and Use](#).

userid

SYSTEM

is the z/VM user ID of the person being authorized as a link-authorized alternate operator. The same user can be authorized to operate more than one link, but the same user cannot be authorized as both a link-authorized alternate operator and a system-authorized alternate operator. The user ID of the RSCS virtual machine cannot be specified here, or an error will result.

If you specify SYSTEM, RSCS authorizes a console operator with any desired RSCS authority level (CP, link, and so forth).

nodeid

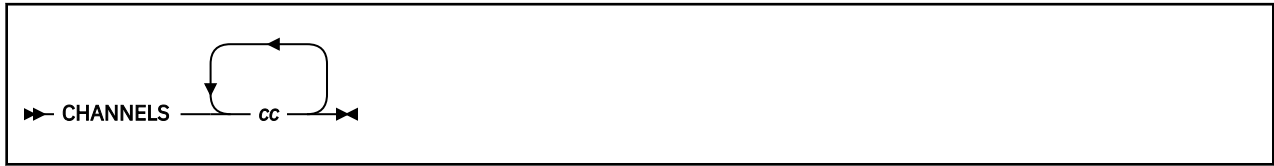
is the node ID of the specified alternate operator.

ANY

indicates userid is authorized from any node.

If you specify an asterisk (*) or omit this operand, RSCS uses its own local node ID.

CHANNELS Statement



Purpose

The CHANNELS statement reserves one or more virtual channels for use by RSCS as a virtual unit record device pool. RSCS allocates devices as needed from this pool for each spool file it sends or receives. Each reserved channel provides 256 virtual device addresses that can be used (X'cc00' - X'ccFF').

Specification

One statement is allowed. It is optional and can be placed anywhere in the configuration file after the LOCAL statement (if one exists) and before all LINKDEFINE statements.

Operands

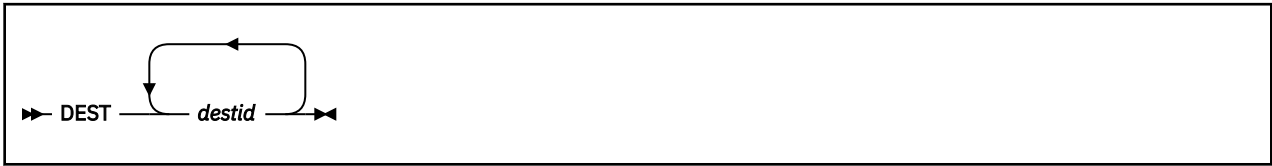
cc

is a hexadecimal number, 01 - 1F, that identifies a channel to be reserved for exclusive use by RSCS. (Addresses on this channel cannot be used for any other purpose.) Channel numbers do not have to be specified in any special order; however, RSCS allocates channels in descending sequence (1F - 01). No channel number may be specified more than once.

Usage Notes

1. If you omit the CHANNELS statement, RSCS will use channel X'0F' as the default for its virtual unit record device pool.
2. Not all of the devices reserved by the CHANNELS statement will necessarily be used. The virtual device block structure in CP limits the number of virtual devices a virtual machine can define.
3. Each virtual machine on a z/VM system can have a maximum of 24,576 virtual devices. Because of real storage constraints, the actual limit allowed by CP might be less.

DEST Statement



Purpose

The DEST statement specifies Print Services Facility/VM (PSF) printer destinations on the local system.

Specification

Include as many statements as needed; they are optional. DEST statements can be placed anywhere in the configuration file after the LOCAL statement (if one exists).

Operands

destid

is the name of a destination on the local system that is a page-image printer under the control of PSF. You can specify multiple names.

Usage Notes

1. To dynamically add or delete destinations after RSCS has been initialized, use the DEST command. For information about the DEST command, see [z/VM: RSCS Networking Operation and Use](#).

DUMP Statement



Purpose

The DUMP statement tells RSCS what type of dump to take when an RSCS problem (abend) occurs and to which user (virtual machine) RSCS should send the dump.

Specification

One statement is allowed. It is optional and can be placed anywhere in the configuration file after the LOCAL statement (if one exists).

Operands

CP

specifies that the CP DUMP command will be used to produce all dumps.

VM

specifies that the CP VMDUMP command will be used to produce all dumps. This is the type required by the Dump Viewing Facility and RSCS service personnel.

*

userid

is the z/VM user ID to which RSCS should send all RSCS dumps.

If you specify an asterisk (*) or omit this operand, RSCS sends:

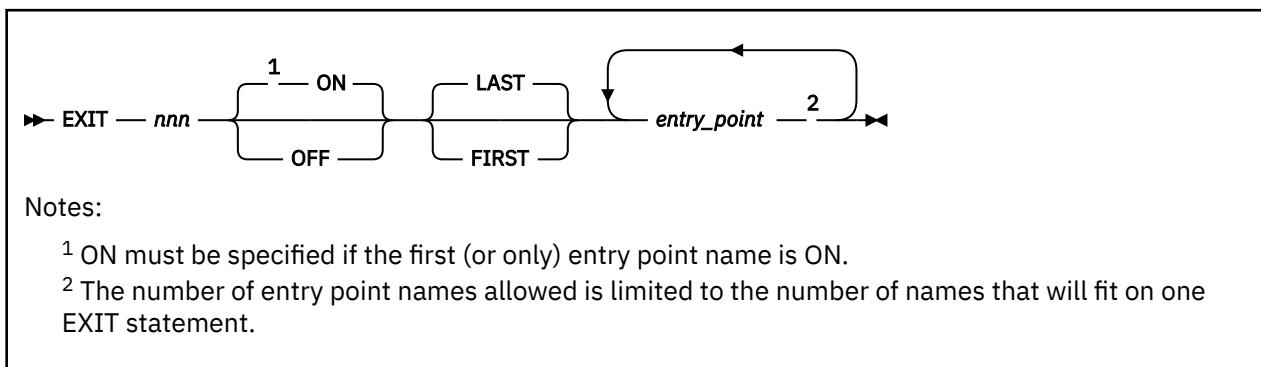
- CP dumps to the RSCS virtual printer
- VM dumps to the user ID specified in the DUMP operand of the SYSTEM_USERIDS configuration statement.

Usage Notes

1. If you omit the DUMP statement, a VM-type dump will be produced.
2. Under certain circumstances, a GCS dump may be taken instead of an RSCS dump. For example, if something goes wrong during RSCS initialization and RSCS has not defined its own ESTAE exit, an RSCS dump will not be taken. Instead, a GCS dump will be sent to the GCS Common Dump Receiver specified in the definition of the GCS group.
3. The RSCS program directory contains information to help you report a problem to IBM. It also describes the types of traces, logs, and dumps you should have.

EXIT Statement

PI



Purpose

The EXIT statement defines and loads RSCS exit routines during RSCS initialization. The exit points can be enabled at initialization or allowed to remain disabled.

Specification

Include as many statements as needed; they are optional. EXIT statements can be placed anywhere in the configuration file after the LOCAL statement (if one exists). You can have more than one EXIT statement per exit point.

Operands

nnn

is the exit point to be loaded. This value is a decimal number, 0 - 255. Leading zeros are not significant.

ON OFF

specifies the status of the exit point after loading.

ON

Enabled. This is the default. ON must be specified if the first (or only) entry point name is ON.

OFF

Disabled. The exit point can be enabled later using the EXIT command. For more information, see [z/VM: RSCS Networking Operation and Use](#).

LAST FIRST

specifies where RSCS should place the specified entry points. RSCS builds the current list of entry point names during initialization and then calls them in the order specified.

LAST

At the end of the current list of entry point names associated with this exit point. This is the default.

FIRST

At the beginning of the current list of entry point names associated with this exit point.

entry_point

is the name of an entry point in the load module that contains the exit routines for handling the exit point. Each entry point name corresponds to an exit routine that will be called from within the RSCS code when the associated exit point is processed. As many entry point names as will fit on one EXIT

EXIT

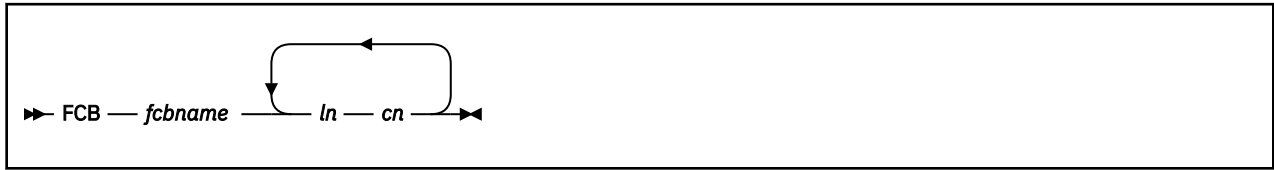
statement are allowed. The entry points will be called in order, based on the current list. RSCS builds the list during initialization.

Usage Notes

1. There are 48 IBM-defined exit points. For information about these exit points, see [z/VM: RSCS Networking Exit Customization](#).
2. RSCS will not load entry points that are marked as nonexecutable, cannot be located, or are overlays. If one or more entry points specified for a given exit point cannot be loaded, RSCS discards the entire EXIT statement, including any valid entry points.

PI end

FCB Statement



Purpose

The FCB statement defines a forms control buffer (FCB) image which RSCS simulates on SNA3270P-type, TN3270E-type, and 3270P-type links. RSCS supports all twelve channel skips.

Specification

Include as many statements as needed; they are optional. FCB statements can be placed anywhere in the configuration file after the LOCAL statement (if one exists) and before any LINKDEFINE statements that refer to them.

Operands

fcbname

is the 1- to 4-character name of the FCB image.

ln cn

is the line number (*ln*) and channel number (*cn*) of a channel of the FCB image. The line number and channel number must be specified as a pair. You can specify more than one pair if the pairs are separated by at least one blank character and the statement fits on one line of the configuration file.

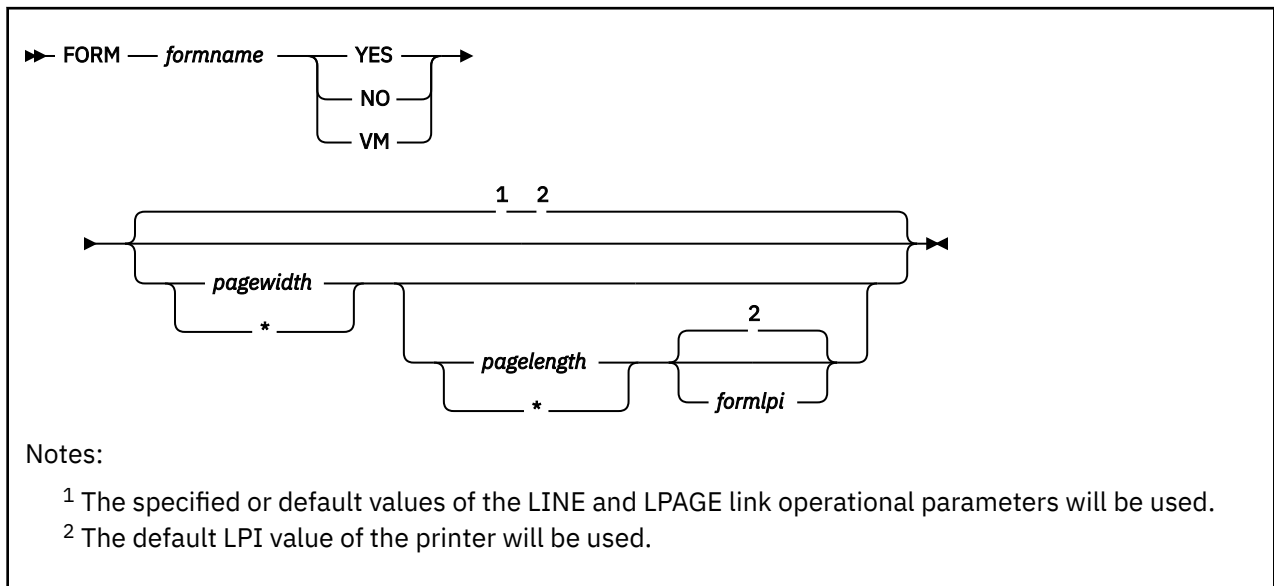
A line number is a decimal number, 1 - 256, and cannot be duplicated.

A channel number is a decimal number, 1 - 12. You can specify channel numbers more than once and in any order.

Usage Notes

1. The FCB statement is valid only for SNA3270P-type, 3270P-type, and TN3270E-type links.
2. If you specify a line/channel pair with a line number greater than the value specified on the LPAGE operational parameter of the SNA3270P-type, 3270P-type, or TN3270E-type link, RSCS uses the LPAGE value.
3. RSCS always assumes that channel 1 is a skip to the top of the page. If you redefine channel 1 in an FCB statement, the results are unpredictable.

FORM Statement



Purpose

The FORM statement specifies the characteristics of a form name.

Specification

Include as many statements as needed; they are optional. FORM statements can be placed anywhere in the configuration file after the LOCAL statement (if one exists).

Operands

formname

is the 1- to 8-character operator form name.

YES

NO

VM

specifies the type of separator page to be printed when a form is mounted on a printer or workstation link.

YES

Short RSCS-style separator page.

NO

No header or trailer page.

VM

Full VM-style separator page.

For descriptions of the separator page formats, see [z/VM: RSCS Networking Operation and Use](#).

pagewidth

is the width of the form in the number of print positions. This value can be 1 - 220. This value is used in place of the LLINE parameter on printer links. If you specify an asterisk (*) or omit this operand, the specified or default value of the LLINE link parameter is used for this form on the printer link.

The FORM statement is a positional statement. If you do not wish to specify *pagewidth* but you do wish to specify *pagelength* or *formlpi*, you must specify an asterisk (*) as a place holder.

pagelength

is the length of the form in the number of print lines. This value can be 0 - 99. This value is used in place of the LPAGE parameter on printer links. If you specify an asterisk (*) or omit this operand, the specified or default value of the LPAGE link parameter is used for this form on the printer link.

The FORM statement is a positional statement. If you do not wish to specify *pagelength* but you do wish to specify *formlpi*, you must specify an asterisk (*) as a place holder.

formlpi

is the number of lines per inch that the form will print. This value can be 3, 4, 6, or 8. This value is used in place of the LPI parameter on printer links. If you omit this operand, the number of lines per inch is assigned by the printer.

Note: This parameter affects SNA LU_T1 printers and workstations when SETPAGE=YES.

Usage Notes

1. When a spool file is opened by a printer or workstation link driver, the form name of the spool file is compared to the names in the form table. (The form table is created from the FORM statements in the configuration file when RSCS is initialized.) If a match is found, the characteristics of the form (that is, separator type and form size) are used by the link to process the file (generate the separator page, find the top of the page, truncate output lines, and so forth). If a match is not found, the link uses the default parameters, specified at the time the link driver is started, to process the file.
2. If you specify a form name when starting a link, be aware that the form name characteristics always override any SEP, LLINE, and LPAGE specifications and will override LPI specifications when SETPAGE=YES.

HIDECHARACTER Statement

```
►► HIDEcharacter — c ◄◄
```

Purpose

The HIDECHARACTER statement changes the default hide character. Use the hide character to mark sections of link parameter definitions in the RSCS configuration file or the RSCS events file that you do not want displayed in response to an RSCS QUERY command.

Specification

One statement is allowed. It is optional and can be placed anywhere in the configuration file after the LOCAL statement (if one exists).

Operands

c

is the character to be used as the hide character. Do not use a letter, a number, a single quote ('), an equal sign (=), or the continuation character (#).

Usage Notes

1. If you omit the HIDECHARACTER statement, the default hide character is a backward slash (\).
2. The hide character can be used in the following RSCS statements and commands:
 - PARM statement
 - UPARM statement
 - DEFINE command (PARM and UPARM operands)
 - SCHEDULE command
 - START command (PARM and OPARM operands)

See [z/VM: RSCS Networking Operation and Use](#) for more information about the DEFINE, SCHEDULE, and START commands.
3. Use the hide character anywhere after the link ID on a PARM or UPARM statement to mark a section of the statement as nondisplayable. If you try to surround PARM *linkid* with the hide character, RSCS flags the PARM statement as an error.
4. When you use the hide character on a PARM or UPARM statement and a user issues a QUERY command to display that information, RSCS replaces the information with a string of X characters. For example:

If you specify:

```
parm newyork pass=abcdefgh
parm newyork pass=\abcdefgh\
```

The QUERY response will be:

```
PASS=ABCDEFGH
PASS=XXXXXXXX
```

IMBED Statement



Purpose

The IMBED statement lets you include another file into the RSCS configuration file.

Specification

Include as many statements as needed; they are optional. IMBED statements can be placed anywhere in the configuration file.

Operands

ddname

is the data definition name of the imbedded file. If you specify a *ddname*, you must identify the file to RSCS by specifying the *ddname* on a GCS FILEDEF command during RSCS initialization.

fn ft

is the CMS file name and file type of the imbedded file. If you specify a file name and file type, you do not have to identify the file on a GCS FILEDEF command.

Usage Notes

1. The rules that apply to the RSCS master configuration file also apply to all imbedded files. That is, the file must contain fixed or variable length records of no more than 255 characters. For a complete list of rules, see [“Rules for Coding the RSCS Configuration File”](#) on page 129.
2. If you specify a CMS file name and file type, RSCS does not require you to identify the file on a GCS FILEDEF command. For example, if your master file name was RSCS CONFIG and you had two imbedded files (IMBEDONE CONFIG and IMBEDTWO CONFIG), you would specify the following GCS FILEDEF command before initializing RSCS:

```
filedef config disk rscs config *
```

You would include the following statements in RSCS CONFIG:

```

:
imbed imbedone config
:
imbed imbedtwo config
:

```

However, if you specify a *ddname*, you must also include a GCS FILEDEF command in the PROFILE GCS exec for the RSCS virtual machine (or manually specify the FILEDEF command). Using the same example, you would specify the following GCS FILEDEF commands before initializing RSCS:

```

filedef config disk rscs config *
filedef imbed1 disk imbedone config *
filedef imbed2 disk imbedtwo config *

```

You would include the following statements in RSCS CONFIG:

```

:
imbed imbed1
:

```

```
imbed imbed2
:
```

- RSCS treats all of the records of the imbedded file as being in the master configuration file at the point where you place the IMBED statement. “RSCSSMPL CONFIG” on page 298 shows an example of a master configuration file with IMBED statements. Figure 50 on page 144 shows an example of a configuration file with imbedded files (left) and without imbedded files (right). RSCS considers both configuration files to be equal.

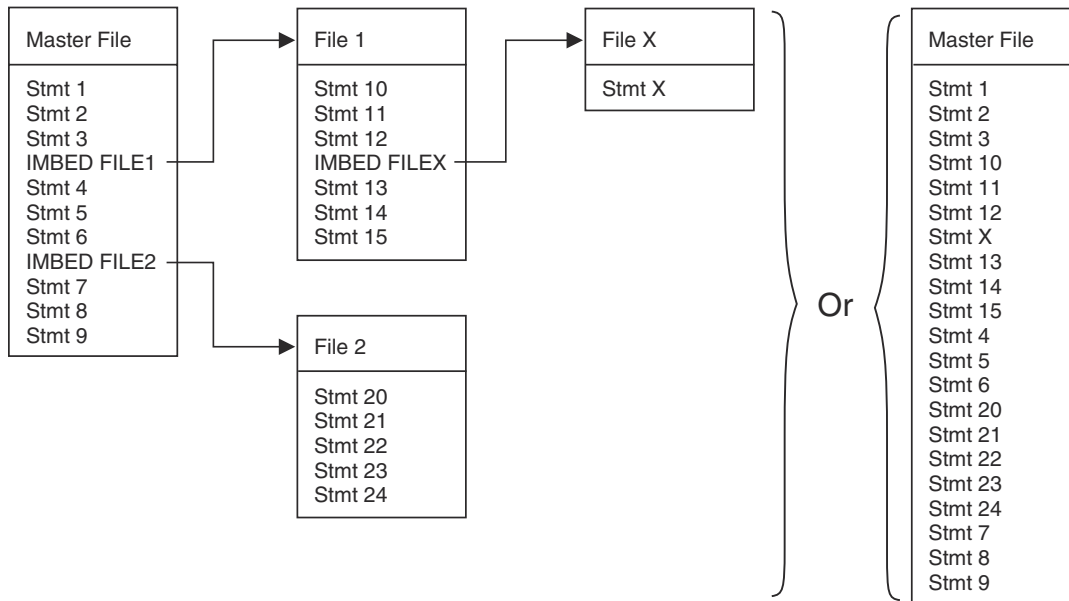
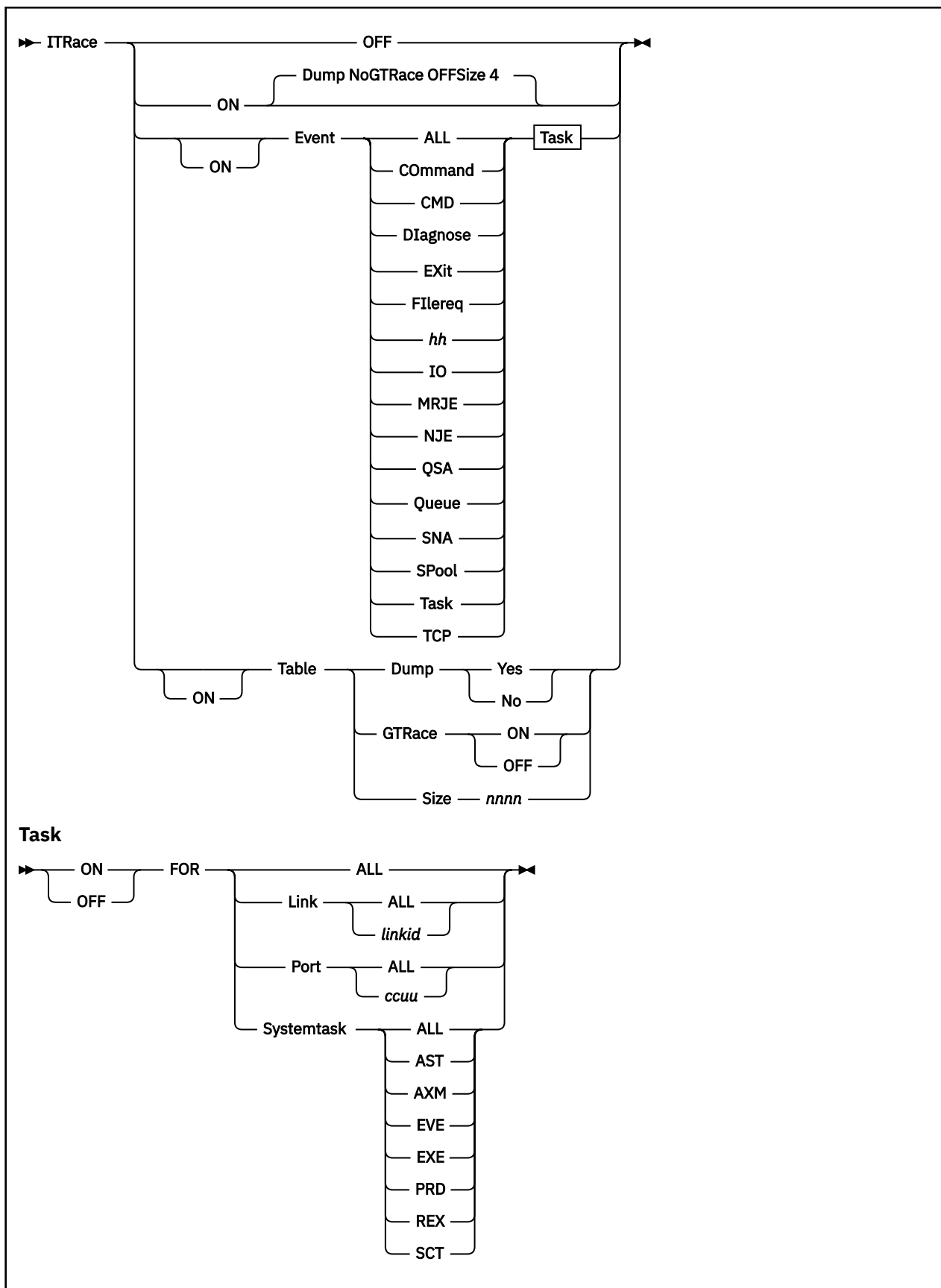


Figure 50. An RSCS Configuration File With and Without Imbedded Files

- You can nest up to 10 levels of IMBED files (the master configuration file plus nine levels of imbedded files).
- If RSCS fails to find the IMBED file during initialization, RSCS issues an error message and continues initialization processing. RSCS terminates initialization processing only when you place a TOLERANCE OFF statement somewhere before the IMBED statement. For more information, see the “TOLERANCE Statement” on page 266.
- You can also use the IMBED statement in the RSCS events file, and you may be able to use it in user exit files.

ITRACE Statement



Purpose

The ITRACE statement defines the initial state of the RSCS internal trace table, defines the events RSCS will trace, and optionally starts tracing when RSCS initializes.

Specification

Include as many statements as needed; they are optional. ITRACE statements can be placed anywhere in the configuration file after the LOCAL statement (if one exists).

Operands

OFF

ON

specifies the status of tracing when RSCS initializes.

Event

tells RSCS to trace a specific type of event in the internal trace table. You can specify only one event on each ITRACE statement (unless you specify ALL). If you need to specify more than one event, you must specify more than one ITRACE statement.

The valid event types are:

ALL

All events.

COmmand

CMD

All commands that RSCS runs (either operand can be used).

DIagnose

All CP Diagnose code functions that RSCS requests.

EXit

All calls to exit routines from RSCS.

FIlereq

All calls to open and close spool files and transmission algorithms.

hh

Record ID of a specific event. (Each record type has a unique 2-digit hexadecimal record ID.) For more information about the record types, see usage note [“3” on page 148](#).

IO

All I/O requests and interrupts that RSCS processes.

MRJE

Data buffer transmission before I/O initiation and after the completion of I/O operation. WRITE and READ data are collected, because both use the same buffer.

NJE

All NJE records that RSCS processes or builds.

QSA

All quick storage allocations and deallocations that RSCS performs.

Queue

Any elements that RSCS places on and removes from the internal command and message queues.

SNA

Each time a VTAM request is made or an exit routine is scheduled.

SPool

All records read from and written to the CP spool.

Task

Any tasks that RSCS starts or ends.

TCP

TCP SOCKBLOK, SOCKCBLK, and PRDBLOK before and after IUCV calls to the TCP/IP virtual machine.

**ON FOR
OFF FOR**

starts or stops tracing the specified event for the specified task. The valid task values are:

ALL

All tasks (links, dial-out ports, and system tasks) in the internal trace table.

Link ALL**Link *linkid***

Events associated with all links or with a specific link. You cannot specify a generic link ID (for example, LINK NY*). Only specific link IDs are allowed.

Port ALL**Port *ccuu***

Events for all auto-answer ports or for a specific auto-answer port.

Systemtask

System task in the internal trace table. You can specify only one task on each ITRACE statement (unless you specify ALL). If you need to specify more than one task, you must specify more than one ITRACE statement. The valid system task values are:

ALL

All system tasks.

AST

Auto-start task, which attaches auto-dial links and monitors ITO and retry intervals.

AXM

Spool manager task, which monitors the reader of the RSCS virtual machine and processes CP and RSCS spool-related commands.

EVE

Event scheduling task, which monitors and processes all scheduled events.

EXE

EXEC task, which processes all RSCS EXEC command requests.

REX

Communications task, which processes RSCS commands.

SCT

SNA control task, which maintains the RSCS ACB and contains most of the VTAM exit routines.

Table

sets global defaults for the RSCS internal trace table.

Dump Yes**Dump No**

specifies whether RSCS should dump the internal trace table (using the CP DUMP or VMDUMP command) to the virtual machine specified on the DUMP statement before the trace table wraps to the beginning. For more information, see the [“DUMP Statement”](#) on page 136.

GTRace ON**GTRace OFF**

specifies whether RSCS should place the specified event in the internal RSCS trace table and in the GCS trace table using a GCS ITRACE GTRACE command. To copy these events into a CPTRAP file, use the GCS ETRACE command. For more information about the GCS ETRACE and ITRACE commands, see [z/VM: Group Control System](#).

Size *nnnn*

specifies the size of the RSCS internal trace table. The number you specify is the number of 4096-byte pages of storage that RSCS should get from GCS for the internal trace table. This value can be any decimal number, 4 - 9999. The default is 4. If RSCS cannot get the amount of storage

you specify, it tries to get the default size and issues message DMT438W to let you know how much storage it obtained.

Usage Notes

1. RSCS starts tracing after initialization only when you specify the ON operand (either by itself or with the EVENT or TABLE operands). If no ITRACE statements specify ON, then RSCS will not start tracing after initialization. If one ITRACE statement specifies ON and the other ITRACE statements do not specify either ON or OFF, RSCS will start tracing after initialization. If one ITRACE statement specifies ON and one specifies OFF, RSCS uses the last statement specified.
2. RSCS always uses the latest ITRACE specification. If you specify conflicting options, RSCS uses the most recent. For example, if you specified:

```
itrace event exit on for link toronto
:
itrace event exit off for link toronto
```

RSCS would not trace calls to user exit routines for link TORONTO.

If you specify options that do not conflict, RSCS adds the latest specification to the list of items to trace. For example, if you specified:

```
itrace event cmd on for link all
itrace on event diagnose on for all
itrace table dump yes
```

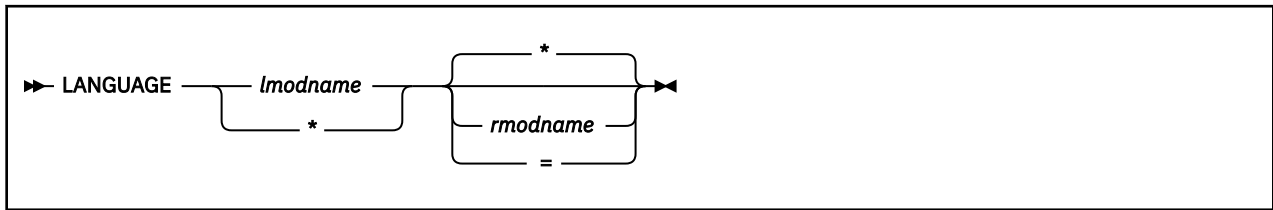
RSCS would:

- Trace all commands it processed for all links
 - Trace all CP Diagnose codes it requests for all tasks
 - Dump the internal trace table before wrapping to the beginning
 - Invoke tracing after initialization
3. RSCS traces events using trace records. These trace records are identified by a 2-digit hexadecimal number. The following table lists the trace records, their event type, and their meaning:

Trace Record	Event Type	What RSCS is tracing
00	Task	Start of a task
01	Task	End of a task
02	EXit	Call to a GCS exit routine
03	IO	I/O interrupt
04	FIlreq	Call to request a file
05	Queue	Call to add an element to a queue
06	Queue	Call to remove a queue element
07	—	Call to issue a message
08	QSA	Call to allocate or deallocate storage
09	COmmand, CMD	Command entry
0B	SNA	Initial call to VTAM
0C	SNA	Completion of VTAM call
0D	EXit	Call to an exit (EXITCALL)
0E	EXit	Exit point return
0F	SPOOL	Call to input spool file routine

Trace Record	Event Type	What RSCS is tracing
10	SPOOL	Call to output unit record device routine
11	IO	Call to I/O processing routine
12	NJE	NJE record sent and received
13	COMMAND, CMD	A scheduled event
14	SPOOL	Data buffers sent and received
15	DIAGNOSE	Diagnose code
16	SNA	VTAM exit routine
17	NJE	NJE job header
18	NJE	NJE data set header
19	NJE	NJE job trailer
1A	NJE	NJE request to open
1B	NJE	NJE permission granted
1C	NJE	NJE end of file
1D	NJE	NJE receiver on-line
1E	NJE	NJE reject
1F	NJE	NJE abend
20	NJE	NJE connection, RJE connection, or sign-on record
21	EXIT	Call to exit routine
22	QSA	Call to deallocate storage
23	NJE	NJE record received
24	TCP	IUCV calls to TCP/IP
25	TCP	Normal IUCV completion from TCP/IP
26	TCP	IUCV immediate error returns
27	MRJE	Send and receive buffer trace for MRJE
28 - EF	—	Reserved for IBM use
F0 - FF	—	Reserved for customer use

LANGUAGE Statement



Purpose

The LANGUAGE statement specifies the name of a load module that contains a compiled RSCS message file.

Specification

One statement is allowed. It is optional and can be placed anywhere in the configuration file after the LOCAL statement (if one exists).

Operands

lmodname

*

is the name or alias of a load module (contained in a CMS load library identified by a GCS GLOBAL LOADLIB command before RSCS is initialized) that contains messages to be issued to users at the local node. These users include CP and RSCS operators, printers, and workstations connected to RSCS. This value can be 1 - 8 characters.

If you specify an asterisk (*), RSCS will use the default language contained in the RSCS load module as the local node language; that is, no message module will be loaded.

*

rmodname

=

is the name or alias of a load module (contained in a CMS load library identified by a GCS GLOBAL LOADLIB command before RSCS is initialized) that contains messages to be issued to users at remote NJE nodes. This value can be 1 - 8 characters.

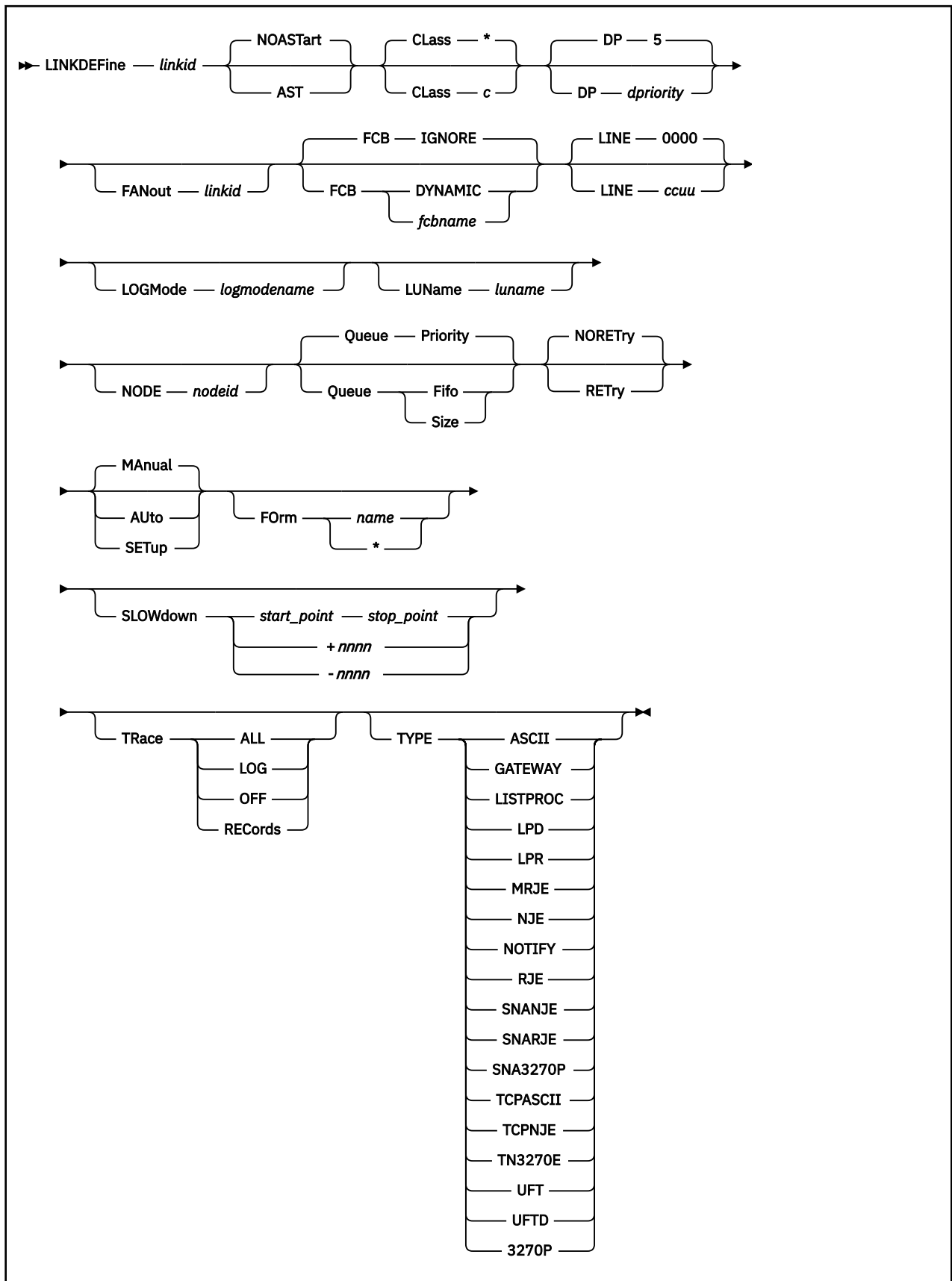
If you specify an asterisk (*) or omit this operand, RSCS will use the default language contained in the RSCS load module as the remote node language; that is, no message module will be loaded.

If you specify an equals sign (=), the language at the local node will be used for the remote node.

Usage Notes

1. When RSCS finds this statement, it attempts to load the message file and use it for all following messages. To be available to RSCS, the message file must be successfully compiled and link edited into a CMS loadlib available during RSCS initialization.
2. American English is the default language shipped with RSCS. However, a message load module for uppercase English is also included in the RSCS LOADLIB. To use uppercase English, you do not need to specify an additional load library on the GLOBAL LOADLIB statement in the PROFILE GCS file. Just change the *lmodname* or *rmodname* operand on the LANGUAGE statement to DMTUCENG.

LINKDEFINE Statement



Purpose

The LINKDEFINE statement defines the default attributes of a single RSCS link. These link attributes apply to the link when it is started, unless overridden by values specified on the DEFINE or START command.

LINK Statement

The LINKDEFINE statement replaces the LINK statement. LINKDEFINE is more powerful than LINK because it supports more features. Although LINK is still supported, you should not use it to define new links. If you are migrating from a previous version of RSCS, you should change any LINK statements to LINKDEFINE statements. The LINK statement is not documented.

Additional Link Attributes

The “OPTION Statement” on page 163 defines defaults for various system and link attributes. Depending on the type of link you are defining with the LINKDEFINE statement, you may be able to override OPTION defaults for this link by specifying values on the “PARM Statement” on page 168. The PARM statement defines additional attributes for each link type. If you do not define a PARM statement for this link, all of the PARM defaults for the link type will apply to this link, unless overridden by a DEFINE or START command. Some PARM attributes do not have defaults; if not defined on the PARM statement, they must be specified on the DEFINE or START command, or the link driver will not start.

Specification

Include as many statements as needed; they are optional. LINKDEFINE statements can be placed anywhere in the configuration file after the LOCAL, CHANNELS, and OPFORM statements (if these statements exist). If the LINKDEFINE statement references a link ID or an FCB name, you must place LINKDEFINE after the corresponding link definition or FCB statement.

You can specify multiple LINKDEFINE statements for the same link. If you specify more than one LINKDEFINE statement with the same attribute, RSCS uses the attribute from the last LINKDEFINE statement in the configuration file.

Operands

linkid

is the 1- to 8-character name of the link. If the link defines a connection to a peer node, the link ID must be the name of the node (node ID) to which the link connects, unless you specify the NODE operand. If you specify NODE, you can use a nickname as the link ID.

NOASStart

AStart

specifies whether the link is defined as an auto-start link. An auto-start link will be started automatically when a file is queued for transmission on the link while it is inactive. Printer links (ASCII-type, LPR-type, MRJE-type, RJE-type, SNA3270P-type, TCPASCII-type, TN3270E-type, UFT-type, and 3270P-type) will be auto-started only if the queued file has a class and form name that match those defined for the link. The default is NOASSTART.

Class *

Class *class*

specifies the classes of files that can be processed by the active link. You can specify an asterisk (*) to indicate all file classes can be processed, or you can specify a 1- to 4-character string (*c*, *cc*, *ccc*, or *cccc*), where each character is one of the valid z/VM spool classes, A - Z and 0 - 9. RSCS processes files in the order that the classes are specified on this operand, left to right, and processes files in priority order within each class. The default is CLASS *. If you specify * or allow it to default, RSCS processes files in priority sequence only.

DP *dpriority*

specifies a number, 1 - 9, that allows GCS to alter the dispatching priority of this link in relation to the dispatching priorities of other links. Priority 1 is the highest and 9 is the lowest. The default is 5.

FANout *linkid*

specifies the name of the next link that is physically attached to the local system and through which RSCS routes files to other systems. RSCS uses the fanout link as a shared path when distributing multiple data set header files over the network. The fanout link must have been previously defined in the configuration file.

FCB

specifies the forms control buffer (FCB) processing mode.

IGNORE

Ignore all FCB names. This is the default. If you specify FCB IGNORE for a SNA3270P-type, TN3270E-type, or 3270P-type link, RSCS converts all channel skips (other than channel one) to a "line-space-one".

DYNAMIC

Process files with any FCB name. For SNA3270P-type, TN3270E-type, and 3270P-type links, RSCS will perform channel skip simulation. If the file does not specify an FCB name, RSCS treats the file as if IGNORE was specified (RSCS converts all channel skips to a "line-space-one"). Before using an FCB name, you must define the FCB image to RSCS by using the FCB statement. For more information, see the ["FCB Statement"](#) on page 139.

fcfname

Process only files with this 1- to 4-character FCB image name. Before channel skip simulation can occur on SNA3270P-type, TN3270E-type, and 3270P-type links, you must define the FCB name to RSCS by using the FCB statement. For more information, see the ["FCB Statement"](#) on page 139.

FCB is valid only for ASCII-type, MRJE-type, RJE-type, SNARJE-type, SNA3270P-type, TCPASCII-type, TN3270E-type, and 3270P-type links. If you specify FCB for any other link type, RSCS ignores the operand.

LINE *ccuu*

specifies the virtual device number of a telecommunication adapter or printer for this link. Valid addresses are X'0000' and X'0002' - X'FFFF'. The virtual addresses cannot be within the RSCS virtual unit record device pool (default X'0F00' - X'0FFF', or as specified on the CHANNELS statement). The default is X'0000'.

If specified for a GATEWAY-type, LISTPROC-type, LPD-type, LPR-type, NOTIFY-type, SNANJE-type, SNARJE-type, SNA3270P-type, TCPASCII-type, TCPNJE-type, UFT-type, or UFTD-type link, RSCS checks the syntax of the virtual device number, but does not use it.

LOGMode *logmodename*

specifies the 1- to 8-character name of the logon mode table entry for an SNA LU. This allows the default logon mode table defined in the VTAM configuration to be overridden. If you omit this operand, the name will be kept as blank, causing the default logon mode table to be used. If you specify this operand for a non-SNA link, the syntax is checked, but it is not used.

LUName *luname*

specifies the 1- to 8-character logical unit name for the device or system connected to this link. If you omit this operand, the logical unit name is assumed to be the same as the link ID. If you specify this operand for a non-SNA link, the syntax is checked, but it is not used.

Note: You cannot use an LU name that was previously specified for another link. If you specify an already-defined LU name, RSCS rejects the LINKDEFINE statement and issues message DMT713E.

NODE *nodeid*

specifies the 1- to 8-character name of the node at the other end of the link. If you omit this operand, the link ID must match the node ID of the node at the other end of this link. Otherwise, RSCS has no way of knowing what node this link connects to.

If you specify this operand, you have told RSCS the name of the node this link connects to, and therefore the link ID does not need to match the node ID. This lets you specify a nickname for the link and lets you define more than one link to the node. For more information about using NODE, see usage notes ["1"](#) on page 156, ["2"](#) on page 157, and ["3"](#) on page 157.

Note: The NODE operand applies only to GATEWAY-type, NJE-type, SNANJE-type, and TCPNJE-type links. It is ignored for all other link types.

Queue

specifies how the files will be queued for transmission on the target link ID.

Priority

By priority (low-numbered priority first) and by size within priority (small files first). This is the default.

Fifo

In order of arrival (the time the file was written into the CP spool). Files that were already queued before the link is defined will not be reordered in the queue.

Size

By size (small files first).

NORETry

RETry

specifies whether RSCS should try to restart an auto-dial or SNA link after certain temporary error conditions have occurred. These error conditions include VTAM errors and time outs on a dial to a remote system or workstation. The default is NORETRY.

If you specify RETRY on an auto-dial or SNA link, RSCS will (by default) automatically try to restart the link after 1 minute. If the restart fails, RSCS will wait 10 minutes before trying again. If that restart fails, RSCS will wait 19 minutes before trying again. RSCS will continue to retry the link indefinitely at increasing intervals. The intervals are 1, 10, 19, 27, 34, 40, 45, 49, 52, 54, and 55 minutes. Once RSCS reaches the 55 minute interval between restarts, all restarts after that point will be 55 minutes apart until the link restarts or RSCS terminates.

To change the default retry intervals for your local system, use the RETRY statement described [“RETRY Statement” on page 257](#).

MAnual

specifies that the link will process only files with the current form name. This is the default. (If you specify the form name operand as FORM *, RSCS will process files with any form name.)

AUto

specifies that the link will process files with any form name and issue automatic prompting messages for any forms changes.

SEtUp

specifies that the link will process files with any form name and issue automatic prompting messages for any forms changes. Also, the workstation operator is allowed to check and adjust forms alignment when changing forms. If you specify this operand for an LPR-type link, it is ignored.

FOrm name

FOrm *

specifies the 1- to 8-character operator form name of files that may be processed for this link. If you specify an asterisk (*), any file may be processed, regardless of the form name.

SLOWdown

specifies the file inventory levels when slowdown mode will be started and stopped for this link. (Slowdown mode applies only to a GATEWAY-type, NJE-type, SNANJE-type, or TCPNJE-type link.) You can define *absolute* (base) or *relative* slowdown values.

Absolute (base) slowdown values for the link are defined by specifying the actual number of files:

start_point

Number of files when slowdown mode will be started. This value can be 1 - 10000. The start value must be greater than the stop value.

stop_point

Number of files when slowdown mode will be stopped. This value can be 1 - 9999.

You can also set absolute slowdown values for a specific link by using the SLOWDOWN operand on the DEFINE command. You can set absolute slowdown values for the system by using the SLOWDOWN statement or command.

Relative slowdown values for the link are defined by applying a delta value to the current absolute slowdown values for the system:

+nnnn

Number of files to be added to the absolute start and stop values for the system. This delta value can be 1 - 9999.

-nnnn

Number of files to be subtracted from the absolute start and stop values for the system. This delta value can be 1 - 9999.

You can also set relative slowdown values for a specific link by using the SLOWDOWN operand on the DEFINE command.

Note:

1. When you change the absolute slowdown values for your system, RSCS will automatically change any relative values you assigned for a link.
2. If there are no absolute slowdown values for your system, you cannot assign relative slowdown values for a link. If you try, RSCS issues error message DMT489E.
3. For more information about specifying absolute system slowdown values for your system, see the [“SLOWDOWN Statement”](#) on page 265 or the SLOWDOWN command (see [z/VM: RSCS Networking Operation and Use](#)).

TRace

starts or stops tracing for specific events. The trace records give additional details about the data and its flow across a link.

ALL

Starts the full-buffer logging of activity on the link. (This operand is recommended by the IBM Support Center for diagnostic purposes.)

LOG

Starts the partial-buffer logging of activity on the link. Only the first 16 bytes and last 16 bytes in the data buffer are logged. LOG is the default function.

OFF

Stops the reporting of line activity, stops the logging of I/O activity (and initiates the automatic printout of the log data), and resets the routing of the log output.

RECOords

Starts the logging of NJE record segments sent and received on this link. This option is valid only for GATEWAY-type, LISTPROC-type, NJE-type, SNANJE-type, and TCPNJE-type links. If you specify RECORDS for any other link type, RSCS issues message DMT204E.

TYPE

specifies the type of telecommunication protocol to be used to communicate with the remote workstation or system. The valid types are:

ASCII

Connection to an ASCII printer or plotter using an ASCII control unit.

GATEWAY

Connection using an installation-defined protocol.

LISTPROC

List processor.

LPD

Connection that enables RSCS to receive print data streams from an LPR client in a TCP/IP network for distribution in the RSCS network.

LINKDEFINE

LPR

Connection that enables RSCS to send print data streams to a line printer daemon in a TCP/IP network.

MRJE

BSC connection to a multi-leaving workstation or host system.

NJE

BSC, CTCA, ESCON or FICON CTC, or 3088 connection to a peer NJE system.

NOTIFY

Note generator.

RJE

BSC connection to a BSC workstation.

SNANJE

Connection to an NJE system through VTAM.

SNARJE

Connection to an SNA System/36 RJE workstation through VTAM.

SNA3270P

Connection to a 3270 Information Display System Printer through VTAM.

TCPASCII

Connection to an ASCII printer or plotter attached to a terminal server within a TCP/IP network.

TCPNJE

Connection to a remote peer NJE system in a TCP/IP network.

TN3270E

Connection to a remote 3270 printer attached within a TCP/IP network.

UFT

Connection that enables RSCS to send files to a remote UFT daemon in a TCP/IP network.

UFTD

Connection that enables RSCS to receive files from a remote UFT Client in a TCP/IP network for distribution in the RSCS network.

3270P

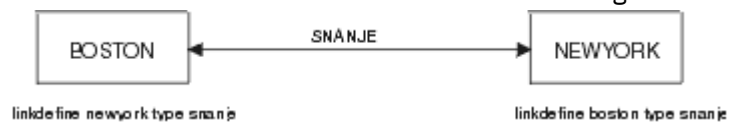
Connection to a 3270 printer using a BSC adapter or channel attachment.

Note:

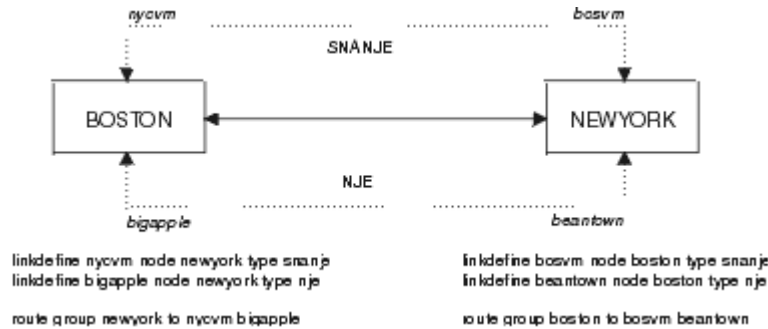
1. If you have defined a new link type (using the LINKTYPE statement), you can also specify that link type.
2. The LINKTYPE statement must appear before any LINKDEFINE statements that refer to it.
3. If you omit the TYPE operand (and it is not specified on another LINKDEFINE statement), RSCS sets the link type to UNDEFIND. In that case, the link type must be specified on a DEFINE or START command for the link.

Usage Notes

1. You can use the NODE parameter to define more than one link to an adjacent node. Without the NODE parameter, a typical link and the associated LINKDEFINE statements might be:



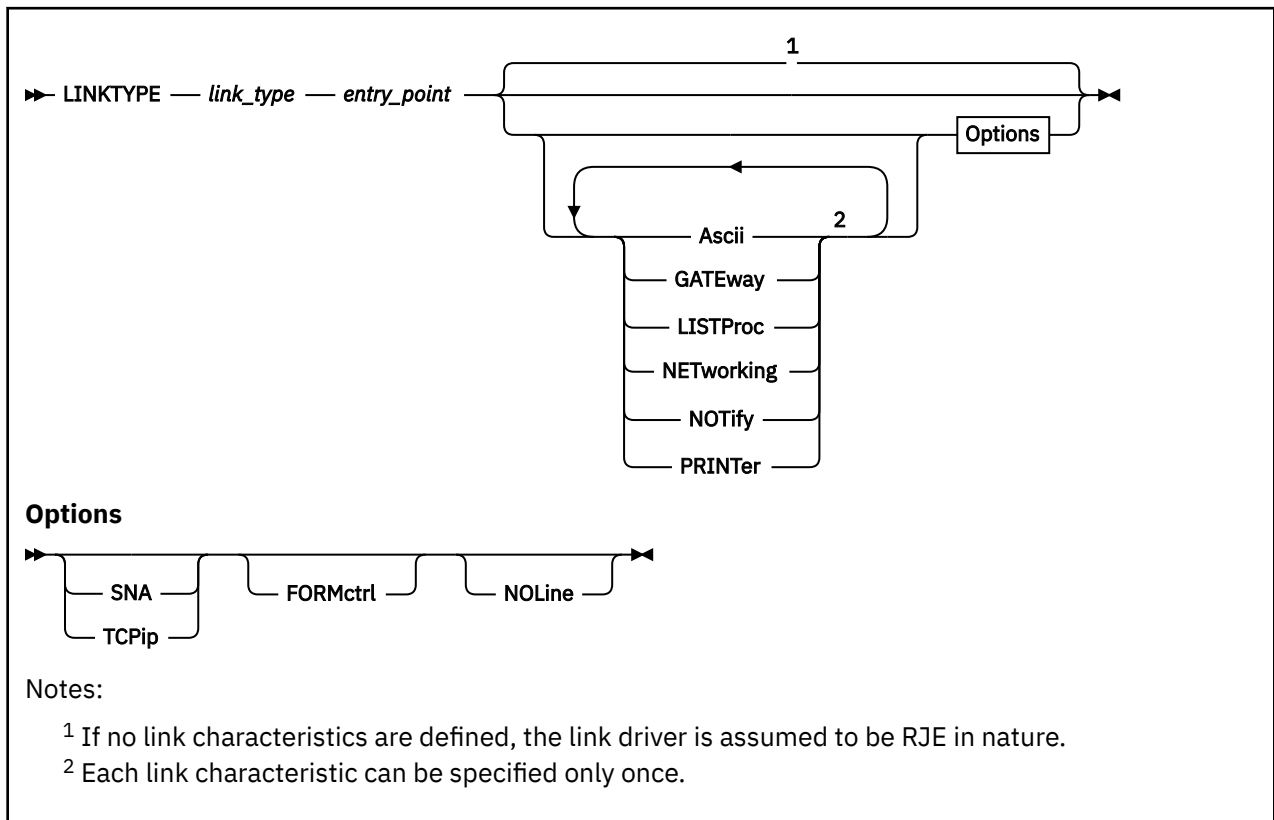
If you used the NODE parameter, you could define two (or more) links to the same node. Then, you could use the ROUTE statement to create a routing group for the links. The link and the associated LINKDEFINE and ROUTE statements would be:



After setting up multiple links to the same node and creating a routing group that contains those links, users can send files to the node and have RSCS decide which link to take.

2. An immediate routing loop occurs when a file arrives and departs over the same link. RSCS flags this condition so that you can take the appropriate action. If you use the NODE operand to define more than one link to an adjacent node, an immediate routing loop can occur when a file arrives on one link and departs on the other. Although the link IDs are different for these two links, RSCS will also flag this as an immediate routing loop. Using the example in usage note "1" on page 156, if node BOSTON sends a file to node NEWYORK over link NYCVM and node NEWYORK sends the file back to node BOSTON over link BEANTOWN, RSCS will detect the routing loop and flag the file as looping.
3. *MULTI* files (files with multiple data set headers) normally use the link ID as the file name of the file. If you create more than one link to a node (using the NODE operand), RSCS uses the node ID, instead of the link ID, as the file name. For example, if node BOSTON needs to send a *MULTI* file to node NEWYORK, RSCS creates a NEWYORK *MULTI* file (not a NYCVM *MULTI* or BIGAPPLE *MULTI* file). RSCS can then send this file to node NEWYORK over either link (NYCVM or BIGAPPLE).
4. Only one SNANJE-type link can be defined between one RSCS node and the same remote NJE node.
5. Before implementing a TCPNJE-type link to JES2, you should check the JES2 PSP buckets for required service.

LINKTYPE Statement



Purpose

The LINKTYPE statement lets you dynamically define additional link drivers to the base RSCS system.

Attention

Writing a link driver requires a thorough knowledge of system programming, RSCS programming conventions, and RSCS internals. Without having this knowledge, if you write a link driver and use the LINKTYPE command to define it to RSCS, you can seriously degrade system performance and cause system problems.

Specification

Include as many statements as needed; they are optional. LINKTYPE statements can be placed anywhere in the configuration file after the LOCAL statement (if one exists) and before any LINKDEFINE statements that refer to them.

Operands

link_type

is the 1- to 8-character name of the link driver type.

PI

entry_point

is the 1- to 8-character name of the entry point of the link driver module.

PI end

Ascii

specifies that this link driver type is ASCII in nature.

GATEway

specifies that this link driver type is a GATEWAY link driver.

LISTProc

specifies that this link driver type is a list processor link driver.

NETworking

specifies that this link driver type will carry out NJE protocol and interact with the common networking function of RSCS.

NOTify

specifies that this link driver type is a NOTIFY link driver.

PRINTer

specifies that this link driver type is a printer link driver.

SNA

specifies that this link driver type is an SNA session driver. If you omit this operand, RSCS assumes the link driver type is a non-SNA link driver.

TCPip

specifies that this link driver type will be used for TCP/IP connections.

FORMctrl

specifies that this link driver type will use forms control. If you omit this operand, RSCS assumes the link driver will not have forms control.

NOLine

specifies that this link driver type does not have a specific line address associated with it.

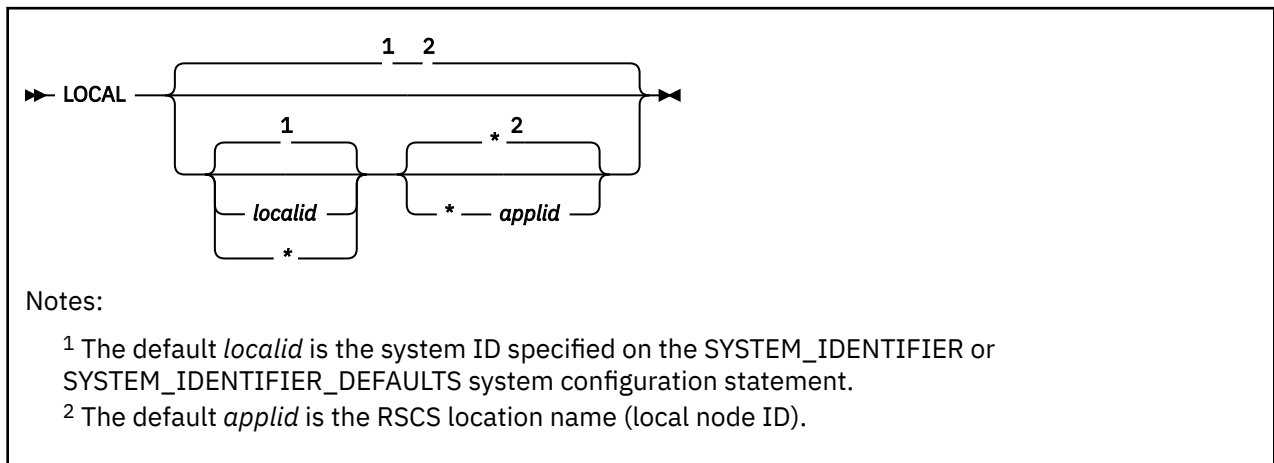
Usage Notes

1. If you specify *link_type* and *entry_point_name* without specifying any of the link driver characteristics (ASCII, GATEWAY, LISTPROC, NETWORKING, NOLINE, NOTIFY, PRINTER, SNA, TCP, FORMCTRL, or NOLINE), RSCS assumes this link driver will be RJE in nature.
2. The LINKTYPE statement lets you add your own link driver type to the list that RSCS recognizes. Because you are supplying the link driver, RSCS does not check the link driver characteristics. This means you can specify as many characteristics as you want. For example, if you want to define a link type that is both networking and SNA in nature, the LINKTYPE statement would be:

```
linktype netsna dmtnslep networking sna
```

Some characteristics do not go together (for example, NETWORKING and PRINTER). However, because RSCS does not check the characteristics, you could specify them on the same LINKTYPE statement.

LOCAL Statement



Purpose

The LOCAL statement defines the name by which the local RSCS node is known to other nodes in the network.

Specification

One statement is allowed. If specified, it must be the first executable statement in the configuration file. You can place only blank lines, comment lines, IMBED statements, and TOLERANCE statements before the LOCAL statement.

Operands

localid

*

is the 1- to 8-character name of the local RSCS node. This name must be unique within the RSCS network of interconnected devices and systems.

If you specify an asterisk (*) or omit this operand, RSCS uses the system ID specified on the SYSTEM_IDENTIFIER or SYSTEM_IDENTIFIER_DEFAULTS statement in the z/VM system configuration file. If a system ID was not specified when the z/VM system was generated, RSCS initialization terminates and message DMT493T is issued.

* *applid*

is the 1- to 8-character name by which VTAM knows RSCS. This name is the same as that specified by an APPL definition statement when VTAM was configured. If not specified here or on the RSCS NETWORK command, the name defaults to the RSCS location name (local node ID).

Note: The asterisk (*) specified before the *applid* has no meaning. It is a placeholder to maintain compatibility with RSCS V1.

MSGNOH Statement

► MSGNOH ◄

Purpose

The MSGNOH statement tells RSCS to use the CP MSGNOH command when issuing messages to local users.

Specification

One statement is allowed. It is optional and can be placed anywhere in the configuration file after the LOCAL statement (if one exists).

Usage Notes

1. If you specify this statement in the configuration file, the IDENTITY statement in the RSCS virtual machine definition must specify the CP privilege class that includes the MSGNOH command. (The z/VM-provided default privilege class that contains MSGNOH is class B.) For more information, see “RSCS Virtual Machine” on page 31.
2. If you do not specify this statement in the configuration file, RSCS uses the RSCS MSG command to send messages to local users. For information about the MSG command, see *z/VM: RSCS Networking Operation and Use*.

Special Note about Columnar Messages

Many of the responses to RSCS QUERY commands are issued as columnar messages. Columnar messages were designed with the idea that you would use MSGNOH to send messages to local users. This means that messages from the local RSCS virtual machine can be up to 78 characters wide, and messages from a remote RSCS virtual machine can be up to 63 characters wide, before RSCS must wrap the message to the next line. When a columnar message wraps, it is difficult to read.

Also, RSCS must truncate columnar messages longer than 132 characters. It does not matter whether the message is from a local or remote RSCS virtual machine. When RSCS finds a columnar message longer than 132 characters, it truncates the message at the end of the last column that is less than 132 characters. This means that you will not see any part of the column that would make the message too long, nor will you see any other columns after that point.

If you do not use MSGNOH, be aware that RSCS QUERY command responses may be affected.

OPFORM Statement



Purpose

The OPFORM statement allows the RSCS installation to specify the default operator form name for printer forms control for print files. RSCS uses the OPFORM statement to determine what type of file form to select for processing.

Specification

One statement is allowed. It is optional and can be placed anywhere in the configuration file after the LOCAL statement (if one exists) and before all LINKDEFINE statements.

Operands

formname

*

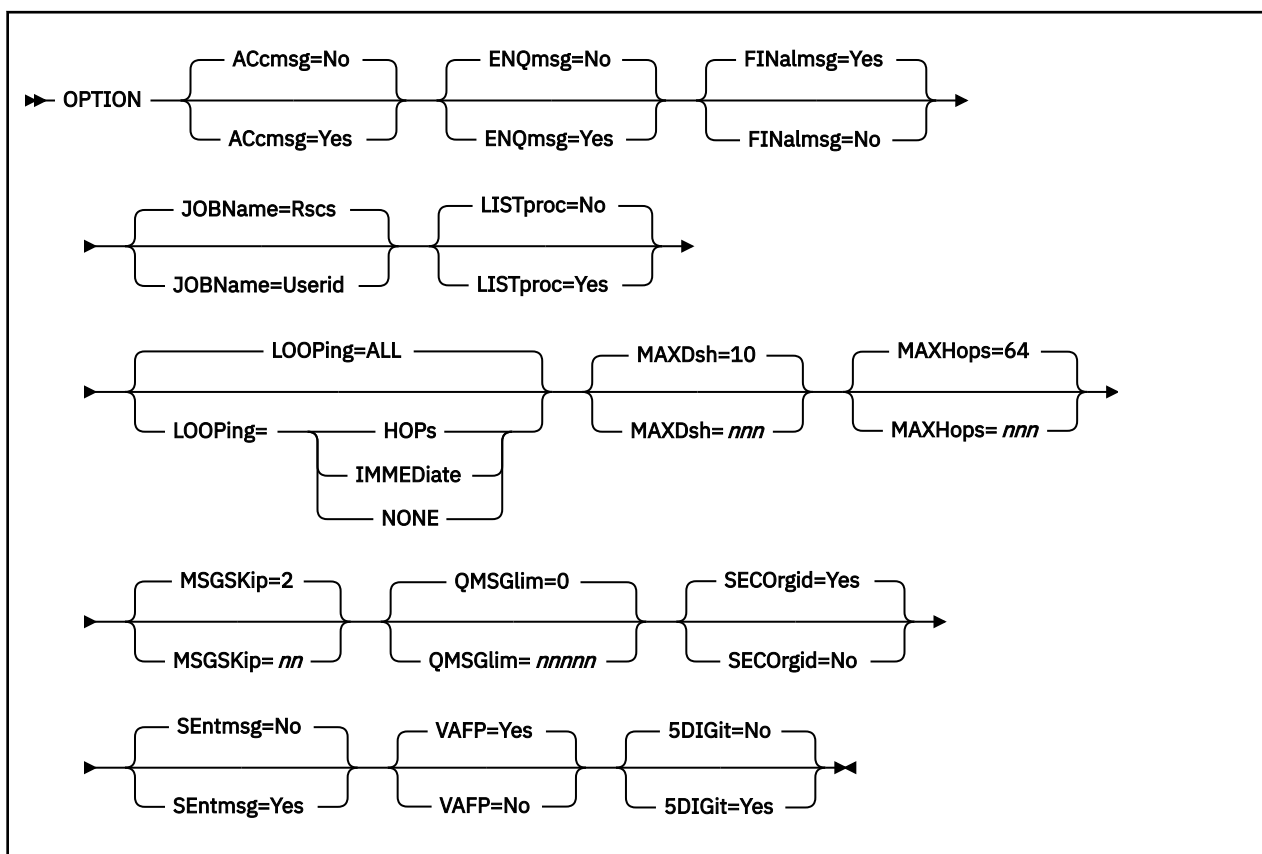
is the 1- to 8-character name to be used as the default operator form name for print files if none is specified when a link is started. For consistency between z/VM and RSCS operations, this form name should match the operator form name that corresponds to the user form name specified on the FORM_DEFAULT statement in the z/VM system configuration file.

If you specify an asterisk (*), printer forms control is not performed; that is, any file will be selected for transmission regardless of form name.

Usage Notes

1. If you omit the OPFORM statement, RSCS uses STANDARD as the default operator form name. This is the same name that CP uses if no user form name is specified on the FORM_DEFAULT system configuration statement.
2. The USERFORM system configuration statement creates a list of user form names and their corresponding operator form names. For more information about the FORM_DEFAULT and USERFORM statements, see [z/VM: CP Planning and Administration](#).

OPTION Statement



Purpose

The `OPTION` statement allows you to define defaults for various RSCS system and link parameters. Many of these values are used as defaults on the `LINKDEFINE` statement for certain types of links unless overridden on a `PARM` statement or on a `DEFINE` or `START` command. The description of each `OPTION` statement operand indicates how you can override the defined or defaulted value.

Specification

Include as many statements as needed; they are optional. `OPTION` statements can be placed anywhere in the configuration file after the `LOCAL` statement (if one exists).

Operands

ACcmsg=No

ACcmsg=Yes

specifies whether the local RSCS system will generate the initial file-accepted message (DMT102I). This message tells you that RSCS has accepted your file for transmission to its ultimate destination. It does not tell you on which link (or links) the file is queued. The default is NO.

You can override the defined or defaulted `ACCMMSG` value by specifying the `ACCMMSG` operand on the `CP TAG` command or in the address record. For information on the `TAG` command, see [z/VM: RSCS Networking Operation and Use](#).

ENQmsg=No

ENQmsg=Yes

specifies whether the local RSCS system will generate the initial file-queued message (DMT101I). This message tells you on which link (or links) your file is queued. The default is NO.

OPTION

You can override the defined or defaulted ENQMSG value by specifying the ENQMSG operand on the CP TAG command or in the address record. For information on the TAG command, see [z/VM: RSCS Networking Operation and Use](#).

Note: If you are using the support that lets you route a node through multiple links and you specify ENQMSG=YES, users will receive one message for each link on which the file is queued. Sending out one file and receiving more than one file-queued message in return is potentially confusing to users who are not aware of the multiple-link support. To avoid this confusion, specify ENQMSG=NO and ACCMSG=YES.

FINALmsg=

specifies whether RSCS will deliver the confirmation-of-arrival message from the destination RSCS system (message DMT104I for a destination user ID, DMT147I for a printer or workstation).

Yes

Both the originating and destination virtual machines will receive the message. This is the default.

No

Only the destination virtual machine will receive the message. (This applies only to message DMT104I.)

You can override the defined or defaulted FINALMSG value by specifying the FINALMSG operand on the CP TAG command or in the address record. For information on the TAG command, see [z/VM: RSCS Networking Operation and Use](#).

JOBName=

specifies the type of job name RSCS will create for all networking link drivers (GATEWAY, LISTPROC, NJE, SNANJE, and TCPNJE). RSCS creates job headers for all files originating at your local node and all files arriving at your local node without a job header (that is, those coming from a workstation link).

Rscs

RSCS will generate a job name of RSCSnnnn, where nnnn is the origin spool ID of the file. This is the default.

Userid

RSCS will use the origin user ID as the job name.

You can override the defined or defaulted JOBNAME value for a specific GATEWAY-type, LISTPROC-type, NJE-type, SNANJE-type, or TCPNJE-type link by specifying the JOBNAME operand on the PARM statement for the link or by specifying the JOBNAME operational parameter on the DEFINE or START command. You can override the defined or defaulted JOBNAME value for a specific file by specifying the JOBNAME operand on the CP TAG command. For information on the DEFINE, START, and TAG commands, see [z/VM: RSCS Networking Operation and Use](#).

LISTproc=

specifies whether the connected system can handle an unlimited number of data set headers.

No

The link will use the number of data set headers specified on the MAXDSH operand. This is the default.

Yes

The connected system can handle an unlimited number of data set headers.

You can override the defined or defaulted LISTPROC value for a specific GATEWAY-type, NJE-type, SNANJE-type, or TCPNJE-type link by specifying the LISTPROC operand on the PARM statement for the link or by specifying the LISTPROC operational parameter on the DEFINE or START command. For information on the DEFINE and START commands, see [z/VM: RSCS Networking Operation and Use](#).

The following table summarizes how non-list processor systems will handle list processor files.

Table 16. How Non-List Processor Systems Handle List Processor Files

Non-List Processor Systems	How the File Will Be Handled
<ul style="list-style-type: none"> • RSCS V1.3* • JES3 • VSE/POWER 	These systems will immediately split the file into separate spool files. These files will be either distributed to local user IDs or sent on to other links.
<ul style="list-style-type: none"> • RSCS V2.2* (without list processor support • RSCS V2.3* or later* (without a list processor defined) • JES2 	These systems will receive the file and split out separate files for local user IDs. They will also split out one file containing data set headers to be sent to each link on the distribution list.

Note: * These systems will handle only a limited number of data set headers.

LOOPING=

specifies the level of checking to be performed by the RSCS loop checking facility.

ALL

Checks for both HOPS and IMMEDIATE loops. This is the default.

HOPS

Checks for and flags files that have passed through more nodes than was specified on the MAXHOPS operand.

IMMEDIATE

Checks for and flags files, commands, and messages routed through the link on which they arrived.

NONE

Does not perform any loop checking.

You can override the defined or defaulted LOOPING value by using the LOOPING command. For information on the LOOPING command, see [z/VM: RSCS Networking Operation and Use](#).

MAXDsh=nnn

specifies the maximum number of data set headers that the link driver will place on the link when processing a list processor file. This value is intended for connections to NJE subsystems that do not have list processor capabilities. This operand is ignored if LISTPROC=YES is specified.

This value can be 0 - 100. The default is 10. A value of 0 is a special case that means the determination for the number of data set headers placed on the link is based on the actual number of data set headers contained in the spool file. A value of 0 should be specified only with full understanding of the number of data set headers that the remote system can handle.

You can override the defined or defaulted MAXDSH value for a specific GATEWAY-type, NJE-type, SNANJE-type, or TCPNJE-type link by specifying the MAXDSH operand on the PARM statement for the link or by specifying the MAXDSH operational parameter on the DEFINE or START command. For information on the DEFINE and START commands, see [z/VM: RSCS Networking Operation and Use](#).

MAXHops=nnn

specifies the minimum number of nodes that a file may traverse before RSCS recognizes that the file is caught in a routing loop. RSCS checks the file's hop count to determine if the file is caught in a routing loop. Some nodes in the loop may not update the hop count, so the actual number of traversed nodes may be higher than the hop count indicates. This value can be 2 - 256. The default is 64.

You cannot override the MAXHOPS value after RSCS initializes, but you can control loop checking by using the LOOPING command. For information on the LOOPING command, see [z/VM: RSCS Networking Operation and Use](#).

MSGSKip=nn

specifies the number of file buffers that are sent on NJE-type links before a message buffer is sent (unless there are no file buffers available). This value can be 0 - 15. The default is 2.

A value of 0 is a special case that tells RSCS to send a message buffer whenever messages are available to send on the link, regardless of whether file buffers are available.

You can override the defined or defaulted MSGSKIP value for a specific NJE-type, SNANJE-type, or TCPNJE-type link by specifying the MSGSKIP operand on the PARM statement for the link or by specifying the MSGSKIP operational parameter on the DEFINE or START command. For information on the DEFINE and START commands, see *z/VM: RSCS Networking Operation and Use*.

Note: If a link determines during sign-on that the peer system supports the mixed RCB feature of NJE, RSCS ignores the MSGSKIP setting for that link. The mixed RCB feature lets RSCS transmit buffers containing a mixture of information (files and messages). If RSCS can transmit files and messages in one buffer, there is no need to use the MSGSKIP operand to tell RSCS how many file buffers to send before sending a message buffer. For more information about the mixed RCB feature of NJE, see *z/OS: Network Job Entry (NJE) Formats and Protocols* ([https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa320988/\\$file/hasa600_v2r5.pdf](https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa320988/$file/hasa600_v2r5.pdf)).

QMSGlim=nnnnn

specifies the limit on the number of response messages that will be issued for an individual RSCS QUERY command.

0

No limit is in effect. All QUERY response messages will be issued. This is the default.

nnnnn

Maximum number of QUERY response messages that will be issued. This value can be 1 - 32700. If this limit is in effect, it will be checked for every QUERY command. If the limit is reached, message DMT007W will be issued to let the command issuer know that not all responses have been returned.

You cannot override the QMSGGLIM value after RSCS initializes.

SECorgid=Yes**SECorgid=No**

specifies whether RSCS should load the secure origin ID support. The default is YES.

When you load the secure origin ID support, RSCS issues CP DIAGNOSE code X'F8' to:

- Get origin user ID information from the CP spool before sending a file to another node
- Write origin user ID information to the CP spool before writing an incoming file to spool

Note:

1. If you omit SECORGID and DIAGNOSE code X'F8' is not available on your local system, RSCS issues message DMT502W and continues initialization.
2. If you specify SECORGID=YES and DIAGNOSE code X'F8' is not available on your local system, or RSCS is not authorized to issue the diagnose code, RSCS issues message DMT501T and terminates initialization.

You cannot override the SECORGID value after RSCS initializes.

SEntmsg=No**SEntmsg=Yes**

specifies whether confirmation-of-forwarding messages from intermediate RSCS systems (DMT147I) will be delivered. The default is NO.

You can override the defined or defaulted SENTMSG value for a specific file by specifying the SENTMSG operand on the CP TAG command or in the address record. For information on the TAG command, see *z/VM: RSCS Networking Operation and Use*.

VAFP=Yes**VAFP=No**

specifies whether RSCS will use the VAFP function, if available, to allow records longer than 204 bytes to be written to the spool. The default is YES.

If you specify VAFP=NO, RSCS will reject the request to send any VAFP files on NJE-type links. VAFP files that are spooled and tagged to RSCS for local processing, such as for sending to a local printer, will be accepted for processing as long as they are not destined for sending on an NJE-type link.

You cannot override the VAFP value after RSCS initializes.

5DIGit=No**5DIGit=Yes**

specifies whether RSCS will display the origin job ID of a file as 5 digits. The default is NO.

You cannot override the 5DIGIT value after RSCS initializes.

PARM Statement

Purpose

The PARM statement defines operational characteristics to be used by the link driver handling the specified link. The PARM information is made available to the driver when the link is started. The format and content of PARM information are different for each type of link.

- [“PARM Statement for an ASCII-Type Link” on page 169](#)
- [“PARM Statement for a GATEWAY-Type Link” on page 172](#)
- [“PARM Statement for a LISTPROC-Type Link” on page 177](#)
- [“PARM Statement for an LPD-Type Link” on page 181](#)
- [“PARM Statement for an LPR-Type Link” on page 183](#)
- [“PARM Statement for an MRJE-Type Link” on page 188](#)
- [“PARM Statement for an NJE-Type Link” on page 193](#)
- [“PARM Statement for a NOTIFY-Type Link” on page 201](#)
- [“PARM Statement for an RJE-Type Link” on page 203](#)
- [“PARM Statement for an SNANJE-Type Link” on page 208](#)
- [“PARM Statement for an SNARJE-Type Link” on page 214](#)
- [“PARM Statement for an SNA3270P-Type Link” on page 217](#)
- [“PARM Statement for a TCPASCII-Type Link” on page 224](#)
- [“PARM Statement for a TCPNJE-Type Link” on page 227](#)
- [“PARM Statement for a TN3270E-Type Link” on page 234](#)
- [“PARM Statement for a UFT-Type Link” on page 240](#)
- [“PARM Statement for a UFTD-Type Link” on page 243](#)
- [“PARM Statement for a 3270P-Type Link” on page 245](#)

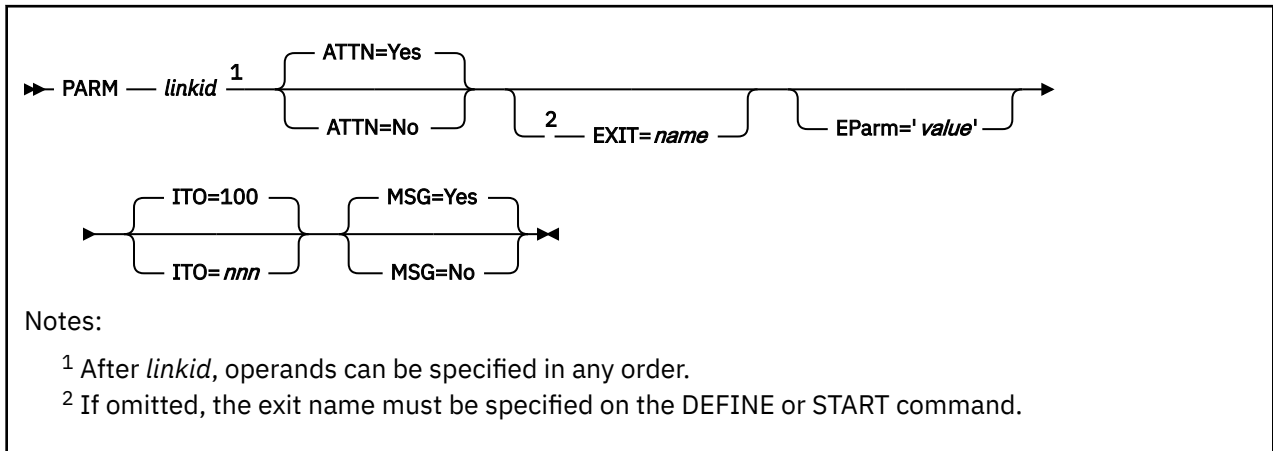
Specification

Include as many statements as needed; they are optional. However, you can have only one PARM statement for each link ID. The PARM statement may be placed anywhere in the configuration file after the LINKDEFINE statement to which it refers. If PARM statement operands are intended to override OPTION statement defaults, the PARM statement must be placed after the OPTION statement.

Attention

Although the PARM statement is not required, it is recommended. If you do not define a PARM statement for a link, all of PARM defaults for that link type will apply to the link. You can override PARM attributes for the link by specifying values on the DEFINE or START command. Some required link attributes do not have PARM defaults; if a value is not defined on the PARM statement, it must be specified on the DEFINE or START command, or the link will not activate on the START command. For information on the DEFINE and START commands, see [z/VM: RSCS Networking Operation and Use](#).

PARM Statement for an ASCII-Type Link



Purpose

This PARM statement defines the operational characteristics for a specific ASCII-type link. An ASCII-type link connects the local RSCS system to a remote ASCII printer or plotter.

The link can be started:

- Automatically during system start-up
- Automatically when a file is queued for transmission on an auto-start link
- Manually by an operator or authorized user

Note: If you do not define a PARM statement for an ASCII-type link, all of the indicated defaults will apply to the link. You can override PARM attributes by specifying values on the DEFINE or START command. Some required link attributes do not have PARM defaults; if a value is not defined on the PARM statement, it must be specified on the DEFINE or START command, or the link will not activate on the START command. For information on the DEFINE and START commands, see [z/VM: RSCS Networking Operation and Use](#).

Operands

linkid

is the name of the ASCII-type link, as defined on a LINKDEFINE statement.

ATTN=Yes

ATTN=No

specifies whether RSCS should wait for an attention interrupt after printing each buffer transmitted to the ASCII control unit. The default is YES.

If you have a 7171 ASCII Device Attachment Control Unit, you should not specify ATTN=NO.

If you have a 9370 ASCII Subsystem Controller, the value of ATTN depends on whether the controller was configured to respond to the host with an attention interrupt after printing each buffer. To configure the controller, use the **QFADEF** menu of the Device Definition Table for that particular device. Choose the **Send ATTN after transparent write** option. Usually, it is recommended that you specify ATTN=NO for 9370 ASCII Subsystem Controllers.

PI

EXIT=name

specifies the name of the exit routine module that contains the exit routines supporting the specific ASCII device. The exit routine module must reside in a LOADLIB that has been identified by using the GCS GLOBAL command. If you omit this operand and the exit name is not specified on the DEFINE or START command, or if RSCS cannot find the exit routine module, the link driver will not start.

PARM (ASCII-Type)

IBM supplies the following sample exit routine modules in the RSCSEXIT LOADLIB:

Module

Supported Device

ASCXDSOE

DS180 Matrix Printer from Datasouth.

ASCXDWRE

LA120 DECwriter Printer from DEC.

ASCXONE

Generic ASCII printer.

ASCXPROP

IBM Proprinter.

ASCXPSE

PostScript printer.

ASCXSPWE

NEC 3515 Spinwriter Printer.

ASCXZETE

Nicolet Zeta 8 Plotter.

ASCX749E

IBM Instruments XY/749 Multipen Digital Plotter.

For more information about the ASCII exit routines, see [z/VM: RSCS Networking Exit Customization](#).

EParm='value'

specifies a parameter value associated with the exit routine module. This value is a character string up to 239 bytes in length, enclosed in single quotation marks. Any characters are allowed between the single quotation marks except another single quotation mark. The character string cannot be continued to the next line in the configuration file. The actual value of the character string is defined by the exit routine module.

For information about the EPARM parameters supported for use with the ASCXONE and ASCXPSE sample modules, see [z/VM: RSCS Networking Exit Customization](#).

PI end

ITO=nnn

specifies the inactivity time out (ITO) value, which defines the length of time that can occur with no activity (that is, no file transmission or reception) before the link is deactivated. If a link is ended because its ITO value has been exceeded, its restart exec will not be processed.

The ITO value can be 0 - 100:

0

The link will be deactivated immediately when there is no activity on it. However, before deactivating the link, RSCS will first send files that are queued for transmission, or finish reception of the file currently being received.

1 - 99

Approximate number of minutes until the link will be deactivated if there has been no activity on it.

100

No ITO will be in effect. The link will stay active until it is deactivated manually or an unrecoverable error occurs. This is the default.

MSG=

specifies whether RSCS will allow the link driver to dequeue messages destined for transmission to the remote printer.

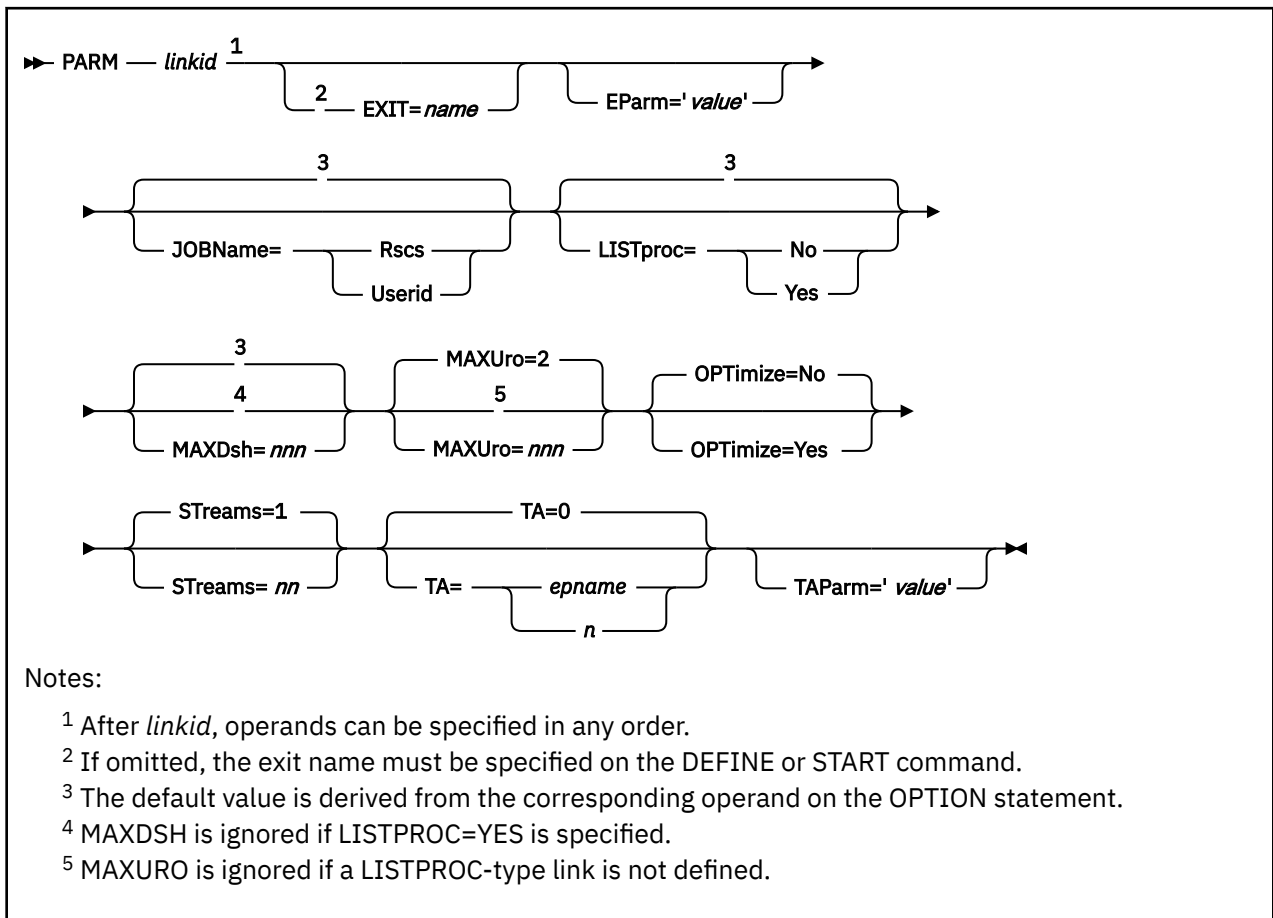
Yes

RSCS will select all messages for transmission. This is the default.

No

RSCS will select messages, but they will not be sent to the remote printer. The messages will be purged.

PARM Statement for a GATEWAY-Type Link



Purpose

This PARM statement defines the operational characteristics for a specific GATEWAY-type link. A GATEWAY-type link is an installation-defined link that lets an installation customize a network gateway.

The link can be started:

- Automatically during system start-up
- Automatically when a file is queued for transmission on an auto-start link
- Manually by an operator or authorized user

Note: If you do not define a PARM statement for a GATEWAY-type link, all of the indicated defaults will apply to the link. You can override PARM attributes by specifying values on the DEFINE or START command. Some required link attributes do not have PARM defaults; if a value is not defined on the PARM statement, it must be specified on the DEFINE or START command, or the link will not activate on the START command. For information on the DEFINE and START commands, see [z/VM: RSCS Networking Operation and Use](#).

Operands

linkid

is the name of the GATEWAY-type link, as defined on a LINKDEFINE statement.

PI

EXIT=name

specifies the name of the exit routine module that contains the gateway program. The exit routine module must reside in a LOADLIB that has been identified by using the GCS GLOBAL command. If you omit this operand and the exit name is not specified on the DEFINE or START command, or if RSCS cannot find the exit routine module, the link driver will not start.

IBM supplies a sample exit routine module called GPI in the RSCSEXIT LOADLIB. For more information about the gateway programming interface, see *z/VM: RSCS Networking Exit Customization*.

EPARM='value'

specifies a parameter value associated with the exit routine module. This value is a character string up to 80 bytes in length, enclosed in single quotation marks. Any characters are allowed between the single quotation marks except another single quotation mark. The character string cannot be continued to the next line in the configuration file. The actual value of the character string is defined by the exit routine module.

PI end

JOBName=

specifies the type of job name RSCS will create for all files originating at your local node and all files arriving at your local node without a job header (that is, those coming from a workstation link).

Rscs

RSCS will generate a job name of RSCSnnnn, where nnnn is the origin spool ID of the file.

Userid

RSCS will use the origin user ID as the job name.

If you specify this operand, it will override any previously defined default for this link. If you omit this operand, the default value is derived from the corresponding operand on the OPTION statement.

To override the default job name for a specific file, use the JOBNAM operand on the CP TAG command. For more information, see *z/VM: RSCS Networking Operation and Use*.

LISTproc=No

LISTproc=Yes

specifies whether the connected system can handle an unlimited number of data set headers. If you specify NO, the link will use the number of data set headers specified on the MAXDSH operand.

If you specify this operand, it will override any previously defined default for this link. If you omit this operand, the default value is derived from the corresponding operand on the OPTION statement.

The following table summarizes how non-list processor systems will handle list processor files.

Table 17. How Non-List Processor Systems Handle List Processor Files

Non-List Processor Systems	How the File Will Be Handled
<ul style="list-style-type: none"> • RSCS V1.3* • JES3 • VSE/POWER 	<p>These systems will immediately split the file into separate spool files. These files will be either distributed to local user IDs or sent on to other links.</p>
<ul style="list-style-type: none"> • RSCS V2.2* (without list processor support • RSCS V2.3* or later* (without a list processor defined) • JES2 	<p>These systems will receive the file and split out separate files for local user IDs. They will also split out one file containing data set headers to be sent to each link on the distribution list.</p>

Note: * These systems will handle only a limited number of data set headers.

MAXDsh=nnn

specifies the maximum number of data set headers the link driver will put on the link when processing a file with multiple data sets. This value is intended for connections to NJE subsystems that do not have list processor capabilities. This operand is ignored if LISTPROC=YES is specified.

This value can be 0 - 100. A value of 0 is a special case that means the determination for the number of data set headers placed on the link is based on the actual number of data set headers contained in the spool file. A value of 0 should be specified only with full understanding of the number of data set headers that the remote system can handle.

If you specify this operand, it will override any previously defined default for this link. If you omit this operand, the default value is derived from the corresponding operand on the OPTION statement.

MAXUro=nnn

specifies the maximum number of unit record output devices (virtual printers or punches) for each stream that the link driver will use at any one time while receiving a file. RSCS uses these devices to fan out files containing data set headers, which the link driver determines must be split off the original file. All remaining data set headers (and the file contents) are written to an additional unit record device and queued on the list processor link. This operand is ignored if a LISTPROC-type link is not defined.

This value can be 0 - 255. The default is 2. Specifying a value of 0 will cause all received data set headers and file contents to be queued to the list processor task. Under standard conditions, a link will use one more device per stream than was specified. If you specify 0 and the first data set header received in a file is for the local system or for a non-networking link attached to the local system, an extra device will be defined to hold the remaining data set headers.

OPTimize=No

OPTimize=Yes

specifies whether optimization (merging of CCWs) should be used for files that originate on the local system and files that are processed by the local system. (This does not include store-and-forward files that are passing through the local node.) The default is NO.

STreams=nn

specifies the maximum number of files that may be transmitted simultaneously. This value can be 1 - 32. The default is 1.

Note:

1. RSCS cannot actually send more than one file at a time. When several files are active on different streams, RSCS sends the file with the smallest number of blocks and credits the other files for having to wait. After the file is transmitted, RSCS subtracts the credits from the number of blocks left to send for each active file. RSCS then chooses the file with the lowest number to send. The longer a large file has to wait, the more credits it will accumulate. When the difference between the accumulated credits and the actual file size becomes small enough, RSCS transmits the large file.

PI

2. Transmission algorithm 1 can handle only seven streams. If you specify more than seven streams, you cannot specify TA=1. Instead, you must define your own transmission algorithm and identify it by number or entry point name on the TA operand.

TA=

specifies the transmission algorithm, which tells RSCS how to select files for transmission on the link. This can be specified as either of the following:

epname

is the name of the entry point (outside of the RSCS LOADLIB) where your installation-defined algorithm resides.

n

is a hexadecimal value, 0 - F, that identifies the exit routine in the DMTAXA module (DMTAXAGn):

0

Identifies an IBM-supplied algorithm that allows files to be selected regardless of size. This is the default.

Note: If you specify TA=0 or allow it to default, do not specify the TAPARM operand.

1

Identifies an IBM-supplied algorithm that specifies a default size for each stream specified on the STREAMS operand, up to seven streams. You can override the default by specifying the stream size on the TAPARM operand.

2 - F

Identifies an algorithm defined by your installation.

Note: If you update the DMTAXA module, it must be reassembled.

TAParm='value'

specifies a parameter value associated with the transmission algorithm. This value is a character string up to 80 bytes in length, enclosed in single quotation marks. (The single quotation marks are optional if you do not include blanks in the character string.) Any characters are allowed between the single quotation marks except another single quotation mark. The character string cannot be continued to the next line in the configuration file.

The value that you specify for TAPARM depends on the value that you specify for TA:

TA=

TAPARM=

0

Do not specify TAPARM; it will be rejected with an error message.

epname

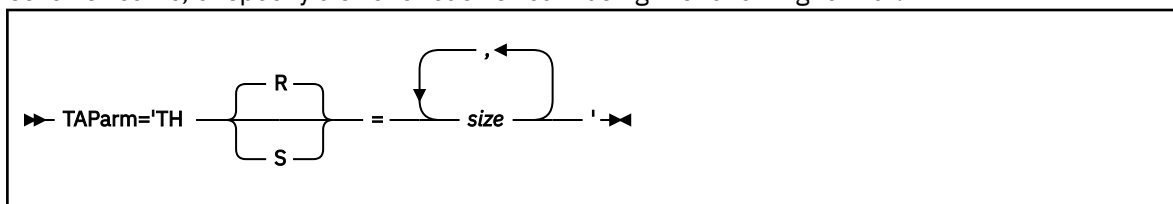
Specify a TAPARM defined by your installation.

2 - F

Specify a TAPARM defined by your installation.

1

Either omit TAPARM to accept the default size for each stream (listed in the table below), up to seven streams, or specify a size for each stream using the following format:



TH=

THR=

specifies the threshold number of records. THR is the default.

THS=

specifies the threshold number of spool file blocks. Specify THS when you expect to send files with very large records, such as printer output files, containing page image data, that are destined for a PSF-controlled printer.

size

is the file size. This can be specified as either of the following:

nnnnnnnn

Maximum file size RSCS should send on the stream. The minimum file size is one more than the maximum file size from the previous stream specification. If there is no previous stream specification, there is no minimum file size.

(min,max)

Minimum and maximum file size RSCS should send on the stream. If you do not want to specify a minimum file size, omit *min*. If you do not want to specify a maximum file

PARAM (GATEWAY-Type)

size, omit *max*. Specifying (,) means you do not want to specify either a minimum or a maximum file size; any file is eligible for transmission on the stream.

For example, if you specified STREAMS=3, you can choose the TAPARM default by omitting TAPARM or you can specify any one of the following:

TAPARM='TH=*size1*'

TAPARM='TH=*size1,size2*'

TAPARM='TH=*size1,size2,size3*'

TAPARM='THR=*size1*'

TAPARM='THR=*size1,size2*'

TAPARM='THR=*size1,size2,size3*'

TAPARM='THS=*size1*'

TAPARM='THS=*size1,size2*'

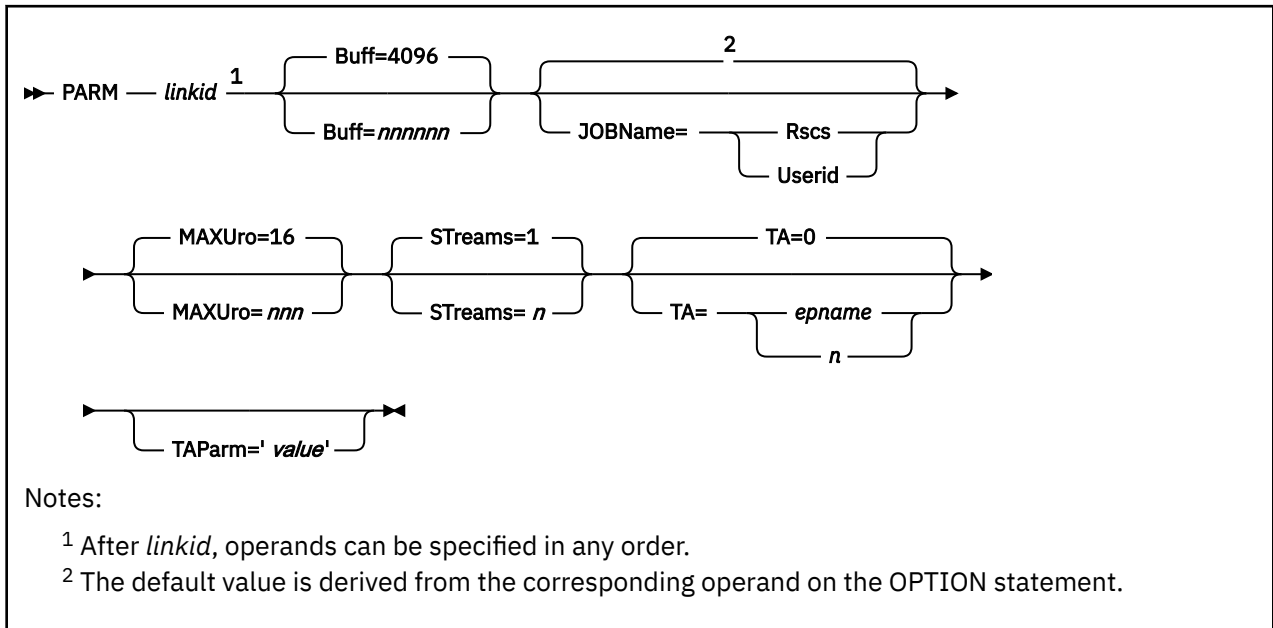
TAPARM='THS=*size1,size2,size3*'

If you specify TA=1 and omit TAPARM, these are the defaults:

If you specified	THR= Default for Stream 1	THR= Default for Stream 2	THR= Default for Stream 3	THR= Default for Stream 4	THR= Default for Stream 5	THR= Default for Stream 6	THR= Default for Stream 7
STREAMS=1	All files	—	—	—	—	—	—
STREAMS=2	(,100)	(101,)	—	—	—	—	—
STREAMS=3	(,100)	(101,)	(101,)	—	—	—	—
STREAMS=4	(,100)	(101,)	(101,)	(101,)	—	—	—
STREAMS=5	(,100)	(101,)	(101,)	(101,)	(101,)	—	—
STREAMS=6	(,100)	(101,)	(101,)	(101,)	(101,)	(101,)	—
STREAMS=7	(,100)	(101,)	(101,)	(101,)	(101,)	(101,)	(101,)

PI end

PARM Statement for a LISTPROC-Type Link



Purpose

This PARM statement defines the operational characteristics for a specific LISTPROC-type link. A LISTPROC-type link is a list processor link, which processes information to be distributed by the networking links.

The link can be started:

- Automatically during system start-up
- Automatically when a file is queued for transmission on an auto-start link
- Manually by an operator or authorized user

Note: If you do not define a PARM statement for a LISTPROC-type link, all of the indicated defaults will apply to the link. You can override PARM attributes by specifying values on the DEFINE or START command. For information on the DEFINE and START commands, see [z/VM: RSCS Networking Operation and Use](#).

Operands

linkid

is the name of the LISTPROC-type link, as defined on a LINKDEFINE statement.

Buff=nnnnn

specifies the size of the buffer used internally by the link driver for reading and writing records. This value can be 300 - 32765 bytes. The default is 4096.

JOBName=

specifies the type of job name RSCS will create for all files originating at your local node and all files arriving at your local node without a job header (that is, those coming from a workstation link).

Rscs

RSCS will generate a job name of RSCSnnnn, where nnnn is the origin spool ID of the file.

Userid

RSCS will use the origin user ID as the job name.

If you specify this operand, it will override any previously defined default for this link. If you omit this operand, the default value is derived from the corresponding operand on the OPTION statement.

PARM (LISTPROC-Type)

To override the default job name for a specific file, use the JOBNAME operand on the CP TAG command. For more information, see [z/VM: RSCS Networking Operation and Use](#).

MAXUro=nnn

specifies the maximum number of unit record output devices (virtual printers or punches) for each stream that the list processor will use at any one time while processing a distribution list. This value can be 1 - 255. The default is 16.

Streams=n

specifies the maximum number of files that may be transmitted simultaneously on different streams. This value can be 1 - 7. The default is 1.

Note: RSCS cannot actually send more than one file at a time. When several files are active on different streams, RSCS sends the file with the smallest number of blocks and credits the other files for having to wait. After the file is transmitted, RSCS subtracts the credits from the number of blocks left to send for each active file. RSCS then chooses the file with the lowest number to send. The longer a large file has to wait, the more credits it will accumulate. When the difference between the accumulated credits and the actual file size becomes small enough, RSCS transmits the large file.

PI

TA=

specifies the transmission algorithm, which tells RSCS how to select files for transmission on the link. This can be specified as either of the following:

epname

is the name of the entry point (outside of the RSCS LOADLIB) where your installation-defined algorithm resides.

n

is a hexadecimal value, 0 - F, that identifies the exit routine in the DMTAXA module (DMTAXAGn):

0

Identifies an IBM-supplied algorithm that allows files to be selected regardless of size. This is the default.

Note: If you specify TA=0 or allow it to default, do not specify the TAPARM operand.

1

Identifies an IBM-supplied algorithm that specifies a default size for each stream specified on the STREAMS operand. You can override the default by specifying the stream size on the TAPARM operand.

2 - F

Identifies an algorithm defined by your installation.

Note: If you update the DMTAXA module, it must be reassembled.

TAParm='value'

specifies a parameter value associated with the transmission algorithm. This value is a character string up to 80 bytes in length, enclosed in single quotation marks. (The single quotation marks are optional if you do not include blanks in the character string.) Any characters are allowed between the single quotation marks except another single quotation mark. The character string cannot be continued to the next line in the configuration file.

The value that you specify for TAPARM depends on the value that you specify for TA:

TA=

TAPARM=

0

Do not specify TAPARM; it will be rejected with an error message.

epname

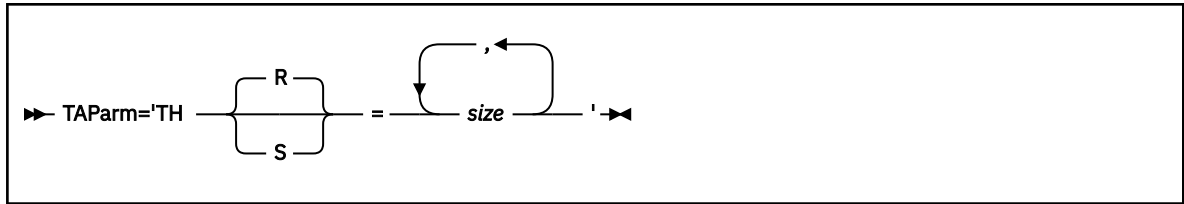
Specify a TAPARM defined by your installation.

2 - F

Specify a TAPARM defined by your installation.

1

Either omit TAPARM to accept the default size for each stream (listed in the table below) or specify a size for each stream using the following format:



TH=

THR=

specifies the threshold number of records. THR is the default.

THS=

specifies the threshold number of spool file blocks. Specify THS when you expect to send files with very large records, such as printer output files, containing page image data, that are destined for a PSF-controlled printer.

size

is the file size. This can be specified as either of the following:

nnnnnnnn

Maximum file size RSCS should send on the stream. The minimum file size is one more than the maximum file size from the previous stream specification. If there is no previous stream specification, there is no minimum file size.

(min,max)

Minimum and maximum file size RSCS should send on the stream. If you do not want to specify a minimum file size, omit *min*. If you do not want to specify a maximum file size, omit *max*. Specifying (,) means you do not want to specify either a minimum or a maximum file size; any file is eligible for transmission on the stream.

For example, if you specified STREAMS=3, you can choose the TAPARM default by omitting TAPARM or you can specify any one of the following:

TAPARM='TH=size1'

TAPARM='TH=size1,size2'

TAPARM='TH=size1,size2,size3'

TAPARM='THR=size1'

TAPARM='THR=size1,size2'

TAPARM='THR=size1,size2,size3'

TAPARM='THS=size1'

TAPARM='THS=size1,size2'

TAPARM='THS=size1,size2,size3'

If you specify TA=1 and omit TAPARM, these are the defaults:

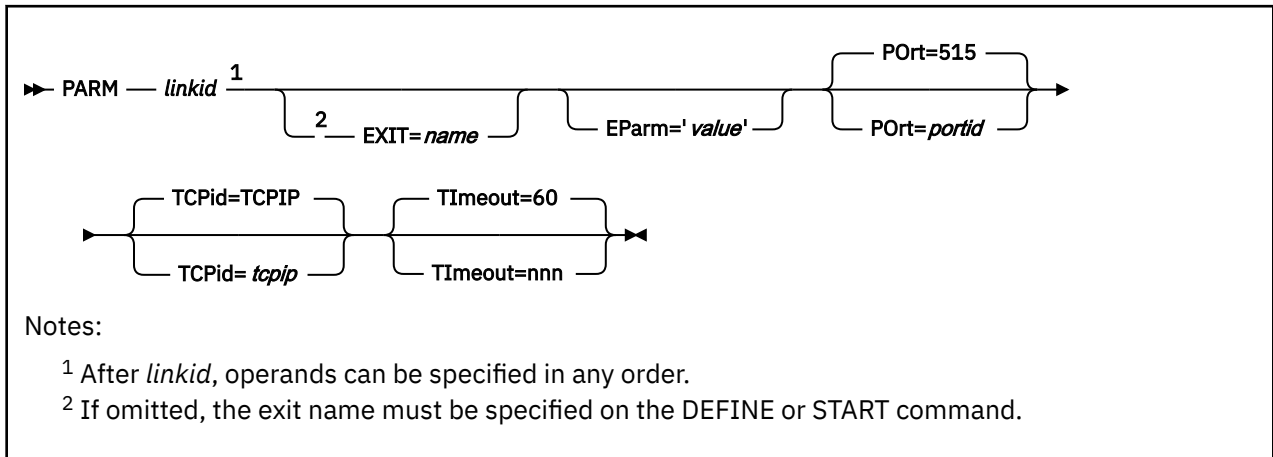
If you specified	THR= Default for Stream 1	THR= Default for Stream 2	THR= Default for Stream 3	THR= Default for Stream 4	THR= Default for Stream 5	THR= Default for Stream 6	THR= Default for Stream 7
STREAMS=1	All files	—	—	—	—	—	—
STREAMS=2	(,100)	(101,)	—	—	—	—	—

PARM (LISTPROC-Type)

If you specified	THR= Default for Stream 1	THR= Default for Stream 2	THR= Default for Stream 3	THR= Default for Stream 4	THR= Default for Stream 5	THR= Default for Stream 6	THR= Default for Stream 7
STREAMS=3	(,100)	(101,)	(101,)	–	–	–	–
STREAMS=4	(,100)	(101,)	(101,)	(101,)	–	–	–
STREAMS=5	(,100)	(101,)	(101,)	(101,)	(101,)	–	–
STREAMS=6	(,100)	(101,)	(101,)	(101,)	(101,)	(101,)	–
STREAMS=7	(,100)	(101,)	(101,)	(101,)	(101,)	(101,)	(101,)

PI end

PARM Statement for an LPD-Type Link



Purpose

This PARM statement defines the operational characteristics for a specific LPD-type link. An LPD-type link enables the local RSCS system to receive print data streams from an LPR client in a TCP/IP network.

The link can be started:

- Automatically during system start-up
- Manually by an operator or authorized user

Attention

Any file queued to this link will be ignored (not processed).

Note: If you do not define a PARM statement for an LPD-type link, all of the indicated defaults will apply to the link. You can override PARM attributes by specifying values on DEFINE or START command. Some required link attributes do not have PARM defaults; if a value is not defined on the PARM statement, it must be specified on the DEFINE or START command, or the link will not activate on the START command. For information on the DEFINE and START commands, see *z/VM: RSCS Networking Operation and Use*.

Operands

linkid

is the name of the LPD-type link, as defined on a LINKDEFINE statement.

PI

EXIT=name

specifies the name of the exit routine module that contains the exit routines supporting this LPD link driver. The exit routine module must reside in a LOADLIB that has been identified by using the GCS GLOBAL command. If you omit this operand and the exit name is not specified on the DEFINE or START command, or if RSCS cannot find the exit routine module, the link driver will not start.

IBM supplies a sample exit routine module called LPDXMANY in the RSCSEXIT LOADLIB. For more information about the LPD exit routines, see *z/VM: RSCS Networking Exit Customization*.

EParm='value'

specifies a parameter value associated with the exit routine module. This value is a character string up to 239 bytes in length, enclosed in single quotation marks. Any character is allowed between the single quotation marks except another single quotation mark. The character string cannot be continued to the next line in the configuration file. The actual value of the character string is defined by the exit routine module.

PARM (LPD-Type)

For information about the EPARM parameters supported for use with the LPDXMANY sample module, see [z/VM: RSCS Networking Exit Customization](#).

PI end

Port=portid

specifies the port number on the local host where RSCS will listen for connect requests. The default is 515.

Note: By default, TCP/IP restricts the use of well-known ports (1 - 1023). To use port 515 or any other well-known port when usage is restricted, the RSCS server must be authorized on a PORT statement in the TCP/IP configuration file (PROFILE TCPIP). For more information, see [z/VM: TCP/IP Planning and Customization](#).

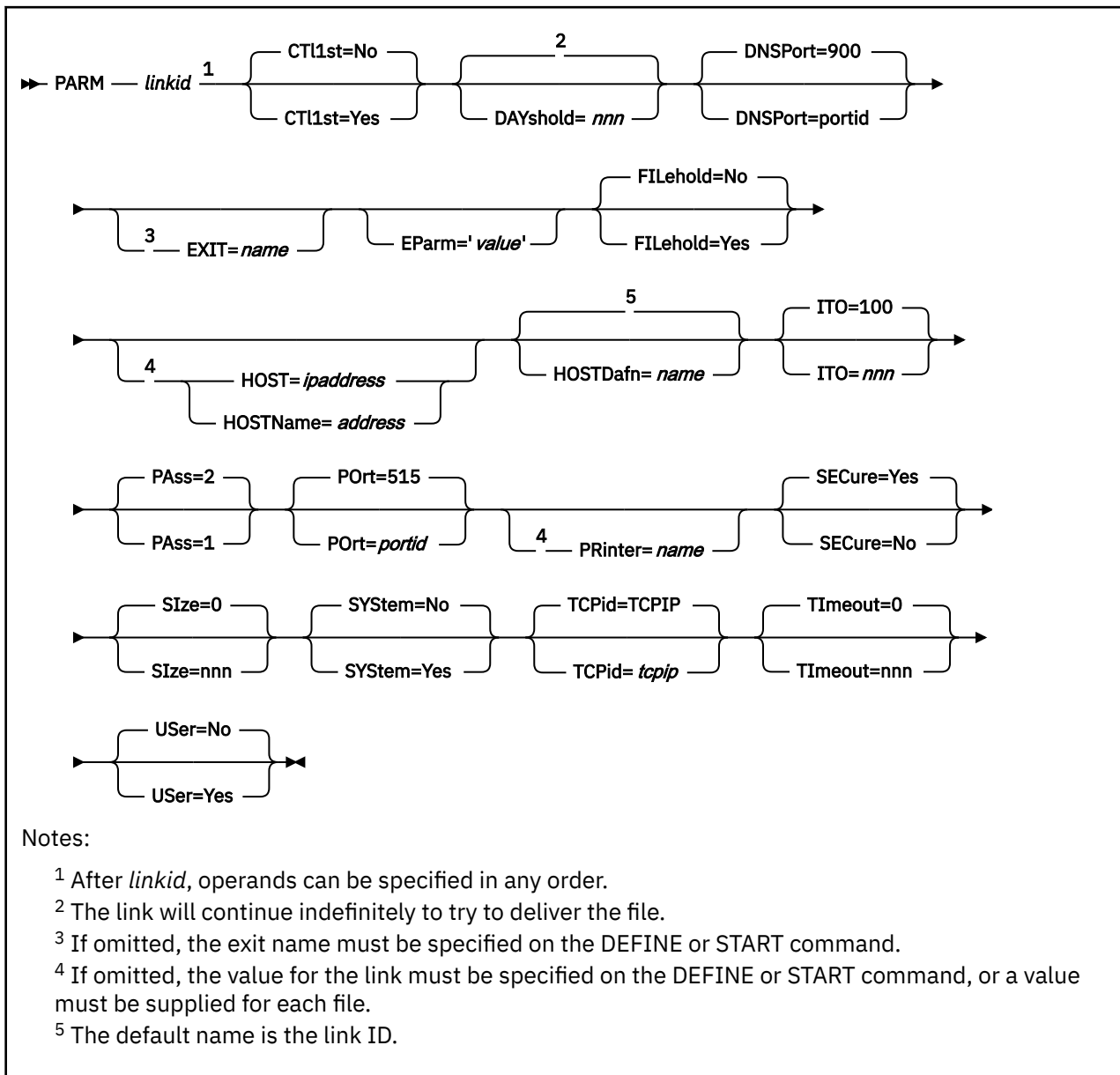
TCPid=tcPIP

specifies the name of the TCP/IP virtual machine. The default name is TCPIP.

Timeout=nnn

specifies the maximum amount of time, in seconds, that RSCS will wait for a response when receiving data from a TCP/IP LPR client prior to breaking the socket connection. The default is 60. After the LPD link driver breaks this connection, it will remain operational waiting for another LPR client to connect.

PARM Statement for an LPR-Type Link



Purpose

This PARM statement defines the operational characteristics for a specific LPR-type link. An LPR-type link enables the local RSCS system to connect to a remote line printer daemon (LPD) in a TCP/IP network.

The link can be started:

- Automatically during system start-up
- Automatically when a file is queued for transmission on an auto-start link
- Manually by an operator or authorized user

Note: If you do not define a PARM statement for an LPR-type link, all of the indicated defaults will apply to the link. You can override PARM attributes by specifying values on the DEFINE or START command. Some required link attributes do not have PARM defaults; if a value is not defined on the PARM statement, it must be specified on the DEFINE or START command, or it must be supplied with the file or by the

LPRXFORM exit. For information on the DEFINE and START commands and the LPRXFORM exit, see [z/VM: RSCS Networking Operation and Use](#).

Operands

linkid

is the name of the link, as defined on a LINKDEFINE statement.

CTI1st=No

CTI1st=Yes

specifies whether the control file will be sent before the data file. The default is NO, which means the data file will be sent first.

DAYshold=nnn

specifies the number of days that an undeliverable file will be placed on hold before it is re-queued to a NOTIFY-type link. This value can be 0 - 365. In order to requeue files to a NOTIFY-type link, a route for *LPRH* must be defined; for example:

```
ROUTE *LPRH* TO *LPRHOLD
```

This operand must be used with an events file entry that periodically changes any files on hold to nohold for the LPR-type link. If you specify a value of 0, the LPR-type link will requeue files to a NOTIFY-type link every time it places them on hold. If you omit this operand, the LPR-type link will continue indefinitely to try to deliver the file.

DNSPort=portid

specifies the port number on the local system where the RSCS domain name server (RSCSDNS), if running, listens for connect requests. The default is 900.

Note: By default, TCP/IP restricts the use of well-known ports (1 - 1023). To use port 900 or any other well-known port when usage is restricted, the RSCSDNS server must be authorized on a PORT statement in the TCP/IP configuration file (PROFILE TCPIP). For more information, see [z/VM: TCP/IP Planning and Customization](#).

PI

EXIT=name

specifies the name of the exit routine module that contains the exit routines supporting this LPR link driver. The exit routine module must reside in a LOADLIB that has been identified by using the GCS GLOBAL command. If you omit this operand and the exit name is not specified on the DEFINE or START command, or if RSCS cannot find the exit routine module, the link driver will not start.

IBM supplies the following sample exit routine modules in the RSCSEXIT LOADLIB:

Module

Supported Device

LPRXONE

Non-PostScript printer.

LPRXPSE

PostScript printer.

For more information about the LPR exit routines, see [z/VM: RSCS Networking Exit Customization](#).

EParm='value'

specifies a parameter value associated with the exit routine module. This value is a character string up to 239 bytes in length, enclosed in single quotation marks. Any character is allowed between the single quotation marks except another single quotation mark. The character string cannot be continued to the next line in the configuration file. The actual value of the character string is defined by the exit routine module.

For information about the EPARM parameters supported for use with the LPRXONE and LPRXPSE sample modules, see [z/VM: RSCS Networking Exit Customization](#).

PI end

FILEhold=No**FILEhold=Yes**

specifies whether the link should place a file on hold when it is unable to connect to a remote daemon after approximately 1 minute. The default is NO, which means the link will continue attempting to connect until successful or the link is stopped (with the STOP command).

If you specify YES, the link will attempt to connect to the remote daemon at 1, 2, 4, 8, 16, and 32 second intervals. If the link is still unable to connect, the file will be placed on hold. The RSCS events file can be used to periodically change any files on hold to nohold, allowing for another delivery attempt. For example, an RSCS events file entry which will change all files from hold to nohold on an LPR-type link named LPR every 10 minutes would look like this:

```
*      M:010      *      *      *      *      CHANGE LPR HOLD NOHOLD
```

HOST=ipaddress

specifies the IP address (in dotted decimal format) of the remote host to which to connect. This value may be overridden for individual files by the LPRXFORM exit (if SYSTEM=YES is specified) or passed by a user with the file (if USER=YES is specified). If a HOST or HOSTNAME value for the link is not defined on the PARM statement, it must be specified on the DEFINE or START command, or a value must be supplied for each file.

HOSTName=address

specifies the fully qualified domain name address (up to 200 characters) of the remote host to which to connect. This value may be overridden for individual files by the LPRXFORM exit (if SYSTEM=YES is specified) or passed by a user with the file (if USER=YES is specified). When overridden, the host name address can be up to 255 characters. If a HOST or HOSTNAME value for the link is not defined on the PARM statement, it must be specified on the DEFINE or START command, or a value must be supplied for each file.

Note: Use of this operand requires the RSCS domain name server, which will resolve the domain name address to an IP address.

HOSTDafn=name

specifies the 1- to 8-character name to be used as the host name portion of the control and data file names. A control or data file name is used within the **Receive control file** and **Receive data file** subcommands of the daemon's **Receive job** command. The name should start with cfa (control file) or dfa (data file), followed by a 3-digit job number and the name of the host that has constructed the control or data file.

If you omit this operand, RSCS uses the link ID for the host name portion of the file names. Certain daemons do not tolerate a name which does not match an existing defined host name. This operand allows you to supply a defined name instead of using the link ID.

ITO=nnn

specifies the inactivity time out (ITO) value, which defines the length of time that can occur with no activity (that is, no file transmission or reception) before the link is deactivated. If a link is ended because its ITO value has been exceeded, its restart exec will not be processed.

The ITO value can be 0 - 100:

0

The link will be deactivated immediately when there is no activity on it. However, before deactivating the link, RSCS will first send files that are queued for transmission, or finish reception of the file currently being received.

1 - 99

Approximate number of minutes until the link will be deactivated if there has been no activity on it.

100

No ITO will be in effect. The link will stay active until it is deactivated manually or an unrecoverable error occurs. This is the default.

PAss=

specifies the number of passes that RSCS will make through a spool file when sending it on the link.

2

RSCS will make two passes through the file. This is the default. The TAG, Record, and EOF file exits will be called to determine the byte count of the file. This value is required by most host systems to which the file will be sent. A second pass will then be made to send the file itself. The byte count determined for the file must match for each pass made over the file.

1

RSCS will send a byte count of 0 to the host and then make only one pass through the spool file as it is sent. No binary zero characters must be sent, as this will indicate the end of the file.

POrt=portid

specifies the port number to which RSCS connects on the remote host. The default is 515.

Note: By default, TCP/IP restricts the use of well-known ports (1 - 1023). To use port 515 or any other well-known port when usage is restricted, the RSCS server must be authorized on a PORT statement in the TCP/IP configuration file (PROFILE TCPIP). For more information, see [z/VM: TCP/IP Planning and Customization](#).

PRinter=name

specifies the default printer queue name. If you omit this operand, the printer name for the link must be specified on the DEFINE or START command, or a value must be supplied for each file.

The default value may be overridden for individual files by the LPRXFORM exit (if SYSTEM=YES is specified) or passed by a user with the file (if USER=YES is specified).

Note: Depending on the line printer daemon, the printer name may be case-sensitive. You can use mixed-case letters to specify the printer name in the RSCS CONFIG file. However, CP converts to uppercase all data that is entered from the RSCS console or sent on the SMSG command. To ensure that the correct letters are used, you can create an exec to specify a case-sensitive printer name to RSCS on the SMSG command. Issue the following EXECIO command:

```
EXECIO * CP (STEM CPMSG. STRING SMSG rscsid options)
```

where:

rscsid

is the CP name of the RSCS virtual machine.

options

is the uppercase RSCS command and any options that might need to be in mixed case, such as the printer name.

For more information about EXECIO, see [z/VM: CMS Commands and Utilities Reference](#).

SECure=Yes**SECure=No**

specifies whether only local ports 721 - 731 are to be used. The default is YES.

Note: By default, TCP/IP restricts the use of well-known ports (1 - 1023). To use ports 721 - 731 when usage is restricted, the RSCS server must be authorized on a PORT statement in the TCP/IP configuration file (PROFILE TCPIP). For more information, see [z/VM: TCP/IP Planning and Customization](#).

If you specify NO, any available stack-selected port will be used. SECURE=NO is the recommended setting to avoid the TCP/IP port restrictions.

SIze=nnn

specifies the maximum number of records the spool file may contain in order to be processed. This value can be 0 - 1048576. The default is 0, which means there is no limit. If the number of records exceeds the maximum allowed amount, an error message will be issued to the user and the console, and the file will be placed on hold.

SYStem=No**SYStem=Yes**

specifies whether RSCS should search for a FORM table provided by the LPRXFORM exit, using the form name the file was spooled with. The default is NO. When sending a print file to a remote line

printer daemon, the FORM table can provide keywords to define the printer queue name, remote host address, translate table, filter, substitute form, prefix string, separator page setting, and suffix string to be used for the file, overriding the values defined for the link.

TCPid=*tcpip*

specifies the name of the TCP/IP virtual machine. The default name is TCPIP.

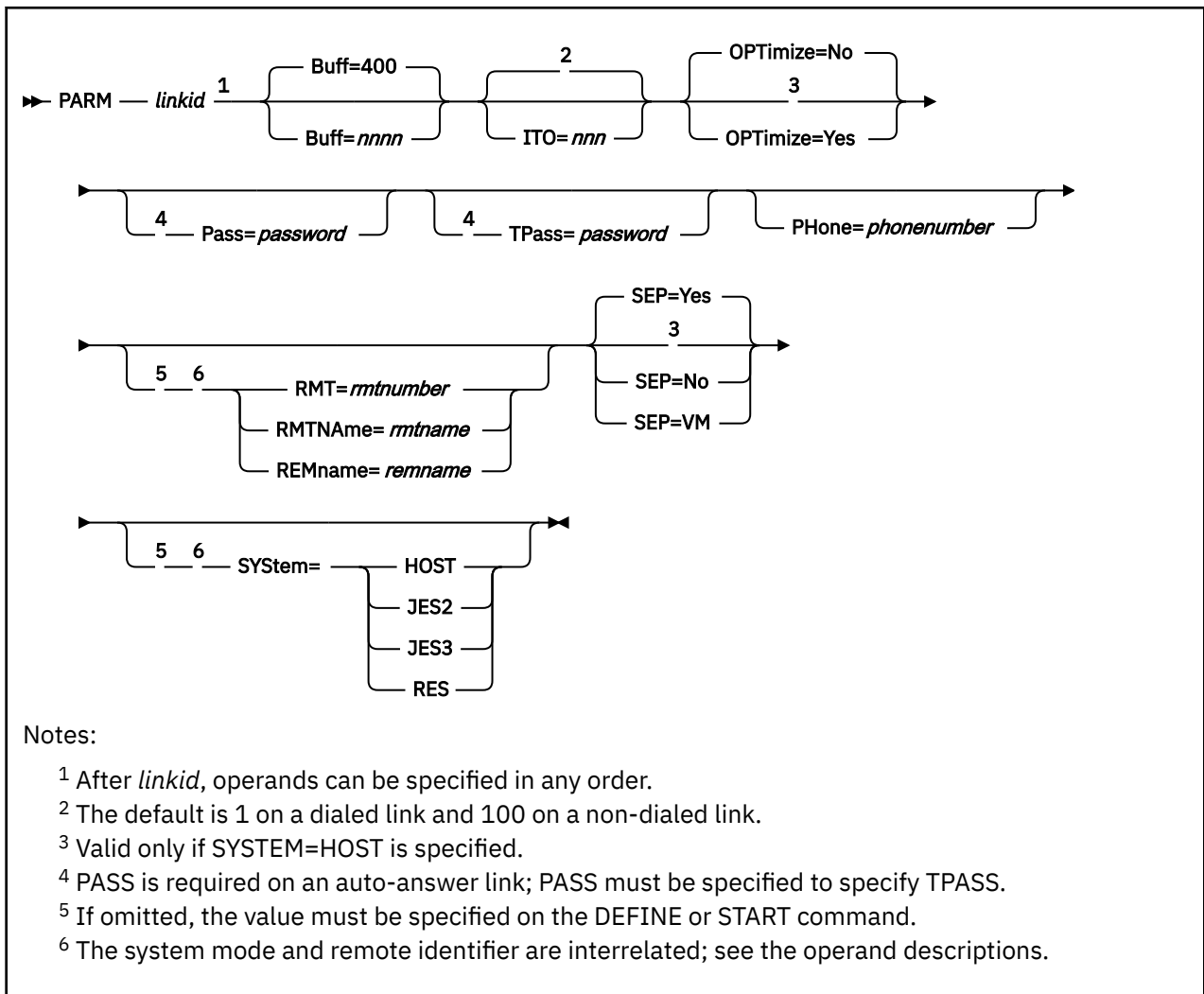
TImeout=*nnn*

specifies the maximum amount of time, in seconds, to wait for a response from the remote host before halting delivery of the file and placing it on hold. The default is 0, which means the link will wait indefinitely.

USer=No**USer=Yes**

specifies whether RSCS will allow keywords to be defined and looked for at the beginning of a print file. The default is NO. When sending a print file to a remote line printer daemon, these keywords can define the printer queue name, remote host address, translate table, filter, prefix string, separator page setting, and suffix string to be used for the file, overriding the values defined for the link.

PARM Statement for an MRJE-Type Link



Purpose

This PARM statement defines the operational characteristics for a specific MRJE-type link. An MRJE-type link is a workstation link. It can connect a system node to:

- (In `HOST` mode) a remote workstation node
- (Acting as a workstation in `JES2`, `JES3`, or `RES` mode) another system that is acting as the host

The link can be started:

- Automatically during system start-up
- Automatically when a file is queued for transmission on an auto-start link
- Automatically when a call is received over an enabled (auto-answer) switched telecommunication line
- Manually by an operator or authorized user

For other than a link on an auto-answer line, the host side of the link must be started first, before a remote station can sign on.

Note: If you do not define a PARM statement for an MRJE-type link, all of the indicated defaults will apply to the link. You can override PARM attributes by specifying values on the `DEFINE` or `START` command. Some required link attributes do not have PARM defaults; if a value is not defined on the PARM statement, it must be specified on the `DEFINE` or `START` command, or the link will not activate on the `START`

command. For information on the DEFINE and START commands, see [z/VM: RSCS Networking Operation and Use](#).

Operands

linkid

is the name of the MRJE-type link, as defined on a LINKDEFINE statement.

Buff=nnnn

specifies the size of the telecommunication buffer. This value can be 300 - 1017 bytes. The default is 400.

ITO=nnn

specifies the inactivity time out (ITO) value, which defines the length of time that can occur with no activity (that is, no file transmission or reception) before the link is deactivated. If a link is ended because its ITO value has been exceeded, its restart exec will not be processed.

The ITO value can be 0 - 100:

0

The link will be deactivated immediately when there is no activity on it. However, before deactivating the link, RSCS will first send files that are queued for transmission, or finish reception of the file currently being received.

1 - 99

Approximate number of minutes until the link will be deactivated if there has been no activity on it.

100

No ITO will be in effect. The link will stay active until it is deactivated manually or an unrecoverable error occurs.

If you omit this operand, the default is:

- 1 (one minute) for a dialed link
- 100 (no ITO in effect) for a link that is not dialed

OPTimize=No

OPTimize=Yes

specifies whether optimization (merging of CCWs) should be used for print files being transmitted over this workstation link. The default is NO.

Specify NO, or allow it to default, when using a workstation link to communicate with a device that will use specialized file data such as page formatting information, or that requires preservation of original carriage control information.

This operand is valid only if SYSTEM=HOST is also specified; otherwise, it will be ignored regardless of the setting.

Pass=password

specifies an identification that validates the start of a session. You must specify the PASS operand to use the TPASS operand.

If this end of the link is acting as a workstation (SYSTEM=JES2, SYSTEM=JES3, or SYSTEM=RES), the PASS identification is passed along to the remote system as part of the sign-on record:

- If SYSTEM=JES2 or SYSTEM=JES3, the identification is a 1- to 8-character line password.
- If SYSTEM=RES, the identification consists of a 1- to 8-character user ID, followed by a slash and a 1- to 8-character password (for a maximum of 17 characters).

If this end of the link is acting as a host (SYSTEM=HOST), the PASS identification is a 1- to 8-character password that the remote station must supply on its sign-on card to connect to RSCS.

For an auto-answer link (automatically started when a call is received on an enabled port), SYSTEM=HOST is forced, PASS is mandatory, and RMT and RMTNAME are ignored.

TPass=password

specifies an identification that can be used with the PASS value to validate the start of a session. To use the TPASS operand, you must also specify the PASS operand.

If this end of the link is acting as a workstation (SYSTEM=JES2 or SYSTEM=JES3), the TPASS identification is passed along to the remote system as part of the sign-on record:

- If SYSTEM=JES2, the identification is a 1- to 8-character password starting in column 73.
- If SYSTEM=JES3, the identification is a 1- to 8-character password starting in column 35.

If SYSTEM=RES, the TPASS operand is ignored.

If this end of the link is acting as a host (SYSTEM=HOST), the TPASS identification is a 1- to 8-character password that the remote station must supply on its sign-on card (starting in column 35 or 73) to connect to RSCS.

PHone=phonenumber

specifies the phone number to be used when starting an auto-dial link. This value is a string up to 32 characters long, optionally enclosed within single quotation marks (' '). Only the following characters may be used:

0 1 2 3 4 5 6 7 8 9

Decimal numerals are used for the actual dial digits of the phone number. For example:

```
PHONE=5551234
```

(blank)

Blanks make the phone number easier to read. They are not transmitted to the Automatic Calling Unit (ACU). If you use blanks, you must enclose the phone number within single quotation marks. For example:

```
PHONE='1 607 555 1234'
```

—

Hyphens also make the phone number easier to read. They are not transmitted to the ACU. For example:

```
PHONE='1-607 555-1234'
```

*

The asterisk is a separator character you can include following an access digit. For a private automatic branch exchange (PABX) extension, this character delays dialing to allow the dial tone to be returned from the public exchange. For example:

```
PHONE='9*1-607 555-1234'
```

.

The period is an end-of-number character that marks the end of the dial-digit group for those ACUs that require such an end character. The type of ACU in use determines whether you need an end-of-number character. For example:

```
PHONE='9*1-607 555-1234.'
```

RMT=rmtnumber

specifies the number that RSCS uses to generate the name of the workstation. This operand must be specified if SYSTEM=RES is specified. It may be specified if SYSTEM=JES2 or SYSTEM=HOST is specified. It is not valid if SYSTEM=JES3 is specified.

If SYSTEM=JES2 or SYSTEM=RES is specified, the generated name is placed on the sign-on record that will be sent to the host system. If SYSTEM=JES2, the range for *rmtnumber* is 1 - 999, and the generated name is REMOTE*nnn*. If SYSTEM=RES, the range for *rmtnumber* is 1 - 99, and the generated name is REMOTE*nn*. A leading zero is significant; for instance, RMT=01 produces a remote name of

REMOTE01, while RMT=1 produces a remote name of REMOTE1. This name must correspond to the name that was defined at the remote system for the RSCS system identifier.

If SYSTEM=HOST is specified, the generated name is the name of the remote workstation that RSCS expects to sign on. A leading zero is not significant; a remote workstation named either REMOTE1 or REMOTE01 could sign on to an RSCS system where either RMT=01 or RMT=1 was specified. If neither RMT, RMTNAME, nor REMNAME is specified for a HOST system, the workstation identifier on the sign-on record must be the remote workstation's node name.

For an auto-answer link (automatically started when a call is received on an enabled port), the RMT operand is ignored.

RMTName=rmtname

specifies the 5-character alphanumeric remote workstation name that was assigned to RSCS when the remote system was generated. This operand must be specified if SYSTEM=JES3 is specified. It may be specified if SYSTEM=JES2 or SYSTEM=HOST is specified. It is not valid if SYSTEM=RES is specified.

For an auto-answer link (automatically started when a call is received on an enabled port), the RMTNAME operand is ignored.

REMname=remname

specifies a 2- to 9-character alphanumeric name that is acceptable to JES2 as a remote identifier. This operand may be specified if SYSTEM=JES2 or SYSTEM=HOST is specified. It is not valid if SYSTEM=JES3 or SYSTEM=RES is specified.

SEP=

specifies the type of separator page that will be inserted before each print file sent across the link. This operand is valid only if SYSTEM=HOST is specified.

Yes

RSCS-style separator page. This is the default.

No

No separator page.

VM

VM-style separator page.

For information about separator page formats, see *z/VM: RSCS Networking Operation and Use*.

Note: This operand applies only to print files. When transmitting a punch file to a remote workstation, RSCS creates a header line on the output. For information about the format of this header line, see *z/VM: RSCS Networking Operation and Use*.

SYStem=

specifies the operation mode of the link.

HOST

Operates as a host to a remote workstation signing on either as REMOTE*rmtnumber* or with the remote workstation's node name.

JES2

Operates as a workstation into a remote JES2 system, signing on with the defined remote name (REMOTE*rmtnumber*, *rmtname*, or *remname*).

JES3

Operates as a workstation into a remote JES3 system, signing on as remote *rmtname*.

RES

Operates as a workstation into the Remote Entry Subsystem (RES) of OS/VS1, signing on as REMOTE*rmtnumber*.

For an auto-answer link (automatically started when a call is received on an enabled port), SYSTEM=HOST is forced.

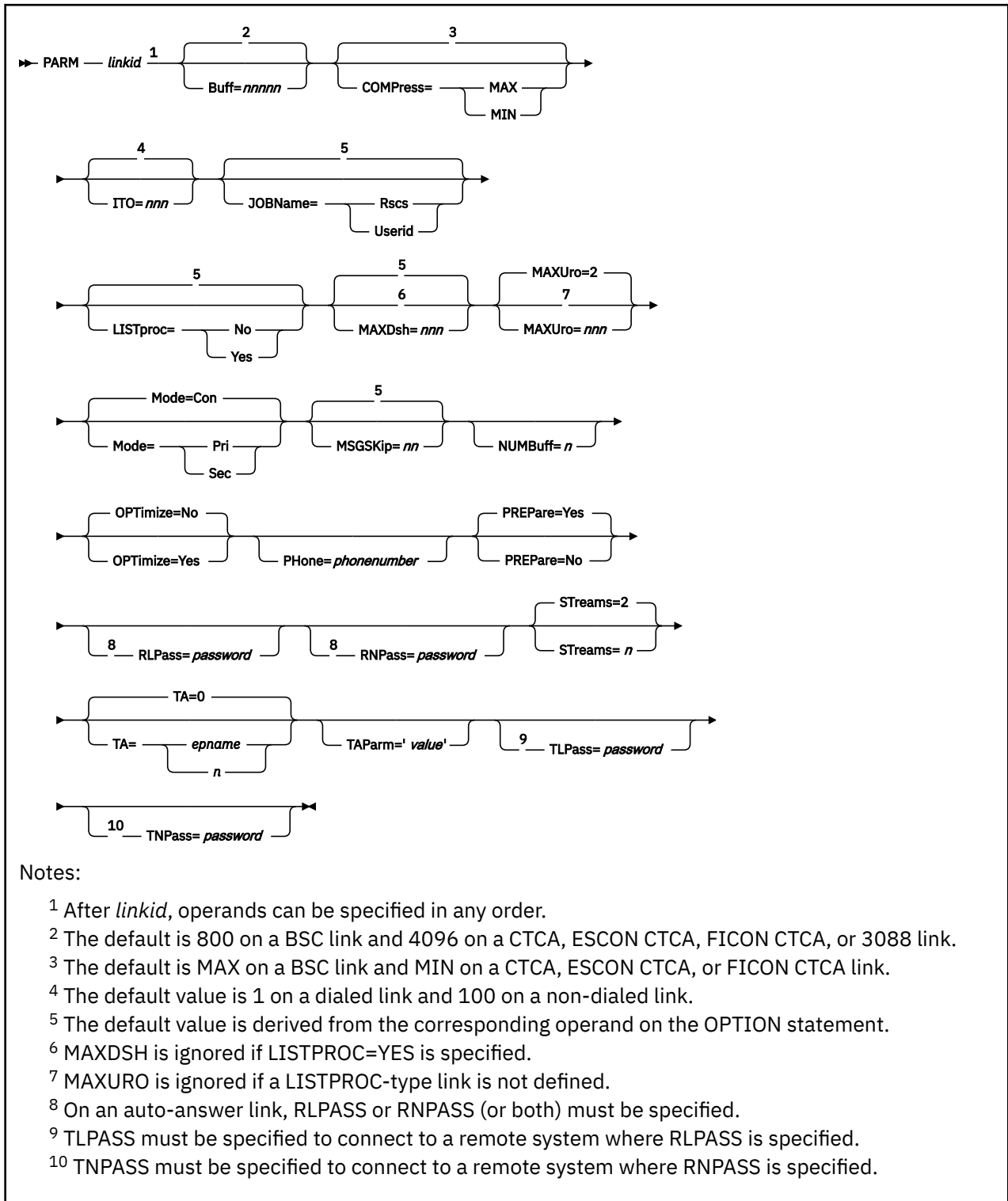
Usage Notes

When you define a form name (using the FORM statement), you can define these characteristics:

- Separator page style
- Line length
- Page length
- Lines per inch (only for LU_T1 printers and workstations)

If you specify a form name when starting a link, be aware that the form name characteristics always override any SEP, LLINE, and LPAGE specifications and will override LPI specifications when SETPAGE=YES.

PARM Statement for an NJE-Type Link



Purpose

This PARM statement defines the operational characteristics for a specific NJE-type link. An NJE-type link is a networking link. It manages BSC, CTCA, ESCON CTCA, FICON CTCA, or 3088 communications between z/VM and either another z/VM system or a non-z/VM system.

PARM (NJE-Type)

The link can be started:

- Automatically during system start-up
- Automatically when a file is queued for transmission on an auto-start link
- Automatically when a call is received over an enabled (auto-answer) switched telecommunication line
- Manually by an operator or authorized user

Note: If you do not define a PARM statement for an NJE-type link, all of the indicated defaults will apply to the link. You can override PARM attributes by specifying values on the DEFINE or START command. For information on the DEFINE and START commands, see [z/VM: RSCS Networking Operation and Use](#).

Operands

linkid

is the name of the NJE-type link, as defined on a LINKDEFINE statement.

Buff=nnnnn

specifies the maximum buffer size this link can accept from the communications adapter. This value can be 300 - 32765 bytes. The default is 800 on a BSC link and 4096 on a CTCA, ESCON or FICON CTCA, or 3088 link. The actual buffer size used for other than a JES3 system is determined at sign-on time and is the smaller of the buffer sizes specified (or defaulted to) by the two connected systems. If a small buffer size is specified (or defaulted to) by the remote node, it should be encouraged to specify a larger buffer.

The value for BUFF must be selected based on the type of data link. If the link is over a BSC line, the baud rate must also be considered. Usually, the largest buffer sizes can be used only on CTCA, ESCON CTCA, and FICON CTCA links. When starting a link to a JES3 system, both sides must specify the same buffer size.

Note: Other systems may have software limitations that determine the largest acceptable buffer size. The maximum valid buffer size can be used only if RSCS is connected to an RSCS Version 2 or later system.

COMPRESS=

specifies the degree of data compression to be used when transmitting the file.

MAX

Maximum data compression. This is the default on a BSC link.

MIN

Minimum data compression. This is the default on a CTCA, ESCON CTCA, or FICON CTCA link.

ITO=nnn

specifies the inactivity time out (ITO) value, which defines the length of time that can occur with no activity (that is, no file transmission or reception) before the link is deactivated. If a link is ended because its ITO value has been exceeded, its restart exec will not be processed.

The ITO value can be 0 - 100:

0

The link will be deactivated immediately when there is no activity on it. If ITO=0 is specified for the local and remote nodes of the link, and files are queued on one end of the link, the link will deactivate before the files are sent. If both sides of the link have files to process, the files will be sent.

1 - 99

Approximate number of minutes until the link will be deactivated if there has been no activity on it.

100

No ITO will be in effect. The link will stay active until it is deactivated manually or an unrecoverable error occurs.

If you omit this operand, the default is:

- 1 (one minute) for a dialed link

- 100 (no ITO in effect) for a link that is not dialed

JOBName=

specifies the type of job name RSCS will create for all files originating at your local node and all files arriving at your local node without a job header (that is, those coming from a workstation link).

Rscs

RSCS will generate a job name of RSCSnnnn, where nnnn is the origin spool ID of the file.

Userid

RSCS will use the origin user ID as the job name.

If you specify this operand, it will override any previously defined default for this link. If you omit this operand, the default value is derived from the corresponding operand on the OPTION statement.

To override the default job name for a specific file, use the JOBNAME operand on the CP TAG command. For more information, see *z/VM: RSCS Networking Operation and Use*.

LISTproc=No

LISTproc=Yes

specifies whether the connected system can handle an unlimited number of data set headers. If you specify NO, the link will use the number of data set headers specified on the MAXDSH operand.

If you specify this operand, it will override any previously defined default for this link. If you omit this operand, the default value is derived from the corresponding operand on the OPTION statement.

The following table summarizes how non-list processor systems will handle list processor files.

Table 18. How Non-List Processor Systems Handle List Processor Files

Non-List Processor Systems	How the File Will Be Handled
<ul style="list-style-type: none"> • RSCS V1.3* • JES3 • VSE/POWER 	<p>These systems will immediately split the file into separate spool files. These files will be either distributed to local user IDs or sent on to other links.</p>
<ul style="list-style-type: none"> • RSCS V2.2* (without list processor support • RSCS V2.3* or later* (without a list processor defined) • JES2 	<p>These systems will receive the file and split out separate files for local user IDs. They will also split out one file containing data set headers to be sent to each link on the distribution list.</p>

Note: * These systems will handle only a limited number of data set headers.

MAXDsh=nnn

specifies the maximum number of data set headers the link driver will send on the link when processing a file with multiple data sets. This value is intended for connections to NJE subsystems that do not have list processor capabilities. This operand is ignored if LISTPROC=YES is specified.

This value can be 0 - 100. A value of 0 is a special case that means the determination for the number of data set headers placed on the link is based on the actual number of data set headers contained in the spool file. A value of 0 should be specified only with full understanding of the number of data set headers that the remote system can handle.

If you specify this operand, it will override any previously defined default for this link. If you omit this operand, the default value is derived from the corresponding operand on the OPTION statement.

MAXUro=nnn

specifies the maximum number of unit record output devices (virtual printers or punches) for each stream that the link driver will use at any one time while receiving a file. RSCS uses these devices to fan out files containing data set headers, which the link driver determines must be split off the original file. All remaining data set headers (and the file contents) are written to an additional unit record

PARM (NJE-Type)

device and queued on the list processor link. This operand is ignored if a LISTPROC-type link is not defined.

This value can be 0 - 255. The default is 2. Specifying a value of 0 will cause all received data set headers and file contents to be queued to the list processor task. Under standard conditions, a link will always use one more device per stream than specified. If a value of 0 is specified and the first data set header received in a file is for the local system or for a non-networking link attached to the local system, an extra device will be defined to hold the remaining data set headers.

Mode=

specifies the mode of the local RSCS system in a BSC connection. This operand has no meaning for a CTCA, ESCON CTCA, FICON CTCA, or 3088 connection.

CON

Specifies contention mode. This is the default. This option is intended for nonswitched line connections, where both sides of the connection can act as the primary when the initial connection is being made. A contention situation will exist, with the loser becoming the secondary.

PRI

Identifies RSCS as the primary system and transmits a Type I sign-on response record to establish communications. If PRI is specified for both sides of a connection, the system will not connect.

SEC

Identifies RSCS as the secondary system and transmits a Type J sign-on response record to establish communication. If SEC is specified for both sides of a connection, the system will not connect.

For best results, if the remote system is:

- RSCS V1 system (which is always primary), specify MODE=SEC
- RSCS V2 or later auto-answer system (which is automatically started with MODE=SEC when it receives the call on an enabled port), specify MODE=PRI
- JES2 system started by a \$S LNE*n* command, specify either MODE=PRI or MODE=CON
- JES2 system started by a \$SN LNE*n* command, specify either MODE=SEC or MODE=CON
- JES3 or VSE/POWER system (which can be either primary or secondary, but cannot be forced into a particular mode), specify any of the three modes

MSGSKIP=*nn*

specifies the number of file buffers that RSCS will send on the link before sending a message buffer (unless there are no file buffers available). This value can be 0 - 15. Specifying 0 is a special case that tells RSCS to send a message buffer whenever messages are available to send on the link, regardless of whether file buffers are available.

If you specify this operand, it will override any previously defined default for this link. If you omit this operand, the default value is derived from the corresponding operand on the OPTION statement.

Note: If this link determines during sign-on that the peer system supports the mixed RCB feature of NJE, RSCS ignores the MSGSKIP setting. The mixed RCB feature lets RSCS send buffers containing a mixture of information (files and messages). If RSCS can send files and messages in one buffer, there is no need to use the MSGSKIP operand to tell RSCS how many file buffers to send before sending a message buffer. For more information about the mixed RCB feature of NJE, see [z/OS: Network Job Entry \(NJE\) Formats and Protocols \(https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa320988/\\$file/hasa600_v2r5.pdf\)](https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa320988/$file/hasa600_v2r5.pdf).

NUMBuff=*n*

has no meaning and exists only to enable the current level of RSCS to maintain compatibility with RSCS Version 2. Whether you specify or omit this operand, RSCS gives each link two transmission buffers.

OPTimize=No**OPTimize=Yes**

specifies whether optimization (merging of CCWs) should be used for files that originate on the local system and files that are processed by the local system. (This does not include store-and-forward files that are passing through the local node.) The default is NO.

PHone=phonenumber

specifies the phone number to be used when starting an auto-dial link. This string can be up to 32 characters long, optionally enclosed within single quotation marks (' '). Only the following characters may be used:

0 1 2 3 4 5 6 7 8 9

Decimal numerals are used for the actual dial digits of the phone number. For example:

```
PHONE=5551234
```

(blank)

Blanks make the phone number easier to read. They are not transmitted to the Automatic Calling Unit (ACU). If you use blanks, you must enclose the phone number within single quotation marks. For example:

```
PHONE='1 607 555 1234'
```

-

Hyphens also make the phone number easier to read. They are not transmitted to the ACU. For example:

```
PHONE='1-607 555-1234'
```

*

The asterisk is a separator character you can include following an access digit. For a private automatic branch exchange (PABX) extension, this character delays dialing to allow the dial tone to be returned from the public exchange. For example:

```
PHONE='9*1-607 555-1234'
```

.

The period is an end-of-number character that marks the end of the dial-digit group for those ACUs that require such an end character. The type of ACU in use determines whether you need an end-of-number character. For example:

```
PHONE='9*1-607 555-1234.'
```

PREPare=

specifies whether the NJE PREPARE protocol should be used between two RSCS nodes during idle link time. The PREPARE protocol defines a method by which non-SNA links do not have to send null buffers to each other every one or two seconds.

Yes

PREPARE sequences will be carried out if the opposite node agrees through sign-on concurrence. This is the default.

Note:

1. The remote node must have RSCS V2.2 or later installed.
2. Using PREPARE sequences reduces needless I/O. However, it is recommended that you not use PREPARE if you meet error recovery problems when using the 3088 Channel Unit or the 3725, 3735, or 3745 Transmission Control Units.

No

PREPARE sequences will not be used. Each node will send null buffers every one or two seconds.

RLPass=password

specifies the 1- to 8-character line password that is required from the remote system on its sign-on record before transmission of data is initiated. If the correct password is not received at sign-on time, RSCS deactivates the link. On an auto-answer link, RLPASS or RNPASS (or both) must be defined.

RNPass=password

specifies the 1- to 8-character node password that is required from the remote system on its sign-on record before transmission of data is initiated. If the correct password is not received at sign-on time, RSCS deactivates the link. On an auto-answer link, RLPASS or RNPASS (or both) must be defined.

STreams=n

specifies the maximum number of files that may be transmitted simultaneously on different streams. This value can be 1 - 7. The default is 2.

Note:

1. RSCS cannot actually send more than one file at a time. When you have several files active on different streams, RSCS sends the file with the smallest number of blocks and credits the other files for having to wait. After the file is transmitted, RSCS subtracts the credits from the number of blocks left to send for each active file. RSCS then chooses the file with the lowest number to send. The longer a large file has to wait, the more credits it will accumulate. When the difference between the accumulated credits and the actual file size becomes small enough, RSCS transmits the large file.
2. To allow the link to operate in a mode compatible with RSCS V1 (that is, only one file will be transmitted at a time and all files are eligible for selection), specify a stream number of 1 and a transmission algorithm number of 0 (TA=0, which is the default).

PI

TA=

specifies the transmission algorithm, which tells RSCS how to select files for transmission on the link.

epname

is the name of the entry point (outside of the RSCS LOADLIB) where your installation-defined algorithm resides.

n

is a hexadecimal value, 0 - F, that identifies the exit routine in the DMTAXA module (DMTAXAGn):

0

Identifies an IBM-supplied algorithm that allows files to be selected regardless of size. This is the default.

Note: If you specify TA=0 or allow it to default, do not specify the TAPARM operand.

1

Identifies an IBM-supplied algorithm that specifies a default size for each stream specified on the STREAMS operand. You can override the default by specifying the stream size on the TAPARM operand.

2 - F

Identifies an algorithm defined by your installation.

Note: If you update the DMTAXA module, it must be reassembled.

TAParm='value'

specifies a parameter value associated with the transmission algorithm. This value is a character string up to 80 bytes in length, enclosed in single quotation marks. (The single quotation marks are optional if you do not include blanks in the character string.) Any characters are allowed between the single quotation marks except another single quotation mark. The character string cannot be continued to the next line in the configuration file.

The value that you specify for TAPARM depends on the value that you specify for TA:

TA=

TAPARM=

0

Do not specify TAPARM; it will be rejected with an error message.

epname

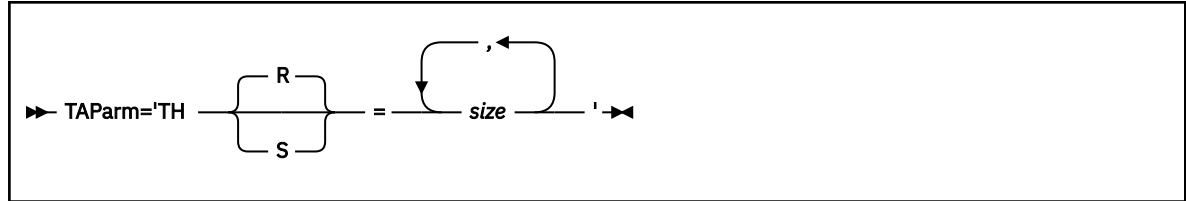
Specify a TAPARM defined by your installation.

2 - F

Specify a TAPARM defined by your installation.

1

Either omit TAPARM to accept the default size for each stream (listed in the table below) or specify a size for each stream using the following format:



TH=

THR=

specifies the threshold number of records. THR is the default.

THS=

specifies the threshold number of spool file blocks. Specify THS when you expect to send files with very large records, such as printer output files, containing page image data, that are destined for a PSF-controlled printer.

size

is the file size. This can be specified as either of the following:

nnnnnnnn

Maximum file size RSCS should send on the stream. The minimum file size is one more than the maximum file size from the previous stream specification. If there is no previous stream specification, there is no minimum file size.

(min,max)

Minimum and maximum file size RSCS should send on the stream. If you do not want to specify a minimum file size, omit *min*. If you do not want to specify a maximum file size, omit *max*. Specifying (,) means you do not want to specify either a minimum or a maximum file size; any file is eligible for transmission on the stream.

For example, if you specified STREAMS=3, you can choose the TAPARM default by omitting TAPARM or you can specify any one of the following:

TAPARM='TH=*size1*'

TAPARM='TH=*size1,size2*'

TAPARM='TH=*size1,size2,size3*'

TAPARM='THR=*size1*'

TAPARM='THR=*size1,size2*'

TAPARM='THR=*size1,size2,size3*'

TAPARM='THS=*size1*'

TAPARM='THS=*size1,size2*'

TAPARM='THS=*size1,size2,size3*'

If TA=1 is specified and TAPARM is omitted, these are the defaults:

PARM (NJE-Type)

If you specified	THR= Default for Stream 1	THR= Default for Stream 2	THR= Default for Stream 3	THR= Default for Stream 4	THR= Default for Stream 5	THR= Default for Stream 6	THR= Default for Stream 7
STREAMS=1	All files	—	—	—	—	—	—
STREAMS=2	(,100)	(101,)	—	—	—	—	—
STREAMS=3	(,100)	(101,)	(101,)	—	—	—	—
STREAMS=4	(,100)	(101,)	(101,)	(101,)	—	—	—
STREAMS=5	(,100)	(101,)	(101,)	(101,)	(101,)	—	—
STREAMS=6	(,100)	(101,)	(101,)	(101,)	(101,)	(101,)	—
STREAMS=7	(,100)	(101,)	(101,)	(101,)	(101,)	(101,)	(101,)

PI end

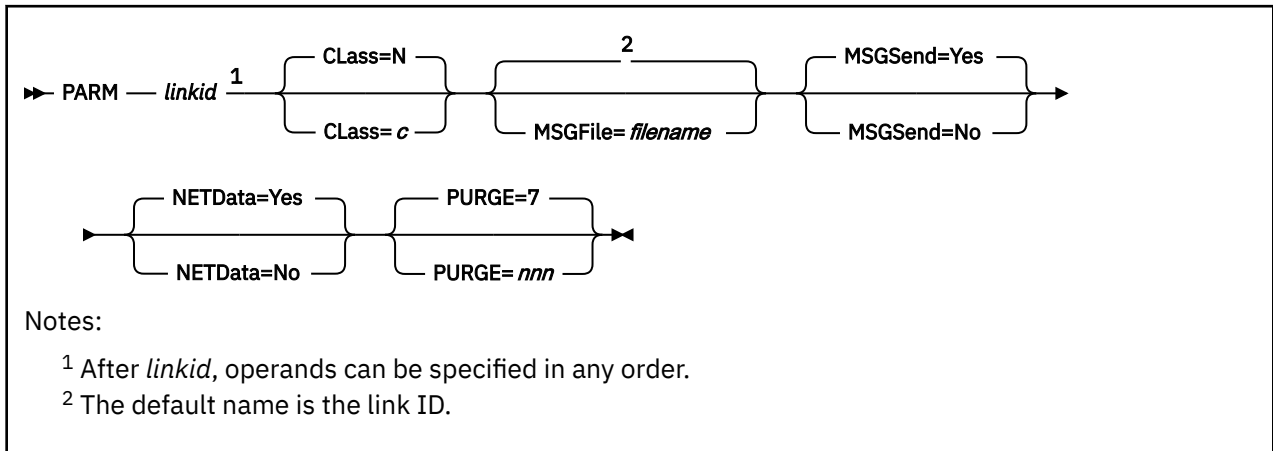
TLPass=password

specifies the 1- to 8-character line password to be transmitted to the remote system at sign-on time. TLPASS must be specified to connect to a remote system where RLPASS is defined.

TNPass=password

specifies the 1- to 8-character node password to be transmitted to the remote system at sign-on time. TNPASS must be specified to connect to a remote system where RNPASS is defined.

PARM Statement for a NOTIFY-Type Link



Purpose

This PARM statement defines the operational characteristics for a specific NOTIFY-type link. A NOTIFY-type link holds files, generates notes, and purges files. For example, a NOTIFY-type link can be used to handle misdirected files. A misdirected file is a file routed to an incorrect user ID or incorrect node.

The link can be started:

- Automatically during system start-up
- Automatically when a file is queued for transmission on an auto-start link
- Manually by an operator or authorized user

Note: If you do not define a PARM statement for a NOTIFY-type link, all of the indicated defaults will apply to the link. You can override PARM attributes by specifying values on the DEFINE or START command. For information on the DEFINE and START commands, see [z/VM: RSCS Networking Operation and Use](#).

Operands

linkid

is the name of the NOTIFY-type link, as defined on a LINKDEFINE statement.

Class=c

specifies the CP spool file class to be used for the NOTICE file created by this link. The valid CP spool file classes are A - Z and 1 - 9. The default is N.

MSGFile=filename

specifies the 1- to 8-character file name of the message template for the link. The file must have a file type of TEMPLATE. The message template determines the content and destination of notes generated by the NOTIFY link driver. If RSCS cannot find the file, the link will not start. The default file name is the link ID.

MSGSend=Yes

MSGSend=No

specifies whether an active NOTIFY-type link should notify users when they send messages to undefined locations. The default is YES; RSCS sends message DMT310E to the user who sent the message to the undefined location. After receiving a message destined for an undefined location and notifying (or not notifying) the message sender, the NOTIFY-type link purges the message.

NETData=Yes

NETData=No

specifies whether the NOTIFY-type link should generate notes in NETDATA format. The default is YES. If you specify NO, the link will generate notes in 80-byte punch format. However, if the logical record length of the note template exceeds 80 bytes, the note file will be truncated to 80 bytes.

PARM (NOTIFY-Type)

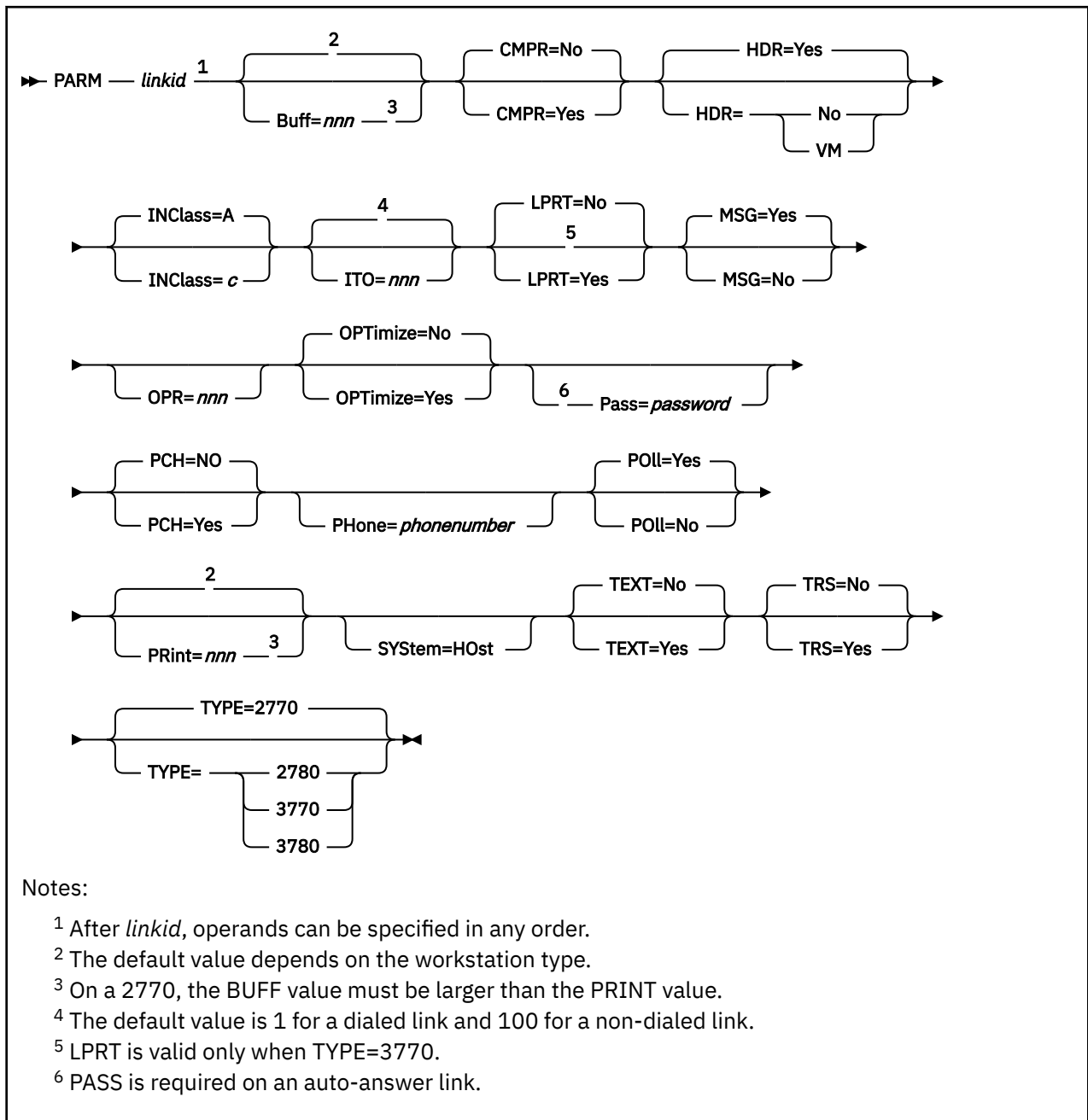
PURGE=*nnn*

specifies the number of days that RSCS will retain a file after sending a notice to the originator. This value can be 0 - 365. The default is 7. Specifying a value of 0 tells RSCS not to purge the file.

Usage Notes

Files that originate from the SYSTEM user ID will automatically be held on NOTIFY-type links until the specified purge interval has expired. However, RSCS will not send a NOTIFY message to the file originator. This will prevent two or more NOTIFY-type links from constantly sending each other messages.

PARM Statement for an RJE-Type Link



Purpose

This PARM statement defines the operational characteristics for a specific RJE-type link. An RJE-type link is a workstation link, connecting a system node to a workstation node over a BSC line.

The link can be started:

- Automatically during system start-up
- Automatically when a file is queued for transmission on an auto-start link
- Automatically when a call is received over an enabled (auto-answer) switched telecommunication line
- Manually by an operator or authorized user

PARM (RJE-Type)

For other than a link on an auto-answer line, the host side of the link must be started first, before a remote station can sign on.

Note: If you do not define a PARM statement for an RJE-type link, all of the indicated defaults will apply to the link. You can override PARM attributes by specifying values on the DEFINE or START command. For information on the DEFINE and START commands, see [z/VM: RSCS Networking Operation and Use](#).

Operands

linkid

is the name of the RJE-type link, as defined on a LINKDEFINE statement.

Buff=nnn

specifies the size of the line buffer, in bytes. This value depends on the workstation type:

Workstation	Acceptable Values	Default
2770	128, 256, or 512	128
2780	170 or 400	170
3770	256	256
3780	512	512

Note: To avoid print-line truncation on a 2770, the number of bytes (BUFF=nnn) must be larger than the number of print positions (PRINT=nnn). The buffer must be large enough to contain 6 bytes of control information and the characters to be printed.

CMPR=No

CMPR=Yes

specifies whether the 2770, 3770, or 3780 workstation has the blank compression feature. The default is NO.

HDR=

specifies the type of separator page to be inserted before each print file sent across the link.

Yes

RSCS-style separator page. This is the default.

No

No separator page.

VM

VM-style separator page.

For information about separator page formats, see [z/VM: RSCS Networking Operation and Use](#).

INClass=c

specifies the CP spool file class to be used for files received from the remote workstation. The valid CP spool file classes are A - Z and 1 - 9. The default is A.

ITO=nnn

specifies the inactivity time out (ITO) value, which defines the length of time that can occur with no activity (that is, no file transmission or reception) before the link is deactivated. If a link is ended because its ITO value has been exceeded, its restart exec will not be processed.

The ITO value can be 0 - 100:

0

The link will be deactivated immediately when there is no activity on it. However, before deactivating the link, RSCS will first send files that are queued for transmission, or finish reception of the file currently being received.

1 - 99

Approximate number of minutes until the link will be deactivated if there has been no activity on it.

100

No ITO will be in effect. The link will stay active until it is deactivated manually or an unrecoverable error occurs.

If you omit this operand, the default is:

- 1 (one minute) for a dialed link
- 100 (no ITO in effect) for a link that is not dialed

LPRT=

specifies where RSCS will direct print files. This operand is valid only when TYPE=3770.

No

RSCS will direct all print files to the console printer. This is the default.

Yes

RSCS will direct all print files to the line printer.

MSG=

specifies whether RSCS will allow the link driver to dequeue messages destined for transmission to the remote workstation.

Yes

RSCS will select all messages for transmission. This is the default.

No

RSCS will select messages, but they will not be sent to the remote workstation. The messages will be purged.

OPR=nnn

specifies the printer width (number of characters) to override the value specified on the PRINT operand. RSCS will truncate the file to the specified OPR value. This value can be 1 - 255. For files created in CMS or a guest virtual machine, the minimum value is 132.

OPTimize=No**OPTimize=Yes**

specifies whether optimization (merging of CCWs) should be used for print files being transmitted over this workstation link. The default is NO.

Specify NO when using a workstation link to communicate with a device that will use specialized file data such as page formatting information, or that requires preservation of original carriage control information.

Pass=password

specifies the 1- to 8-character password that the remote workstation must supply to be allowed to connect to RSCS. If the correct password is not received at sign-on time, RSCS deactivates the link. If a password is not defined on the PARM statement or on the DEFINE or START command, the remote workstation is not required to supply a password at sign-on time.

For an auto-answer link (automatically started when a call is received on an enabled port), PASS is mandatory.

PCH=No**PCH=Yes**

specifies whether a punch is available at the remote workstation. The default is NO.

PHone=phonenumber

specifies the phone number to be used when starting an auto-dial link. This string can be up to 32 characters long, optionally enclosed within single quotation marks (' '). Only the following characters may be used:

0 1 2 3 4 5 6 7 8 9

Decimal numerals are used for the actual dial digits of the phone number. For example:

```
PHONE=5551234
```

PARM (RJE-Type)

(blank)

Blanks make the phone number easier to read. They are not transmitted to the Automatic Calling Unit (ACU). If you use blanks, you must enclose the phone number within single quotation marks. For example:

```
PHONE='1 607 555 1234'
```

–

Hyphens also make the phone number easier to read. They are not transmitted to the ACU. For example:

```
PHONE='1-607 555-1234'
```

*

The asterisk is a separator character you can include following an access digit. For a private automatic branch exchange (PABX) extension, this character delays dialing to allow the dial tone to be returned from the public exchange. For example:

```
PHONE='9*1-607 555-1234'
```

.

The period is an end-of-number character that marks the end of the dial-digit group for those ACUs that require such an end character. The type of ACU in use determines whether you need an end-of-number character. For example:

```
PHONE='9*1-607 555-1234.'
```

POLL=

specifies whether the link will poll the remote workstation when the link does not have work to perform.

Yes

The link will poll continuously. This is the default.

No

The link will remain idle until it receives work.

PRint=nnn

specifies the number of print positions available at the workstation printer. This value depends on the workstation type:

Workstation	Acceptable Values	Default
2770	120, 132, or 144	120
2780	120 or 144	120
3770	132	132
3780	120 or 144	120

Note: To avoid print-line truncation for the 2770, the number of print positions (PRINT=nnn) must be *smaller* than the number of bytes in the buffer (BUFF=nnn). This is because the buffer must be large enough to contain 6 bytes of control information in addition to the characters to be printed.

SYStem=HOst

specifies that the link will operate as a host to a remote workstation. The workstation can consist of a card reader, a printer, and optionally a card punch.

TEXT=

specifies whether characters on a TN text printing chain should be transmitted to the remote printer in a mixture of uppercase and lowercase.

No

RSCS will translate any lowercase characters to uppercase before transmitting them to the remote printer. This is the default.

Yes

RSCS will transmit both uppercase and lowercase characters to the remote printer.

TRS=No**TRS=Yes**

specifies whether the remote workstation has the transparency feature. This feature is only for punch files, not print files. The default is NO.

Note: Usually, RSCS translates characters less than X'40' and greater than X'FE' into blanks. If you do not want RSCS to translate these characters into blanks when it sends them to the remote workstation, specify TRS=YES.

TYPE=

specifies the workstation type. The supported values are:

2770

2770 Data Communication System (the default).

2780

2780 Data Transmission Terminal.

3770

3770 Data Communication System.

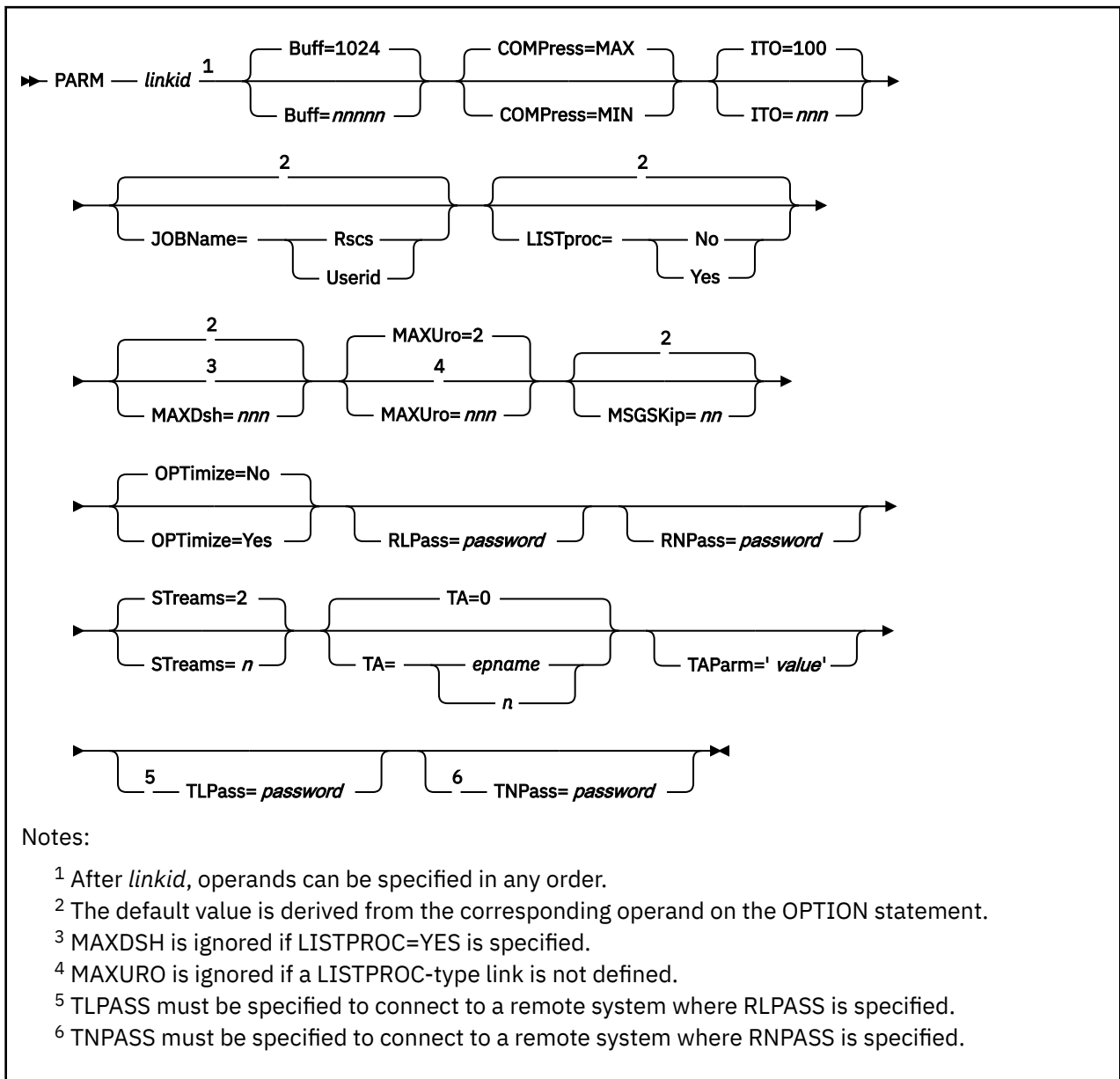
3780

3780 Data Communication Terminal.

Usage Notes

1. If you specify the BUFF, CMPR, LPRT, TRS, or TYPE operand but you do not specify the PASS operand, RSCS will internally perform the sign-on for this specific remote workstation. (That is, it will be an auto-sign-on link.) Once RSCS establishes the connection, the workstation receives all appropriate messages indicating the successful completion of this sign-on.
2. When you specify PASS operand, the remote workstation must sign on to this host so that passwords are validated.
3. Most remote RJE nodes are printer (output) workstations. To assist with operating with real or true emulation of hardware, the POLL=NO and MSG=NO options are recommended for RJE-type links. The OPR and INCLASS options are also recommended. These options enable you to select the type of files that should be processed on the link. You can define an RJE-type link that will process only printer and punch files, without processing message files. In addition, POLL=NO reduces the number of I/O operations needed on the link and within CP.

PARM Statement for an SNANJE-Type Link



Purpose

This PARM statement defines the operational characteristics for a specific SNANJE-type link. An SNANJE-type link starts a session with a remote NJE system connected to an SNA network.

The link can be started:

- Automatically during system start-up
- Automatically when a file is queued for transmission on an auto-start link
- Automatically if the remote system initiates the session
- Manually by an operator or authorized user

Note: If you do not define a PARM statement for an SNANJE-type link, all of the indicated defaults will apply to the link. You can override PARM attributes by specifying values on the DEFINE or START command. For information on the DEFINE and START commands, see [z/VM: RSCS Networking Operation and Use](#).

Operands

linkid

is the name of the SNANJE-type link, as defined on a LINKDEFINE statement.

Buff=nnnn

specifies the maximum size RU (Request Response Unit) this link can accept from VTAM. This value can be 300 - 32765 bytes. The default is 1024. The actual buffer size used for transmission is determined at sign-on time and is the smaller of the buffer sizes specified (or defaulted to) by the two connected systems.

COMPRESS=

specifies the degree of data compression to be used when transmitting the file.

MAX

Maximum data compression. This is the default.

MIN

Minimum data compression.

ITO=nnn

specifies the inactivity time out (ITO) value, which defines the length of time that can occur with no activity (that is, no file transmission or reception) before the link is deactivated. If a link is ended because its ITO value has been exceeded, its restart exec will not be processed.

The ITO value can be 0 - 100:

0

The link will be deactivated immediately when there is no activity on it. If ITO=0 is specified for the local and remote nodes of the link and files are queued on one end of the link, the link will deactivate before the files are sent. If both sides of the link have files to process, the files will be sent.

1 - 99

Approximate number of minutes until the link will be deactivated if there has been no activity on it.

100

No ITO will be in effect. The link will stay active until it is deactivated manually or an unrecoverable error occurs. This is the default.

JOBName=

specifies the type of job name RSCS will create for all files originating at your local node and all files arriving at your local node without a job header (that is, those coming from a workstation link).

Rscs

RSCS will generate a job name of RSCSnnnn, where nnnn is the origin spool ID of the file.

Userid

RSCS will use the origin user ID as the job name.

If you specify this operand, it will override any previously defined default for this link. If you omit this operand, the default value is derived from the corresponding operand on the OPTION statement.

To override the default job name for a specific file, use the JOBNAM operand on the CP TAG command. For more information about the CP TAG command, see [z/VM: RSCS Networking Operation and Use](#).

LISTproc=No

LISTproc=Yes

specifies whether the connected system can handle an unlimited number of data set headers. If you specify NO, the link will use the number of data set headers specified on the MAXDSH operand.

If you specify this operand, it will override any previously defined default for this link. If you omit this operand, the default value is derived from the corresponding operand on the OPTION statement.

The following table summarizes how non-list processor systems will handle list processor files.

Table 19. How Non-List Processor Systems Handle List Processor Files

Non-List Processor Systems	How the File Will Be Handled
<ul style="list-style-type: none"> • RSCS V1.3* • JES3 • VSE/POWER 	<p>These systems will immediately split the file into separate spool files. These files will be either distributed to local user IDs or sent on to other links.</p>
<ul style="list-style-type: none"> • RSCS V2.2* (without list processor support • RSCS V2.3* or later* (without a list processor defined) • JES2 	<p>These systems will receive the file and split out separate files for local user IDs. They will also split out one file containing data set headers to be sent to each link on the distribution list.</p>

Note: * These systems will handle only a limited number of data set headers.

MAXDsh=nnn

specifies the maximum number of data set headers the link driver will send on the link when processing a file with multiple data sets. This value is intended for connections to NJE subsystems that do not have list processor capabilities. This operand is ignored if LISTPROC=YES is specified.

This value can be 0 - 100. A value of 0 is a special case that means the determination for the number of data set headers placed on the link is based on the actual number of data set headers contained in the spool file. A value of 0 should be specified only with full understanding of the number of data set headers that the remote system can handle.

If you specify this operand, it will override any previously defined default for this link. If you omit this operand, the default value is derived from the corresponding operand on the OPTION statement.

MAXUro=nnn

specifies the maximum number of unit record output devices (virtual printers or punches) for each stream that the link driver will use at any one time while receiving a file. RSCS uses these devices to fan out files containing data set headers, which the link driver determines must be split off the original file. All remaining data set headers (and the file contents) are written to an additional unit record device and queued on the list processor link. This operand is ignored if a LISTPROC-type link is not defined.

This value can be 0 - 255. The default is 2. Specifying a value of 0 will cause all received data set headers and file contents to be queued to the list processor task. Under usual conditions, a link will always use one more device per stream than specified. If a value of 0 is specified, and the first data set header received in a file is for the local system or for a non-networking link attached to the local system, an extra device will be defined to hold the remaining data set headers.

MSGSKIP=nn

specifies the number of file buffers that RSCS will send on the link before sending a message buffer (unless there are no file buffers available). This value can be 0 - 15. Specifying 0 is a special case that tells RSCS to send a message buffer whenever messages are available to send on the link, regardless of whether file buffers are available.

If you specify this operand, it will override any previously defined default for this link. If you omit this operand, the default value is derived from the corresponding operand on the OPTION statement.

Note: If this link determines during sign-on that the peer system supports the mixed RCB feature of NJE, RSCS ignores the MSGSKIP setting. The mixed RCB feature lets RSCS send buffers containing a mixture of information (files and messages). If RSCS can send files and messages in one buffer, there is no need to use the MSGSKIP operand to tell RSCS how many file buffers to send before sending a message buffer. For more information about the mixed RCB feature of NJE, see [z/OS: Network Job Entry \(NJE\) Formats and Protocols \(https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa320988/\\$file/hasa600_v2r5.pdf\)](https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa320988/$file/hasa600_v2r5.pdf).

OPTimize=No

OPTimize=Yes

specifies whether optimization (merging of CCWs) should be used for files that originate on the local system and files that are processed by the local system. (This does not include store-and-forward files that are passing through the local node.) The default is NO.

RLPass=password

specifies the 1- to 8-character line password that is required from the remote system before transmission of data is initiated. If the correct password is not received at sign-on time, RSCS deactivates the link.

RNPass=password

specifies the 1- to 8-character node password that is required from the remote system before transmission of data is initiated. If the correct password is not received at sign-on time, RSCS deactivates the link.

STreams=n

specifies the maximum number of files that may be transmitted simultaneously on different streams. This value can be 1 - 7. The default is 2.

Note: RSCS cannot actually send more than one file at a time. When several files are active on different streams, RSCS sends the file with the smallest number of blocks and credits the other files for having to wait. After the file is transmitted, RSCS subtracts the credits from the number of blocks left to send for each active file. RSCS then chooses the file with the lowest number to send. The longer a large file has to wait, the more credits it will accumulate. When the difference between the accumulated credits and the actual file size becomes small enough, RSCS transmits the large file.

PI

TA=

specifies the transmission algorithm, which tells RSCS how to select files for transmission on the link.

epname

is the name of the entry point (outside of the RSCS LOADLIB) where your installation-defined algorithm resides.

n

is a hexadecimal value, 0 - F, that identifies the exit routine in the DMTAXA module (DMTAXAG n):

0

Identifies an IBM-supplied algorithm that allows files to be selected regardless of size. This is the default.

Note: If you specify TA=0 or allow it to default, do not specify the TAPARM operand.

1

Identifies an IBM-supplied algorithm that specifies a default size for each stream specified on the STREAMS operand. You can override the default by specifying the stream size on the TAPARM operand.

2 - F

Identifies an algorithm defined by your installation.

Note: If you update the DMTAXA module, it must be reassembled.

TAParm='value'

specifies a parameter value associated with the transmission algorithm. This value is a character string up to 80 bytes in length, enclosed in single quotation marks. (The single quotation marks are optional if you do not include blanks in the character string.) Any characters are allowed between the single quotation marks except another single quotation mark. The character string cannot be continued to the next line in the configuration file.

The value that you specify for TAPARM depends on the value that you specify for TA:

TA=

TAPARM=

PARM (SNANJE-Type)

0

Do not specify TAPARM; it will be rejected with an error message.

epname

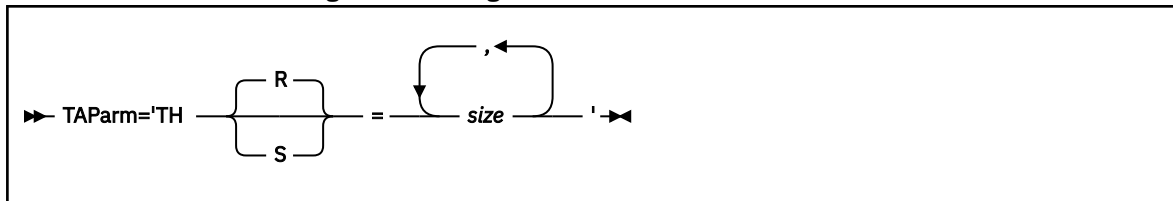
Specify a TAPARM defined by your installation.

2 - F

Specify a TAPARM defined by your installation.

1

Either omit TAPARM to accept the default size for each stream (listed in the table below) or specify a size for each stream using the following format:



TH=

THR=

specifies the threshold number of records. THR is the default.

THS=

specifies the threshold number of spool file blocks. Specify THS when you expect to send files with very large records, such as printer output files, containing page image data, that are destined for a PSF-controlled printer.

size

is the file size. This can be specified as either of the following:

nnnnnnnn

Maximum file size RSCS should send on the stream. The minimum file size is one more than the maximum file size from the previous stream specification. If there is no previous stream specification, there is no minimum file size.

(min,max)

Minimum and maximum file size RSCS should send on the stream. If you do not want to specify a minimum file size, omit *min*. If you do not want to specify a maximum file size, omit *max*. Specifying (,) means you do not want to specify either a minimum or a maximum file size; any file is eligible for transmission on the stream.

For example, if you specified STREAMS=3, you can choose the TAPARM default by omitting TAPARM or you can specify any one of the following:

TAPARM='TH=*size1*'

TAPARM='TH=*size1,size2*'

TAPARM='TH=*size1,size2,size3*'

TAPARM='THR=*size1*'

TAPARM='THR=*size1,size2*'

TAPARM='THR=*size1,size2,size3*'

TAPARM='THS=*size1*'

TAPARM='THS=*size1,size2*'

TAPARM='THS=*size1,size2,size3*'

If TA=1 is specified and TAPARM is omitted, these are the defaults:

If you specified	THR= Default for Stream 1	THR= Default for Stream 2	THR= Default for Stream 3	THR= Default for Stream 4	THR= Default for Stream 5	THR= Default for Stream 6	THR= Default for Stream 7
STREAMS=1	All files	—	—	—	—	—	—

If you specified	THR= Default for Stream 1	THR= Default for Stream 2	THR= Default for Stream 3	THR= Default for Stream 4	THR= Default for Stream 5	THR= Default for Stream 6	THR= Default for Stream 7
STREAMS=2	(,100)	(101,)	–	–	–	–	–
STREAMS=3	(,100)	(101,)	(101,)	–	–	–	–
STREAMS=4	(,100)	(101,)	(101,)	(101,)	–	–	–
STREAMS=5	(,100)	(101,)	(101,)	(101,)	(101,)	–	–
STREAMS=6	(,100)	(101,)	(101,)	(101,)	(101,)	(101,)	–
STREAMS=7	(,100)	(101,)	(101,)	(101,)	(101,)	(101,)	(101,)

PI end

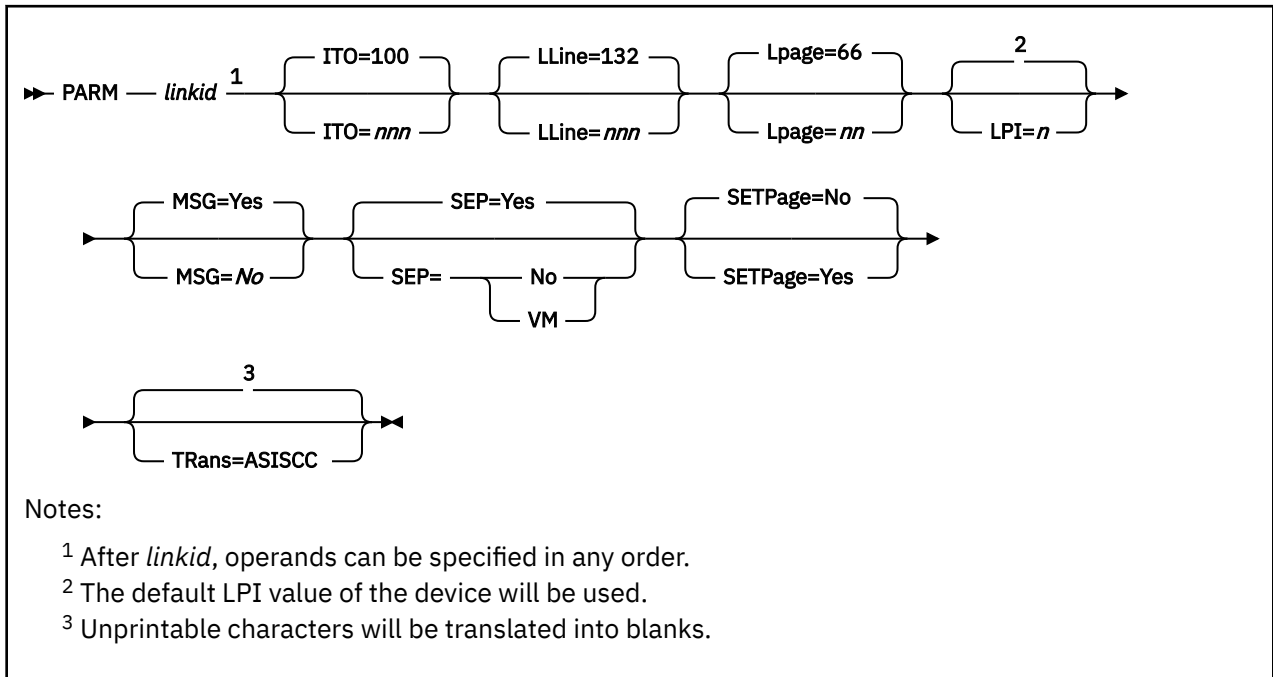
TLPass=password

specifies the 1- to 8-character line password to be transmitted to the remote system at sign-on time. TLPASS must be specified to connect to a remote system where RLPASS is defined.

TNPass=password

specifies the 1- to 8-character node password to be transmitted to the remote system at sign-on time. TNPASS must be specified to connect to a remote system where RNPASS is defined.

PARM Statement for an SNARJE-Type Link



Purpose

This PARM statement defines the operational characteristics for a specific SNARJE-type link. An SNARJE-type link is a workstation link, connecting a system node to an SNA System/36 RJE workstation.

The link can be started:

- Automatically during system start-up
- Automatically when a file is queued for transmission on an auto-start link
- Automatically if the remote workstation initiates the session
- Manually by an operator or authorized user

Note: If you do not define a PARM statement for an SNARJE-type link, all of the indicated defaults will apply to the link. You can override PARM attributes by specifying values on the DEFINE or START command. For information on the DEFINE and START commands, see [z/VM: RSCS Networking Operation and Use](#).

Operands

linkid

is the name of SNARJE-type link, as defined on a LINKDEFINE statement.

ITO=nnn

specifies the inactivity time out (ITO) value, which defines the length of time that can occur with no activity (that is, no file transmission or reception) before the link is deactivated. If a link is ended because its ITO value has been exceeded, its restart exec will not be processed.

The ITO value can be 0 - 100:

0

The link will be deactivated immediately when there is no activity on it. However, before deactivating the link, RSCS will first send files that are queued for transmission, or finish reception of the file currently being received.

1 - 99

Approximate number of minutes until the link will be deactivated if there has been no activity on it.

100

No ITO will be in effect. The link will stay active until it is deactivated manually or an unrecoverable error occurs. This is the default.

LLine=nnn

specifies the maximum line length (number of characters per line). LLINE is a logical maximum, which is determined by the size of the form being used in the printer. This value can be 1 - 220. The default is 132. All print lines are truncated at the length specified by this value. If a number less than 74 is specified for LLINE, no separator page is printed.

Lpage=nn

specifies the number of lines per page on the type of form inserted in the printer. This value can be 0 - 99. The default is 66. A value of 0 specifies that no page ejects will be done when a page eject is found in the file.

Note: RSCS uses this value only to determine the current vertical position on a page while printing a file. Any actual page ejects are determined by Skip-to-Channel-1 CCWs within the file. If there are no Skip-to-Channel-1 CCWs in the file, no page ejects are done, except between files. The actual number of lines contained in the file between consecutive Skip-to-Channel-1 CCWs must be equal to or less than this value.

LPI=n

specifies the number of lines per inch to be printed on the form for an LU_T1 session. This value can be 3, 4, 6, or 8. If you omit this operand, RSCS sets the value for LPI in the SNA Character String (SCS) to X'00'. According to LU_T1 protocol, X'00' for LPI causes the device to use its default value for lines per inch. See the printer documentation for information about what this default value is.

Note: RSCS sends SCS orders to the device to set page characteristics by overriding defaults. RSCS will not send the SCS orders unless you also specify SETPAGE=YES.

MSG=

specifies whether RSCS will allow the link driver to dequeue messages destined for transmission to the remote workstation.

Yes

RSCS will select all messages for transmission. This is the default.

No

RSCS will select messages, but they will not be sent to the remote workstation. The messages will be purged.

SEP=

specifies the type of separator page to be inserted before each print file sent across the link.

Yes

RSCS-style separator page. This is the default.

No

No separator page.

VM

VM-style separator page.

For descriptions of the separator page formats, see [z/VM: RSCS Networking Operation and Use](#).

Note: This operand applies only to print files. When transmitting a punch file to a remote workstation, RSCS creates a header line on the output. For information about the format of this header line, see [z/VM: RSCS Networking Operation and Use](#).

SETPage=

specifies whether RSCS will send SCS orders to the device to set page characteristics for printing for an LU_T1 session. These page characteristics are based on the values specified on the LLINE, LPAGE, and LPI operands.

PARM (SNARJE-Type)

No

RSCS will not set page characteristics. No SCS orders will be sent. This is the default.

Yes

RSCS will automatically set page characteristics for the device. SCS orders will override hardware defaults and those set manually on the device.

Note: If the device is being shared with another VTAM application, you should specify SETPAGE=YES to ensure proper page alignment when RSCS regains control of the session.

TRans=ASISCC

specifies that RSCS will not translate unprintable characters in files into blanks. This allows the user to pass SCS orders to the remote workstation.

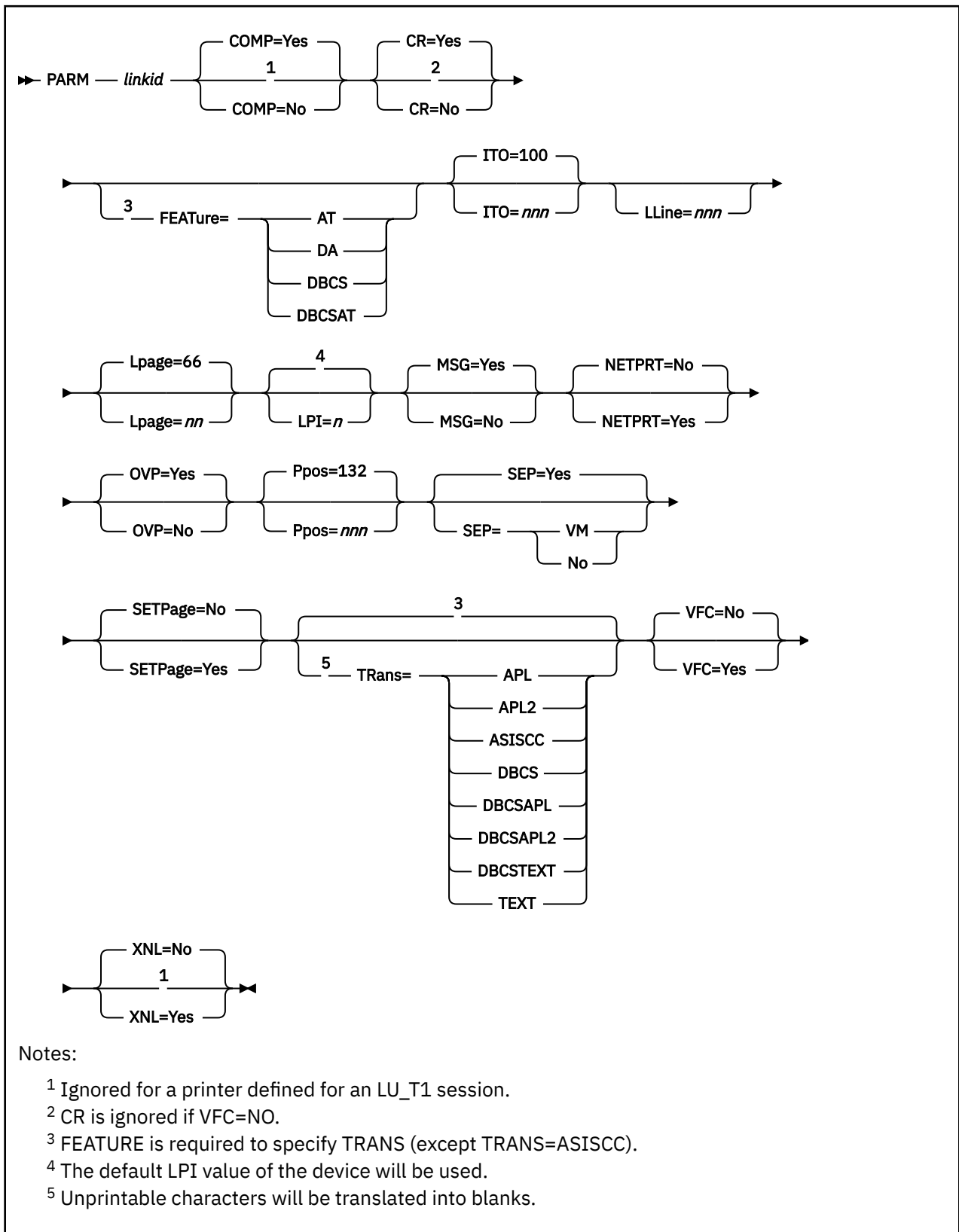
Usage Notes

When you define a form name (using the FORM statement), you can define the following characteristics:

- Separator page style
- Line length
- Page length
- Lines per inch (only for LU_T1 printers and workstations)

If you specify a form name when starting a link, be aware that the form name characteristics always override any SEP, LLINE, and LPAGE specifications and will override LPI specifications when SETPAGE=YES.

PARM Statement for an SNA3270P-Type Link



Purpose

This PARM statement defines the operational characteristics for a specific SNA3270P-type link. An SNA3270P-type link starts a session with a remote IBM 3270 Information Display System printer connected to an SNA network.

The link can be started:

- Automatically during system start-up
- Automatically when a file is queued for transmission on an auto-start link
- Automatically if the remote workstation initiates the session
- Manually by an operator or authorized user

Note: If you do not define a PARM statement for an SNA3270P-type link, all of the indicated defaults will apply to the link. You can override PARM attributes by specifying values on the DEFINE or START command. For information on the DEFINE and START commands, see [z/VM: RSCS Networking Operation and Use](#).

Operands

linkid

is the name of the SNA3270P-type link, as defined on a LINKDEFINE statement.

COMP=Yes

COMP=No

specifies whether the link will perform blank compression. The default is YES; occurrences of five or more blanks will be compressed. You should specify NO for a locally attached printer. This operand is ignored for a printer defined for an LU_T1 session, and no blank compression will occur.

CR=Yes

CR=No

specifies whether the link will send a 3270 carriage return (CR) order after a forms feed (FF) when VFC=YES is specified. The default is YES. This operand has no meaning unless VFC=YES is also specified. Specify CR=NO when VFC=YES is specified and cut paper is used in a 5210 printer.

FEATure=

specifies that the printer has the indicated feature.

AT

APL/Text feature.

DA

Data Analysis-APL feature.

DBCS

Double-Byte Character Set feature.

DBCSAT

AT and DBCS features.

For the valid combinations of FEATURE and TRANS settings, see [Table 20 on page 222](#).

Note:

1. To use the TRANS operand and the PRT option on the CP TAG command, FEATURE must be specified. This rule does not apply to TRANS=ASISCC or to the ASISCC, GRAF, GRAPH, and NOTR settings on the PRT option, because they have no corresponding FEATURE setting.
2. Specifying the FEATURE operand as a default is helpful, even if you are specifying TRANS=ASISCC, because it lets you specify a TRANS setting as a holding position. You can then override the TRANS setting for a file by using the PRT option on the TAG command. If TRANS is not specified, then the only PRT settings allowed will be ASISCC, GRAF, GRAPH, and NOTR.
3. The DBCS feature processes the double-byte character set but does not translate any APL or TEXT representation that may exist in the file. The only translation done with the DBCS setting is 3270 control character removal. If you want APL or TEXT translation, use FEATURE=DBCSAT with

TRANS=DBCSAPL, TRANS=DBCSAPL2, or TRANS=DBCSTEXT. If APL, APL2®, or TEXT is specified, the DBCS characters will not be verified or translated to the correct format for the printer output.

For more information about the PRT option of the CP TAG command, see *z/VM: RSCS Networking Operation and Use*.

ITO=nnn

specifies the inactivity time out (ITO) value, which defines the length of time that can occur with no activity (that is, no file transmission or reception) before the link is deactivated. If a link is ended because its ITO value has been exceeded, its restart exec will not be processed.

The ITO value can be 0 - 100:

0

The link will be deactivated immediately when there is no activity on it. However, before deactivating the link, RSCS will first send files that are queued for transmission, or finish reception of the file currently being received.

1 - 99

Approximate number of minutes until the link will be deactivated if there has been no activity on it.

100

No ITO will be in effect. The link will stay active until it is deactivated manually or an unrecoverable error occurs. This is the default.

LLine=nnn

specifies the maximum line length (number of characters per line). LLINE is a logical maximum, which is determined by the size of the form being used in the printer. This value can be 1 - 220. All print lines are truncated at the length specified by this value.

If LLINE is not specified, or if the LLINE value is greater than the PPOS value, LLINE is ignored and the PPOS value is used. If a file is sent to a printer with a character width equal to the LLINE value, the output will be truncated to the PPOS width.

If the LLINE value is less than the PPOS value, the LLINE value will override the PPOS value and will remain as the printer default for its active duration. If a number less than 74 is specified for LLINE, no separator page is printed.

Note: The 5210 printer accepts print wheels that allow it to print in various size fonts. Therefore, the font size on the print wheel must be considered when you specify the LLine value. For example, if the 5210 is using a 10-pitch font print wheel, the LLINE value should be 85. However, if a 12-pitch font print wheel is being used, the LLINE value should be greater than 85.

Lpage=nn

specifies the number of lines per page on the type of form inserted in the printer. This value can be 0 - 99. The default is 66. A value of 0 specifies that no page eject will be done when a page eject is found in the file.

Note: RSCS uses this value only to determine the current vertical position on a page while printing a file. Any actual page ejects are determined by Skip-to-Channel-1 CCWs within the file. If there are no Skip-to-Channel-1 CCWs in the file, no page ejects are done, except between files. The actual number of lines contained in the file between consecutive Skip-to-Channel-1 CCWs must be equal to or less than this value.

LPI=n

specifies the number of lines per inch to be printed on the form for an LU_T1 session. This value can be 3, 4, 6, or 8. If you omit this operand, RSCS sets the value for LPI in the SNA Character String (SCS) to X'00'. According to LU_T1 protocol, X'00' for LPI causes the printer to use its default value for lines per inch. See the printer documentation for information about what this default value is.

Note: RSCS sends SCS orders to the printer to set page characteristics by overriding printer defaults. RSCS will not send the SCS orders unless you also specify SETPAGE=YES.

MSG=

specifies whether RSCS will allow the link driver to dequeue messages destined for transmission to the remote printer.

Yes

RSCS will select all messages for transmission. This is the default.

No

RSCS will select messages, but they will not be sent to the remote printer. The messages will be purged.

NETPRT=No

NETPRT=Yes

specifies whether an IBM Network printer 4312, 4317, or 4324 is attached to this link. The default is NO. You must specify YES if one of these printers is attached and it is to print IDPS graphic files. Specify NO if any other printer type is attached.

OVP=Yes

OVP=No

specifies whether the link will allow overprinting to occur (such as underscored words or overstruck characters for highlighting). The default is YES. Specify NO for a printer or control unit that does not support the 3270 carriage return (CR) order.

Ppos=nnn

specifies the maximum number of print positions available on the printer. This value can be 1 - 220, depending on the printer. The default is 132.

PPOS is a hardware maximum determined by the number of characters the printer can print on each line, independent of the size of the form used in the printer. Each printer has its own PPOS value; see the printer documentation for more information. If the PPOS value you specify does not match the printer's PPOS value, some types of printed output (including separator pages) will not print correctly.

SEP=

specifies the type of separator page to be inserted before each print file sent across the link.

Yes

RSCS-style separator page. This is the default.

No

No separator page.

VM

VM-style separator page.

For descriptions of the separator page formats, see [z/VM: RSCS Networking Operation and Use](#).

Note: This operand applies only to print files. When transmitting a punch file to a remote workstation, RSCS creates a header line on the output. For information about the format of this header line, see [z/VM: RSCS Networking Operation and Use](#).

SETPage=

specifies whether RSCS will send SCS orders to the printer to set page characteristics for printing for an LU_T1 session. These page characteristics are based on the values specified on the LLINE, LPAGE, and LPI operands.

No

RSCS will not set page characteristics. No SCS orders will be sent. This is the default.

Yes

RSCS will automatically set page characteristics on the printer. SCS orders will override printer hardware defaults and those set manually on the printer.

Note: If the printer is being shared with another VTAM application, you should specify SETPAGE=YES (and VFC=YES if the printer supports vertical forms control and the 3270 carriage control order) to ensure proper page alignment when RSCS regains control of the printer.

TRans=

specifies the default translation mode for files being transmitted to a printer that has the AT, DA, DBCS, or DBCSAT feature (as specified on the FEATURE operand). The translation mode tells RSCS that all files may contain the indicated special characters and that RSCS should translate these special characters accordingly. Alternatively, you can use this operand to specify that unprintable characters should not be translated into blanks.

The default translation mode you specify on TRANS can be overridden for a specific file by using the PRT option on the CP TAG command. For more information about the CP TAG command, see [z/VM: RSCS Networking Operation and Use](#).

The valid translation modes are:

APL

If RSCS detects internally represented EBCDIC special APL characters in a file, RSCS will translate those characters into the appropriate 2-byte I/O interface codes.

APL2

If RSCS detects internally represented EBCDIC special APL2 characters in a file, RSCS will translate those characters into the appropriate 2-byte I/O interface codes.

This mode is valid only when FEATURE is set to AT or DBCSAT. If TRANS=APL2 is specified with any other FEATURE setting, it is marked as an incorrect combination and an error message is issued.

ASISCC

RSCS will not translate unprintable characters into blanks. This lets the user pass SCS orders to the remote printer or workstation. [Table 20 on page 222](#) shows that ASISCC can be used with all FEATURE settings.

TRANS=ASISCC means the default is no translation. If a file needs translation, you can override this default by using the PRT option on the CP TAG command (assuming that you specified the correct FEATURE setting). If FEATURE is not specified, then only the ASISCC, GRAF, GRAPH, or NOTR setting can be specified on the TAG command.

Note: TRANS=ASISCC can be specified without a FEATURE setting.

DBCS

RSCS will verify double-byte EBCDIC character strings delimited by SO/SI (shift-out/shift-in) characters for valid DBCS syntax before transmitting a file.

DBCSAPL

RSCS will translate internally represented EBCDIC special APL characters into the appropriate 2-byte I/O interface codes and will verify all double-byte EBCDIC character strings delimited by SO/SI (shift-out/shift-in) characters for valid DBCS syntax before transmitting a file.

DBCSAPL2

RSCS will translate internally represented EBCDIC special APL2 characters into the appropriate 2-byte I/O interface codes and will verify all double-byte EBCDIC character strings delimited by SO/SI (shift-out/shift-in) characters for valid DBCS syntax before transmitting a file.

This mode is valid only when FEATURE is set to DBCSAT. If TRANS=DBCSAPL2 is specified with any other FEATURE setting, it is marked as an incorrect combination and an error message is issued.

DBCSTEXT

RSCS will translate internally represented EBCDIC special TEXT characters into the appropriate 2-byte I/O interface codes and will verify double-byte EBCDIC character strings delimited by SO/SI (shift-out/shift-in) characters for valid DBCS syntax before transmitting a file.

TEXT

RSCS will translate internally represented EBCDIC special TEXT characters into the appropriate 2-byte I/O interface codes.

For the valid combinations of FEATURE and TRANS settings, see [Table 20 on page 222](#).

Note:

1. If you omit the TRANS operand, the usual character translation occurs (unprintable characters will be translated into blanks).
2. RSCS ignores the TRANS operand unless you also specify the corresponding FEATURE setting. TRANS=ASISCC is the exception; it does not require a FEATURE setting.
3. RSCS does not perform blank compression within DBCS SO/SI characters.
4. RSCS translates only those characters identified in the list above. RSCS will not change the standard EBCDIC data.

VFC=

specifies whether the printer will use the 3270 Vertical Forms Control (VFC) feature for vertical spacing.

No

Vertical spacing will be achieved through multiple new line (NL) orders. This is the default.

Yes

The printer will use the 3270 VFC feature and the 3270 carriage control (CC) order. VFC=YES is valid only for those printers and control units that support the 3270 CC order.

Note: Generally, IBM recommends that you specify VFC=YES. If users send GDDM or IPDS graphic files to this printer, you must specify VFC=YES to ensure that the pages align correctly.

XNL=No

XNL=Yes

specifies whether the link is to include an extra new line (NL) order following each line that is as long as the maximum number of print positions. The default is NO. Specify YES for those printers (such as the 3262, 3289, 4234, and 6262) that do not generate an extra new line function for a maximum length line. This operand is ignored if the printer is defined for LU_T1 sessions.

Usage Notes

1. When you define a form name (using the FORM statement), you can define these characteristics:
 - Separator page style
 - Line length
 - Page length
 - Lines per inch (only for LU_T1 printers and work stations)
2. If you specify a form name when starting a link, be aware that the form name characteristics always override any SEP, LLINE, and LPAGE specifications and will override LPI specifications when SETPAGE=YES.
3. If you specify an incorrect combination of FEATURE and TRANS settings, the SCO receives error message DMT807E and the link is deactivated.
4. Table 20 on page 222 indicates the valid combinations of FEATURE and TRANS settings.

Table 20. Valid Combinations of FEATURE and TRANS Settings

TRans=	FEATure= DA	FEATure= AT	FEATure= DBCS	FEATure= DBCSAT
APL	X	X		X
APL2		X		X
TEXT	X	X		X
DBCS			X	X
DBCSAPL				X

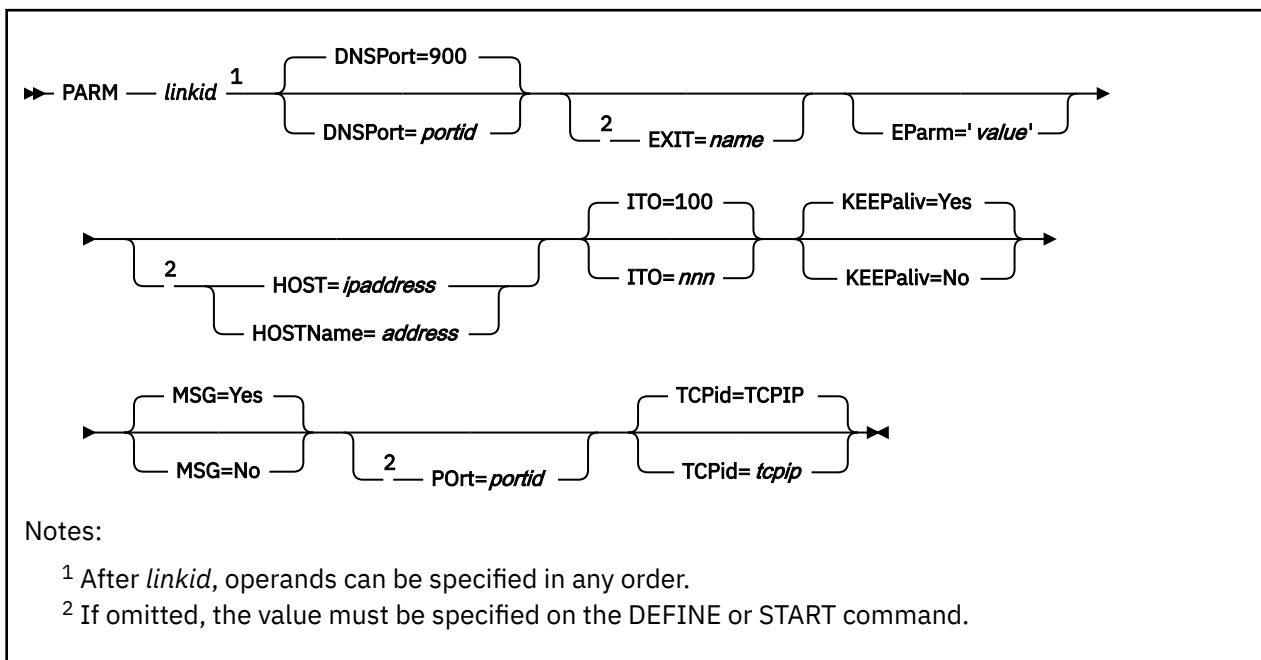
Table 20. Valid Combinations of FEATURE and TRANS Settings (continued)

TRans=	FEATure= DA	FEATure= AT	FEATure= DBCS	FEATure= DBCSAT
DBCSAPL2				X
DBCSTEXT				X
ASISCC	X	X	X	X

Note:

- a. If APL, APL2, or TEXT is specified on the TRANS operand, and the FEATURE operand is set to DBCSAT, then there is no DBCS translation of the file, only APL, APL2, or TEXT. The DBCS translation will occur with APL, APL2, or TEXT translation only when the TRANS setting is DBCSAPL, DBCSAPL2, or DBCSTEXT.
- b. TRANS=ASISCC does not require the FEATURE operand to be specified.

PARM Statement for a TCPASCII-Type Link



Purpose

This PARM statement defines the operational characteristics for a specific TCPASCII-type link. A TCPASCII-type link begins a session with a remote ASCII printer or plotter attached to a print server in a TCP/IP network.

The link can be started:

- Automatically during system start-up
- Automatically when a file is queued for transmission on an auto-start link
- Manually by an operator or authorized user

Note: If you do not define a PARM statement for a TCPASCII-type link, all of the indicated defaults will apply to the link. You can override PARM attributes by specifying values on the DEFINE or START command. Some required link attributes do not have PARM defaults; if a value is not defined on the PARM statement, it must be specified on the DEFINE or START command, or the link will not activate on the START command. For information on the DEFINE and START commands, see [z/VM: RSCS Networking Operation and Use](#).

Operands

linkid

is the name of the TCPASCII-type link, as defined on a LINKDEFINE statement.

DNSPort=portid

specifies the port number on the local system where the RSCS domain name server (RSCSDNS), if running, listens for connect requests. The default is 900.

Note: By default, TCP/IP restricts the use of well-known ports (1 - 1023). To use port 900 or any other well-known port when usage is restricted, the RSCSDNS server must be authorized on a PORT statement in the TCP/IP configuration file (PROFILE TCPIP). For more information, see [z/VM: TCP/IP Planning and Customization](#).

PI

EXIT=name

specifies the name of the exit routine module that contains the exit routines supporting the specific ASCII device. The exit routine module must reside in a LOADLIB that has been identified by using the GCS GLOBAL command. If you omit this operand and the exit name is not specified on the DEFINE or START command, or if RSCS cannot find the exit routine module, the link driver will not start.

IBM supplies the following sample exit routine modules in the RSCSEXIT LOADLIB:

Module**Supported Device****ASCXDSOE**

DS180 Matrix Printer from Datasouth.

ASCXDWRE

LA120 DECwriter Printer from DEC.

ASCXONE

Generic ASCII printer.

ASCXPROP

IBM Proprinter.

ASCXPSE

PostScript printer.

ASCXSPWE

NEC 3515 Spinwriter Printer.

ASCXZETE

Nicolet Zeta 8 Plotter.

ASCX749E

IBM Instruments XY/749 Multipen Digital Plotter.

For more information about the ASCII exit routines, see [z/VM: RSCS Networking Exit Customization](#).

EParm='value'

specifies a parameter value associated with the exit routine module. This value is a character string up to 239 bytes in length, enclosed in single quotation marks. Any characters are allowed between the single quotation marks except another single quotation mark. The character string cannot be continued to the next line in the configuration file. The actual value of the character string is defined by the exit routine module.

For information about the EPARM parameters supported for use with the ASCXONE and ASCXPSE sample modules, see [z/VM: RSCS Networking Exit Customization](#).

PI end**HOST=ipaddress**

specifies the IP address (in dotted decimal format) of the remote host to which to connect.

If a HOST or HOSTNAME value is not defined on the PARM statement, it must be specified on the DEFINE or START command.

HOSTName=address

specifies the fully qualified domain name address (up to 200 characters) of the remote host to which to connect.

Note: Use of this operand requires the RSCS domain name server, which will resolve the domain name address to an IP address.

If a HOST or HOSTNAME value is not defined on the PARM statement, it must be specified on the DEFINE or START command.

ITO=nnn

specifies the inactivity time out (ITO) value, which defines the length of time that can occur with no activity (that is, no file transmission or reception) before the link is deactivated. If a link is ended because its ITO value has been exceeded, its restart exec will not be processed.

PARM (TCPASCII-Type)

The ITO value can be 0 - 100:

0

The link will be deactivated immediately when there is no activity on it. However, before deactivating the link, RSCS will first send files that are queued for transmission, or finish reception of the file currently being received.

1 - 99

Approximate number of minutes until the link will be deactivated if there has been no activity on it.

100

No ITO will be in effect. The link will stay active until it is deactivated manually or an unrecoverable error occurs. This is the default.

KEEPaliv=Yes

KEEPaliv=No

specifies whether keep alive packets will be sent when a connection has been idle for the keep alive interval. This allows the link driver to detect that the other side has terminated or become unreachable even when no data is being exchanged. The keep alive interval is specified in the configuration file for the TCP/IP virtual machine. The default is YES.

Msg=

specifies whether RSCS will allow the link driver to dequeue messages destined for transmission to the remote printer.

Yes

RSCS will select all messages for transmission. This is the default.

No

RSCS will select messages, but they will not be sent to the remote printer. The messages will be purged.

Port=portid

specifies the port number to which to connect on the host.

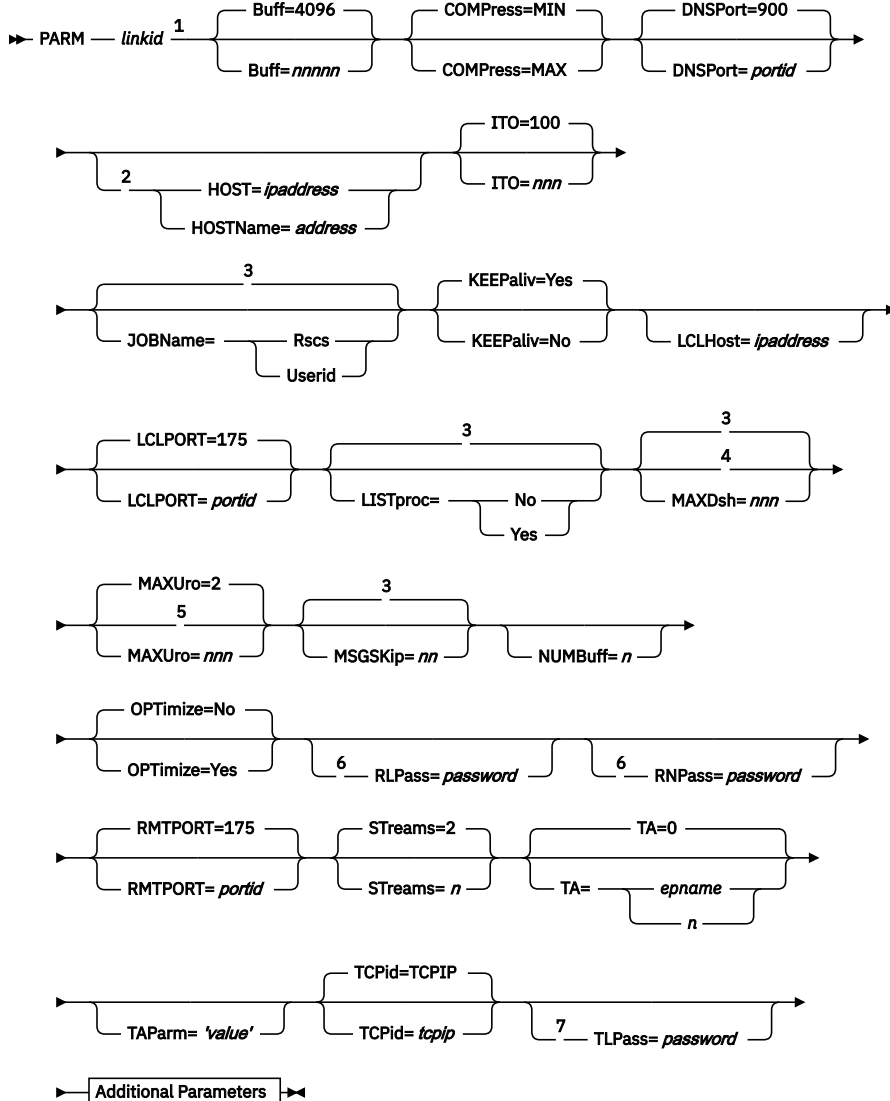
Note: By default, TCP/IP restricts the use of well-known ports (1 - 1023). To use a well-known port when usage is restricted, the RSCS server must be authorized on a PORT statement in the TCP/IP configuration file (PROFILE TCPIP). For more information, see [z/VM: TCP/IP Planning and Customization](#).

If the port number is not defined on the PARM statement, it must be specified on the DEFINE or START command.

TCPid=tcPIP

specifies the name of the TCP/IP virtual machine. The default name is TCPIP.

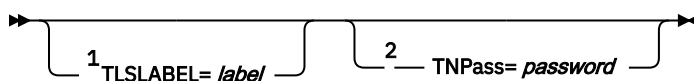
PARM Statement for a TCPNJE-Type Link



Notes:

- 1 After *linkid*, operands can be specified in any order.
- 2 If omitted, the value must be specified on the DEFINE or START command.
- 3 The default value is derived from the corresponding operand on the OPTION statement.
- 4 MAXDSH is ignored if LISTPROC=YES is specified.
- 5 MAXURO is ignored if a LISTPROC-type link is not defined.
- 6 On an auto-answer link, RLPASS or RNPASS (or both) must be specified.
- 7 TLPASS must be specified to connect to a remote system where RLPASS is specified.

Additional Parameters



Notes:

- 1 TlsLabel input must be uppercase.

² TNPASS must be specified to connect to a remote system where RNPASS is specified.

Purpose

This PARM statement defines the operational characteristics for a specific TCPNJE-type link. A TCPNJE-type link connects the local RSCS system to a remote NJE system in a TCP/IP network.

The link can be started:

- Automatically during system start-up
- Automatically when a file is queued for transmission on an auto-start link
- Manually by an operator or authorized user

Note: If you do not define a PARM statement for a TCPNJE-type link, all of the indicated defaults will apply to the link. You can override PARM attributes by specifying values on the DEFINE or START command. Some required link attributes do not have PARM defaults; if a value is not defined on the PARM statement, it must be specified on the DEFINE or START command, or the link will not activate on the START command. For information on the DEFINE and START commands, see [z/VM: RSCS Networking Operation and Use](#).

Operands

linkid

is the name of the TCPNJE-type link, as defined on a LINKDEFINE statement.

Buff=nnnnn

specifies the maximum buffer size the link can accept. This value can be 300 - 32765 bytes. The default is 4096. The actual buffer size used is determined at sign-on time and is the smaller of the buffer sizes specified or defaulted to by the two connected systems. If a small buffer size is specified or defaulted to by the remote node, it should be encouraged to specify a larger buffer.

The transfer rate of files across TCPNJE-type links is dependent on the buffer sizes defined for the link and within the TCP/IP stack. The available buffer sizes within the TCP/IP stack should match or be larger than the buffer size defined on the TCPNJE-type link PARM statement.

COMPRESS=

specifies the degree of data compression to be used when transmitting the file.

MIN

Minimum data compression. This is the default.

MAX

Maximum data compression.

DNSPort=portid

specifies the port number on the local system where the RSCS domain name server (RSCSDNS), if running, listens for connect requests. The default is 900.

Note: By default, TCP/IP restricts the use of well-known ports (1 - 1023). To use port 900 or any other well-known port when usage is restricted, the RSCSDNS server must be authorized on a PORT statement in the TCP/IP configuration file (PROFILE TCPIP). For more information, see [z/VM: TCP/IP Planning and Customization](#).

HOST=ipaddress

specifies the IP address (in dotted decimal format) of the remote host to which to connect.

If a HOST or HOSTNAME value is not defined on the PARM statement, it must be specified on the DEFINE or START command.

HOSTName=address

specifies the fully qualified domain name address (up to 200 characters) of the remote host to which to connect.

Note: Use of this operand requires the RSCS domain name server, which will resolve the domain name address to an IP address.

If a HOST or HOSTNAME value is not defined on the PARM statement, it must be specified on the DEFINE or START command.

ITO=nnn

specifies the inactivity time out (ITO) value, which defines the length of time that can occur with no activity (that is, no file transmission or reception) before the link is deactivated. If a link is ended because its ITO value has been exceeded, its restart exec will not be processed.

The ITO value can be 0 - 100:

0

The link will be deactivated immediately when there is no activity on it. If ITO=0 is specified for the local and remote nodes of the link and files are queued on one end of the link, the link will deactivate before the files are sent. If both sides of the link have files to process, the files will be sent.

1 - 99

Approximate number of minutes until the link will be deactivated if there has been no activity on it.

100

No ITO will be in effect. The link will stay active until it is deactivated manually or an unrecoverable error occurs. This is the default.

JOBName=

specifies the type of job name RSCS will create for all files originating at your local node and all files arriving at your local node without a job header (that is, those coming from a workstation link).

Rscs

RSCS will generate a job name of RSCSnnnn, where nnnn is the origin spool ID of the file.

Userid

RSCS will use the origin user ID as the job name.

If you specify this operand, it will override any previously defined default for this link. If you omit this operand, the default value is derived from the corresponding operand on the OPTION statement.

To override the default job name for a specific file, use the JOBNAM operand on the CP TAG command. For more information, see *z/VM: RSCS Networking Operation and Use*.

KEEPaliv=Yes

KEEPaliv=No

specifies whether keep alive packets will be sent when a connection has been idle for the keep alive interval. This allows the link driver to detect that the other side has terminated or become unreachable, even when no data is being exchanged. The keep alive interval is specified in the configuration file for the TCP/IP virtual machine. The default is YES.

LCLHost=ipaddress

specifies the IP address (in dotted decimal format) of the local OSA interface on which RSCS socket connect requests should be sent for this link. This IP address must match the IP address provided with the HOST operand on the remote side.

LCLPORT=portid

specifies the port number on the local system where the link listens for connect requests from the remote host. The default is 175.

Note: By default, TCP/IP restricts the use of well-known ports (1 - 1023). To use port 175 or any other well-known port when usage is restricted, the RSCS server must be authorized on a PORT statement in the TCP/IP configuration file (PROFILE TCP/IP). For more information, see *z/VM: TCP/IP Planning and Customization*.

LISTproc=No

LISTproc=Yes

specifies whether the connected system can handle an unlimited number of data set headers. If you specify NO, the link will use the number of data set headers specified on the MAXDSH operand.

If you specify this operand, it will override any previously defined default for this link. If you omit this operand, the default value is derived from the corresponding operand on the OPTION statement.

The following table summarizes how non-list processor systems will handle list processor files.

Table 21. How Non-List Processor Systems Handle List Processor Files

Non-List Processor Systems	How the File Will Be Handled
<ul style="list-style-type: none"> • RSCS V1.3* • JES3 • VSE/POWER® 	These systems will immediately split the file into separate spool files. These files will be either distributed to local user IDs or sent on to other links.
<ul style="list-style-type: none"> • RSCS V2.2* (without list processor support • RSCS V2.3* or later* (without a list processor defined) • JES2 	These systems will receive the file and split out separate files for local user IDs. They will also split out one file containing data set headers to be sent to each link on the distribution list.

Note: * These systems will handle only a limited number of data set headers.

MAXDsh=nnn

specifies the maximum number of data set headers the link driver will send on the link when processing a file with multiple data sets. This value is intended for connections to NJE subsystems that do not have list processor capabilities. This operand is ignored if LISTPROC=YES is specified.

This value can be 0 - 100. A value of 0 is a special case that means the determination for the number of data set headers placed on the link is based on the actual number of data set headers contained in the spool file. A value of 0 should be specified only with full understanding of the number of data set headers that the remote system can handle.

If you specify this operand, it will override any previously defined default for this link. If you omit this operand, the default value is derived from the corresponding operand on the OPTION statement.

MAXUro=nnn

specifies the maximum number of unit record output devices (virtual printers or punches) for each stream that the link driver will use at any one time while receiving a file. RSCS uses these devices to fan out files containing data set headers, which the link driver determines must be split off the original file. All remaining data set headers (and the file contents) are written to an additional unit record device and queued on the list processor link. This operand is ignored if a LISTPROC-type link is not defined.

This value can be 0 - 255. The default is 2. Specifying a value of 0 will cause all received data set headers and file contents to be queued to the list processor task. Under standard conditions, a link will always use one more device per stream than specified. If a value of 0 is specified and the first data set header received in a file is for the local system or for a non-networking link attached to the local system, an extra device will be defined to hold the remaining data set headers.

MSGSKip=nnn

specifies the number of file buffers that RSCS will send on the link before sending a message buffer (unless there are no file buffers available). This value can be 0 - 15. Specifying 0 is a special case that tells RSCS to send a message buffer whenever messages are available to send on the link, regardless of whether file buffers are available.

If you specify this operand, it will override any previously defined default for this link. If you omit this operand, the default value is derived from the corresponding operand on the OPTION statement.

Note: If this link determines during sign-on that the peer system supports the mixed RCB feature of NJE, RSCS ignores the MSGSKIP setting. The mixed RCB feature lets RSCS send buffers containing a mixture of information (files and messages). If RSCS can send files and messages in one buffer, there is no need to use the MSGSKIP operand to tell RSCS how many file buffers to send before sending

a message buffer. For more information about the mixed RCB feature of NJE, see *z/OS: Network Job Entry (NJE) Formats and Protocols* ([https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa320988/\\$file/hasa600_v2r5.pdf](https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa320988/$file/hasa600_v2r5.pdf)).

NUMBuff=*n*

has no meaning, and exists only to maintain compatibility with levels of RSCS prior to Version 3. Whether you specify or omit this operand, RSCS gives each link two transmission buffers.

OPTimize=No**OPTimize=Yes**

specifies whether optimization (merging of CCWs) should be used for files that originate on the local system and files that are processed by the local system. (This does not include store-and-forward files that are passing through the local node.) The default is NO.

RLPass=*password*

specifies the 1- to 8-character line password that is required from the remote system on its sign-on record before transmission of data is initiated. If the correct password is not received at sign-on time, RSCS deactivates the link. On an auto-answer link, RLPASS or RNPASS (or both) must be defined.

RNPass=*password*

specifies the 1- to 8-character node password that is required from the remote system on its sign-on record before transmission of data is initiated. If the correct password is not received at sign-on time, RSCS deactivates the link. On an auto-answer link, RLPASS or RNPASS (or both) must be defined.

RMTPORT=*portid*

specifies the port number to which to connect on the remote host. The default is 175.

Note: By default, TCP/IP restricts the use of well-known ports (1 - 1023). To use port 175 or any other well-known port when usage is restricted, the RSCS server must be authorized on a PORT statement in the TCP/IP configuration file (PROFILE TCPIP). For more information, see *z/VM: TCP/IP Planning and Customization*.

STreams=*n*

specifies the maximum number of files that may be transmitted simultaneously on different streams. This value can be 1 - 7. The default is 2.

Note:

1. RSCS cannot actually send more than one file at a time. When several files are active on different streams, RSCS sends the file with the smallest number of blocks and credits the other files for having to wait. After the file is transmitted, RSCS subtracts the credits from the number of blocks left to send for each active file. RSCS then chooses the file with the lowest number to send. The longer a large file has to wait, the more credits it will accumulate. When the difference between the accumulated credits and the actual file size becomes small enough, RSCS transmits the large file.
2. To allow the link to operate in a mode compatible with RSCS V1 (that is, only one file will be transmitted at a time and all files are eligible for selection), specify a stream number of 1 and a transmission algorithm number of 0 (TA=0, which is the default).

PI**TA=**

specifies the transmission algorithm, which tells RSCS how to select files for transmission on the link.

epname

is the name of the entry point (outside of the RSCS LOADLIB) where your installation-defined algorithm resides.

n

is a hexadecimal value, 0 - F, that identifies the exit routine in the DMTAXA module (DMTAXAG*n*):

0

Identifies an IBM-supplied algorithm that allows files to be selected regardless of size. This is the default.

Note: If you specify TA=0 or allow it to default, do not specify the TAPARM operand.

PARM (TCPNJE-Type)

1

Identifies an IBM-supplied algorithm that specifies a default size for each stream specified on the STREAMS operand. You can override the default by specifying the stream size on the TAPARM operand.

2 - F

Identifies an algorithm defined by your installation.

Note: If you update the DMTAXA module, it must be reassembled.

TAParm='value'

specifies a parameter value associated with the transmission algorithm. This value is a character string up to 80 bytes in length, enclosed in single quotation marks. (The single quotation marks are optional if you do not include blanks in the character string.) Any characters are allowed between the single quotation marks except another single quotation mark. The character string cannot be continued to the next line in the configuration file.

The value that you specify for TAPARM depends on the value that you specify for TA:

TA=

TAPARM=

0

Do not specify TAPARM; it will be rejected with an error message.

epname

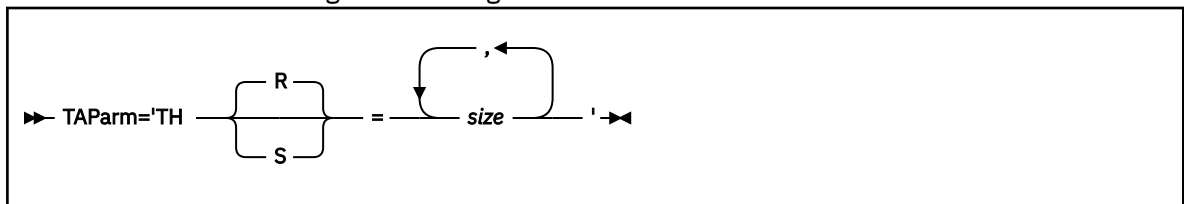
Specify a TAPARM defined by your installation.

2 - F

Specify a TAPARM defined by your installation.

1

Either omit TAPARM to accept the default size for each stream (listed in the table below) or specify a size for each stream using the following format:



TH=

THR=

specifies the threshold number of records. THR is the default.

THS=

specifies the threshold number of spool file blocks. Specify THS when you expect to send files with very large records, such as printer output files, containing page image data, that are destined for a PSF-controlled printer.

size

is the file size. This can be specified as either of the following:

nnnnnnnn

Maximum file size RSCS should send on the stream. The minimum file size is one more than the maximum file size from the previous stream specification. If there is no previous stream specification, there is no minimum file size.

(min,max)

Minimum and maximum file size RSCS should send on the stream. If you do not want to specify a minimum file size, omit *min*. If you do not want to specify a maximum file size, omit *max*. Specifying (,) means you do not want to specify either a minimum or a maximum file size; any file is eligible for transmission on the stream.

For example, if you specified STREAMS=3, you can choose the TAPARM default by omitting TAPARM or you can specify any one of the following:

TAPARM='TH=*size1*'
 TAPARM='TH=*size1,size2*'
 TAPARM='TH=*size1,size2,size3*'

TAPARM='THR=*size1*'
 TAPARM='THR=*size1,size2*'
 TAPARM='THR=*size1,size2,size3*'

TAPARM='THS=*size1*'
 TAPARM='THS=*size1,size2*'
 TAPARM='THS=*size1,size2,size3*'

If TA=1 is specified and TAPARM is omitted, these are the defaults:

If you specified	THR= Default for Stream 1	THR= Default for Stream 2	THR= Default for Stream 3	THR= Default for Stream 4	THR= Default for Stream 5	THR= Default for Stream 6	THR= Default for Stream 7
STREAMS=1	All files	—	—	—	—	—	—
STREAMS=2	(,100)	(101,)	—	—	—	—	—
STREAMS=3	(,100)	(101,)	(101,)	—	—	—	—
STREAMS=4	(,100)	(101,)	(101,)	(101,)	—	—	—
STREAMS=5	(,100)	(101,)	(101,)	(101,)	(101,)	—	—
STREAMS=6	(,100)	(101,)	(101,)	(101,)	(101,)	(101,)	—
STREAMS=7	(,100)	(101,)	(101,)	(101,)	(101,)	(101,)	(101,)

PI end

TCPid=*tcpip*

specifies the name of the TCP/IP virtual machine. The default name is TCPIP.

TLPass=*password*

specifies the 1- to 8-character line password to be transmitted to the remote system at sign-on time. TLPASS must be specified to connect to a remote system where RLPASS is defined.

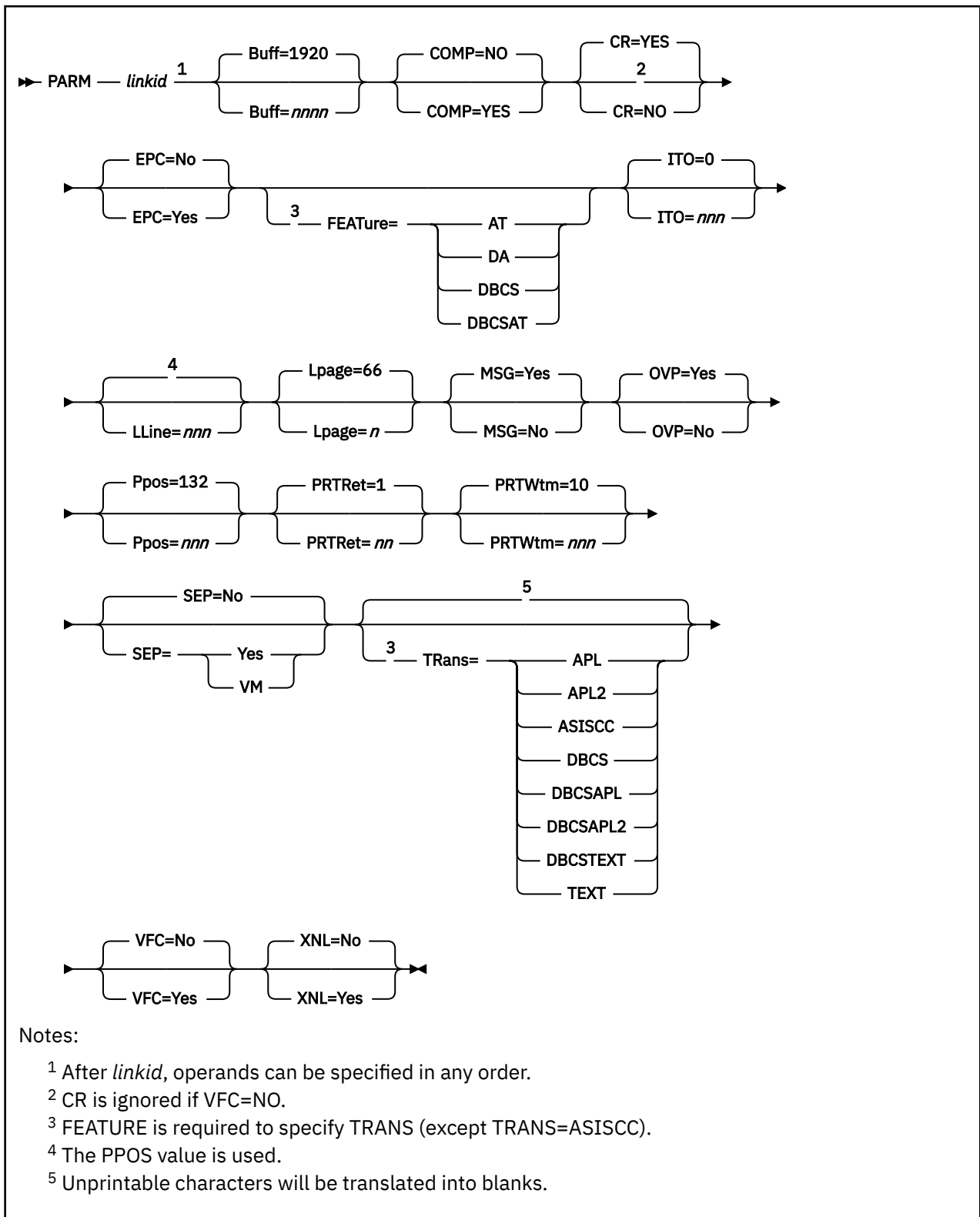
TLSLABEL=*label*

specifies the 1- to 8-character label of an x.509 digital certificate that will be used to encrypt/decrypt all data flowing over the link using TLS protocols. The specified certificate and its corresponding TLSLABEL must exist in the TLS/SSL Server certificate database. For additional information on the TLS/SSL Server and managing its certificate database, see [z/VM: TCP/IP User's Guide](#) and [z/VM: TCP/IP Planning and Customization](#).

TNPass=*password*

specifies the 1- to 8-character node password to be transmitted to the remote system at sign-on time. TNPASS must be specified to connect to a remote system where RNPASS is defined.

PARM Statement for a TN3270E-Type Link



Purpose

This PARM statement defines the operational characteristics for a specific TN3270E-type link. A TN3270E-type link starts a session with a remote IBM 3270 Information Display System printer connected to a TCP/IP network.

The link can be started:

- Automatically during system start-up
- Automatically when a file is queued for transmission on an auto-start link
- Manually by an operator or authorized user

Note: If you do not define a PARM statement for a TN3270E-type link, all of the indicated defaults will apply to the link. You can override PARM attributes by specifying values on the DEFINE or START command. For information on the DEFINE and START commands, see [z/VM: RSCS Networking Operation and Use](#).

Operands

linkid

is the name of the TN3270E-type link, as defined on a LINKDEFINE statement.

Buff=nnnn

specifies the size of the buffer in the 3271, 3272, 3274, or 3276 control unit being used. This value can be 480, 960, 1920, 2560, 3440, or 3564 bytes. The default is 1920.

COMP=No

COMP=Yes

specifies whether the link will perform blank compression. The default is NO. If you specify YES, occurrences of five or more blanks will be compressed. You should specify NO for a locally attached printer.

CR=Yes

CR=No

specifies whether the link should send a 3270 carriage return (CR) order after a forms feed (FF) when VFC=YES is specified. The default is YES. This operand has no meaning unless VFC=YES is also specified. Specify CR=NO when VFC=YES is specified and cut paper is used in a 5210 printer.

EPC=No

EPC=Yes

specifies whether Early Print Complete should be used by the link driver (for example, when you have a 6262 printer). The default is NO.

FEATure=

specifies that the printer has the indicated feature.

AT

APL/Text feature.

DA

Data Analysis-APL feature.

DBCS

Double-Byte Character Set feature.

DBCSAT

AT and DBCS features.

For the valid combinations of FEATURE and TRANS operand settings, see [Table 22 on page 239](#).

Note:

1. To use the TRANS operand and the PRT option on the CP TAG command, FEATURE must be specified. This rule does not apply to TRANS=ASISCC or to the ASISCC, GRAF, GRAPH, and NOTR settings on the PRT option, because they have no corresponding FEATURE setting.

2. Specifying the FEATURE operand as a default is helpful, even if you are specifying TRANS=ASISCC, because it lets you specify a TRANS setting as a holding position. You can then override the TRANS setting for a file by using the PRT option on the TAG command. If TRANS is not specified, then the only PRT settings allowed will be ASISCC, GRAPH, GRAF, and NOTR.
3. The DBCS feature processes the double-byte character set but does not translate any APL or TEXT representation that may exist in the file. The only translation done with the DBCS setting is 3270 control character removal. If you want APL or TEXT translation, use FEATURE=DBCSTAT with TRANS=DBCAPL, TRANS=DBCAPL2, or TRANS=DBCSTEXT. If APL, APL2, or TEXT is specified, the DBCS characters will not be verified or translated to the correct format for the printer output.

For more information about the PRT option of the CP TAG command, see [z/VM: RSCS Networking Operation and Use](#).

ITO=nnn

specifies the inactivity time out (ITO) value, which defines the length of time that can occur with no activity (that is, no file transmission or reception) before the link is deactivated. If a link is ended because its ITO value has been exceeded, its restart exec will not be processed.

The ITO value can be 0 - 100:

0

The link will be deactivated immediately when there is no it. This is the default. However, before deactivating the link, RSCS will first send files that are queued for transmission, or finish reception of the file currently being received.

1 - 99

Approximate number of minutes until the link will be deactivated if there has been no activity on it.

100

No ITO will be in effect. The link will stay active until it is deactivated manually or an unrecoverable error occurs.

LLine=nnn

specifies the maximum line length (number of characters per line). LLINE is a logical maximum, which is determined by the size of the form being used in the printer. This value can be 1 - 220. All print lines are truncated at the length specified by this value.

If LLINE is not specified, or if the LLINE value is greater than the PPOS value, LLINE is ignored and the PPOS value is used. If a file is sent to a printer with a character width equal to the LLINE value, the output will be truncated to the PPOS width.

If the LLINE value is less than the PPOS value, the LLINE value will override the PPOS value and will remain as the printer default for its active duration. If a number less than 74 is specified for LLINE, no separator page is printed.

Note: The 5210 printer accepts print wheels that allow it to print in various size fonts. Therefore, the font size on the print wheel must be considered when you specify the LLINE value. For example, if the 5210 is using a 10-pitch font print wheel, the LLINE value should be 85. However, if a 12-pitch font print wheel is being used, the LLINE value should be greater than 85.

Lpage=nn

specifies the number of lines per page on the type of form inserted in the printer. This value can be 0 - 99. The default is 66. A value of 0 specifies that no page eject will be done when a page eject is found in the file.

Note: This value is used by RSCS only to determine the current vertical position on a page while printing a file. Any actual page ejects are determined by Skip-to-Channel-1 CCWs within the file. If there are no Skip-to-Channel-1 CCWs in the file, then no page ejects are done, except between files. The actual number of lines contained in the file between consecutive Skip-to-Channel-1 CCWs must be equal to or less than this value.

MSG=

specifies whether RSCS will allow the link driver to dequeue messages destined for transmission to the remote printer.

Yes

RSCS will select all messages for transmission. This is the default.

No

RSCS will select messages, but they will not be sent to the remote printer. The messages will be purged.

OVP=Yes**OVP=No**

specifies whether the link will allow overprinting to occur (such as underscored words or overstruck characters for highlighting). The default is YES. Specify NO for a printer or control unit that does not support the 3270 carriage return (CR) order.

Ppos=nnn

specifies the maximum number of print positions available on the printer. This value can be 1 - 220, depending on the printer. The default is 132.

PPOS is a hardware maximum determined by the number of characters the printer can print on each line, independent of the size of the form used in the printer. Each printer has its own PPOS value. See the printer documentation for more information. If the PPOS value you specify does not match the printer's PPOS value, some types of printed output (including separator pages) will not print correctly.

PRTRet=nn

specifies the number of hours the link will attempt error recovery due to an unattached device before terminating. This value can be 1 - 100. The default is 1.

PRTWtm=nnn

specifies the number of seconds between recovery attempts due to an unattached device. The default is 10.

SEP=

specifies the type of separator page to be inserted before each print file sent across the link.

No

No separator page. This is the default.

Yes

RSCS-style separator page.

VM

VM-style separator page.

For information on separator page formats, see [z/VM: RSCS Networking Operation and Use](#).

TRans=

specifies the default translation mode for files being transmitted to a printer that has the AT, DA, DBCS, or DBCSAT feature (as specified on the FEATURE operand). The translation mode tells RSCS that all files may contain the indicated special characters and that RSCS should translate these special characters accordingly. Alternatively, you can use this operand to specify that unprintable characters should not be translated into blanks.

The default translation mode you specify on TRANS can be overridden for a specific file by using the PRT option on the CP TAG command. For more information about the CP TAG command, see [z/VM: RSCS Networking Operation and Use](#).

The valid translation modes are:

APL

If RSCS detects internally represented EBCDIC special APL characters in a file, RSCS will translate those characters into the appropriate 2-byte I/O interface codes.

APL2

If RSCS detects internally represented EBCDIC special APL2 characters in a file, RSCS will translate those characters into the appropriate 2-byte I/O interface codes.

This mode is valid only when FEATURE is set to AT or DBCSAT. If TRANS=APL2 is specified with any other FEATURE setting, it is marked as an incorrect combination and an error message is issued.

ASISCC

RSCS will not translate unprintable characters into blanks. [Table 22 on page 239](#) shows that ASISCC can be used with all FEATURE settings.

TRANS=ASISCC means the default is no translation. If a file needs translation, you can override this default by using the PRT option on the CP TAG command (assuming that you specified the correct FEATURE setting). If FEATURE is not specified, then only the ASISCC, GRAF, GRAPH, or NOTR setting can be used on the TAG command.

Note: TRANS=ASISCC can be specified without a FEATURE setting.

DBCS

RSCS will verify double-byte EBCDIC character strings delimited by SO/SI (shift-out/shift-in) characters for valid DBCS syntax before transmitting a file.

DBCSAPL

RSCS will translate internally represented EBCDIC special APL characters into the appropriate 2-byte I/O interface codes and will verify all double-byte EBCDIC character strings delimited by SO/SI (shift-out/shift-in) characters for valid DBCS syntax before transmitting a file.

DBCSAPL2

RSCS will translate internally represented EBCDIC special APL2 characters into the appropriate 2-byte I/O interface codes and will verify all double-byte EBCDIC character strings delimited by SO/SI (shift-out/shift-in) characters for valid DBCS syntax before transmitting a file.

This mode is valid only when FEATURE is set to DBCSAT. If TRANS=DBCSAPL2 is specified with any other FEATURE setting, it is marked as an incorrect combination and an error message is issued.

DBCSTEXT

RSCS will translate internally represented EBCDIC special TEXT characters into the appropriate 2-byte I/O interface codes and will verify double-byte EBCDIC character strings delimited by SO/SI (shift-out/shift-in) characters for valid DBCS syntax before transmitting a file.

TEXT

RSCS will translate internally represented EBCDIC special TEXT characters into the appropriate 2-byte I/O interface codes.

For the valid combinations of FEATURE and TRANS settings, see [Table 22 on page 239](#).

Note:

1. If you omit the TRANS operand, the usual character translation occurs (unprintable characters will be translated into blanks).
2. RSCS ignores the TRANS operand unless you also specify the corresponding FEATURE setting. TRANS=ASISCC is the exception; it does not require a FEATURE setting.
3. RSCS does not perform blank compression within DBCS SO/SI characters.
4. RSCS translates only those characters identified in the list above. RSCS will not change the standard EBCDIC data.

VFC=

specifies whether the printer will use the 3270 Vertical Forms Control (VFC) feature for vertical spacing.

No

Vertical spacing will be achieved through multiple 3270 new line (NL) orders. This is the default.

Yes

The printer will use the 3270 VFC feature and the 3270 carriage control (CC) order. VFC=YES is valid only for those printers and control units that support the 3270 CC order.

Note: Generally, IBM recommends that you specify VFC=YES. If users send GDDM or IPDS graphic files to this printer, you must specify VFC=YES to ensure that the pages align correctly.

XNL=No

XNL=Yes

specifies whether the link will include an extra new line (NL) order following each line that is as long as the maximum number of print positions. The default is NO. Specify YES for those printers (such as the 3262, 3289, 4234, and 6262) that do not generate an extra new line function for a maximum length line.

Usage Notes

1. When you define a form name using the FORM statement, you can define the these characteristics:

- Separator page style
- Line length
- Page length

If you specify a form name when starting a link, be aware that the form name characteristics always override any SEP, LLINE, and LPAGE specifications and will override LPI specifications when SETPAGE=YES.

2. If you specify an incorrect combination of FEATURE and TRANS settings, the SCO receives error message DMT807E and the link is deactivated.
3. [Table 22 on page 239](#) indicates the valid combinations of FEATURE and TRANS settings.

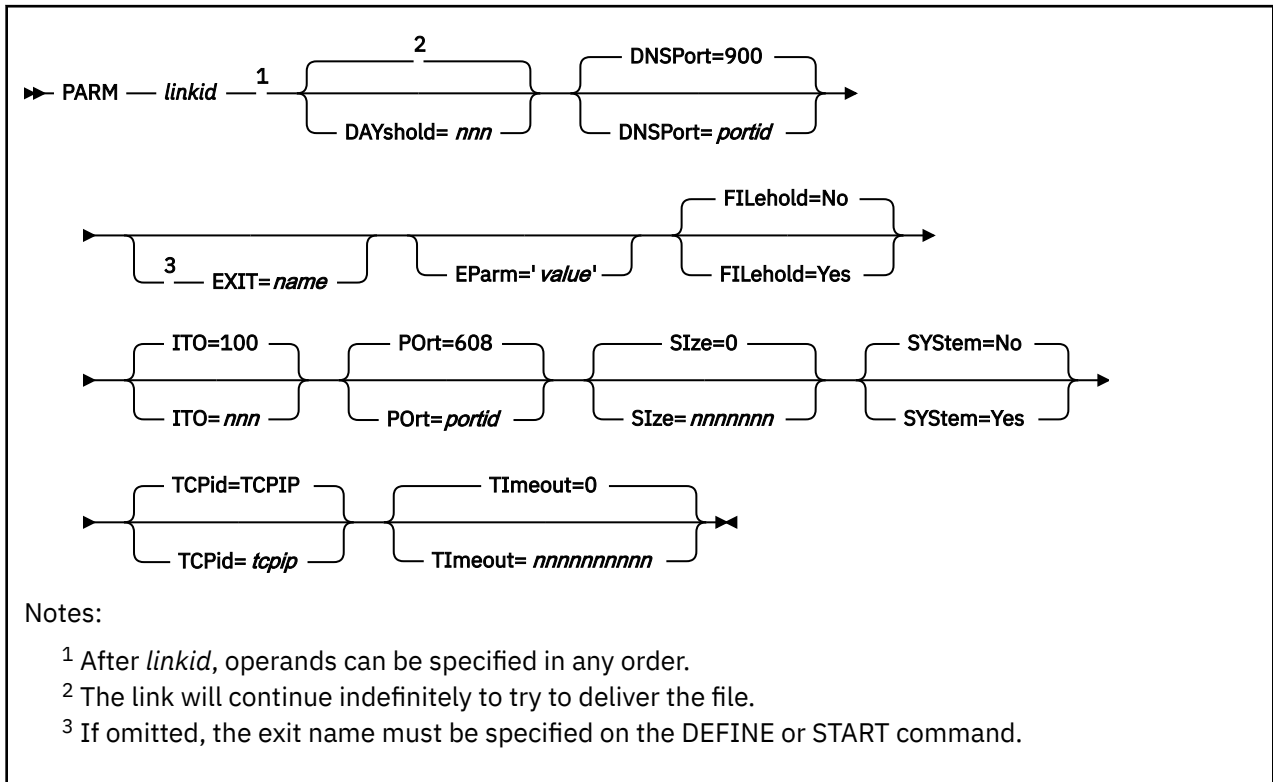
Table 22. Valid Combinations of FEATURE and TRANS Settings

TRans=	FEATure= DA	FEATure= AT	FEATure= DBCS	FEATure= DBCSAT
APL	X	X		X
APL2		X		X
TEXT	X	X		X
DBCS			X	X
DBCSAPL				X
DBCSAPL2				X
DBCSTEXT				X
ASISCC	X	X	X	X

Note:

- a. If APL, APL2, or TEXT is specified for the TRANS operand, and the FEATURE operand is set to DBCSAT, then there is no DBCS translation of the file, only APL, APL2, or TEXT. The DBCS translation will occur with APL, APL2, or TEXT translation only when the TRANS setting is DBCSAPL, DBCSAPL2, or DBCSTEXT.
- b. TRANS=ASISCC does not require the FEATURE operand to be specified.

PARM Statement for a UFT-Type Link



Purpose

This PARM statement defines the operational characteristics for a specific UFT-type link. A UFT-type link connects the local RSCS system to a remote UFT daemon in a TCP/IP network.

The link can be started:

- Automatically during system start-up
- Automatically when a file is queued for transmission on an auto-start link
- Manually by an operator or authorized user

Note: If you do not define a PARM statement for a UFT-type link, all of the indicated defaults will apply to the link. You can override PARM attributes by specifying values on the DEFINE or START command. Some required link attributes do not have PARM defaults; if a value is not defined on the PARM statement, it must be specified on the DEFINE or START command, or the link will not activate on the START command. For information on the DEFINE and START commands, see [z/VM: RSCS Networking Operation and Use](#).

Operands

linkid

is the name of the UFT-type link, as defined on a LINKDEFINE statement.

DAYshold=*nnn*

specifies the number of days that an undeliverable file will be placed on hold before RSCS re-queues it to a NOTIFY-type link. This value can be 0 - 365. In order to re-queue files to a NOTIFY-type link, a route for *UFTH* must be defined; for example:

```
ROUTE *UFTH* TO *UFTHOLD
```

This operand must be used with an events file entry which periodically changes any files on hold to nohold for the UFT-type link. If you specify a value of 0, the UFT-type link will requeue files to a

NOTIFY-type link every time it places them on hold. If you omit this operand, the UFT-type link will continue indefinitely to try to deliver the file.

DNSPort=portid

specifies the port number on the local system where the RSCS domain name server (RSCSDNS), if running, listens for connect requests. The default is 900.

Note: By default, TCP/IP restricts the use of well-known ports (1 - 1023). To use port 900 or any other well-known port when usage is restricted, the RSCSDNS server must be authorized on a PORT statement in the TCP/IP configuration file (PROFILE TCP/IP). For more information, see [z/VM: TCP/IP Planning and Customization](#).

PI**EXIT=name**

specifies the name of the exit routine module that contains the exit routines supporting this UFT link driver. The exit routine module must reside in a LOADLIB that has been identified by using the GCS GLOBAL command. If you omit this operand and the exit name is not specified on the DEFINE or START command, or if RSCS cannot find the exit routine module, the link driver will not start.

IBM supplies a sample exit routine module called UFTXOUT in the RSCSEXIT LOADLIB. For information about the UFT exit routines, see [z/VM: RSCS Networking Exit Customization](#).

EParm='value'

specifies a parameter value associated with the exit routine module. This value is a character string up to 239 bytes in length, enclosed in single quotation marks. Any character is allowed between the single quotation marks except another single quotation mark. The character string cannot be continued to the next line in the configuration file. The actual value of the character string is defined by the exit routine module.

For information about the EPARM parameters supported for use with the UFTXOUT sample module, see [z/VM: RSCS Networking Exit Customization](#).

PI end**FILEhold=No****FILEhold=Yes**

specifies whether the link should place a file on hold when it is unable to connect to a remote UFT daemon after approximately 1 minute. The default is NO, which means the link will continue attempting to connect until successful or the link is stopped (with the STOP command).

If you specify YES, the link will attempt to connect to the remote daemon at 1, 2, 4, 8, 16, and 32 second intervals. If the link is still unable to connect, the file will be placed on hold. The RSCS events file can be used to periodically change any files on hold to nohold, allowing for another delivery attempt. For example, an RSCS events file entry which will change all files on hold to nohold for a link called UFT every 10 minutes would look like this:

```
*      M:010      *      *      *      *      CHANGE UFT HOLD NOHOLD
```

ITO=nnn

specifies the inactivity time out (ITO) value, which defines the length of time that can occur with no activity (that is, no file transmission or reception) before the link is deactivated. If a link is ended because its ITO value has been exceeded, its restart exec will not be processed.

The ITO value can be 0 - 100:

0

The link will be deactivated immediately when there is no activity on it. However, before deactivating the link, RSCS will first send files that are queued for transmission, or finish reception of the file currently being received.

1 - 99

Approximate number of minutes until the link will be deactivated if there has been no activity on it.

100

No ITO will be in effect. The link will stay active until it is deactivated manually or an unrecoverable error occurs. This is the default.

Port=portid

specifies the port number to which to connect on the remote host. The default is 608.

Note: By default, TCP/IP restricts the use of well-known ports (1 - 1023). To use port 608 or any other well-known port when usage is restricted, the RSCS server must be authorized on a PORT statement in the TCP/IP configuration file (PROFILE TCP/IP). For more information, see [z/VM: TCP/IP Planning and Customization](#).

Size=nnnnnnnn

specifies the maximum number of records the spool file may contain in order to be processed. This value can be 0 - 1048576. The default is 0, which indicates there is no limit. If the number of records exceeds the maximum allowed amount, an error message will be issued to the user and the console, and the file will be placed on hold.

SYStem=No

SYStem=Yes

specifies whether RSCS should search for a FORM table provided by the LPRXFORM exit, using the form name the file was spooled with. The default is NO. When sending a print file to a remote UFT daemon, the FORM table can provide keywords to define the remote host address, translate table, and substitute form to be used for the file, overriding the values defined for the link.

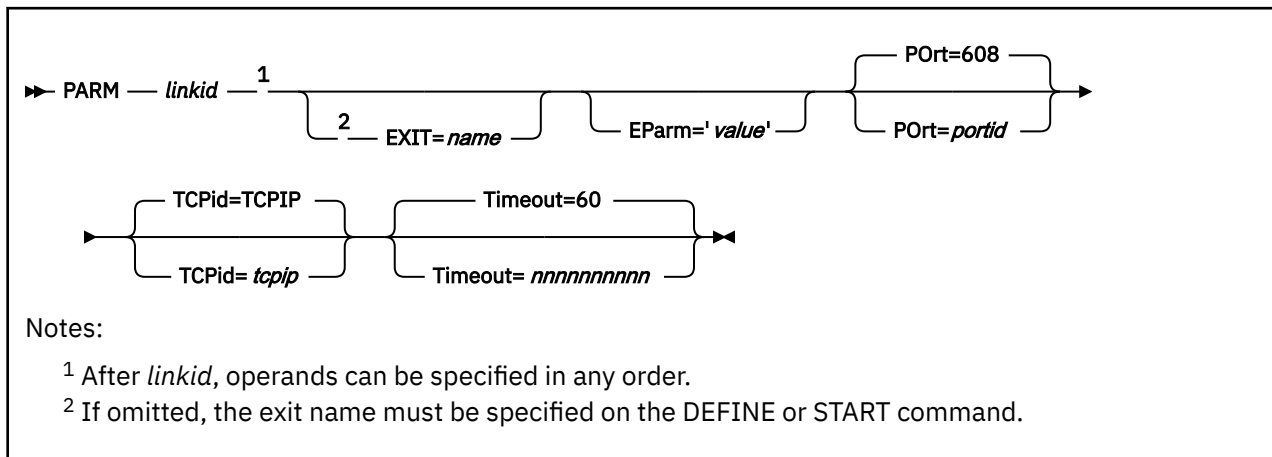
TCPid=tcPIP

specifies the name of the TCP/IP virtual machine. The default name is TCPIP.

TImeout=nnnnnnnnnn

specifies the maximum amount of time, in seconds, to wait for a response from the remote daemon before halting delivery of the file and placing it on hold. This value can be 0 - 2147483647. The default is 0, which means the link will wait indefinitely.

PARM Statement for a UFTD-Type Link



Purpose

This PARM statement defines the operational characteristics for a specific UFTD-type link. A UFTD-type link acts as a gateway, allowing the local RSCS system to accept data streams from a UFT client in a TCP/IP network.

The link can be started:

- Automatically during system start-up
- Manually by an operator or authorized user

Attention
Any file queued to this link will be ignored (not processed).

Note: If you do not define a PARM statement for a UFTD-type link, all of the indicated defaults will apply to the link. You can override PARM attributes by specifying values on the DEFINE or START command. Some required link attributes do not have PARM defaults; if a value is not defined on the PARM statement, it must be specified on the DEFINE or START command, or the link will not activate on the START command. For information on the DEFINE and START commands, see [z/VM: RSCS Networking Operation and Use](#).

Operands

linkid

is the name of the UFTD-type link, as defined on a LINKDEFINE statement.

PI

EXIT=name

specifies the name of the exit routine module that contains the exit routines supporting this UFTD link driver. The exit routine module must reside in a LOADLIB that has been identified by using the GCS GLOBAL command. If you omit this operand and the exit name is not specified on the DEFINE or START command, or if RSCS cannot find the exit routine module, the link driver will not start.

IBM supplies a sample exit routine module called UFTXIN in the RSCSEXIT LOADLIB. For information about the UFTD exit routines, see [z/VM: RSCS Networking Exit Customization](#).

EParm='value'

specifies a parameter value associated with the exit routine module. This value is a character string up to 239 bytes in length, enclosed in single quotation marks. Any character is allowed between the single quotation marks except another single quotation mark. The character string cannot be

PARM (UFTD-Type)

continued to the next line in the configuration file. The actual value of the character string is defined by the exit routine module.

For information about the EPARM parameters supported for use with the UFTXIN sample module, see [*z/VM: RSCS Networking Exit Customization*](#).

PI end

Port=portid

specifies the port number on the local host where RSCS will listen for connect requests. The default is 608.

Note: By default, TCP/IP restricts the use of well-known ports (1 - 1023). To use port 608 or any other well-known port when usage is restricted, the RSCS server must be authorized on a PORT statement in the TCP/IP configuration file (PROFILE TCP/IP). For more information, see [*z/VM: TCP/IP Planning and Customization*](#).

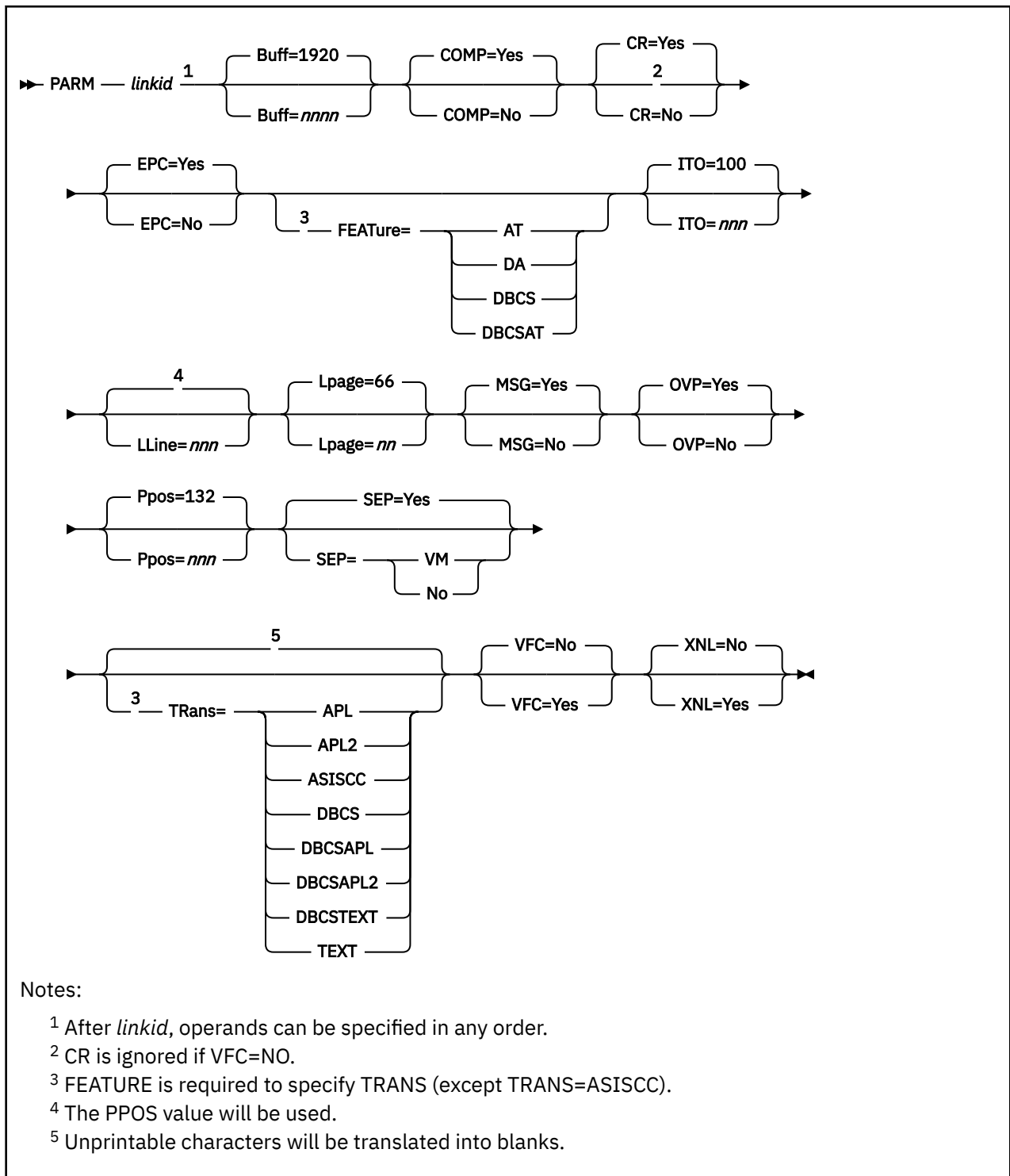
TCPid=tcPIP

specifies the name of the TCP/IP virtual machine. The default name is TCPIP.

Timeout=nnnnnnnnnn

specifies the maximum amount of time, in seconds, that RSCS will wait for data from a TCP/IP UFT client before breaking the socket connection. This value can be 0 - 2147483647. The default is 60. A value of 0 means to wait indefinitely. After the UFTD link driver breaks the connection, it will remain operational waiting for another UFT client to connect.

PARM Statement for a 3270P-Type Link



Purpose

This PARM statement defines the operational characteristics for a specific 3270P-type link. A 3270P-type link starts a session with a remote IBM 3270 Information Display System printer.

The link can be started:

- Automatically during system start-up

PARM (3270P-Type)

- Automatically when a file is queued for transmission on an auto-start link
- Manually by an operator or authorized user

Note: If you do not define a PARM statement for a 3270P-type link, all of the indicated defaults will apply to the link. You can override PARM attributes by specifying values on the DEFINE or START command. For information on the DEFINE and START commands, see [z/VM: RSCS Networking Operation and Use](#).

Operands

linkid

is the name of the 3270P-type link, as defined on a LINKDEFINE statement.

Buff=nnnn

specifies the size of the buffer in the 3271, 3272, 3274, or 3276 control unit. This value can be 480, 960, 1920, 2560, 3440, or 3564 bytes. The default is 1920. The correct buffer size must be specified to match the buffer size on the control unit being used.

COMP=Yes

COMP=No

specifies whether the link will perform blank compression. The default is YES; occurrences of five or more blanks will be compressed. Specify NO for a locally attached printer.

CR=Yes

CR=No

specifies whether the link should send a 3270 carriage return (CR) order after a forms feed (FF) when VFC=YES is specified. The default is YES. This operand has no meaning unless VFC=YES is also specified. Specify CR=NO when VFC=YES is specified and cut paper is used in a 5210 printer.

EPC=Yes

EPC=No

specifies whether Early Print Complete should be used by the link driver (for example, when you have a 6262 printer). The default is YES.

FEATure=

specifies that the printer has the specified feature.

AT

APL/Text feature.

DA

Data Analysis-APL feature.

DBCS

Double-Byte Character Set feature.

DBCSAT

AT and DBCS features.

For the valid combinations of FEATURE and TRANS operand settings, see [Table 23 on page 250](#).

Note:

1. To use the TRANS operand and the PRT option on the CP TAG command, FEATURE must be specified. This rule does not apply to TRANS=ASISCC or to the ASISCC, GRAF, GRAPH, and NOTR settings on the PRT option, because they have no corresponding FEATURE setting.
2. Specifying the FEATURE operand as a default is helpful, even if you are specifying TRANS=ASISCC, because it lets you specify a TRANS setting as a holding position. You can then override the TRANS setting for a file by using the PRT option on the TAG command. If TRANS is not specified, then the only PRT settings allowed will be ASISCC, GRAF, GRAPH, and NOTR.
3. The DBCS feature processes the double-byte character set but does not translate any APL or TEXT representation that may exist in the file. The only translation done with the DBCS setting is 3270 control character removal. If you want APL or TEXT translation, use FEATURE=DBCSAT with TRANS=DBCSAPL, TRANS=DBCSAPL2, or TRANS=DBCSTEXT. If APL, APL2, or TEXT is specified, the DBCS characters will not be verified or translated to the correct format for the printer output.

For more information about the PRT option of the CP TAG command, see *z/VM: RSCS Networking Operation and Use*.

ITO=nnn

specifies the inactivity time out (ITO) value, which defines the length of time that can occur with no activity (that is, no file transmission or reception) before the link is deactivated. If a link is ended because its ITO value has been exceeded, its restart exec will not be processed.

The ITO value can be 0 - 100:

0

The link will be deactivated immediately when there is no activity on it. However, before deactivating the link, RSCS will first send files that are queued for transmission, or finish reception of the file currently being received.

1 - 99

Approximate number of minutes until the link will be deactivated if there has been no activity on it.

100

No ITO will be in effect. The link will stay active until it is deactivated manually or an unrecoverable error occurs. This is the default.

LLine=nnn

specifies the maximum line length (number of characters per line). LLINE is a logical maximum, which is determined by the size of the form being used in the printer. This value can be 1 - 220. All print lines are truncated at the length specified by this value.

If LLINE is not specified, or if the value of LLINE is greater than the PPOS value, LLINE is ignored and the PPOS value is used. If a file is sent to a printer with a character width equal to the LLINE value, the output will be truncated to the PPOS width.

If the LLINE value is less than the PPOS value, the LLINE value will override the PPOS value and will remain as the printer default for its active duration. If a number less than 74 is specified for LLINE, no separator page is printed.

Note: The 5210 printer accepts several print wheels that allow it to print in various size fonts. Therefore, the font size on the print wheel must be considered when you specify the LLINE value. For example, if the 5210 is using a 10-pitch font print wheel, the LLINE value should be 85. However, if a 12-pitch font print wheel is being used, the LLINE value should be greater than 85.

Lpage=nn

specifies the number of lines per page on the type of form inserted in the printer. This value can be 0 - 99. The default is 66. A value of 0 specifies that no page ejects will be done when a page eject is found in the file.

Note: This value is used by RSCS only to determine the current vertical position on a page while printing a file. Any actual page ejects are determined by Skip-to-Channel-1 CCWs within the file. If there are no Skip-to-Channel-1 CCWs in the file, no page ejects are done, except between files. The actual number of lines contained in the file between consecutive Skip-to-Channel-1 CCWs must be equal to or less than this value.

MSG=

specifies whether RSCS will allow the link driver to dequeue messages destined for transmission to the local printer.

Yes

RSCS will select all messages for transmission. This is the default.

No

RSCS will select messages, but they will not be sent to the local printer. The messages will be purged.

OVP=Yes

OVP=No

specifies whether the link will allow overprinting to occur (such as underscored words or overstruck characters for highlighting). The default is YES. Specify NO for a printer or control unit that does not support the 3270 carriage return (CR) order.

Ppos=nnn

specifies the maximum number of print positions available on the printer. This value can be 1 - 220, depending on the printer. The default is 132.

PPOS is a hardware maximum determined by the number of characters the printer can print on each line, independent of the size of the form used in the printer. Each printer has its own PPOS value. See the printer documentation for more information. If the PPOS value you specify does not match the printer's PPOS value, some types of printed output (including separator pages) will not print correctly.

SEP=

specifies the type of separator page to be inserted before each print file sent across the link.

Yes

RSCS-style separator page. This is the default.

No

No separator page.

VM

VM-style separator page.

For information on separator page formats, see *[z/VM: RSCS Networking Operation and Use](#)*.

Note: This operand applies only to print files. When transmitting a punch file to a remote workstation, RSCS creates a header line on the output. For information about the format of this header line, see *[z/VM: RSCS Networking Operation and Use](#)*.

TRans=

specifies the default translation mode for files being transmitted to a printer that has the AT, DA, DBCS, or DBCSAT feature (as specified on the FEATURE operand). The translation mode tells RSCS that all files may contain the indicated special characters and that RSCS should translate these special characters accordingly. Alternatively, you can use this operand to specify that unprintable characters should not be translated into blanks.

The default translation mode you specify on TRANS can be overridden for a specific file by using the PRT option on the CP TAG command. For more information about the CP TAG command, see *[z/VM: RSCS Networking Operation and Use](#)*

The valid translation modes are:

APL

If RSCS detects internally represented EBCDIC special APL characters in a file, RSCS will translate those characters into the appropriate 2-byte I/O interface codes.

APL2

If RSCS detects internally represented EBCDIC special APL2 character in a file, RSCS will translate those characters into the appropriate 2-byte I/O interface codes.

This mode is valid only when FEATURE is set to AT or DBCSAT. If TRANS=APL2 is specified with any other FEATURE setting, it is marked as an incorrect combination and an error message is issued.

ASISCC

RSCS will not translate unprintable characters into blanks. [Table 23 on page 250](#) shows that ASISCC can be used with all FEATURE settings.

TRANS=ASISCC means the default is no translation. If a file needs translation, you can override this default by using the PRT option on the CP TAG command (assuming you specified the correct FEATURE setting). If FEATURE is not specified, then only the ASISCC, GRAF, GRAPH, or NOTR setting can be used on the TAG command.

Note: TRANS=ASISCC can be specified without a FEATURE option.

DBCS

RSCS will verify double-byte EBCDIC character strings delimited by SO/SI (shift-out/shift-in) characters for valid DBCS syntax before transmitting a file.

DBCSAPL

RSCS will translate internally represented EBCDIC special APL characters into the appropriate 2-byte I/O interface codes and will verify all double-byte EBCDIC character strings delimited by SO/SI (shift-out/shift-in) characters for valid DBCS syntax before transmitting a file.

DBCSAPL2

RSCS will translate internally represented EBCDIC special APL2 characters into the appropriate 2-byte I/O interface codes and will verify all double-byte EBCDIC character strings delimited by SO/SI (shift-out/shift-in) characters for valid DBCS syntax before transmitting a file.

This mode is only valid when FEATURE is set to DBCSAT. If TRANS=DBCSAPL2 is specified with any other FEATURE setting, it is marked as an incorrect combination and an error message is issued.

DBCSTEXT

RSCS will translate internally represented EBCDIC special TEXT characters into the appropriate 2-byte I/O interface codes and will verify double-byte EBCDIC character strings delimited by SO/SI (shift-out/shift-in) characters for valid DBCS syntax before transmitting a file.

TEXT

RSCS will translate internally represented EBCDIC special TEXT characters into the appropriate 2-byte I/O interface codes.

For the valid combinations of TRANS and FEATURE operand settings, see [Table 23 on page 250](#).

Note:

1. If you omit the TRANS operand, the usual character translation occurs (unprintable characters will be translated into blanks).
2. RSCS ignores the TRANS setting unless you also specify the corresponding FEATURE setting. TRANS=ASISCC is the exception; it does not require a FEATURE setting.
3. RSCS does not perform blank compression within DBCS SO/SI characters.
4. RSCS translates only those characters identified in the list above. RSCS will not change the standard EBCDIC data.

VFC=

specifies whether the printer will use the 3270 Vertical Forms Control (VFC) feature for vertical spacing.

No

Vertical spacing will be achieved through multiple 3270 new line (NL) orders. This is the default.

Yes

The printer will use the 3270 VFC feature and the 3270 carriage control (CC) order. The page length must be set manually on the printer and the LPAGE value must match that setting. VFC=YES is valid only for those printers and control units that support the 3270 CC order.

Note: Generally, IBM recommends that you specify VFC=YES. If users send GDDM or IPDS graphic files to this printer, you must specify VFC=YES to ensure that the pages align correctly.

XNL=No

XNL=Yes

specifies whether the link will include an extra new line (NL) order after each line that is as long as the maximum number of print positions. The default is NO. Specify YES for those printers (such as the 3262, 3289, 4234, and 6262) that do not generate an extra new line function for a maximum length line.

Usage Notes

1. When you define a form name (using the FORM statement), you can define these characteristics:

- Separator page style
- Line length
- Page length

If you specify a form name when starting a link, be aware that the form name characteristics always override any SEP, LLINE, and LPAGE specifications and will override LPI specifications when SETPAGE=YES.

2. If you specify an incorrect combination of TRANS and FEATURE settings, the SCO receives error message DMT807E and the link is deactivated.

3. Table 23 on page 250 shows the valid combinations of TRANS and FEATURE settings.

Table 23. Valid Combinations of TRANS and FEATURE Settings

TRAns=	FEATure= DA	FEATure= AT	FEATure= DBCS	FEATure= DBCSAT
APL	X	X		X
APL2		X		X
TEXT	X	X		X
DBCS			X	X
DBCSAPL				X
DBCSAPL2				X
DBCSTEXT				X
ASISCC	X	X	X	X

Note:

- a. If APL, APL2, or TEXT is specified for the TRANS operand, and the FEATURE operand is set to DBCSAT, then there is no DBCS translation of the file, only APL, APL2, or TEXT. The DBCS translation will occur with APL, APL2, or TEXT translation 0 only when the TRANS setting is DBCSAPL, DBCSAPL2, or DBCSTEXT.
- b. TRANS=ASISCC does not require the FEATURE operand to be specified.

PORT Statement



Purpose

The PORT statement specifies the address of a switched telecommunication line that is to be used for either an auto-answer or auto-dial link. PORT is also used to specify a virtual address for a BSC communications line connection to a remote node. Addresses specified on PORT statements are kept in reserve for use if an address was not specified by the usual means (for example, on a LINK statement, or on a DEFINE or START command for a link). If no address is specified when the link is started, RSCS chooses one from its reserve, for dynamic allocation.

Note: This address is not the same value used on the PORT operand of PARM statements for TCP/IP links.

Specification

Include as many statements as needed; they are optional. PORT statements may be placed anywhere in the configuration file after the LOCAL and CHANNELS statements (if these statements exist).

Operands

ccuu

is the virtual device number, in the range X'0002' - X'FFFF', that represents a switched BSC line. The virtual address specified here cannot be within the RSCS virtual unit record device pool (default X'0F00' - X'0FFF', or as specified on the CHANNELS statement).

Note: Block multiplexer channel operations should be requested for the RSCS virtual machine if either of the following is true:

- Multiple ports are assigned to a channel.
- A port is assigned to a channel on which another RSCS-controlled device is defined.

See [“RSCS Virtual Machine” on page 31](#) for more information.

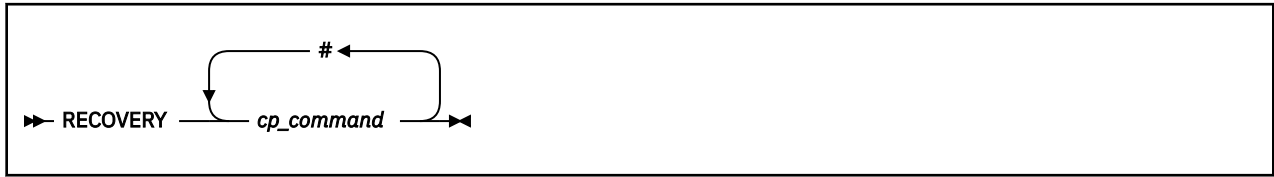
NODIAL

specifies that the port is not available for an auto-dial link. (NODIAL and * are equivalent.) This is the default.

DIAL

specifies that the switched BSC line on the port is connected to the public switched telephone network through a modem that has an attached Automatic Calling Unit (ACU), and that the port may be used for an auto-dial link.

RECOVERY Statement



Purpose

The RECOVERY statement specifies the CP commands that should be issued when an RSCS supervisor task abends, causing RSCS to become disabled.

Specification

One statement is allowed. It is optional and may be placed anywhere in the configuration file after the LOCAL statement (if one exists).

Operands

cp_command

is the CP command that RSCS will issue if RSCS becomes disabled. You must specify at least one CP command. If you specify more than one CP command, each command must be separated by the logical line end symbol (#).

For example, the following statement would cause RSCS to send a message to the operator and re-IPL GCS after the abend of a system task:

```
RECOVERY MSG OPERATOR RSCS abended; time to re-IPL#IPL GCS
```

Note that the logical line end symbol (#) separates the MSG command and the IPL command.

Usage Notes

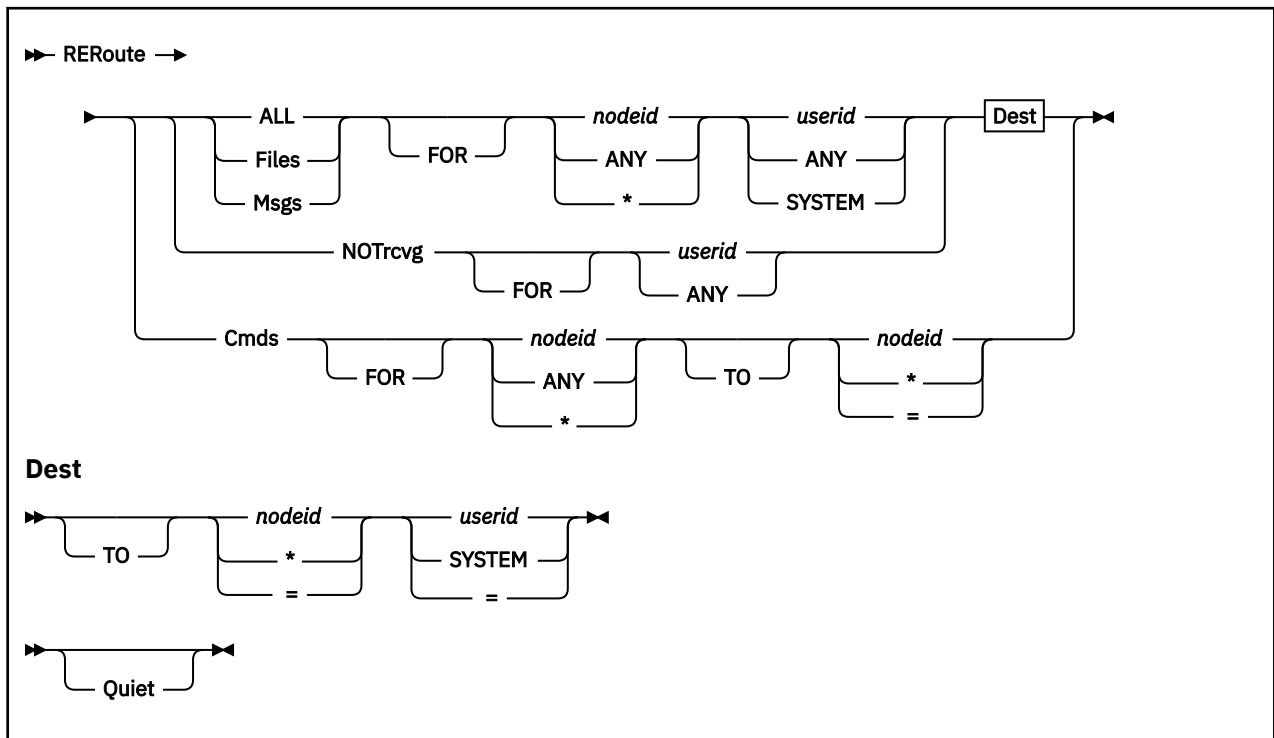
If you specify a CP command that resets the RSCS virtual machine (such as IPL), be aware that any CP commands you specify after that point will be erased. In the example above, the operator would never have received the message if you had reversed the order of the commands:

```
RECOVERY IPL GCS#MSG OPERATOR RSCS abended; time to re-IPL
```

RSCS would pass the IPL command to CP, which would re-IPL GCS, thereby wiping out the RSCS virtual machine and the MSG command.

To avoid this problem, always place commands that reset RSCS at the end of the RECOVERY command.

REROUTE Statement



Purpose

The REROUTE statement modifies the original destination of commands, files, and messages (including command responses) for specific systems and user IDs.

Specification

Include as many statements as needed; they are optional. REROUTE statements may be placed anywhere in the configuration file after the LOCAL statement (if one exists).

Operands

ALL

specifies that all files and messages will be rerouted. ALL has no effect on the rerouting of commands or the rerouting of messages because users are not receiving.

Files

specifies that only files will be rerouted.

Msgs

specifies that only messages and command responses will be rerouted.

FOR

specifies the original destination.

nodeid

is the original node. Instead of a node ID, you can specify:

ANY

indicates all nodes.

*

indicates the local node.

REROUTE

userid

is the original user ID or the link ID of a workstation or printer link. Instead of a user ID, you can specify:

ANY

indicates all users.

SYSTEM

indicates the system unit record device at the specified node or nodes.

NOTrcvg FOR

specifies that messages for a local user will be rerouted when the local user is unable to receive messages because of being logged off, not receiving, disconnected, and so forth. NOTRCVG has no effect on the rerouting of messages to users who are receiving. To reroute messages for a local user to a local user, specify an equal sign (=) as the node in the TO clause of the REROUTE statement.

userid

is the user ID or link ID of a workstation or printer link.

ANY

indicates all users who are not receiving.

TO

specifies the new destination.

nodeid

is the new node. Instead of a node ID, you can specify:

indicates the local node.

=

indicates the original destination node.

userid

is the new user ID or the link ID of a workstation or printer. Instead of a user ID, you can specify:

SYSTEM

indicates the system unit record device at the specified node.

=

indicates the original destination user ID.

Cmnds

specifies that only commands will be rerouted. CMDS has no effect on files or messages.

FOR

specifies the original destination.

nodeid

is the original node.

ANY

indicates all nodes.

indicates the local node.

TO

specifies the new destination.

nodeid

is the new node.

indicates the local node.

=

indicates the original destination node.

Quiet

specifies that RSCS will suppress messages DMT120I, DMT333I, and DMT336I when rerouting a file, message, or command.

Usage Notes

1. You can specify generic nodes and users by coding an asterisk (*) as the last character in the FOR *nodeid* or FOR *userid* clause. This causes RSCS to consider all identifiers that begin with the specified characters. For example, the following statement tells RSCS to reroute all files destined for nodes of the form NYCxxxx:

```
REROUTE FILES FOR NYC* ANY TO ...
```

2. You can specify generic reroutes and specific exceptions to those reroutes in any order. RSCS processes specific reroutes first and generic reroutes last. For example, suppose you had one specific reroute and two generic reroutes:

```
REROUTE ALL FOR NYC* ANY TO NEWYORK =
REROUTE ALL FOR NYCVM2 OPERATOR TO NYCVM1 =
REROUTE ALL FOR NYCVM* OPERATOR TO NYCVM1 OPER1
```

If a file came in for OPERATOR at NYCVM2, RSCS would reroute it to OPERATOR at NYCVM1 because the second reroute is the most specific. If a file came in for OPERATOR at NYCVM1, RSCS would reroute it to OPER1 at NYCVM1 because the third reroute is more specific than the first reroute. If a file came in for OPERATOR at NYCMVS1, RSCS would reroute it to OPERATOR at NEWYORK because the first reroute is the most specific reroute match that RSCS could find.

Note: RSCS processes generic reroutes by matching the destination node ID with the most specific node ID entry in the reroute table. After comparing node IDs, RSCS will compare user ID specifications.

3. You can specify generic reroutes and cancel (or block) a specific reroute. For example:

```
REROUTE ALL FOR NYC* ANY TO NEWYORK =
REROUTE MSGS FOR NYCVM2 OPERATOR TO = =
```

Again, the first statement tells RSCS to reroute all files and messages for all users at all NYCxxxx nodes to the same users at node NEWYORK. The second statement blocks the reroute for user OPERATOR at node NYCVM2 by rerouting those messages back to the same node ID and user ID.

4. When RSCS reroutes a command, file, or message, RSCS issues a message (DMT336I, DMT120I, or DMT333I, respectively) that tells the sender the new destination of the command, file, or message. If you do not want the sender to receive a message saying the command, file, or message was rerouted, use the QUIET operand to suppress the message.

Note: Specifying QUIET affects only one reroute entry at a time. If you specify a specific reroute (a specific node ID and user ID in the FOR clause), RSCS suppresses the reroute messages for that reroute only. If you specify a generic reroute, RSCS suppresses the reroute messages for all reroutes that match the generic specification. **When comparing destinations and generic reroutes, RSCS always sends traffic to the most specific reroute.** The only way to suppress messages for all reroutes is to specify the QUIET operand on every REROUTE statement and command.

5. RSCS reroutes files and messages once per node, unless there is a NOTRCVG reroute in effect, which causes RSCS to reroute a message a second time. For example, suppose you have the following reroutes in effect, and users CHUCK and MATT are not logged on:

```
REROUTE ALL FOR NEWYORK CHUCK TO = MATT
REROUTE ALL FOR NEWYORK MATT TO = JOHN
REROUTE NOTRCVG FOR MATT TO = JACK
```

If a file comes in for CHUCK, RSCS reroutes it to MATT because of the first statement in the example. Although the second statement tells RSCS to reroute everything for MATT to JOHN, RSCS will not reroute the file because it has already been rerouted once from CHUCK to MATT.

REROUTE

If, however, a message comes in for CHUCK, RSCS reroutes it to MATT and then to JACK because of the third statement in the example.

6. RSCS will reroute commands for the local node only if the commands originated from another system or if they are issued in the form:

```
smsg rscsid cmd local_nodeid command_text
```

7. When RSCS receives a propagating QUERY command (for example, QUERY FILES or QUERY *nodeid* PATH), RSCS tests the next node in the propagation to see whether the command should be rerouted.
8. Make sure that reroutes do not refer to each other. Reroute loops involve more than one node, which makes them difficult to find. For example, suppose you had three nodes with one operator at each node:

(At node BOSTON)

```
REROUTE NOTRCVG FOR OPERATOR TO NEWYORK =
```

(At node NEWYORK)

```
REROUTE NOTRCVG FOR OPERATOR TO LONDON =
```

(At node LONDON)

```
REROUTE NOTRCVG FOR OPERATOR TO BOSTON =
```

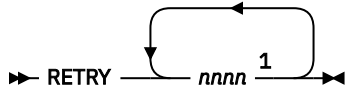
If all three operators were not receiving and a message came in for one of them, you would have a reroute loop.

9. REROUTE might cause problems if the external writer field (set with the CP TAG command) is used for something other than an MVS TSO user ID, because REROUTE changes this field if it matches the destination user field.
10. Use care when rerouting print files for LPR links. For example, the user ID field can be used to control whether the LPRXONE exit translates the data. However, when a user ID field is specified on the REROUTE statement, the LPRXONE exit might not translate the data as expected.
11. For printer links, two reroutes should be active:

```
REROUTE ALL FOR * OLDPRT TO * NEWPRT  
REROUTE ALL FOR OLDPRT ANY TO NEWPRT =
```

This will reroute all files arriving remotely as well as locally.

RETRY Statement



Notes:

¹ Up to 50 intervals can be specified.

Purpose

The RETRY statement tells RSCS when to restart a deactivated auto-dial or SNA link. If you specify the RETRY operand on the LINKDEFINE statement when defining a link, RSCS will automatically try to restart the link at specific intervals. The RETRY statement lets you specify the intervals for your local system.

Specification

One statement is allowed. It is optional and may be placed anywhere in the configuration file after the LOCAL statement (if one exists).

Operands

nnnn

is the retry interval for a deactivated auto-dial or SNA link. This value can be 1 - 1440 minutes. You can specify up to 50 intervals.

Usage Notes

1. If you have specified the RETRY operand for an auto-dial or SNA link and the link deactivates, RSCS will wait for as many minutes as you specify as the first interval. If the link does not restart, RSCS will wait for as many minutes as you specify for the second interval and so on. RSCS will continue to retry the link indefinitely until the link restarts or RSCS terminates. After RSCS has waited the amount of time you specified for the last interval, RSCS will continue to use the last interval as the waiting period. For example, if you specify:

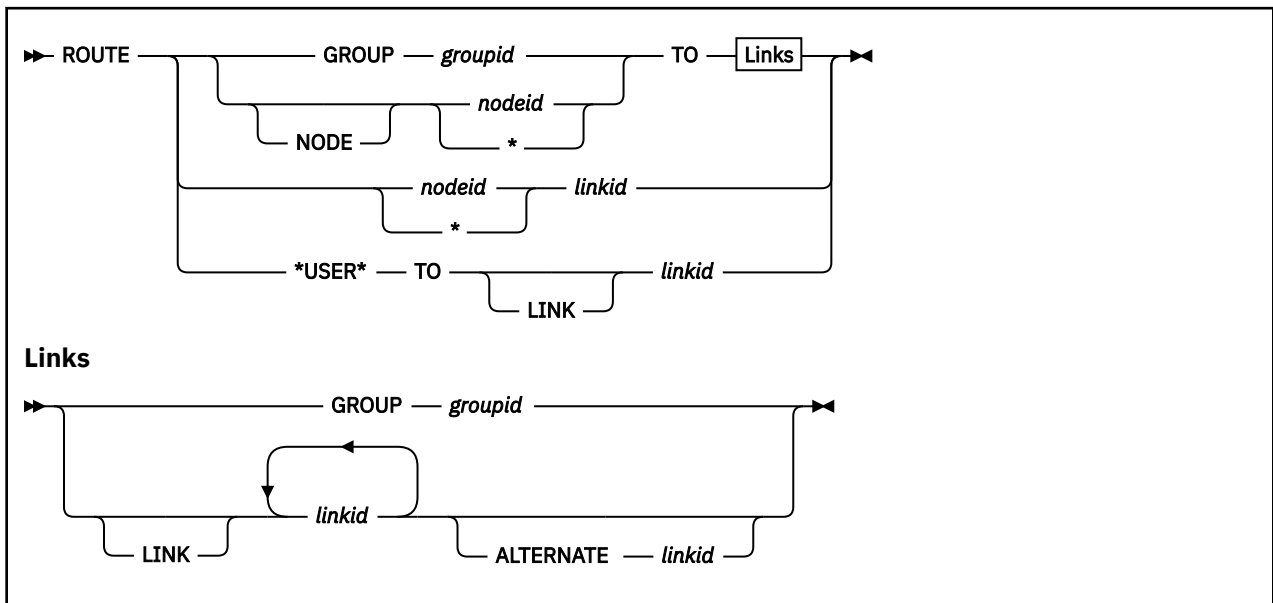
```
RETRY 1 4 5
```

RSCS will try to restart the link after 1 minute. If that fails, RSCS will wait 4 minutes (a total of 5 minutes since the link deactivated) and try to restart the link. If that fails, RSCS will wait 5 minutes (a total of 10 minutes since the link deactivated) and try to restart the link. If that fails, RSCS will continue to retry the link every 5 minutes, indefinitely.

For information about specifying the RETRY operand when defining an auto-dial or SNA link, see the LINKDEFINE statement ([“LINKDEFINE Statement” on page 151](#)). For information about specifying the RETRY operand on the DEFINE or START command, see *z/VM: RSCS Networking Operation and Use*.

2. If you omit the RETRY statement, RSCS will try to restart the link at 1, 10, 19, 27, 34, 40, 45, 49, 52, 54, and 55 minute intervals.

ROUTE Statement



Purpose

The ROUTE statement defines a path to a remote device or system with which RSCS will communicate indirectly through one or more links. Files destined for the remote node may, in fact, be stored and sent on one or more intermediate systems before reaching their destination. RSCS uses routing information to determine the next links in the route to the nonadjacent node. RSCS then sends files, commands, and messages through those links.

Specification

Include as many statements as needed; they are optional. ROUTE statements may be placed anywhere in the configuration file after the LOCAL statement (if one exists) and after any LINKDEFINE statements to which they refer.

Operands

GROUP *groupid*

specifies the name of the group (collection of nodes) whose traffic is being routed.

NODE *nodeid*

NODE *

specifies the name of the node whose traffic is being routed. If you specify an asterisk (*) for the node ID, then traffic for all nodes not explicitly defined will be routed to the specified link. To specify a generic node ID, specify an asterisk (*) as the last character of the node ID. Traffic will be routed for all node IDs that begin with the characters specified to the left of the asterisk.

TO GROUP *groupid*

specifies the name of the group whose links will be used for the routed traffic.

TO LINK *linkid*

specifies the name of a previously defined link through which the traffic will be routed.

ALTERNATE *linkid*

specifies the alternate link that RSCS should use if all the primary links are not in connect state.

nodeid

*

is the name of the destination node. All traffic destined for this node ID will be routed to the specified link. If you specify an asterisk (*) for the node ID, then traffic for all nodes not explicitly defined will be routed to the specified link. To specify a generic node ID, specify an asterisk (*) as the last character of the node ID. Traffic will be routed for all node IDs that begin with the characters specified to the left of the asterisk.

linkid

is the name of a previously defined link. This is the next link through which RSCS will send files, messages, and commands destined for the specified node.

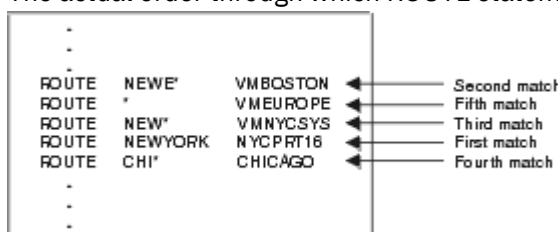
***USER* TO LINK linkid**

specifies the name of the link to be used for traffic destined for unknown user IDs on the local system.

For information on routing traffic to a NOTIFY-type link for processing, see usage note “5” on page 259.

Usage Notes

1. Link definitions take precedence over generic routes.
2. You cannot define a route through which a destination node ID is routed to a LISTPROC-type link.
3. You can specify generic node IDs by coding an asterisk (*) as the last character in the *nodeid*. This will cause all identifiers that begin with the specified characters to be considered. For example, specifying NYC* for a node ID would suggest that files are to be routed for all node IDs of the form NYCxxxxx. When specifying generic node IDs, the length of the character string determines the order through which a match will be made for selecting a route. If multiple matches are possible, longer strings are matched first. Explicit specification of a node ID always takes precedence over a generic specification. The actual order through which ROUTE statements are entered is not a determining factor.



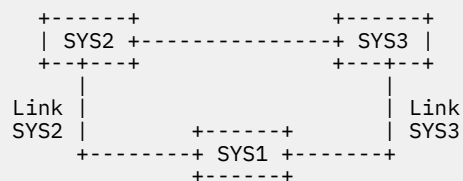
4. If you plan to take advantage of the support that lets you route a node through multiple links, make sure you reserve an appropriate number of tag shadow elements (TASHADOWs). See “SHADOWS Statement” on page 264 for more information about using the SHADOWS statement to reserve tag shadow elements.
5. If your local node has defined a NOTIFY-type link to handle misdirected files, you can use ROUTE statements (or commands) to tell RSCS to route files to that link. For example, suppose you have two NOTIFY-type links called *NOTHERE* and *UNKNOWN*. You could use the following ROUTE statements:

```
ROUTE * TO *UNKNOWN
ROUTE *USER* TO *NOTHERE
```

The first statement would route all files destined to unknown nodes through the *UNKNOWN link. The second statement would route all files destined for unknown users on the local system through the *NOTHERE link.

6. **Important Note for Users Migrating from RSCS V2:** The concept of routing has changed since RSCS V2. The current ROUTE statement is compatible with the RSCS V2 ROUTE statement, with one exception.

ROUTE



Suppose you defined the following LINK and ROUTE statements in the configuration file for system SYS1:

```
LINK SYS2
LINK SYS3
ROUTE SYS2 SYS3
```

If system SYS1 was using RSCS V2, the link between SYS1 and SYS2 would have to go down before the ROUTE statement became effective. Once the link went down, all files destined for node SYS2 would be queued on link SYS3.

If system SYS1 was using RSCS V3 or later, this ROUTE statement would cause all files destined for node SYS2 to be automatically queued on link SYS3 regardless of the connection status of the link between SYS1 and SYS2.

If you are migrating from RSCS V2 and you want your ROUTE statements to perform the same as they have in the past, you would change the previous ROUTE statement to:

```
ROUTE [NODE] SYS2 TO [LINK] SYS2 ALTERNATE SYS3
```

SAFCLASS Statement



Purpose

The SAFCLASS statement specifies an output class that will be used by RSCS when creating store-and-forward spool files. (When the file is transmitted, it is transmitted with the *original* class, not the SAFCLASS-specified class.) This allows RSCS to avoid inadvertently spooling output to a class that has been identified to CP as having top or bottom classification titles.

Specification

One statement is allowed. It is optional and may be placed anywhere in the configuration file after the LOCAL statement (if one exists).

Operands

class

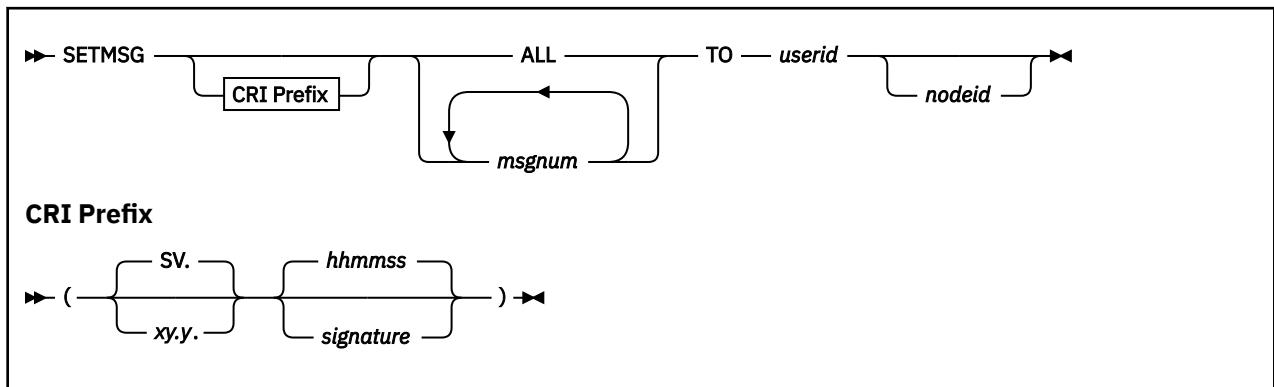
*

is the CP spool class that RSCS will use when creating store-and-forward spool files. The valid CP spool classes are A - Z and 0 - 9. If specified as an asterisk (*), or if the SAFCLASS statement is omitted, the file class remains the same as when the file was received.

Usage Notes

If the PRINTER_TITLE system configuration file statement (or the CP SYSPCLASS macro) has been used on your z/VM system to define a classification title on printed output, you must specify a different class on the SAFCLASS statement to prevent CP from inserting additional title lines.

SETMSG Statement



Purpose

The SETMSG statement tells RSCS to route nonprivate messages to another virtual machine. This lets you select virtual machines to monitor RSCS. For more information about private messages, see [z/VM: RSCS Networking Messages and Codes](#).

SETMSG routes messages based on message number. You can specify more than one SETMSG statement for the same virtual machine. RSCS keeps a list of routed message numbers for each virtual machine and RSCS will add new message numbers to the list as it finds additional SETMSG statements for that virtual machine. After RSCS completes initialization, if a condition occurs that causes RSCS to issue one of the messages in the list, RSCS will route that message to the specified virtual machine.

Specification

Include as many statements as needed; they are optional. SETMSG statements may be placed anywhere in the configuration file after the LOCAL statement (if one exists).

Operands

(xy.signature)

specifies the command response interface (CRI) prefix that RSCS will use when routing the specified messages. The CRI prefix consists of:

()

are the required delimiters of the prefix.

xy.

is the response modifier and its delimiter (a period). The response modifier tells RSCS how to send the responses to the target virtual machine.

x

is a character that indicates which method RSCS will use to send the responses. The valid choices are:

M

RSCS will send the responses using MSG or MSGNOH.

S

RSCS will send the responses using SMSG. This is the default.

y

is a character that indicates which language RSCS will use to build the message text. The valid choices are:

L

Local (national) language.

N

Network (remote) language.

T

Language appropriate to the target virtual machine. That is, if the user is at the local node, RSCS sends the message in the local (national) language. If the user is at a remote node, RSCS sends the message in the remote (network) language.

V

Language independent (machine-readable) format. This is the default.

• is the delimiter between the response modifier and the signature. It must be a period (.) and is required only if you specify the response modifier. If you are specifying just a signature, you can omit the delimiter.

signature

is the string that RSCS will return with every response to the command. This value can be 1 - 6 alphanumeric characters. If you omit this value, the signature will be a time-of-day stamp in the form *hhmmss*.

ALL

specifies that all non-private RSCS messages will be shared.

msgnum

is the 1- to 3-digit number of a non-private RSCS message to be shared. For a list of all the non-private RSCS messages, see [z/VM: RSCS Networking Messages and Codes](#). You can omit leading zeros.

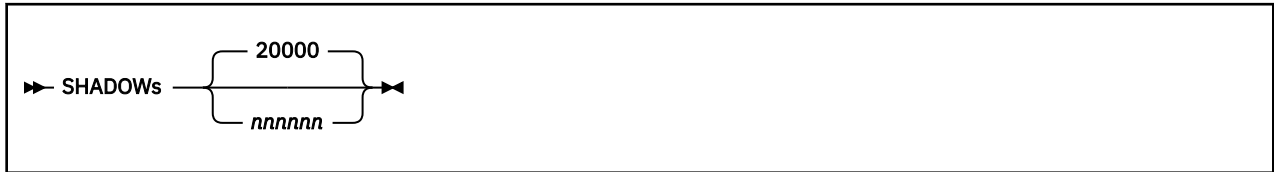
TO *userid nodeid*

specifies the user ID and node ID of the virtual machine that will receive the shared messages. If the node ID is omitted, RSCS sends the messages to the specified user ID on the local system.

Usage Notes

1. RSCS will not begin message routing until initialization completes.
2. RSCS can detect logged-off users only if they are at the same node as the RSCS virtual machine that owns their authorization table entry. If the virtual machine receiving the shared messages becomes disabled, RSCS issues message DMT616E to reflect this situation to the RSCS console.
3. RSCS will continue to send messages to disconnected users to account for the fact that their virtual machines may have console spooling in effect.
4. If you specify a CRI prefix on the SETMSG statement, the message recipient will receive shared messages in the form and with the signature, if any, that you specify in the prefix.
5. RSCS processes the statements in the configuration file sequentially. If you specify more than one SETMSG statement with the same user ID, node ID, and signature combination, RSCS creates a cumulative list of shared messages. If you specify more than one SETMSG statement with the same user ID and node ID combination, but with different signatures, RSCS creates cumulative lists for each signature. The same holds true if an operator issues a SETMSG command with the same user ID, node ID, and signature combination.
6. If you specify a CRI prefix, the signature field is used to match other SETMSG settings for the specified user ID. If you do not specify a CRI signature, the SETMSG setting will affect any messages specified for that user ID. The specified messages will be added to the previous SETMSG setting that has the same signature.

SHADOWS Statement



Purpose

The SHADOWS statement lets you specify the number of tag shadow elements (TASHADOWs) you want RSCS to reserve for routing groups. The tag element of a file is an anchor to the tag shadow elements which are queued on each of the links on which the file is eligible to be queued. These tag shadow elements contain queuing-related information. If the file is queued on a networking link (GATEWAY-type, LISTPROC-type, NJE-type, SNANJE-type, or TCPNJE-type), the tag shadow elements also contain a stream mask. The queuing priority value and stream mask allow each link on which the file is eligible to be queued to have its own queuing method and transmission algorithm.

Specification

One statement is allowed. It is optional and may be placed anywhere in the configuration file after the LOCAL statement (if one exists).

Operands

nnnnnn

is the number of tag shadow elements that RSCS should reserve. This value is a decimal number in the range 20000 - 100000. The default is 20000.

Usage Notes

1. If you do not specify the SHADOWS statement in your configuration file, RSCS reserves 20000 tag shadow elements. If you plan to take advantage of the routing group support, you should increase this number. IBM recommends you multiply the maximum number of links for any route by 10,000. For example, if you have a maximum of 4 links per route, you should reserve 40,000 tag shadow elements.
2. If you do not specify enough tag shadow elements, RSCS will enter a special *degraded* mode. For more information about degraded mode, see [z/VM: RSCS Networking Diagnosis](#).

SLOWDOWN Statement



Purpose

The SLOWDOWN statement lets you specify the absolute (base) system slowdown values. These values tell RSCS when to automatically start or stop slowdown mode for all active GATEWAY-type, NJE-type, SNANJE-type, and TCPNJE-type links.

Slowdown mode starts when the number of files in the RSCS virtual machine's reader grows large enough to match (or surpass) the slowdown start value. To start slowdown mode, RSCS issues HOLD INPUT commands to all active GATEWAY-type, NJE-type, SNANJE-type, and TCPNJE-type links (except those links that have specified their own slowdown values). This prevents incoming traffic so RSCS can concentrate on reducing the size of the current file queue. When the number of files in the reader drops below the stop value, RSCS issues FREE commands to all active GATEWAY-type, NJE-type, SNANJE-type, and TCPNJE-type links and file transmission resumes.

Slowdown mode has no effect on message and command traffic.

Specification

One statement is allowed. It is optional and may be placed anywhere in the configuration file after the LOCAL statement (if one exists) and before all LINKDEFINE statements.

Operands

start

is the absolute (base) slowdown start value for your system. RSCS will start slowdown mode when the total number of files in the RSCS file queue reaches or surpasses this value. This value can be any decimal number, 1 - 10000, but it must be **greater than** the slowdown stop value.

stop

is the absolute (base) slowdown stop value for your system. RSCS will end slowdown mode when the total number of files in the RSCS file queue reaches or becomes less than this value. This value can be any decimal number, 0 - 9999, but it must be **less than** the slowdown start value.

specifies that the slowdown stop value will be calculated as 10% less than the slowdown start value.

Usage Notes

1. This statement only applies to GATEWAY-type, NJE-type, SNANJE-type, and TCPNJE-type links.
2. To avoid excessive starting and stopping of slowdown mode, specify a stop value sufficiently less than the start value. IBM recommends that you set the stop value to at least 10% less than the start value.
3. To set or change the absolute (base) system slowdown values after initialization, use the SLOWDOWN command. For information about the SLOWDOWN command, see [z/VM: RSCS Networking Operation and Use](#).
4. To define or change the start or stop values for a specific link, use the SLOWDOWN operand on either the LINKDEFINE statement or the DEFINE command. For more information, see “LINKDEFINE Statement” on page 151. For information about the DEFINE command, see [z/VM: RSCS Networking Operation and Use](#).

TOLERANCE Statement



Purpose

The TOLERANCE statement lets you mark areas of the configuration file as tolerant or intolerant of errors.

Specification

Include as many statements as needed; they are optional. TOLERANCE statements may be placed anywhere in the configuration file.

Operands

ON

specifies that RSCS will tolerate errors until it reads a TOLERANCE OFF statement. This is the initial setting.

OFF

specifies that RSCS will not tolerate errors until it reads a TOLERANCE ON statement.

Usage Notes

1. If you omit the TOLERANCE statement, the default is to tolerate errors (TOLERANCE ON). This means RSCS will not end initialization unless a T (terminating) error occurs.
2. If you specify TOLERANCE OFF and RSCS finds an error, including W (warning) and E (error), RSCS initialization terminates at that point.
3. If you have sections in your configuration file that are critical to the definition of your network, use the TOLERANCE statement to prevent RSCS from initializing without these sections. For example, you might not want RSCS to start if it found any errors in a section defining:
 - An imbedded file
 - Heavily used links
 - Which users are authorized to use privileged commands
 - Routing groups
 - An exit package (especially one that defines security exit routines)
 - Anything that is important to your installation

TRACEDEST Statement



Purpose

The TRACEDEST statement allows you to specify where RSCS will send trace files.

Specification

One statement is allowed; it is optional. If you specify a local destination (user ID only), the TRACEDEST statement may be placed anywhere in the configuration file after the LOCAL statement (if one exists). If you specify a remote destination (user ID and node ID), the TRACEDEST statement must be placed after the LINKDEFINE or ROUTE statement that identifies the node ID.

Operands

userid

is the user ID of the destination virtual machine.

nodeid

is the node ID of the destination virtual machine.

Usage Notes

1. If you omit the TRACEDEST statement, RSCS sends the trace to the virtual printer associated with the local RSCS virtual machine. If the local system has a system printer, the trace file will print on the system printer. Otherwise, it will remain in the RSCS virtual print queue.
2. The TRACEDEST statement specifies a destination for trace output for **all** links. After initialization, you can use the TO operand of the TRACE command to temporarily override the destination for **one** trace file. After you enter the TRACE command with the TO operand, RSCS changes the destination of the existing trace file or the newly started trace file. When tracing ends, RSCS sends the trace output to the newly specified destination and resets the destination back to the default (the system printer or the destination specified on the TRACEDEST statement). If you open another trace file for that link, RSCS will send it to the default destination unless you override it by entering another TRACE command with the TO operand. For more information about the TRACE command, see [z/VM: RSCS Networking Operation and Use](#).

UPARM Statement

```
▶▶ UPARM — linkid — user_parameters ▶▶
```

Purpose

The UPARM statement lets you specify additional, installation-defined parameters. During RSCS operation, any exit point with access to the RSCS LINKTABL control block may access the parameters defined on a UPARM statement.

Specification

Include as many statements as needed; they are optional. However, you can have only one UPARM statement for each link ID. UPARM statements may be placed anywhere in the configuration file after the LINKDEFINE statement to which they refer.

Operands

linkid

is the link ID from a previous LINKDEFINE statement to which the installation-defined parameters apply.

PI

user_parameters

are parameters defined by your installation. RSCS places these parameters into storage and places a pointer to the storage location in the LUPARM field of the LINKTABL control block. The storage location contains a 2-byte length field and the text of the user parameter as follows:

USERPARM	DSECT		
USERLEN	DS	AL2	Length of 'USERDATA' field
USERDATA	DS	0C	Text specified on UPARM statement

PI end

Usage Notes

1. UPARM parameters have no effect on OPARM or PARM parameters.
2. You may also specify UPARM parameters using the UPARM operand of the DEFINE command. For more information, see [z/VM: RSCS Networking Operation and Use](#).

Chapter 16. RSCS Dynamic Authorization Configuration File Statements

This section contains topics that describe the configuration file statements you can use to define RSCS dynamic authorization (RSCSAUTH) on your system.

Rules for Coding the RSCS Dynamic Authorization Configuration File

The name of the RSCS dynamic authorization configuration file is RSCSAUTH CONFIG. The rules for coding the configuration file are:

1. The AUTHUSER and FILEMODE configuration statements are required. If either statement is omitted or is not valid, the server will not start. All other configuration statements are either optional or provide default values if omitted.
2. You can place configuration statements anywhere in the file, as long as each statement is on a separate line. Each statement must be complete on a line; it cannot be continued to the next line.
3. Comments begin with an asterisk (*). You can include as many comments as needed. You can place a comment on the same line as a configuration statement, following the statement.
4. If a configuration statement is incorrect, the statement is ignored and an error message is issued. Unless the incorrect statement is AUTHUSER or FILEMODE, initialization continues.
5. No configuration statements allow multiple entries. If multiple entries for a particular configuration statement are included in the file, only the first entry in the file is used. The other entries are ignored.

For information about how to read the syntax diagrams used in this chapter, see [“Syntax, Message, and Response Conventions”](#) on page xvii.

AUTHUSER Statement

```
➤ AUTHUSER = userid ➤
```

Purpose

The AUTHUSER statement specifies the user ID that is fully authorized for the RSCSAUTH server. This person can issue RSCSAUTH commands to authorize other users for RSCS commands and links and RSCSAUTH commands.

This statement is required; if it is omitted, the server will not start.

Operands

userid

is the z/VM user ID that is fully authorized for the RSCSAUTH server. The user ID defined in the sample configuration file is MAINT.

FILEMODE Statement

```
►► FILEMODE = fm ◄◄
```

Purpose

The FILEMODE statement specifies the file mode where the RSCSAUTH files (authorization tables and log files) are to be maintained.

This statement is required; if it is omitted, the server will not start.

Operands

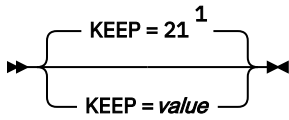
fm

is the file mode where the RSCSAUTH files are to be maintained. This file mode can be a minidisk or an SFS directory; it must be accessed R/W. The file mode defined in the sample configuration file is A.

Usage Notes

1. When setting up this minidisk or SFS directory, consider the impact of your log settings (the LOG and KEEP configuration statements) when determining how much space to allocate.

KEEP Statement



Notes:

¹ The default value is used only if a LOG statement is included in the configuration file.

Purpose

The KEEP statement specifies the maximum number of daily log files to be retained. This statement has meaning only if the LOG statement is specified. When the number of log files exceeds the KEEP value, the oldest log file will be deleted.

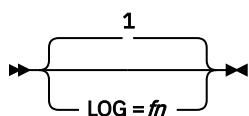
If you omit this statement (and specify the LOG statement), the default number is 21.

Operands

value

is the maximum number of log files to be retained. This value can be 7 - 60. The value defined in the sample configuration file is 7.

LOG Statement



Notes:

¹ No log files are created.

Purpose

The LOG statement specifies the file name of the log file that RSCSAUTH creates to log all messages issued by the server for commands that it processes. The log file is kept on the minidisk or SFS directory specified by the FILEMODE statement. The file type is the creation date, in the form *yyyymmdd*. When an interrupt arrives at the server and the day is changed, the server closes the old log file and creates a new log file.

If you omit this statement, RSCSAUTH will not create log files.

Operands

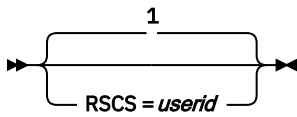
fn

is the file name of the log file. The name defined in the sample configuration file is RSCSAUTH.

Usage Notes

1. With every interrupt arriving at the RSCSAUTH server, RSCSAUTH estimates the available free space on the disk or directory specified on the FILEMODE statement. When the used space exceeds 90% of the allocated space, RSCSAUTH issues a warning message to the user ID specified on the TELL statement. If LOG is enabled, the 90% warning message is logged (only once a day). When the used space exceeds 95% of the allocated space, RSCSAUTH issues another warning message and stops logging if LOG is enabled.

RSCS Statement



Notes:

¹ RSCSAUTH will use the CMS IDENTIFY command to get the user ID.

Purpose

The RSCS statement specifies the user ID of the RSCS server.

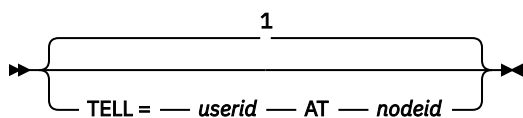
If you omit this statement, RSCSAUTH will use the CMS IDENTIFY command to capture the user ID of the local RSCS virtual machine.

Operands

userid

is the z/VM user ID of the RSCS server. The user ID defined in the sample configuration file is RSCS.

TELL Statement



Notes:

¹ Critical messages will be displayed only at the RSCSAUTH console.

Purpose

The TELL statement specifies the user ID where critical messages will be forwarded. An example is the warning message that the used space on the LOG disk or directory exceeds 90% of the allocated space.

If you omit this statement, critical messages will be displayed only at the RSCSAUTH console.

Attention

If no TELL user is defined, no one will see critical messages that need to be acted upon immediately, because the RSCSAUTH server runs disconnected. Therefore IBM recommends that you define a TELL user.

Operands

userid AT nodeid

specifies the user ID and node ID of the virtual machine where critical messages will be forwarded. The virtual machine defined in the sample configuration file is OPERATOR AT NEWYORK.

TRACERSCSAUTH Statement



Purpose

The TRACERSCSAUTH statement specifies whether RSCSAUTH should trace messages and return codes issued by the RSCSAUTH server. If you specify ON, the messages and return codes will be displayed at the RSCSAUTH console. If you include the LOG statement in the configuration file, the messages and return codes will also be recorded in the log file.

If you omit this statement, the default is OFF.

You can override the TRACERSCSAUTH configuration setting by using the TRACERSCSAUTH command.

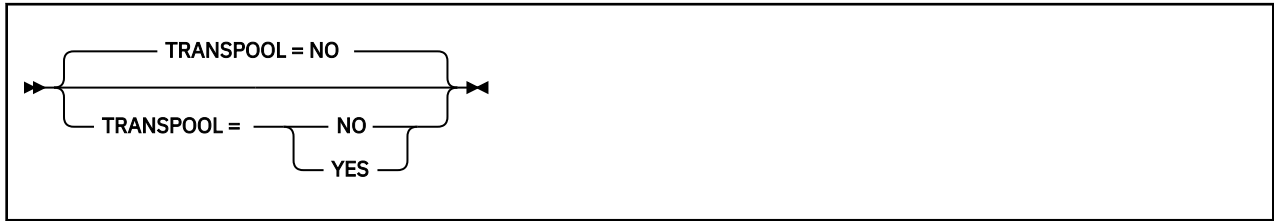
Operands

OFF

ON

specifies whether RSCSAUTH should trace messages and return codes issued by the server. The value defined in the sample configuration file is ON.

TRANSPool Statement



Purpose

The `TRANSPool` statement specifies whether `SPOOL` and `RDR` files will be forwarded to the RSCS server without validation by the `RSCSAUTH` server.

If you omit this statement, the default is `NO`.

Operands

NO

YES

specifies whether `RSCSAUTH` should forward `SPOOL` and `RDR` files without validation. The value defined in the sample configuration file is `YES`.

Chapter 17. RSCS Data Interchange Manager Configuration File Statements

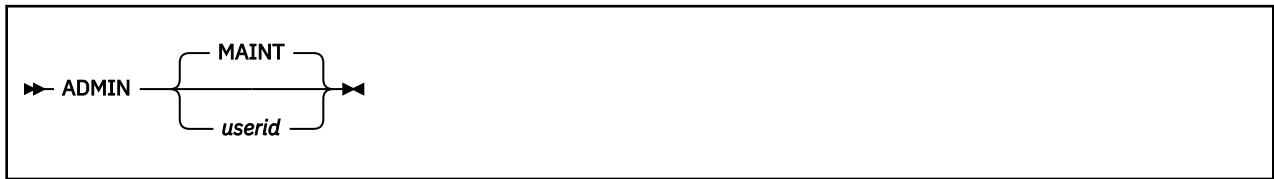
This section contains topics that describe the configuration file statements you can use to define the RSCS Data Interchange Manager (RSCS Interchange) on your system.

Rules for Coding the RSCS Interchange Configuration File

1. The default name for the configuration file is XCHANGE CONFIG.
2. The DOMAIN and RSCSLINK configuration statements are required. If more than one valid DOMAIN or RSCSLINK statement is found in the configuration file, an error will occur on the second valid DOMAIN or RSCSLINK statement, but the server will still initialize. All other configuration statements are either optional or provide default values if omitted.
3. You can place configuration statements anywhere in the file.
4. Comments begin with an asterisk (*). You can use as many comments as needed.
5. If a configuration statement is incorrect, the statement is ignored and an error message is issued. Unless the incorrect statement is DOMAIN or RSCSLINK, initialization continues.
6. Some configuration statements can have multiple entries in the file. However, if multiple entries are included for a configuration statement that can be specified only once, only the last entry in the file is used.
7. If more than the maximum options are used on a configuration statement, the extra options or other information are considered to be comments.

For information about how to read the syntax diagrams used in this chapter, see [“Syntax, Message, and Response Conventions”](#) on page xvii.

ADMIN Statement



Purpose

The ADMIN statement specifies the user ID of the RSCS Interchange server administrator. All error messages are sent to this ID. In addition, all mail failing any security or validation checks are transferred to this ID for inspection and handling.

Operands

MAINT

userid

is the z/VM user ID of the server administrator. The default is MAINT.

AUTHORIZE Statement

```
▶ AUTHORIZE — userid ▶
```

Purpose

The AUTHORIZE statement specifies a user ID that is authorized to issue privileged commands to the RSCS Interchange server.

Operands

userid

is the z/VM user ID of the authorized user.

Usage Notes

1. Only one user ID can be specified on an AUTHORIZE statement, but you can include multiple AUTHORIZE statements in the configuration file.

DOMAIN Statement

```
►► DOMAIN — domain_name ◄◄
```

Purpose

The DOMAIN statement specifies the domain name address of the RSCS Interchange server in the TCP/IP network. It is the portion of the Simple Message Transfer Protocol (SMTP) address used for addressing mail to the gateway from the SMTP environment, as well as the portion of the SMTP address used in constructing the return address for the z/VM user sending mail through the gateway.

The DOMAIN statement must be included and correct for RSCS Interchange to work correctly. If the DOMAIN statement is missing, RSCS Interchange will not complete initialization.

Operands

domain_name

is the name of the RSCS Interchange server in the IP network to be used for addressing names.

The domain name that you specify should be a *sub_domain_name* of the domain name specified in the TCP/IP domain name file. See [“Selecting Domain Names”](#) on page 124 for more information.

Usage Notes

1. The specified domain name must match the RSCS DOMAIN specified on the SMTP virtual machine. RSCS Interchange will append this name to z/VM addresses when delivering the mail to SMTP, as seen in the example.
2. If more than one valid DOMAIN statement is found in the configuration file, an error will occur on all valid statements after the first, but the server will still initialize.

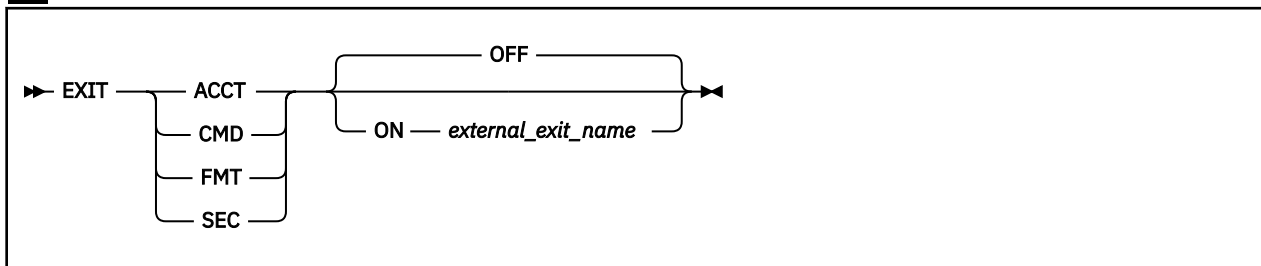
Examples

In the following example, the DOMAIN name is `smtpgate.college.edu`.

```
From: SMITTY@HURRAH.smtpgate.college.edu
```

EXIT Statement

PI



Purpose

The EXIT statement activates and deactivates user-installed exit routines for accounting, commands, format recognition, and security. You can include multiple EXIT statements in the configuration file to specify exits for any or all of the processing areas and to specify multiple exits for a processing area. The exits will be executed in the order defined in the configuration file.

Operands

ACCT

specifies the accounting exit.

CMD

specifies the command exit.

FMT

specifies the format recognition exit.

SEC

specifies the security exit.

OFF

specifies that no user-installed exit routines are to be called for the specified processing area. This is the default.

ON *external_exit_name*

specifies the name of the EXEC, MODULE, or compiled REXX exec (CEXEC) to be called for the exit.

Usage Notes

1. For information about writing RSCS Interchange exit routines, see [z/VM: RSCS Networking Exit Customization](#).
2. EXIT *area* OFF will turn off any previously defined exits within the specified area. This is useful for turning off exits without having to delete them from the configuration file. For example:

```
EXIT CMD ON CMDEX1
EXIT CMD ON CMDEX2
EXIT CMD ON CMDEX3
:
EXIT CMD OFF
```

3. Exit routines can be written in REXX, Assembler, or other languages such as PL/I. If the exit is written in Assembler or another language, it must be compiled and a MODULE file generated. The name of the MODULE must then be specified on the EXIT statement as the *external_exit_name* value.

PI end

FROM_FORMAT Statement



Purpose

The FROM_FORMAT statement specifies whether the local node ID of the RSCS Interchange server will be pre-appended to the From: header of files that are sent to SMTP and destined for a remote domain. When users send notes through SMTP, this enables any replies to be returned through a router at the site.

Operands

NORMAL

includes the local node ID of the RSCS Interchange server in the FROM: header of files sent to a remote domain. This is the default.

DOMAIN_ONLY

removes the local node ID of the RSCS Interchange server from the FROM: header of files sent to a remote domain.

Examples

1. In the following example of the NORMAL setting, the local node ID of the server is included in the FROM: header of a file:

```
user@node.domain
```

2. In the following example of the DOMAIN_ONLY setting, the local node ID of the server has been removed from the FROM: header of a file:

```
user@.domain
```

INFORM_USER Statement

```
►► INFORM_USER — userid ◄◄
```

Purpose

The INFORM_USER statement specifies an additional user ID to be notified whenever RSCS Interchange sends a message to the user ID defined on the ADMIN statement.

Operands

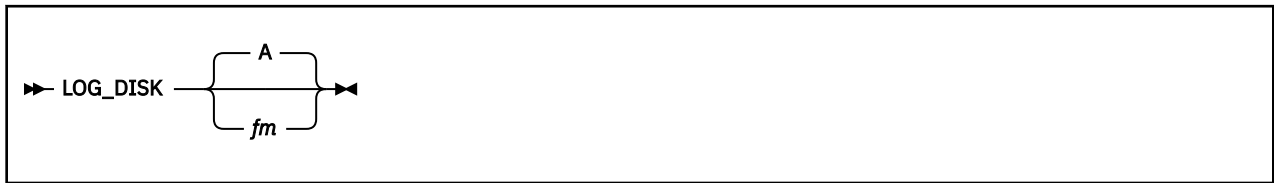
userid

is the z/VM user ID to be notified.

Usage Notes

1. Only one user ID can be defined on an INFORM_USER configuration file statement, but you can specify any number of INFORM_USER statements in the configuration file. If multiple statements are specified, each user ID will receive ADMIN messages.
2. User IDs specified on the INFORM_USER statement do not have authority to enter privileged commands unless you identify them on AUTHORIZE statements as well.

LOG_DISK Statement



Purpose

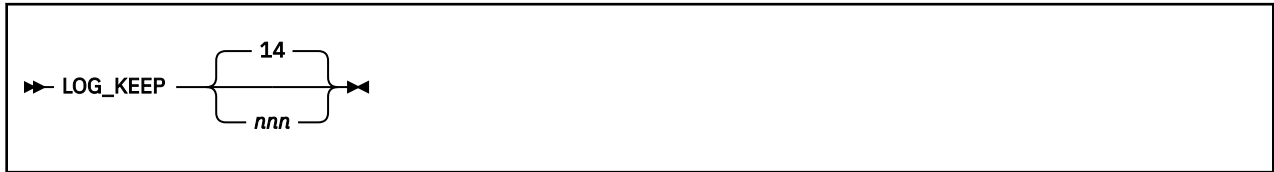
The LOG_DISK statement specifies the file mode where the RSCS Interchange server will create the log files. One log file is created each day with the name *serverid* *yyyymmdd*.

Operands

fm

is the file mode of the log files. The default is A.

LOG_KEEP Statement



Purpose

The LOG_KEEP statement specifies the number of days to keep the RSCS Interchange server log files. Log files older than the specified number of days will be deleted.

Operands

nnn

is the number of days a log file will be kept. This value can be 0 - 364. The default is 14.

Usage Notes

1. LOG_KEEP 0 means the log file is kept only for the current day. You need to move the log file before midnight.

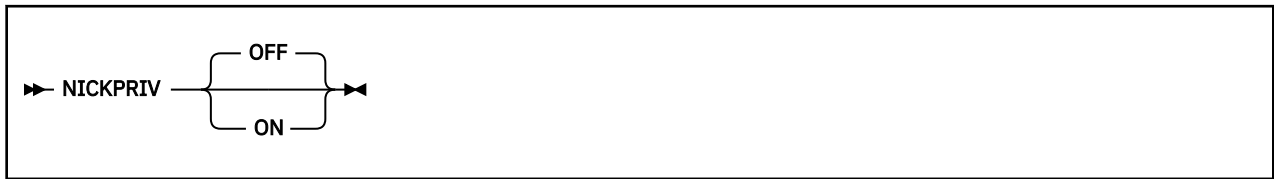
MSGNOH Statement

➤ MSGNOH ➤

Purpose

The MSGNOH statement enables the RSCS Interchange server to use the CP MSGNOH command to issue messages. Privilege CLASS B must be specified in the definition for the RSCS Interchange server virtual machine. If the MSGNOH statement is not included in the configuration file, or if the server is not authorized for CP privilege CLASS B, the CP MSG command will be used to issue messages.

NICKPRIV Statement



Purpose

The NICKPRIV statement specifies whether the NICK command is privileged. This setting determines whether general users may use the NICK command to modify the nickname table.

Operands

OFF

specifies that the NICK command is not privileged and therefore can be used by general users. This is the default.

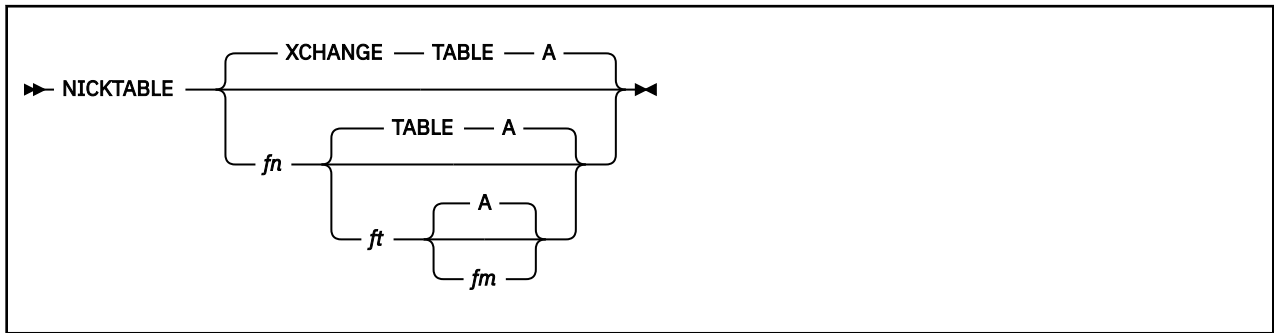
ON

specifies that the NICK command is privileged and therefore cannot be used by general users.

Usage Notes

1. Use NICKPRIV ON if you want to limit who modifies the nickname table.
2. A general user cannot change or delete a nickname table entry defined by another user, regardless of the NICKPRIV setting.

NICKTABLE Statement



Purpose

The NICKTABLE statement identifies the nickname file that RSCS Interchange will use for SMTP address translation.

Operands

fn

is the file name of the nickname file. The default is XCHANGE.

ft

is the file type of the nickname file. The default is TABLE.

fm

is the file mode of the nickname file. The default is A.

RSCSLINK Statement

```
▶▶ RSCSLINK — linkid ◀◀
```

Purpose

The RSCSLINK statement specifies the link that is defined in the RSCS virtual machine for RSCS Interchange. z/VM users specify this link to send mail to SMTP users; for example, MARK @ SMTPGATE, where SMTPGATE is the RSCS Interchange link.

The RSCSLINK statement must be included and correct for RSCS Interchange to work correctly. If the RSCSLINK statement is missing, RSCS Interchange will not complete initialization.

Operands

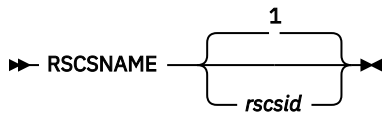
linkid

is the name of the link defined to the RSCS virtual machine for RSCS Interchange.

Usage Notes

1. When more than one valid RSCSLINK statement is found, an error will occur on the second valid statement, but the server will still initialize, using the link ID on the first statement.

RSCSNAME Statement



Notes:

¹ RSCS Interchange will use the CMS IDENTIFY command to get the user ID.

Purpose

The RSCSNAME statement identifies the z/VM user ID of the RSCS server virtual machine that RSCS Interchange will send VM network mail to and receive VM network mail from.

Operands

rscsid

is the z/VM user ID of the RSCS server virtual machine.

SMTPNAME Statement



Purpose

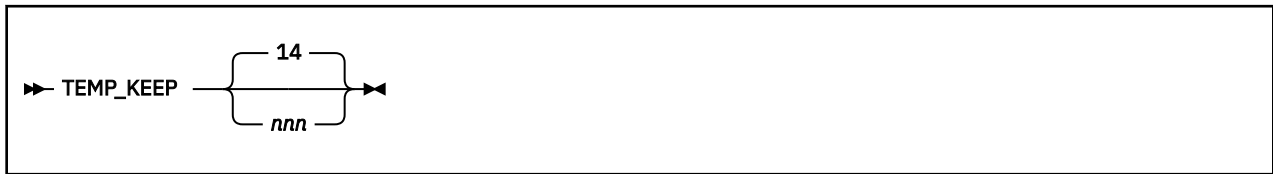
The SMTPNAME statement specifies the z/VM user ID of the Simple Message Transfer Protocol (SMTP) server virtual machine. This is the user ID that handles mail destined to SMTP users.

Operands

userid

is the z/VM user ID of the SMTP server. The default is SMTP.

TEMP_KEEP Statement



Purpose

The TEMP_KEEP statement specifies the number of days to keep the temporary nickname entries used by the RSCS Interchange server. Temporary nicknames older than the specified number of days will be deleted.

Operands

nnn

is the number of days a temporary nickname will be kept. This value can be 0 - 364. The default is 14.

Usage Notes

1. If many temporary nickname entries are created, the system administrator may want to add some of the entries to the permanent nickname table.
2. TEMP_KEEP 0 means the temporary nickname is kept only for the current day. It will be deleted at midnight.

Appendix A. Sample RSCS Configuration File

Before you can use RSCS, you must describe your network to RSCS using the RSCS configuration file. This topic describes a sample network and shows what the corresponding RSCS configuration file would look like. [Figure 51 on page 297](#) shows the nodes for the 17 operating systems in this sample network and shows the links that connect those nodes.

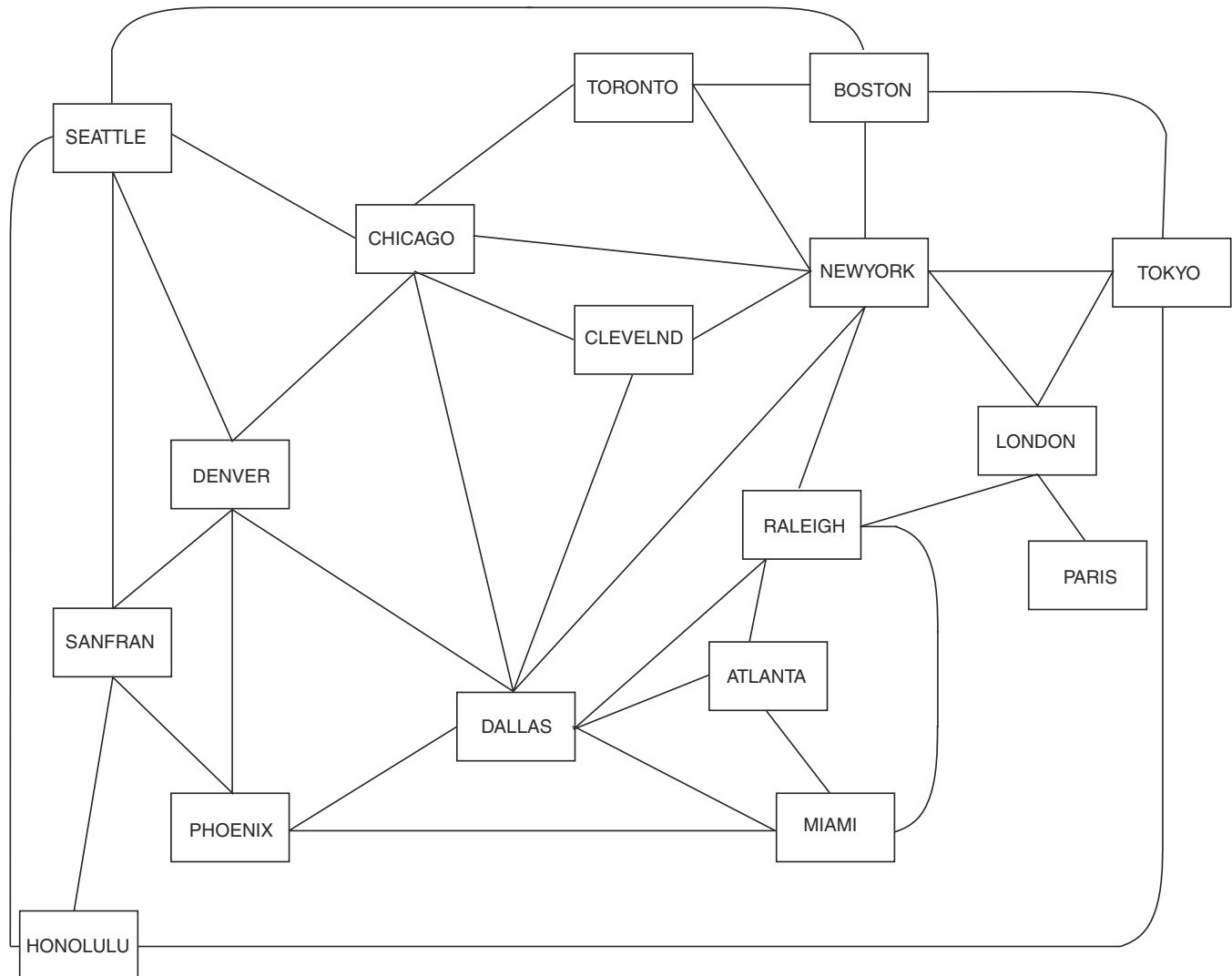


Figure 51. System Nodes in a Sample Network

The following sections show an example of a configuration file for node NEWYORK. “RSCSSMPL CONFIG” on [page 298](#) is the *master* configuration file. The master configuration file contains several IMBED statements that correspond to smaller pieces of the configuration file. Each IMBED statement has a pointer to the appropriate imbedded file. Some imbedded files contain IMBED statements for other imbedded files. [Figure 52 on page 298](#) shows how the imbedded files fit together.

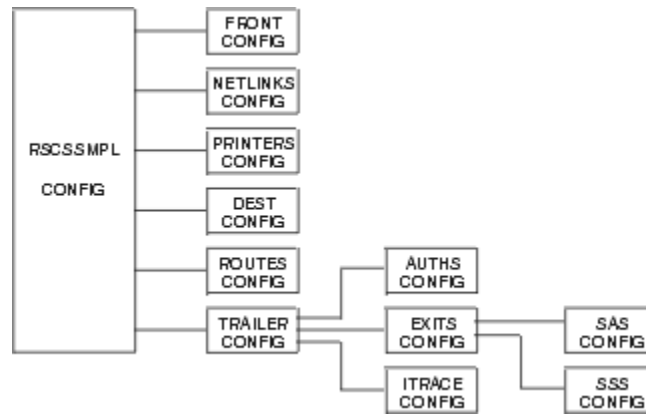


Figure 52. Sample Configuration File for Node NEWYORK

RSCSSMPL CONFIG

RSCSSMPL CONFIG is the sample master configuration file.

```

/*----- (Start of RSCS Configuration File Rules)-----*/
/*
/* 1. All configuration file statements and their operands must be in
/* UPPERCASE. (Comments, like this, can be in lowercase.)
/*
/* 2. All configuration file statements are optional.
/*
/* 3. The configuration file cannot contain sequence numbers.
/*
/* 4. The configuration file can be in either fixed format with a width of
/* 256 characters or in variable format.
/*
/* 5. Statements can be continued to the next physical line by using a
/* continuation character (#). When joined together, this statement
/* cannot exceed the 256 character limit for statement length.
/*
/* 6. REXX style comments (/* ...*/) are supported, but must begin and
/* end on the same physical line.
/*
/* 7. The LOCAL statement, if included, must be the first valid statement
/* in the configuration file. Blank lines, comments, IMBED statements,
/* and TOLERANCE statements are the only items you can place before a
/* LOCAL statement. If you omit the LOCAL statement, RSCS uses the
/* system ID specified on the SYSTEM_IDENTIFIER directory statement or
/* on the SYSID macro when your system was generated (SYSID is only
/* valid on a VM/ESA Version 2 Release 1.0 or earlier system).
/* If the LOCAL statement is omitted and a system ID was not specified,
/* RSCS initialization terminates.
/*
/* 8. The following statements, if included, must be placed before the
/* specified statements:
/* CHANNELS - Before all LINK, LINKDEFINE, and PORT statements
/* FCB - Before all LINKDEFINE statements
/* LINKTYPE - Before corresponding LINK and LINKDEFINE statements
/* OPFORM - Before all LINK and LINKDEFINE statements
/* SLOWDOWN - Before all LINK and LINKDEFINE statements
/*
/* 9. The following statements, if included, must be placed after the
/* corresponding LINK or LINKDEFINE statements:
/* AUTH PARM ROUTE UPARM
/*
/* 10. LINK statements, if included, must follow the CHANNELS and OPFORM
/* statements, but can be defined anywhere else in the file.
/*
/* 11. LINKDEFINE statements, if included, must follow the LOCAL, CHANNELS,
/* and OPFORM statements, if specified, and must follow the corresponding
/* LINK and FCB statements, if specified.
/*
/* 12. PORT statements, if included, must follow the CHANNELS statement, but
/* can be defined anywhere else in the file.
/*
/* 13. The following statements, if included, can be defined anywhere in
/* the configuration file after the LOCAL statement, if included:
/*

```

```

/*      DEST          ITRACE          REROUTE          */
/*      DUMP          LANGUAGE        RETRY           */
/*      EXIT          MSGNOH          SAFCLASS        */
/*      FORM          OPTION          SETMSG          */
/*      HIDECHARACTER RECOVERY        SHADOWS         */
/*      */
/* 14. The position of the TRACEDEST statement, if included, depends upon */
/* the specified destination. For local destinations, the TRACEDEST */
/* statement must follow the LOCAL statement, if specified. For remote */
/* destinations, the TRACEDEST statement must follow the corresponding */
/* LINK, LINKDEFINE, or ROUTE statement. */
/*      */
/*----- (End of Rules) -----*/

/*-----*/
/* The LOCAL Statement (one only). */
/*-----*/
TOLERANCE OFF
LOCAL      NEWYORK          /* Local Node ID */ #
          *                /* Place holder */ #
          SNARSCS          /* Application ID */
TOLERANCE ON

/*-----*/
/* Imbed statements that need to precede link definitions. */
/*-----*/
IMBED FRONT CONFIG          /* See "FRONT CONFIG" on page 299 */

/*-----*/
/* Imbed the files containing the links. */
/*-----*/
TOLERANCE OFF
IMBED NETLINKS CONFIG      /* See "NETLINKS CONFIG" on page 300 */
IMBED PRINTERS CONFIG      /* See "PRINTERS CONFIG" on page 301 */

/*-----*/
/* Imbed the file containing the DEST Statements. */
/*-----*/
IMBED DEST CONFIG          /* See "DEST CONFIG" on page 302 */

/*-----*/
/* Imbed the file containing the routes. */
/*-----*/
IMBED ROUTES CONFIG        /* See "ROUTES CONFIG" on page 302 */
TOLERANCE ON

/*-----*/
/* Imbed the file containing some more options. */
/*-----*/
IMBED TRAILER CONFIG       /* See "TRAILER CONFIG" on page 303 */

```

FRONT CONFIG

FRONT CONFIG is the sample imbed file containing front matter.

```

/*-----*/
/* Front Matter ... statements that must precede link definitions. */
/*-----*/
CHANNELS  E F              /* Reserve 2 channels for UR devices */

* LINKTYPE  MSGR  SMSDRVPE  NOLINE          /* Loadable Link Type */

OPFORM    *                /* Specify operator form */

SAFCLASS  *                /* Use the class of the received file */

SLOWDOWN  7000            /* Stop receiving files at this point */ #
          5000            /* Resume receiving files at this point */

/*----- (RSCS Channel Skip Specifications) -----*/
/*      FCB */
/*      Name  Line Channel  Line Channel  Line Channel  Line Channel  */
/*-----*/
FCB FCB1    1      1      6      2      8      3      10     4      #
           12     5      14     6      16     7      18     8      #
           20     10     22     11     60     9      66     12     #

```

NETLINKS CONFIG

NETLINKS CONFIG is the sample imbed file containing the system links.

```

/*-----*/
/* This file contains the networking links. */
/* These are the links to adjacent system (site) nodes. */
/*-----*/
LINKDEFINE BOSTON TYPE SNANJE QUEUE SIZE LUNAME TN2RSCS ASTART
LINKDEFINE CHICAGO TYPE SNANJE QUEUE SIZE LUNAME N051ZRSC ASTART #
LOGMODE RSCSNJE0
LINKDEFINE CLEVELND TYPE SNANJE QUEUE SIZE LUNAME TNARSCS ASTART
LINKDEFINE DALLAS TYPE SNANJE QUEUE SIZE LUNAME TNHRSCS ASTART
LINKDEFINE LONDON TYPE SNANJE QUEUE SIZE LUNAME TNBRSCS ASTART
LINKDEFINE RALEIGH TYPE SNANJE QUEUE SIZE LUNAME TN7RSCS ASTART #
LOGMODE RSCS07
LINKDEFINE TOKYO TYPE SNANJE QUEUE SIZE LUNAME QHARSCS ASTART #
LOGMODE RSCSNJE0
LINKDEFINE TORONTO TYPE SNANJE QUEUE SIZE LUNAME OVCVRSCS ASTART #
LOGMODE RSCSNJE0

/*-----*/
/* Links to adjacent system (NEWYORK site) nodes. */
/*-----*/
LINKDEFINE NYDEVSYS TYPE NJE LINE 00A4 QUEUE SIZE ASTART
LINKDEFINE RESLAB TYPE NJE LINE 0581 QUEUE SIZE DP 4 ASTART
LINKDEFINE HQARTRS TYPE SNANJE QUEUE SIZE LUNAME WZ2RSCS ASTART #
LOGMODE RSCSNJE0
LINKDEFINE ENDICOTT TYPE TCPNJE QUEUE SIZE RETRY
LINKDEFINE GLENDALE TYPE TCPNJE QUEUE SIZE ASTART

/*-----*/
/* Special links */
/*-----*/
LINKDEFINE *LIST TYPE LISTPROC QUEUE SIZE ASTART
LINKDEFINE *UNKNOWN TYPE NOTIFY QUEUE FIFO
LINKDEFINE *NOTHERE TYPE NOTIFY QUEUE FIFO
LINKDEFINE *LPRHOLD TYPE NOTIFY QUEUE FIFO
LINKDEFINE *UPTHOLD TYPE NOTIFY QUEUE FIFO

/*-----*/
/* UFT Client links */
/*-----*/
LINKDEFINE UFT1 TYPE UFT QUEUE FIFO AST
LINKDEFINE UFT2 TYPE UFT QUEUE FIFO AST
LINKDEFINE UFT3 TYPE UFT QUEUE FIFO AST
LINKDEFINE UFT4 TYPE UFT QUEUE FIFO AST
LINKDEFINE UFT5 TYPE UFT QUEUE FIFO AST
LINKDEFINE UFT6 TYPE UFT QUEUE FIFO AST
LINKDEFINE UFT7 TYPE UFT QUEUE FIFO AST
LINKDEFINE UFT8 TYPE UFT QUEUE FIFO AST

/*-----*/
/* Daemon links */
/*-----*/
LINKDEFINE LPD TYPE LPD
LINKDEFINE UFD TYPE UFTD

/*-----*/
/* Other links */
/*-----*/
LINKDEFINE LNKMRJE TYPE MRJE QUEUE SIZE LINE 0002
LINKDEFINE LNKRJE TYPE RJE QUEUE SIZE LINE 0003

/*-----*/
/* Fanout specifications for the links. */
/*-----*/
LINKDEFINE TOKYO FANOUT LONDON
LINKDEFINE CHICAGO FANOUT CLEVELND

/*-----*/
/* RSCS parameters for the links. */
/*-----*/
PARM BOSTON STREAMS=4 TA=1 TAPARM='THR=(,10),(,35),(,50),(,51,)' BUFF=3200
PARM CHICAGO STREAMS=4 TA=1 TAPARM='THR=(,10),(,35),(,50),(,51,)' BUFF=3200
PARM CLEVELND STREAMS=4 TA=1 TAPARM='THR=(,19),(,39),(,73),(,74,)' BUFF=3200
PARM DALLAS STREAMS=4 TA=1 TAPARM='THR=(,19),(,39),(,73),(,74,)' BUFF=3200
PARM LONDON STREAMS=4 TA=1 TAPARM='THR=(,19),(,39),(,73),(,74,)' BUFF=3200
PARM RALEIGH STREAMS=4 TA=1 TAPARM='THR=(,19),(,39),(,73),(,74,)' BUFF=3200
PARM TOKYO STREAMS=4 TA=1 TAPARM='THR=(,10),(,35),(,50),(,51,)' BUFF=3200
PARM TORONTO STREAMS=4 TA=1 TAPARM='THR=(,10),(,35),(,50),(,51,)' BUFF=3200

```

```

PARAM NYDEVSYS STREAMS=4 TA=1 TAPARM='THR=(,10),(,35),(,50),(51,)' BUFF=3200
PARAM RESLAB STREAMS=4 TA=1 TAPARM='THR=(,10),(,35),(,50),(51,)' BUFF=3200
PARAM HQUARTRS STREAMS=4 TA=1 TAPARM='THR=(,10),(,35),(,50),(51,)' BUFF=3200
PARAM *LIST BUFF=4096 STREAMS=4 TA=0 MAXURO=64
PARAM *NOTHERE MSGFILE=NOTHERE PURGE=3
PARAM *UNKNOWN MSGFILE=UNKNOWN PURGE=3
PARAM *LPRHOLD MSGFILE=LPRHOLD PURGE=3
PARAM *UFTHOLD MSGFILE=UFTHOLD PURGE=3
PARAM ENDICOTT TCPID=TCPIP HOST=9.130.40.72 KEEPALIVE=YES
PARAM GLENDALE TCPID=TCPIP HOST=9.130.63.712 LCLPORT=175
PARAM LNKMRJE SYSTEM=HOST ITO=2 PASS=MRJEPASS PHONE=5551234 RMT=1
PARAM LNKRJE SYSTEM=HOST ITO=2 PASS=RJEPASS PHONE=5556789 TYPE=3780
PARAM UFT1 EXIT=UFTXOUT DAY=2 FILEH=Y TI=60
PARAM UFT2 EXIT=UFTXOUT DAY=2 FILEH=Y TI=60
PARAM UFT3 EXIT=UFTXOUT DAY=2 FILEH=Y TI=60
PARAM UFT4 EXIT=UFTXOUT DAY=2 FILEH=Y TI=60
PARAM UFT5 EXIT=UFTXOUT DAY=2 FILEH=Y TI=60
PARAM UFT6 EXIT=UFTXOUT DAY=2 FILEH=Y TI=60
PARAM UFT7 EXIT=UFTXOUT DAY=2 FILEH=Y TI=60
PARAM UFT8 EXIT=UFTXOUT DAY=2 FILEH=Y TI=60
PARAM LPD EXIT=LPDXMANY TI=300
PARAM UFD EXIT=UFTXIN TI=300

```

PRINTERS CONFIG

PRINTERS CONFIG is the sample imbed file containing the printer links.

```

/*-----*/
/* This file contains the printer links. */
/*-----*/
LINKDEFINE PRT3812A TYPE 3270P LINE 0DFA QUEUE SIZE RETRY
LINKDEFINE PRT3812B TYPE 3270P LINE 0DFB QUEUE SIZE RETRY
LINKDEFINE PRT3812C TYPE 3270P LINE 0DFC QUEUE SIZE RETRY
LINKDEFINE LUNAME TYPE TN3270E LINE 0DFD QUEUE SIZE AST
LINKDEFINE PRT38001 TYPE SNA3270P QUEUE SIZE LUNAME PRT1RSCS ASTART #
LOGMODE SCS3800
LINKDEFINE PRT38002 TYPE SNA3270P QUEUE SIZE LUNAME PRT2RSCS ASTART #
LOGMODE SCS3800
LINKDEFINE PRT38003 TYPE SNA3270P QUEUE SIZE LUNAME PRT3RSCS ASTART #
LOGMODE SCS3800
LINKDEFINE PRT3820A TYPE SNA3270P QUEUE SIZE LUNAME PRTARSCS ASTART #
LOGMODE SCS3820 RETRY
LINKDEFINE PRT3820B TYPE SNA3270P QUEUE SIZE LUNAME PRTBRSCS ASTART #
LOGMODE SCS3820 RETRY
LINKDEFINE PRT3820C TYPE SNA3270P QUEUE SIZE LUNAME PRTCRCSCS ASTART #
LOGMODE SCS3820 RETRY
LINKDEFINE PRT4250A TYPE SNA3270P QUEUE FIFO LUNAME ACRCRSCS ASTART #
LOGMODE SCS4250
LINKDEFINE PRT4250B TYPE SNA3270P QUEUE FIFO LUNAME BCRCRSCS ASTART #
LOGMODE SCS4250
LINKDEFINE PRT4250C TYPE SNA3270P QUEUE FIFO LUNAME CCRCRSCS ASTART #
LOGMODE SCS4250
LINKDEFINE PROPRINT TYPE TCPASCII QUEUE FIFO
LINKDEFINE PROPRNTA TYPE TCPASCII QUEUE FIFO
LINKDEFINE PROPRNTB TYPE TCPASCII QUEUE FIFO
LINKDEFINE LPRP1 TYPE LPR QUEUE FIFO FORM * AST
LINKDEFINE LPRP2 TYPE LPR QUEUE FIFO FORM * AST
LINKDEFINE LPRP3 TYPE LPR QUEUE FIFO FORM * AST
LINKDEFINE LPRP4 TYPE LPR QUEUE FIFO FORM * AST
LINKDEFINE LPRP5 TYPE LPR QUEUE FIFO FORM * AST
LINKDEFINE LPRP6 TYPE LPR QUEUE FIFO FORM * AST
LINKDEFINE LPRP7 TYPE LPR QUEUE FIFO FORM * AST
LINKDEFINE LPRP8 TYPE LPR QUEUE FIFO FORM * AST
LINKDEFINE LPR1 TYPE LPR QUEUE FIFO FORM * AST
LINKDEFINE LPR2 TYPE LPR QUEUE FIFO FORM * AST
LINKDEFINE LPR3 TYPE LPR QUEUE FIFO FORM * AST
LINKDEFINE LPR4 TYPE LPR QUEUE FIFO FORM * AST
LINKDEFINE LPR5 TYPE LPR QUEUE FIFO FORM * AST
LINKDEFINE LPR6 TYPE LPR QUEUE FIFO FORM * AST
LINKDEFINE LPR7 TYPE LPR QUEUE FIFO FORM * AST
LINKDEFINE LPR8 TYPE LPR QUEUE FIFO FORM * AST
/*-----*/
/* RSCS parameters for the links. */
/*-----*/
PARAM PRT3812A SEP=NO VFC=YES EPC=NO
PARAM PRT3812B SEP=NO VFC=YES EPC=NO
PARAM PRT3812C SEP=NO VFC=YES EPC=NO
PARAM LUNAME PRTW=5 PRTR=20 SEP=N COMP=N EPC=N L=0 VFC=Y ITO=0

```

RSCSSMPL CONFIG

```
PARM PRT38001 SEP=NO VFC=YES
PARM PRT38002 SEP=NO VFC=YES
PARM PRT38003 SEP=NO VFC=YES
PARM PRT3820A SEP=NO VFC=YES
PARM PRT3820B SEP=NO VFC=YES
PARM PRT3820C SEP=NO VFC=YES
PARM PRT4250A SEP=NO VFC=YES
PARM PRT4250B SEP=NO VFC=YES
PARM PRT4250C SEP=NO VFC=YES
PARM PROPRINT EXIT=ASCXPROP HOST=9.130.41.73 PORT=21
PARM PROPRNTA EXIT=ASCXPROP HOST=9.130.42.76 PORT=21
PARM PROPRNTB EXIT=ASCXPROP HOST=9.130.54.82 PORT=21
PARM LPRP1 EXIT=LPRXPSE TI=300 DAY=2 FILEH=Y USER=Y
PARM LPRP2 EXIT=LPRXPSE TI=300 DAY=2 FILEH=Y USER=Y
PARM LPRP3 EXIT=LPRXPSE TI=300 DAY=2 FILEH=Y USER=Y
PARM LPRP4 EXIT=LPRXPSE TI=300 DAY=2 FILEH=Y USER=Y
PARM LPRP5 EXIT=LPRXPSE TI=300 DAY=2 FILEH=Y USER=Y
PARM LPRP6 EXIT=LPRXPSE TI=300 DAY=2 FILEH=Y USER=Y
PARM LPRP7 EXIT=LPRXPSE TI=300 DAY=2 FILEH=Y USER=Y
PARM LPRP8 EXIT=LPRXPSE TI=300 DAY=2 FILEH=Y USER=Y
PARM LPR1 EXIT=LPRXONE TI=300 DAY=2 FILEH=Y USER=Y
PARM LPR2 EXIT=LPRXONE TI=300 DAY=2 FILEH=Y USER=Y
PARM LPR3 EXIT=LPRXONE TI=300 DAY=2 FILEH=Y USER=Y
PARM LPR4 EXIT=LPRXONE TI=300 DAY=2 FILEH=Y USER=Y
PARM LPR5 EXIT=LPRXONE TI=300 DAY=2 FILEH=Y USER=Y
PARM LPR6 EXIT=LPRXONE TI=300 DAY=2 FILEH=Y USER=Y
PARM LPR7 EXIT=LPRXONE TI=300 DAY=2 FILEH=Y USER=Y
PARM LPR8 EXIT=LPRXONE TI=300 DAY=2 FILEH=Y USER=Y
```

DEST CONFIG

DEST CONFIG is the sample imbed file containing the DEST statements.

```
/*-----*/
/* The DEST Statement (no limit). */
/*-----*/
DEST PSFPRT1 PSFPRT2
DEST PSFPRT3 PSFPRT4 PSFPRT5
```

ROUTES CONFIG

ROUTES CONFIG is the sample imbed file containing the ROUTE statements.

```
/*-----*/
/* This file contains all the ROUTE statements. */
/*-----*/
/*-----*/
/* Define the routing groups */
/*-----*/
ROUTE GROUP NORTH TO LINK BOSTON CHICAGO CLEVELND TORONTO
ROUTE GROUP USNORTH TO LINK BOSTON CHICAGO CLEVELND
ROUTE GROUP SOUTH TO LINK DALLAS RALEIGH
ROUTE GROUP EAST TO LINK LONDON TOKYO
ROUTE GROUP WEST TO LINK CHICAGO DALLAS
ROUTE GROUP HQUARTRS TO LINK HQUARTRS ALTERNATE NYDEVSYS
ROUTE GROUP DEVELPMT TO LINK NYDEVSYS ALTERNATE RESLAB
ROUTE GROUP RESEARCH TO LINK RESLAB ALTERNATE NYDEVSYS
ROUTE GROUP NYTCP TO LINK GLENDALE ALTERNATE ENDICOTT

/*-----*/
/* Define the paths to nonadjacent nodes */
/*-----*/
ROUTE NODE ATLANTA TO GROUP SOUTH
ROUTE NODE DENVER TO GROUP WEST
ROUTE NODE HONOLULU TO LINK TOKYO ALTERNATE BOSTON
ROUTE NODE MIAMI TO GROUP SOUTH
ROUTE NODE PARIS TO GROUP EAST
ROUTE NODE PHOENIX TO GROUP WEST
ROUTE NODE SANFRAN TO GROUP WEST
ROUTE NODE SEATTLE TO LINK BOSTON CHICAGO TOKYO ALTERNATE DALLAS

/*-----*/
/* Define the paths for misdirected files */
/*-----*/
ROUTE * TO LINK *UNKNOWN /* Files to unknown node */
ROUTE *USER* TO LINK *NOTHERE /* Files to unknown local user */
```

```

/*-----*/
/* Define the paths for undeliverable files */
/*-----*/
ROUTE *LPRH* TO LINK *LPRHOLD          /* Undeliverable LPR files */
ROUTE *UFTH* TO LINK *UFTHOLD          /* Undeliverable UFT files */

/*-----*/
/* Define the paths for input only links */
/*-----*/
ROUTE UFTD   TO LINK *UNKNOWN          /* Don't queue files on UFTD */
ROUTE LPD    TO LINK *UNKNOWN          /* Don't queue files on LPD */

/*-----*/
/* Define the printer pools */
/*-----*/
ROUTE GROUP PRT3812 TO LINK PRT3812A PRT3812B PRT3812C ALTERNATE PRT3820C
ROUTE GROUP PRT3800 TO LINK PRT38001 PRT38002 PRT38003
ROUTE GROUP PRT3820 TO LINK PRT3820A PRT3820B PRT3820C
ROUTE GROUP PRT4250 TO LINK PRT4250A PRT4250B PRT4250C
ROUTE GROUP TCPPRO  TO LINK PROPRINT PROPRNTA ALTERNATE PROPRNTB
ROUTE GROUP LPRP   TO LINK LPRP1 LPRP2 LPRP3 LPRP4 LPRP5 LPRP6 LPRP7 LPRP8
ROUTE GROUP LPR    TO LINK LPR1 LPR2 LPR3 LPR4 LPR5 LPR6 LPR7 LPR8

/*-----*/
/* Define the UFT client pools */
/*-----*/
ROUTE GROUP UFT   TO LINK UFT1 UFT2 UFT3 UFT4 UFT5 UFT6 UFT7 UFT8

```

TRAILER CONFIG

TRAILER CONFIG is the sample imbed file containing the trailer. This file contains several IMBED statements that correspond to additional pieces of the configuration file. Each IMBED statement has a pointer to the appropriate imbedded file.

```

/*-----*/
/* This file contains statements that may need to be placed after link and */
/* node definitions. */
/*-----*/

IMBED AUTHS CONFIG          /* See "AUTHS CONFIG" on page 304 */
DUMP VM OPERATNS           /* Use VMDUMP and send to OPERATNS */
IMBED EXITS CONFIG         /* See "EXITS CONFIG" on page 305 */
HIDECHARACTER \           /* Define character used to hide things */
IMBED ITRACE CONFIG       /* See "ITRACE CONFIG" on page 305 */
LANGUAGE * =              /* Local and Remote Load Module Name */
MSGNOH                    /* Use MSGNOH for message delivery */
RECOVERY MSG OP1 Re-IPLing RSCS#I GCS /* What to do if we abend */
RETRY 1 9 10 10 30        /* Specify time intervals for restarting links */
SETMSG 002 TO OPERATOR    /* Subscribe OPERATOR to msg 2 */
SHADOWS 60000             /* Reserve some tag shadow elements */
TRACEDEST MAINT BOSTON    /* Default trace destination */

/*-----*/
/* RSCS Default Option Specifications */
/*-----*/
OPTION ACCMSG=NO          /* Generate file accepted message? */
OPTION ENQMSG=NO         /* Generate file queued message? */
OPTION FINALMSG=YES      /* Generate file arrival message? */
OPTION JOBNAME=RSCS      /* Specify format of job name */
OPTION LISTPROC=YES      /* Can other systems handle large *MULTI* files? */
OPTION LOOPING=ALL       /* Specify type of loop checking */
* OPTION MAXDSH=10       /* Specify maximum number of data set headers */
OPTION MAXHOPS=64        /* Specify hop count number */
* OPTION MSGSKIP=2       /* Specify how many file buffers sent before msg */
* OPTION SECORGID=YES    /* Invoke Secure Origin ID support? */
OPTION SENTMSG=NO        /* Generate file sent messages? */

```

RSCSSMPL CONFIG

```
/*-----*/
/*                      RSCS Reroute Specifications                      */
/*-----*/
/*-----*/
/* Reroute traffic for node that changed names, */
/* but do not publicize the name change.      */
/*-----*/
REROUTE ALL FOR DEVSYS ANY TO NYDEVSYS = QUIET
REROUTE CMDS FOR DEVSYS TO NYDEVSYS QUIET

/*-----*/
/* Reroute messages for operators */
/*-----*/
REROUTE NOTRCVG FOR OP1 TO * OPER1 QUIET
REROUTE NOTRCVG FOR OP2 TO * OPER2 QUIET
REROUTE NOTRCVG FOR OP3 TO * OPER3 QUIET

/*-----*/
/* User ID change ... make sure she gets her mail */
/*-----*/
REROUTE FILES FOR * JANETM TO * JANETF

/*----- (RSCS Form Name Specifications) -----*/
/*          Separator      Page      Page      Lines      */
/*          Form Name      Page Style  Width    Length    Per Inch  */
/*-----*/
FORM VMSTYLE      VM      132      66      6
FORM RSCSTYLE     YES      80      66      6
FORM NOHDRPG     NO      220     *      8

/*----- (RSCS Port Specifications) -----*/
/*          Virtual      Dial or      */
/*          Address      Nodial?      */
/*-----*/
PORT 0080 NODIAL
PORT 0081 NODIAL
PORT 0082 DIAL
```

AUTHS CONFIG

AUTHS CONFIG is the sample imbed file containing the authorizations.

```
/*-----*/
/* This file contains the operator authorizations.                      */
/*-----*/
/*-----*/
/* System-Authorized Alternate Operators */
/*-----*/
AUTH * DAVE      BOSTON  CP
AUTH * COLLEEN  NYDEVSYS CP
AUTH * COLLEEN  *        CP
AUTH * MAINT    *        CP
AUTH * MARTIN   LONDON  CP
AUTH * MARTIN   *        CP
AUTH * TONY     *        CP
AUTH * DEAN     BOSTON  CP
AUTH * DEAN     *        CP
AUTH * OPERATNS *        CP
AUTH * OPERATOR *        CP
AUTH * OPER1    *        CP
AUTH * OPER2    *        CP
AUTH * OPER3    *        CP
AUTH * OP1      *        CP
AUTH * OP2      *        CP
AUTH * OP3      *        CP
AUTH * RSCSAUTH *        CP

/*-----*/
/* Link-Authorized Alternate Operators */
/*-----*/
AUTH BOSTON BILL * *
AUTH BOSTON LES * *
AUTH CHICAGO SARAH * *
AUTH CHICAGO MIKE * *
AUTH CLEVELND DAWN * *
AUTH CLEVELND JOHN * *
AUTH DALLAS KAREN * *
AUTH LONDON TEDDY * *
```



```

AUTH LONDON KATHY * *
AUTH NYDEVSYS DOROTHY * *
AUTH NYDEVSYS NED * *
AUTH RALEIGH SUZIE * *
AUTH RALEIGH WINSTON * *
AUTH TOKYO KAREN * *
AUTH TOKYO JOSEPH * *
AUTH TORONTO SQUIRMY * *
AUTH TORONTO MATTHEW * *

```

EXITS CONFIG

EXITS CONFIG is the sample imbed file containing the exit points. This file contains several IMBED statements that correspond to additional pieces of the configuration file. Each IMBED statement has a pointer to the appropriate imbedded file.

```

/*-----*/
/* This file contains the exit point specifications. */
/* Note: Use of exit routines is not required for normal RSCS operation. */
/*-----*/
/*-----*/
/* Using the LPRXFORM Exit Support */
/*-----*/
EXIT 0 ON LPRXFM00
EXIT 1 ON LPRXFM01

/*-----*/
/* Imbed Simple Accounting Support */
/*-----*/
IMBED SAS CONFIG /* See "SAS CONFIG" on page 305 */

/*-----*/
/* Imbed Simple Security Support */
/*-----*/
IMBED SSS CONFIG /* See "SSS CONFIG" on page 306 */

```

ITRACE CONFIG

ITRACE CONFIG is the sample imbed file containing internal tracing definitions.

```

/*-----*/
/* This file contains the internal tracing specifications. */
/*-----*/
ITRACE EVENT ALL ON FOR SYSTEMTASK REX
ITRACE EVENT DIAGNOSE ON FOR SYSTEMTASK AXM
ITRACE EVENT FILEREQ ON FOR LINK ALL
ITRACE EVENT FILEREQ OFF FOR LINK WORK1
ITRACE EVENT IO ON FOR PORT ALL
ITRACE EVENT NJE ON FOR LINK SYSTEM1
ITRACE EVENT SNA ON FOR LINK SNAPRT1
ITRACE EVENT MRJE ON FOR LINK LNKMRJE
ITRACE TABLE SIZE 20
ITRACE TABLE DUMP NO
* ITRACE ON

```

SAS CONFIG

SAS CONFIG is the sample imbed file containing simple accounting support.

```

/*-----*/
/* This file contains the EXIT statements needed to install the simple */
/* accounting support. */
/*-----*/
EXIT 02 ON SASPOL02 /* Spool file accept */
EXIT 03 ON FIRST SASPOL03 /* Spool file send */
EXIT 04 ON SASPOL04 /* Spool file purge */
EXIT 05 ON SASPOL05 /* Spool file receive */

```

SSS CONFIG

SSS CONFIG is the sample imbed file containing simple security support.

```
/*-----*/
/* This file contains the EXIT statements needed to install the simple */
/* security support. */
/*-----*/
EXIT 00 ON          SSSECU00          /* RSCS initialization */
EXIT 01 ON FIRST   SSSECU01          /* RSCS termination   */
EXIT 14 ON          SSSECU14          /* Job header reception */
EXIT 15 ON          SSSECU15          /* Dataset header reception */
EXIT 19 ON          SSSECU19          /* Command execution exit */
EXIT 21 ON          SSSECU21          /* File arrival exit   */
EXIT 32 ON          SSSECU32          /* Msg/cmd reception exit */
```

Appendix B. Logon Mode Table Entries

The following example shows the entries supplied by RSCS in the sample RSCSTAB ASSEMBLE file.

RSCSTAB	CSECT		
RSCSTAB	MODETAB		
RSCSNJE0	MODEENT	LOGMODE=RSCSNJE0, ¹	C
		FMPROF=X'03', ²	C
		TSPROF=X'03', ²	C
		PRIPROT=X'72', ²	C
		SECPROT=X'72', ²	C
		COMPROT=X'4020', ²	C
		SSNDPAC=X'03', ¹⁰	C
		SRCVPAC=X'03', ¹⁰	C
		RUSIZES=X'0000', ³	C
		PSNDPAC=X'03', ¹⁰	C
		PSEVIC=X'000000000000000000000000' ⁹	C
RSCSPRT0	MODEENT	LOGMODE=RSCSPRT0, ¹	C
		FMPROF=X'02', ²	C
		TSPROF=X'02', ²	C
		PRIPROT=X'60', ²	C
		SECPROT=X'40', ²	C
		COMPROT=X'0000', ²	C
		SSNDPAC=X'00', ³	C
		SRCVPAC=X'00', ³	C
		RUSIZES=X'C7C7', ²	C
		PSNDPAC=X'80', ¹¹	C
		PSEVIC=X'008000000000185018507F00' ⁶	C
RSCSPRT1	MODEENT	LOGMODE=RSCSPRT1, ¹	C
		FMPROF=X'03', ²	C
		TSPROF=X'03', ²	C
		PRIPROT=X'B1', ²	C
		SECPROT=X'30', ²	C
		COMPROT=X'3080', ²	C
		SSNDPAC=X'01', ¹¹	C
		SRCVPAC=X'01', ⁸	C
		RUSIZES=X'C7C7', ⁴	C
		PSNDPAC=X'81', ¹¹	C
		PSEVIC=X'01000000E100000000000000' ⁵	C
RSCSPR1H	MODEENT	LOGMODE=RSCSPR1H, ¹	C
		FMPROF=X'03', ²	C
		TSPROF=X'03', ²	C
		PRIPROT=X'B1', ²	C
		SECPROT=X'30', ²	C
		COMPROT=X'7080', ²	C
		SSNDPAC=X'01', ¹¹	C
		SRCVPAC=X'01', ⁸	C
		RUSIZES=X'C7C7', ⁴	C
		PSNDPAC=X'81', ¹¹	C
		PSEVIC=X'01000000E100000000000000' ⁵	C
RSCSPRT3	MODEENT	LOGMODE=RSCSPRT3, ¹	C
		FMPROF=X'03', ²	C
		TSPROF=X'03', ²	C
		PRIPROT=X'B1', ²	C
		SECPROT=X'20', ²	C
		COMPROT=X'3080', ²	C
		SSNDPAC=X'00', ³	C
		SRCVPAC=X'00', ³	C
		RUSIZES=X'C7C7', ⁴	C
		PSNDPAC=X'80', ¹¹	C
		PSEVIC=X'038000000000185018507F00' ⁶	C
RSCSRJE1	MODEENT	LOGMODE=RSCSRJE1, ¹	C
		FMPROF=X'03', ²	C
		TSPROF=X'03', ²	C
		PRIPROT=X'A1', ²	C
		SECPROT=X'A1', ²	C
		COMPROT=X'7080', ²	C
		RUSIZES=X'8585', ⁴	C
		SSNDPAC=X'07', ¹¹	C
		SRCVPAC=X'07', ⁸	C
		PSEVIC=X'01102000F100C00000010040' ⁷	C
	MODEEND		
	END		

Notes:

Sample Logon Mode Table

1. This is a suggested logon mode table name.
2. RSCS ignores the contents of this field. When building the Bind, RSCS overrides the values you set in the VTAM logmode table with the value of this field in the sample logon mode table.
3. This entry is the default value and may be omitted.
4. This entry must be changed to match the actual buffer size in the 3270 printer. As [Figure 53](#) on [page 308](#) shows, the acceptable range of RU sizes for RSCS is from 256 to 8192 bytes. For more information about the full range of acceptable values for RU sizes that are valid in the BIND SESSION image, see *Systems Network Architecture: Sessions Between Logical Units*.

Specify RUSIZES in the form *abab* (representing the *sending* and *receiving* buffer sizes, respectively); use the values shown the values shown for *a* and *b* that correspond to the desired size. For example, a buffer size of 256 bytes would be specified as 85.

b a-> v	8	9	A	B	C	D	E	F
5	256	288	320	352	384	416	448	480
6	512	576	640	704	768	832	896	960
7	1024	1152	1280	1408	1536	1664	1792	1920
8	2048	2304	2560	2816	3072	3328	3584	3840
9	4096	4608	5120	5632	6144	6656	7168	7680
A	8192	...						

Figure 53. RU Buffer Size Specification Values

5. The first byte of the PSERVIC must be as shown. The remaining bytes can be set as desired.
6. The first byte of the PSERVIC must be as shown. Bytes 7, 8, 9, and 10 must reflect the buffer size of the printer. For more information about setting the buffer size, consult the *Component Description* document for the specific printer.
7. The first byte of the PSERVIC must be as shown. The values shown in the sample logon mode table in bytes 7, 8, 9, and 10 are recommended.
8. The value of the SRCVPAC field depends on the system configuration and use.
9. The first byte of the PSERVIC must be as shown. The remaining bytes are set to zero by RSCS. When RSCSNJEO is used in an SLU session, it must be set exactly to the values shown. This does not apply to the SSNDPAC, SRCVPAC, and PSNDPAC fields.
10. The SSNDPAC, SRCVPAC, and PSNDPAC fields supply a value to the VTAM pacing facility. For more information on vpadding, see [Table 4](#) on [page 45](#).
11. This is a recommended value; other values may be used, when appropriate.

Logical Unit Sessions for Printer Links

RSCS supports three types of logical unit sessions for printer links. An LU_T0 session permits communications with non-SNA 3270 printers and control units that are attached to VTAM. An LU_T1 session supports only those printers having the SNA Character String (SCS) feature. An LU_T3 session allows communications with SNA 3270 printers that do not have the SCS feature.

LU_T1 Sessions

The installation can specify (using a Logon Mode Table entry) the type of session for a particular printer. It is recommended that LU_T1 be specified, whenever possible. All 3270 printers in current production are capable of communicating to RSCS using LU_T1 sessions. The SCS data stream and

SNA protocols used in LU_T1 sessions permit more efficient operation of the printer, allowing it to perform closer to rated speed (lines per minute) than when supported by LU_T0 and LU_T3 sessions. Also, the control panel keys on the printer (CANCEL PRINT, PA1, and PA2) are supported only in LU_T1 mode.

LU_T0 and LU_T3 Sessions

Definition of a printer for LU_T0 and LU_T3 sessions is intended for GDDM Extended 3270 data stream output and for older printers that are not capable of higher-speed LU_T1 sessions.

Appendix C. Sample RSCS Dynamic Authorization Configuration File

The following example shows the sample RSCS dynamic authorization configuration file (RSCSAUTH CONFIG) supplied with RSCS. Your configuration file may vary depending on site guidelines.

```

*****
* This is a sample config file for RSCSAUTH server.          *
*                                                           *
* Description about individual statements:                   *
*                                                           *
* AUTHUSER - Specifies the user who is fully authorized    *
*             for RSCSAUTH server. Without specifying the  *
*             user, RSCSAUTH server can not run.           *
*                                                           *
*             Note: The user specified will be fully      *
*                   authorized from all the nodes.        *
*                                                           *
* FILEMODE - Specifies the file mode where all RSCSAUTH  *
*             files are to be maintained. Without        *
*             specifying the value, RSCSAUTH server can    *
*             not run.                                     *
*                                                           *
*             Note: The file mode specified should be R/W. *
*                                                           *
* TRACERSCSAUTH - Set to ON/OFF. If set to ON, extra     *
*                 messages like Wakeup RC will be        *
*                 captured.                               *
*                                                           *
* TELL          - All critical messages like disk space   *
*                 constraint will be forwarded to the    *
*                 user specified.                         *
*                                                           *
*             Note: User ID along with the node name     *
*                   should be provided.                  *
*                                                           *
* RSCS          - All RSCS commands are forwarded to the *
*                 specified ID for processing, otherwise *
*                 the commands are forwarded to the local *
*                 RSCS ID.                                *
*                                                           *
* LOG          - User need to provide a filename to      *
*                 record all the activities of RSCSAUTH  *
*                 server.                                 *
*                                                           *
*             Note: Only filename should be provided.    *
*                   File type will be generated by the   *
*                   RSCSAUTH server and will be in the  *
*                   form YYYYMMDD.                       *
*                                                           *
* KEEP         - If specified, the value indicates the   *
*                 maximum number of log file to be      *
*                 retained.                               *
*                                                           *
*             Note: Value should be between 7 & 60,     *
*                   with a default value of 21.         *
*                                                           *
* TRANSPPOOL   - If set to YES, all the SPOOL/RDR files  *
*                 are forwarded to the RSCS server without *
*                 validation.                             *
*                                                           *
*             Note: Valid values are YES/NO.             *
*                                                           *
*****
AUTHUSER = ADMIN
FILEMODE = A
TRACERSCSAUTH = ON
TELL = OPERATOR AT NEWYORK
RSCS = RSCS
LOG = RSCSAUTH
KEEP = 7
TRANSPPOOL = YES
*****

```

Sample RSCSAUTH CONFIG

```
* End of sample RSCSAUTH configuration file.  
*****
```


Appendix D. Sample RSCS Data Interchange Manager Configuration File

The following example shows the sample RSCS Interchange configuration file (XCHANGE CONFIG) supplied with RSCS. Your configuration file may vary depending on site guidelines.

```

*****
* This is a sample RSCS Interchange Configuration File
*   - RSCSNAME will default to RSCS
*   - SMTPNAME will default to SMTP
*   - ADMIN will default to MAINT
*   - There are NO EXITS defined
*****
*
* The following are REQUIRED statements
*
DOMAIN endicott.ibm.com
RSCSLINK SMTPGATE
*
* The following sets up the privileged users
*
AUTHORIZE SYSADMIN
AUTHORIZE BOSS
*
* The following sets up the informed users
*
INFORM_USER VMADMIN
INFORM_USER TCPADMIN
*
* The following sets up the RSCS Interchange environment
*
MSGNOH
NICKTABLE XNICK TABLE A
NICKPRIV ON
NICK_OVERRIDE ALL
LOG_DISK B
LOG_KEEP 90
TEMP_KEEP 60
*
*****

```


Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. 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 IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

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

*IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US*

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

*Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan*

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

*IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US*

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

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, IBM International Program License Agreement or any equivalent agreement between us.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information may contain examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information may contain sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Programming Interface Information

This publication primarily documents information that is NOT intended to be used as Programming Interfaces of z/VM.

This publication also documents intended Programming Interfaces that allow the customer to write programs to obtain the services of z/VM. This information is identified where it occurs, either by an introductory statement to a chapter or section or by the following marking:

PI

<...Programming Interface information...>

PI end

Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com)[®] are trademarks or registered trademarks of International Business Machines Corp., in the United States and/or other countries. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on [IBM Copyright and trademark information](http://www.ibm.com/legal/copytrade) (<https://www.ibm.com/legal/copytrade>).

Adobe and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

The registered trademark Linux is used pursuant to a sublicense from the Linux Foundation, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis.

Terms and Conditions for Product Documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

Personal Use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial Use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

IBM Online Privacy Statement

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user, or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see:

- The section entitled **IBM Websites** at [IBM Privacy Statement](https://www.ibm.com/privacy) (<https://www.ibm.com/privacy>)
- [Cookies and Similar Technologies](https://www.ibm.com/privacy#Cookies_and_Similar_Technologies) (https://www.ibm.com/privacy#Cookies_and_Similar_Technologies)

Bibliography

This topic lists the publications in the z/VM library. For abstracts of the z/VM publications, see [z/VM: General Information](#).

Where to Get z/VM Information

The current z/VM product documentation is available in [IBM Documentation - z/VM \(https://www.ibm.com/docs/en/zvm\)](https://www.ibm.com/docs/en/zvm).

z/VM Base Library

Overview

- [z/VM: License Information](#), GI13-4377
- [z/VM: General Information](#), GC24-6286

Installation, Migration, and Service

- [z/VM: Installation Guide](#), GC24-6292
- [z/VM: Migration Guide](#), GC24-6294
- [z/VM: Service Guide](#), GC24-6325
- [z/VM: VMSES/E Introduction and Reference](#), GC24-6336

Planning and Administration

- [z/VM: CMS File Pool Planning, Administration, and Operation](#), SC24-6261
- [z/VM: CMS Planning and Administration](#), SC24-6264
- [z/VM: Connectivity](#), SC24-6267
- [z/VM: CP Planning and Administration](#), SC24-6271
- [z/VM: Getting Started with Linux on IBM Z](#), SC24-6287
- [z/VM: Group Control System](#), SC24-6289
- [z/VM: I/O Configuration](#), SC24-6291
- [z/VM: Running Guest Operating Systems](#), SC24-6321
- [z/VM: Saved Segments Planning and Administration](#), SC24-6322
- [z/VM: Secure Configuration Guide](#), SC24-6323

Customization and Tuning

- [z/VM: CP Exit Customization](#), SC24-6269
- [z/VM: Performance](#), SC24-6301

Operation and Use

- [z/VM: CMS Commands and Utilities Reference](#), SC24-6260
- [z/VM: CMS Primer](#), SC24-6265
- [z/VM: CMS User's Guide](#), SC24-6266
- [z/VM: CP Commands and Utilities Reference](#), SC24-6268

- [z/VM: System Operation](#), SC24-6326
- [z/VM: Virtual Machine Operation](#), SC24-6334
- [z/VM: XEDIT Commands and Macros Reference](#), SC24-6337
- [z/VM: XEDIT User's Guide](#), SC24-6338

Application Programming

- [z/VM: CMS Application Development Guide](#), SC24-6256
- [z/VM: CMS Application Development Guide for Assembler](#), SC24-6257
- [z/VM: CMS Application Multitasking](#), SC24-6258
- [z/VM: CMS Callable Services Reference](#), SC24-6259
- [z/VM: CMS Macros and Functions Reference](#), SC24-6262
- [z/VM: CMS Pipelines User's Guide and Reference](#), SC24-6252
- [z/VM: CP Programming Services](#), SC24-6272
- [z/VM: CPI Communications User's Guide](#), SC24-6273
- [z/VM: ESA/XC Principles of Operation](#), SC24-6285
- [z/VM: Language Environment User's Guide](#), SC24-6293
- [z/VM: OpenExtensions Advanced Application Programming Tools](#), SC24-6295
- [z/VM: OpenExtensions Callable Services Reference](#), SC24-6296
- [z/VM: OpenExtensions Commands Reference](#), SC24-6297
- [z/VM: OpenExtensions POSIX Conformance Document](#), GC24-6298
- [z/VM: OpenExtensions User's Guide](#), SC24-6299
- [z/VM: Program Management Binder for CMS](#), SC24-6304
- [z/VM: Reusable Server Kernel Programmer's Guide and Reference](#), SC24-6313
- [z/VM: REXX/VM Reference](#), SC24-6314
- [z/VM: REXX/VM User's Guide](#), SC24-6315
- [z/VM: Systems Management Application Programming](#), SC24-6327
- [z/VM: z/Architecture Extended Configuration \(z/XC\) Principles of Operation](#), SC27-4940

Diagnosis

- [z/VM: CMS and REXX/VM Messages and Codes](#), GC24-6255
- [z/VM: CP Messages and Codes](#), GC24-6270
- [z/VM: Diagnosis Guide](#), GC24-6280
- [z/VM: Dump Viewing Facility](#), GC24-6284
- [z/VM: Other Components Messages and Codes](#), GC24-6300
- [z/VM: VM Dump Tool](#), GC24-6335

z/VM Facilities and Features

Data Facility Storage Management Subsystem for z/VM

- [z/VM: DFSMS/VM Customization](#), SC24-6274
- [z/VM: DFSMS/VM Diagnosis Guide](#), GC24-6275
- [z/VM: DFSMS/VM Messages and Codes](#), GC24-6276
- [z/VM: DFSMS/VM Planning Guide](#), SC24-6277

- *z/VM: DFSMS/VM Removable Media Services*, SC24-6278
- *z/VM: DFSMS/VM Storage Administration*, SC24-6279

Directory Maintenance Facility for z/VM

- *z/VM: Directory Maintenance Facility Commands Reference*, SC24-6281
- *z/VM: Directory Maintenance Facility Messages*, GC24-6282
- *z/VM: Directory Maintenance Facility Tailoring and Administration Guide*, SC24-6283

Open Systems Adapter

- *Open Systems Adapter-Express Customer's Guide and Reference* (<https://www.ibm.com/support/pages/node/6019492>), SA22-7935
- *Open Systems Adapter-Express Integrated Console Controller User's Guide* (<https://www.ibm.com/support/pages/node/6019810>), SC27-9003
- *Open Systems Adapter-Express Integrated Console Controller 3215 Support* (https://www.ibm.com/docs/en/SSLTBW_2.1.0/com.ibm.zos.v2r1.ioa/ioa.htm), SA23-2247
- *Open Systems Adapter/Support Facility on the Hardware Management Console* (https://www.ibm.com/docs/en/SSLTBW_2.1.0/com.ibm.zos.v2r1.ioa/ioa.htm), SC14-7580

Performance Toolkit for z/VM

- *z/VM: Performance Toolkit Guide*, SC24-6302
- *z/VM: Performance Toolkit Reference*, SC24-6303

RACF Security Server for z/VM

- *z/VM: RACF Security Server Auditor's Guide*, SC24-6305
- *z/VM: RACF Security Server Command Language Reference*, SC24-6306
- *z/VM: RACF Security Server Diagnosis Guide*, GC24-6307
- *z/VM: RACF Security Server General User's Guide*, SC24-6308
- *z/VM: RACF Security Server Macros and Interfaces*, SC24-6309
- *z/VM: RACF Security Server Messages and Codes*, GC24-6310
- *z/VM: RACF Security Server Security Administrator's Guide*, SC24-6311
- *z/VM: RACF Security Server System Programmer's Guide*, SC24-6312
- *z/VM: Security Server RACROUTE Macro Reference*, SC24-6324

Remote Spooling Communications Subsystem Networking for z/VM

- *z/VM: RSCS Networking Diagnosis*, GC24-6316
- *z/VM: RSCS Networking Exit Customization*, SC24-6317
- *z/VM: RSCS Networking Messages and Codes*, GC24-6318
- *z/VM: RSCS Networking Operation and Use*, SC24-6319
- *z/VM: RSCS Networking Planning and Configuration*, SC24-6320

TCP/IP for z/VM

- *z/VM: TCP/IP Diagnosis Guide*, GC24-6328
- *z/VM: TCP/IP LDAP Administration Guide*, SC24-6329
- *z/VM: TCP/IP Messages and Codes*, GC24-6330

- *z/VM: TCP/IP Planning and Customization*, SC24-6331
- *z/VM: TCP/IP Programmer's Reference*, SC24-6332
- *z/VM: TCP/IP User's Guide*, SC24-6333

Prerequisite Products

Device Support Facilities

- Device Support Facilities (ICKDSF): User's Guide and Reference ([https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5gc350033/\\$file/ickug00_v2r5.pdf](https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5gc350033/$file/ickug00_v2r5.pdf)), GC35-0033

Environmental Record Editing and Printing Program

- Environmental Record Editing and Printing Program (EREP): Reference ([https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5gc350152/\\$file/ifc2000_v2r5.pdf](https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5gc350152/$file/ifc2000_v2r5.pdf)), GC35-0152
- Environmental Record Editing and Printing Program (EREP): User's Guide ([https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5gc350151/\\$file/ifc1000_v2r5.pdf](https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5gc350151/$file/ifc1000_v2r5.pdf)), GC35-0151

Related Products

z/OS

- *Common Programming Interface Communications Reference* (<https://publibfp.dhe.ibm.com/epubs/pdf/c2643999.pdf>), SC26-4399
- z/OS and z/VM: Hardware Configuration Definition Messages ([https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sc342668/\\$file/cbdlm100_v2r5.pdf](https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sc342668/$file/cbdlm100_v2r5.pdf)), SC34-2668
- z/OS and z/VM: Hardware Configuration Manager User's Guide ([https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sc342670/\\$file/eequ100_v2r5.pdf](https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sc342670/$file/eequ100_v2r5.pdf)), SC34-2670
- z/OS: Network Job Entry (NJE) Formats and Protocols ([https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa320988/\\$file/hasa600_v2r5.pdf](https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa320988/$file/hasa600_v2r5.pdf)), SA32-0988
- z/OS: IBM Tivoli Directory Server Plug-in Reference for z/OS ([https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa760169/\\$file/glpa300_v2r5.pdf](https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa760169/$file/glpa300_v2r5.pdf)), SA76-0169
- z/OS: Language Environment Concepts Guide ([https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa380687/\\$file/ceea800_v2r5.pdf](https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa380687/$file/ceea800_v2r5.pdf)), SA38-0687
- z/OS: Language Environment Debugging Guide ([https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5ga320908/\\$file/ceea100_v2r5.pdf](https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5ga320908/$file/ceea100_v2r5.pdf)), GA32-0908
- z/OS: Language Environment Programming Guide ([https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa380682/\\$file/ceea200_v2r5.pdf](https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa380682/$file/ceea200_v2r5.pdf)), SA38-0682
- z/OS: Language Environment Programming Reference ([https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa380683/\\$file/ceea300_v2r5.pdf](https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa380683/$file/ceea300_v2r5.pdf)), SA38-0683
- z/OS: Language Environment Runtime Messages ([https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa380686/\\$file/ceea900_v2r5.pdf](https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa380686/$file/ceea900_v2r5.pdf)), SA38-0686
- z/OS: Language Environment Writing Interlanguage Communication Applications ([https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa380684/\\$file/ceea400_v2r5.pdf](https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa380684/$file/ceea400_v2r5.pdf)), SA38-0684
- z/OS: MVS Program Management Advanced Facilities ([https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa231392/\\$file/ieab200_v2r5.pdf](https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa231392/$file/ieab200_v2r5.pdf)), SA23-1392
- z/OS: MVS Program Management User's Guide and Reference ([https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa231393/\\$file/ieab100_v2r5.pdf](https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa231393/$file/ieab100_v2r5.pdf)), SA23-1393

XL C++ for z/VM

- [XL C/C++ for z/VM: Runtime Library Reference, SC09-7624](#)
- [XL C/C++ for z/VM: User's Guide, SC09-7625](#)

Additional Publications

- *IBM 7171 ASCII Device Attachment Control Unit Reference Manual and Programming Guide*, GA37-0021
- *IPDS Reference*, S544-3417
- *Systems Network Architecture: Formats*, GA27-3136
- *Systems Network Architecture: Sessions Between Logical Units*, GC20-1868
- *Systems Network Architecture: Technical Overview*, GC30-3073
- *Systems Network Architecture: Transaction Programmer's Reference Manual for LU Type 6.2*, GC30-3084
- *VTAM: Programming*, SC31-6496
- *VTAM: Resource Definition Reference*, SC31-6498
- *z/OS: MVS JCL Reference*, SA22-7597

Index

Special Characters

continuation character [129](#)

Numerics

3270 printer
 definition [17](#)
 non-SNA
 local, defining to VTAM [46](#)
 requirements [37](#)
 sharing [18](#)
 SNA
 character string [45](#)
 graphics support [45](#)
 local, defining to VTAM [46](#)
 LU type sessions [45](#)
 requirements [47](#)
 support [45](#)
 tending to [81](#)
3270P-type links
 PARM statement [245](#)
 requirements [37](#)
37xx TCU, defining a [37](#)

A

access priority, system resource [53](#)
accounting records [32](#), [121](#)
ACHAFAIL TEMPLATE file [119](#)
ACHARSCS CONFIG file [119](#)
ACU (Automatic Calling Unit)
 in an auto-dial link [78](#)
 on MRJE-type link [190](#)
 on NJE-type link [197](#)
 on RJE-type link [206](#)
adding multiple links to a node [107](#)
additional title lines, inserting [261](#)
addressing, second-level [89](#)
adjacent node [5](#)
ADMIN statement [280](#)
administration, planning for
 checklist [81](#)
 manipulating files in transit [82](#)
 preparing network information [82](#)
 simplifying jobs [82](#)
 tending to workstations and printers [81](#)
administrator, RSCSAUTH [24](#)
algorithm, transmission
 customizing RSCS using [72](#)
 used in exit routine [102](#)
alternate
 link, defining [69](#)
 network path, creating [108](#)
 operator, authorizing [75](#), [132](#)
alternate links
 defining [258](#)

alternate links (*continued*)
 example [69](#)
APL/Text [235](#)
ASCII
 advantages in a single-system environment [19](#)
 device requirements, non-SNA [38](#)
 devices [17](#)
 exits, used for customizing [72](#)
ASCII-type links
 exits, used for customizing [72](#)
 PARM statement [169](#)
 requirements [38](#)
AUTH statement
 category [56](#)
 description [60](#)
 link-authorized alternate operator syntax [132](#)
 system-authorized alternate operator syntax [132](#)
authorize an alternate operator [132](#)
AUTHORIZE statement [281](#)
AUTHUSER statement [270](#)
auto-answer links
 automatically starting [78](#)
 connecting to [67](#)
 invoking line driver [67](#)
 naming conventions [58](#)
auto-dial links
 connecting to [67](#)
 with automatic calling unit (ACU) [78](#)
auto-sign-on links [78](#)
auto-start link (ASTART) [78](#)
automate
 complex tasks [96](#)
 initialization [77](#)
 ongoing operation [77](#)
 operations with execs [77](#)
 simple events [95](#)
 using event scheduler, preparing to [78](#)
automatic
 link management [78](#)
 load balancing for printers [106](#)
 logon exec, sample [35](#)

B

backlog of files [52](#)
balancing, printer automatic load [106](#)
blank compression [218](#), [235](#), [246](#)
block
 multiplexer channel operations [251](#)
 spool file
 threshold number for GATEWAY-type link [175](#)
 threshold number for LISTPROC-type link [179](#)
 threshold number for NJE-type link [199](#)
 threshold number for SNANJE-type link [212](#)
 threshold number for TCPNJE-type link [232](#)
 virtual device [134](#)
BSC

BSC (*continued*)
 planning for [55](#)
 reserving address for [67](#)
buffer size [308](#)

C

case-sensitive names [186](#)
categories
 statement, configuration file [56](#)
CDRM (Cross Domain Resource Manager) [49](#)
CDRSC (Cross Domain Resource) [48](#)
CHANNELS statement
 category [56](#)
 description [60](#)
 syntax [134](#)
character
 continuation (#) [129](#)
 overprinting [220](#), [248](#)
 string, SNA
 sending [215](#), [220](#)
 used by LU_T1 session (3270 printer) [45](#)
checklist, planning
 administration [81](#)
 configuration [55](#)
 diagnosis [83](#)
 installation [31](#)
 operation [75](#)
coding conventions for configuration file [91](#)
columnar messages [65](#), [161](#)
command
 privileged, authorization [60](#)
 rerouting [253](#)
command, locally defined [117](#)
Comment [131](#)
comments in the RSCS configuration file [131](#)
comments, configuration file (symbols) [59](#), [131](#)
communication
 adapter buffer size [194](#), [228](#)
 between logical units [49](#)
 cross-domain [48](#)
 lines [11](#)
 links for direct, example [64](#)
 program, restrictions imposed by [58](#)
 routes for indirect, example [69](#)
 with nodes [86](#)
complex tasks, automating [96](#)
compressing data [194](#), [209](#), [228](#)
compression, blank [218](#), [235](#), [246](#)
configuration file
 adding a link [107](#)
 coding conventions [91](#)
 coding statements in [90](#)
 contents [26](#)
 CP function [56](#)
 creating alternate link [108](#)
 customization, planning for [56](#)
 customizing [85](#)
 definition [11](#)
 network definition [56](#)
 number and sequence of statements in [91](#)
 RSCS dynamic authorization [115](#)
 RSCS function [56](#)
 RSCS Interchange [119](#)

configuration file (*continued*)
 RSCS, updates for RSCS Interchange [119](#)
 rules for coding statements [129](#)
 sample
 creating a routing group [109](#), [110](#)
 for printer pool [106](#)
 for single system node [105](#)
 multiple system nodes [107](#)
 setting up for customization [86](#)
 configuring a RSCS TN3270E printer link [41](#)
CONNECT statement, use in JES2 [43](#)
connection
 defining non-SNA [36](#)
 non-SNA requirements [37](#)
 SNA, preparing VTAM for [44](#)
considerations, performance [52](#)
console log, closing automatically [95](#)
console operator [23](#)
continuation character (#) [129](#)
conversion repositories [101](#)
CP
 DIAGNOSE code X'F8' [166](#)
 function statements [56](#)
 headers, suppressing [65](#), [161](#)
creating
 events file [93](#)
 network paths [108](#)
 printer pool [106](#)
 routing group [109](#)
 transition link [107](#)
cross domain communications [48](#)
customization, planning for
 checklist [55](#)
 comments, in configuration file [59](#)
 configuration file [56](#)
 statement description
 AUTH [60](#)
 CHANNELS [60](#)
 DEST [60](#)
 DUMP [60](#)
 EXIT [60](#)
 FCB [61](#)
 FORM [61](#)
 HIDECHARACTER [61](#)
 IMBED [62](#)
 ITRACE [62](#)
 LANGUAGE [62](#)
 LINKDEFINE [62](#)
 LINKTYPE [63](#)
 LOCAL [64](#)
 MSGNOH [65](#)
 OPFORM [66](#)
 OPTION [66](#)
 PARM [66](#)
 PORT [67](#)
 RECOVERY [67](#)
 REROUTE [67](#)
 RETRY [68](#)
 ROUTE [68](#)
 SAFCLAS [69](#)
 SETMSG [70](#)
 SHADOWS [70](#)
 SLOWDOWN [70](#)
 TOLERANCE [71](#)

- customization, planning for (*continued*)
 - statement description (*continued*)
 - TRACEDEST [71](#)
 - UPARM [71](#)
 - understanding identifier (ID) naming conventions [57](#)
 - using
 - ASCII exits [72](#)
 - the Gateway Programming Interface (GPI) [72](#)
 - the RSCS exit facility [72](#)
 - transmission algorithms [72](#)
- customizing RSCS
 - automating more complex tasks [96](#)
 - automating simple events [95](#)
 - coding statements in the configuration file [90](#)
 - creating an events file [93](#)
 - defining
 - a NOTIFY-type link [98](#)
 - holidays and other special days [97](#)
 - links and routes [86](#)
 - TEMPLATE files [98](#)
 - displaying events [97](#)
 - exit routines [101](#)
 - RSCS message repositories [101](#)
 - setting up
 - NOTIFY-type links [97](#)
 - RSCS configuration file [86](#)
 - RSCS events file [93](#)
 - specifying destination identifiers [100](#)
 - using second-level addressing to avoid routes [89](#)

D

- DA [235](#)
- daily event setup [93](#)
- Data Analysis-APL [235](#)
- data compression [194](#), [209](#), [228](#)
- Data Exchanges
 - definition [20](#)
 - how RSCS handles files [20](#)
 - how RSCS handles messages [20](#)
- days, defining holidays and special [97](#)
- DBCS option
 - specifying on TN3270E-type link [235](#)
- defining
 - 37xx TCU [37](#)
 - alternate link [69](#)
 - cross domain communications [48](#)
 - exit points [60](#), [137](#)
 - form name characteristics [140](#)
 - holidays and other special days [97](#)
 - large network [108](#)
 - links and routes [86](#)
 - local
 - node ID [64](#)
 - non-SNA 3270 printer to VTAM [46](#)
 - SNA 3270 printer to VTAM [46](#)
 - SNA RJE workstation to VTAM [48](#)
 - network with multiple system nodes [106](#)
 - network with single system node [105](#)
 - non-SNA connections [36](#)
 - NOTIFY-type link [98](#)
 - PSF printer destinations [60](#), [135](#)
 - RSCS as a VTAM application [45](#)

- defining (*continued*)
 - RSCS virtual machine [31](#)
 - TEMPLATE files [98](#)
 - virtual storage size [51](#)
 - work hours automatically [96](#)
- defining the structure of your network
 - adding another link to a node [107](#)
 - creating
 - an alternate path [108](#)
 - multiple paths [108](#)
 - printer pool [106](#)
 - routing group [109](#)
 - routing group within a routing group [109](#)
 - defining a large network [108](#)
 - making transitions in links [107](#)
 - specifying fanout links [110](#)
 - using second-level routing to a printer pool [108](#)
 - with a single system node [105](#)
 - with multiple system nodes [106](#)
- degraded mode [70](#)
- delimiter, REXX (/ * */) [59](#), [131](#)
- description of RSCS general users [23](#)
- DEST statement
 - category [56](#)
 - description [60](#)
 - syntax [135](#)
- destination
 - defining PSF printer [135](#)
 - identifiers, specifying [100](#)
- device requirements, ASCII, non-SNA [38](#)
- DIAGNOSE code X'F8' [166](#)
- diagnosing problems using RSCS facilities [83](#)
- diagnosis, planning for
 - checklist [83](#)
 - displaying dump information [84](#)
 - establishing procedures [84](#)
 - producing dumps [84](#)
 - using RSCS facilities [83](#)
 - using tracing facility [83](#)
- diagnostic
 - error messages [83](#)
 - procedures [84](#)
- directory definition
 - RSCS domain name server [113](#)
 - RSCS dynamic authorization server [115](#)
 - RSCS Interchange [120](#)
 - RSCS Interchange server [120](#)
 - RSCS server [31](#)
- displaying
 - dump information [84](#)
 - events [97](#)
- DMTAMENG MSGS translation repository [101](#)
- DMTMGC MCONV conversion repository [101](#)
- domain names, selecting [124](#)
- DOMAIN statement [282](#)
- DUMP statement
 - category [56](#)
 - definition [60](#)
 - statement [136](#)
 - syntax [136](#)
- Dump Viewing Facility
 - displaying dump information [84](#)
 - required dump format [60](#)
- dumps, producing and displaying [84](#)

DUMPSCAN command [84](#)

E

electronic mail [25](#)

elements, reserving tag shadow [70](#)

end node [6](#)

error messages [83](#)

events

 automate simple [95](#)

 displaying [97](#)

 file

 creating [93](#)

 setting up [93](#)

 tracing internal [62](#)

example

 alternative paths in a NJE network [89](#)

 closing console log automatically [95](#)

 comments in a file [59](#)

 creating multiple primary links [108](#)

 defining work hours automatically [96](#)

 large network [109](#)

 links for direct communications [64](#)

 multiple system nodes network [107](#)

 NETCDRM VTAMLST [49](#)

 NETCDRSC VTAMLST [48](#)

 NETCTC VTAMLST [48](#)

 OTHERSYS VTAMLST [49](#)

 PATH0102 VTAMLST [49](#)

 ROUTE [108](#)

 routes for indirect communications [69](#)

 RSCS CONFIG [64](#)

 RSCSNET VTAMLST [64](#)

 RSCSPRT VTAMLST [46](#)

 RSCSPRTN VTAMLST [46](#)

 RSCSRJE VTAMLST [48](#)

 simple two-system network [87](#)

 single system node network [105](#)

 statements

 HIDECHARACTER [61](#)

 LINKDEFINE [105](#), [107](#)

 RECOVERY [67](#)

 RETRY [68](#)

 ROUTE [69](#), [106](#), [107](#), [109](#), [110](#)

 SLOWDOWN [70](#)

 TOLERANCE [71](#)

 systems in TCP/IP network [88](#)

 three-system network [88](#)

execs

 automate initialization using PROFILE GCS [77](#)

 automate operations with [77](#)

 for logging on RSCS automatically [35](#)

exit

 ASCII

 used for customizing [72](#)

 using [102](#)

 facility, customizing RSCS using [72](#)

 routines [101](#)

EXIT statement

 category [56](#)

 description [60](#)

 syntax [137](#)

external event setup [93](#)

F

fanout links, specifying [110](#)

FCB statement

 category [56](#)

 description [61](#)

 syntax [139](#)

file

 imbedding [62](#)

 limit on spool [70](#)

 manipulating, in transit [82](#)

 rerouting [67](#), [253](#)

 spool, blocks, threshold number

 for GATEWAY-type link [175](#)

 for LISTPROC-type link [179](#)

 for NJE-type link [199](#)

 for SNANJE-type link [212](#)

 for TCPNJE-type link [232](#)

 SYSTEM NETID [33](#)

 TEMPLATE, defining and sample [98](#)

FILEMODE statement [271](#)

form name characteristics, defining [140](#)

FORM statement

 category [56](#)

 description [61](#)

 syntax [140](#)

FORM_DEFAULT, system configuration statement [66](#), [162](#)

format

 events file [93](#)

 RSCSV2 dump [84](#)

FROM_FORMAT statement [284](#)

G

gateway

 definition [25](#)

GATEWAY-type links

 preparing for [43](#)

GCS

 FILEDEF command [93](#)

 operating environment [34](#)

 supervisor services [3](#)

 used in customizing [86](#)

 virtual machine group [3](#)

GDDM output [309](#)

GETHOSTC MODULE [114](#)

GPI, using for customizing [72](#)

group

 ID naming convention [57](#)

 routing [69](#)

H

headers, suppressing CP [65](#), [161](#)

hide character [142](#)

HIDECHARACTER statement

 category [56](#)

 description [61](#)

 example [61](#)

 syntax [142](#)

holidays, defining in events file [97](#)

host system [18](#)

I

IBM-defined exits [101](#)
identifier
 destination, specifying [100](#)
 naming conventions [57](#)
identifying RSCS to z/VM [33](#)
IMBED statement
 category [56](#)
 description [62](#)
 syntax [143](#)
imbedding file [62](#)
INFORM_USER statement [285](#)
information collecting procedures [84](#)
initialization, automating [77](#)
initiating SNA sessions [76](#)
inserting additional title lines [261](#)
installation, planning for
 checklist [31](#)
 cross-domain support [48](#)
 defining non-SNA connections [36](#)
 identifying RSCS to z/VM [33](#)
 joining the GCS virtual machine group [34](#)
 logging on the RSCS virtual machine automatically [35](#)
 preparing
 for networking links [43](#)
 for TCP/IP connections [38](#)
 VTAM for SNA connections [44](#)
 RACF toleration [33](#)
 requirements for
 non-SNA connections [37](#)
 non-SNA networking links
 [43](#)
 SNA networking links [44](#)
 SNA 3270 printer support [45](#)
instruction processing [52](#)
internal event setup
 tracing [62](#)
internet, definition [25](#)
IP address, defining [39](#)
IPv4 [10](#)
IPv6 [10](#)
ITRACE statement
 category [56](#)
 description [62](#)
 syntax [145](#)
IUCV connections [39](#)

J

JES2 subsystems [43](#)
jobs, simplifying [82](#)

K

KEEP statement [272](#)

L

LANGUAGE statement
 category [56](#)
 description [62](#)
 syntax [150](#)

limits, spool file [70](#)
lines, inserting additional title [261](#)
link
 adding another, to a node [107](#)
 alternate, defining [69](#)
 authorized alternate operator [132](#)
 auto-answer
 connecting to [67](#)
 invoking line driver [67](#)
 naming conventions [58](#)
 auto-dial
 connecting to [67](#)
 with automatic calling unit (ACU) [78](#)
 auto-sign-on [78](#)
 auto-start (ASTART) [78](#)
 automatically starting [78](#)
 creating transition [107](#)
 defining, when customizing [86](#)
 definition [5](#), [7](#)
 driver
 multistreaming [72](#)
 fanout, specifying [110](#)
 ID naming convention [57](#)
 management, automatic [78](#)
 multiple primary [69](#)
 networking, preparing for [43](#)
 non-SNA types
 networking, requirements [43](#)
 operating [76](#)
 pools [69](#)
 printer, LU sessions for [308](#)
 SNA types
 networking requirements [44](#)
LINK statement [152](#)
link-authorized alternate operator [23](#)
LINKDEFINE statement
 category [56](#)
 defining links and routes [86](#)
 definition [62](#)
 example [105](#), [107](#)
 syntax [151](#)
LINKTYPE statement
 category [56](#)
 definition [63](#)
 syntax [158](#)
list processor [11](#)
LISTPROC-type links
 PARM statement [177](#)
 preparing for [43](#)
loading exit points [60](#), [137](#)
local node [5](#)
local node ID, defining [64](#)
LOCAL statement
 category [56](#)
 description [64](#)
 syntax [160](#)
local/remote IP address [41](#)
LOG statement [273](#)
LOG_DISK statement [286](#)
LOG_KEEP statement [287](#)
logging on RSCS automatically [35](#)
logon
 automatic, sample exec [35](#)
 mode table entries

- logon (*continued*)
 - mode table entries (*continued*)
 - samples RSCSTAB ASSEMBLE file [307](#)
 - supplying [49](#)
 - loops, avoiding routing [108](#)
 - LPD connections [40](#)
 - LPD-Type Link PARM statement [181](#)
 - LPR-type links
 - exits, used for customizing [72](#)
 - PARM statement [183](#)
 - requirements [40](#)
 - LU
 - 3270 printer, type sessions [45](#)
 - ID naming convention [58](#)
 - sessions for printer links [308](#)

M

- manipulating files in transit [82](#)
- MAXCONN value recommendation [39](#)
- message
 - columnar [65](#), [161](#)
 - file, RSCS [71](#)
 - nonprivate, routing [262](#)
 - repositories, RSCS [101](#)
 - rerouting [67](#), [253](#)
 - sending [65](#)
 - suppressing CP, headers [65](#), [161](#)
 - template variables, NOTIFY [98](#)
- message examples, notation used in [xix](#)
- midnight event setup [93](#)
- mode, degraded [70](#)
- MSGNOH statement
 - category [56](#)
 - description [65](#)
 - syntax [161](#)
- multiple
 - advantages of RSCS [22](#)
 - definition [19](#)
 - links to a node, adding [107](#)
 - network paths, creating [108](#)
 - primary links [69](#)
 - system node network [106](#)
- multistreaming [12](#), [72](#)

N

- naming conventions, identifier (ID) [57](#)
- NETCDRM VTAMLST, example [49](#)
- NETCDRSC VTAMLST, example [48](#)
- NETCTC VTAMLST, example [48](#)
- network
 - cross-domain support [48](#)
 - defining
 - a large [108](#)
 - single system node [105](#)
 - structure [105](#)
 - with multiple system nodes [106](#)
 - definition [5](#)
 - definition statements [56](#)
 - examples
 - alternative paths in a NJE [89](#)
 - large [109](#)

- network (*continued*)
 - examples (*continued*)
 - multiple system nodes [107](#)
 - simple two-system [87](#)
 - single system node [105](#)
 - system nodes in a [297](#)
 - TCP/IP [88](#)
 - three-system [88](#)
 - information, preparing [82](#)
 - links, preparing for [43](#)
 - requirements
 - non-SNA links [43](#)
 - SNA links [44](#)
 - NETWORK START command [65](#)
 - network, definition [25](#)
 - NICK_OVERRIDE statement [289](#)
 - NICKPRIV statement [290](#)
 - NICKTABLE statement [291](#)
 - NJE-type links
 - automatic calling unit (ACU) [197](#)
 - PARM statement [193](#)
 - preparing for [43](#)
 - requirements [38](#), [43](#)
 - node
 - adding another link to a [107](#)
 - adjacent [5](#)
 - characteristics in a network [6](#)
 - definition [5](#)
 - ID naming convention [57](#)
 - multiple system, network [106](#)
 - single system, network [105](#)
 - store-and-forward [6](#)
 - system [6](#)
 - non-SNA
 - connections, defining [36](#)
 - connections, requirements [37](#)
 - networking links requirements [43](#)
 - remote workstation requirements [38](#)
 - nonadjacent node [5](#)
 - nonprivate message, routing [262](#)
 - notation used in message and response examples [xix](#)
 - NOTIFY-type links
 - message template variables [98](#)
 - PARM statement [201](#)
 - setting up and defining [97](#)
 - number and sequence of statements [91](#)

O

- ongoing operation, automating [77](#)
- operating links [76](#)
- operation, planning for
 - automating initialization [77](#)
 - automating ongoing operation [77](#)
 - checklist [75](#)
 - initiating SNA sessions [76](#)
 - operating links [76](#)
 - operating the RSCS virtual machine [75](#)
 - preparing to automate operations with execs [77](#)
 - preparing to automate using event scheduler [78](#)
 - shutting down RSCS [78](#)
- operational exchanges, definition [21](#)
- operator
 - console [23](#)

- operator (*continued*)
 - link-authorized alternate [23](#)
 - remote [23](#)
 - RSCS Interchange [24](#)
 - RSCS Interchange privileged user [24](#)
 - RSCSAUTH [24](#)
 - RSCSAUTH administrator [24](#)
 - RSCSDNS [23](#)
 - system-authorized alternate [23](#)
- operator exchange, definition [21](#)
- operator, authorizing alternate [132](#)
- OPFORM statement
 - category [56](#)
 - definition [66](#)
 - syntax [162](#)
- OPTION statement
 - category [56](#)
 - description [66](#)
 - syntax [163](#)
- origin ID support, secure [166](#)
- OTHERSYS VTAMLST, example [49](#)
- overprinting [220](#), [237](#), [248](#)

P

- pacings
 - facility [308](#)
 - value, specifying [45](#), [308](#)
- page fault support, support [53](#)
- PARM statement
 - 3270P-Type Link [245](#)
 - ASCII-Type Link [169](#)
 - GATEWAY-Type Link [172](#)
 - LISTPROC-Type Link [177](#)
 - LPD-Type Link [181](#)
 - LPR-Type Link [183](#)
 - MRJE-Type Link [188](#)
 - NJE-Type Link [193](#)
 - NOTIFY-Type Link [201](#)
 - RJE-Type Link [203](#)
 - SNA3270P-Type Link [217](#)
 - SNANJE-Type Link [208](#)
 - SNARJE-Type Link [214](#)
 - TCPASCII-Type Link [224](#)
 - TCPNJE-Type Link [227](#)
 - TN3270E-Type Link [234](#)
 - UFT-Type Link [240](#)
 - UFTD-Type Link [243](#)
- PATH0102 VTAMLST, example [49](#)
- paths, creating network [108](#)
- peer system [5](#)
- performance considerations
 - common problems [52](#)
 - instruction processing [52](#)
 - processor storage [53](#)
 - resource considerations [52](#)
- permanent event setup [93](#)
- planning
 - adding RSCS to a z/VM system [31](#)
 - administration [81](#)
 - configuration [55](#)
 - diagnosis [83](#)
 - operation [75](#)

- pool
 - creating printer [106](#)
 - link [69](#)
- port redirector task [39](#)
- PORT statement
 - category [56](#)
 - definition [67](#)
 - syntax [251](#)
- prefix, CRI [262](#)
- preparing
 - networking links [43](#)
 - non-SNA connections [36](#)
 - TCP/IP connections [38](#)
 - VTAM for SNA connection [44](#)
 - z/VM for RSCS [31](#)
- primary links, multiple [69](#)
- printer
 - automatic load balancing [106](#)
 - destinations, defining PSF [60](#), [135](#)
 - links, LU sessions for [308](#)
 - page-image, identify [60](#)
 - pool
 - creating [106](#)
 - using second-level routing [108](#)
 - sharing [17](#)
- printer queue names [186](#)
- PRINTER_TITLE, system configuration statement [69](#), [261](#)
- privileged command authorization [60](#)
- problem diagnosis, RSCS facilities [83](#)
- procedures, information collecting [84](#)
- processor storage, reserving [53](#)
- processor, list [11](#)
- producing dumps [84](#)
- PROFILE GCS
 - exec for recovery machine [35](#)
- Programmable Operator Facility (PROP)
 - handling routine system operations [21](#)
- pseudo page fault support [53](#)
- PSF
 - defining printer destinations [135](#)
 - destination identifiers, specifying [100](#)
 - printer destinations, defining [60](#)

Q

- quick dispatch option [52](#)

R

- RACF toleration [33](#)
- RACF, preparing for RSCS Interchange [123](#)
- recovery machine, PROFILE GCS exec for [35](#)
- RECOVERY statement
 - category [56](#)
 - description [67](#)
 - example [67](#)
 - syntax [252](#)
- remote job entry (RJE) [18](#)
- remote workstation operator [23](#)
- remote workstations
 - advantages of [18](#)
 - definition [18](#)
 - host system [18](#)

- remote workstations (*continued*)
 - location [18](#)
 - node [5](#)
 - submitting jobs from [18](#)
- remote workstations requirements, non-SNA [38](#)
- repetitive event setup [93](#)
- repositories, RSCS message [101](#)
- requirements
 - non-SNA
 - 3270 printer [37](#)
 - ASCII device [38](#)
 - connections [37](#)
 - networking links [43](#)
 - remote workstation [38](#)
 - SNA
 - 3270 printers [47](#)
 - networking links [44](#)
 - RJE workstation [47](#)
 - virtual storage [50](#)
- REROUTE statement
 - category [56](#)
 - description [67](#)
 - syntax [253](#)
- rerouting commands, files, and messages [253](#)
- reserving
 - channels [60](#), [134](#)
 - tag shadow elements [70](#), [264](#)
- resource considerations [52](#)
- response examples, notation used in [ix](#)
- restrictions
 - customizing RSCS [102](#)
 - imposed by communications programs [58](#)
 - using commands [76](#)
- RETRY statement
 - category [56](#)
 - description [68](#)
 - syntax [257](#)
- REXX delimiter (*/ * **) [59](#), [131](#)
- RJE (remote job entry) [18](#)
- RJE-type links
 - automatic calling unit (ACU) [206](#)
 - PARM statement [203](#)
- ROUTE statement
 - category [56](#)
 - description [68](#)
 - example [69](#), [106–110](#)
 - syntax [258](#)
- routes, defining when customizing [86](#)
- routines, exit [101](#)
- routing
 - group, creating [109](#)
 - loops, avoiding [108](#)
 - nonprivate messages [262](#)
 - using second-level, to a printer spool [108](#)
- RSCS
 - automating operation [93](#)
 - define as a VTAM application [45](#)
 - definition [3](#)
 - environment [17](#)
 - exit facility [72](#)
 - file transfer management [4](#)
 - function statements [56](#)
 - identifying, to z/VM [33](#)
 - in GCS operating environment [34](#)

- RSCS (*continued*)
 - message file [71](#)
 - operator [23](#)
 - programs supported in SNA networks [44](#)
 - virtual machine environment [3](#)
- RSCS CONFIG, example [64](#)
- RSCS CONFSAMP [86](#)
- RSCS Data Interchange Manager
 - configuration file [119](#)
 - configuring [123](#)
 - connections to SMTP and RSCS, defining [123](#)
 - domain names, selecting [124](#)
 - exits [119](#)
 - failure template [119](#)
 - how it works [26](#)
 - operator [24](#)
 - preparing RACF [123](#)
 - preparing RSCS [121](#)
 - preparing SMTP [123](#)
 - privileged user [24](#)
 - sample configuration file [313](#)
 - sample files [119](#)
 - server profile [119](#)
 - setting up [119](#)
 - statements [279](#)
 - updates to RSCS configuration file [119](#)
 - virtual machine definition, sample [120](#)
- RSCS Data Interchange Manager Configuration File
 - Statements
 - ADMIN [280](#)
 - AUTHORIZE [281](#)
 - DOMAIN [282](#)
 - EXIT [283](#)
 - FROM_FORMAT [284](#)
 - INFORM_USER [285](#)
 - LOG_DISK [286](#)
 - LOG_KEEP [287](#)
 - MSGNOH [288](#)
 - NICK_OVERRIDE [289](#)
 - NICKPRIV [290](#)
 - NICKTABLE [291](#)
 - RSCSLINK [292](#)
 - RSCSNAME [293](#)
 - rules for coding [279](#)
 - sample configuration file [313](#)
 - SMTpname [294](#)
 - TEMP_KEEP [295](#)
- RSCS domain name server
 - description [11](#)
 - GETHOSTC MODULE [114](#)
 - operator [23](#)
 - setting up [113](#)
- RSCS dynamic authorization
 - administrator [24](#)
 - authorized users [23](#)
 - configuration file [115](#)
 - configuration statements [269](#)
 - configuring [116](#), [269](#)
 - description [11](#)
 - identification to RSCS [116](#)
 - operator [24](#)
 - sample configuration file [311](#)
 - sample files [115](#)
 - server profile [115](#)

- RSCS dynamic authorization (*continued*)
 - setting up [115](#)
 - virtual machine definition, sample [115](#)
- RSCS environment
 - multisystem [19](#)
 - single-system [17](#)
- RSCS general users, description [23](#)
- RSCS operator
 - console [23](#)
 - link-authorized alternate [23](#)
 - remote [23](#)
 - RSCS Interchange [24](#)
 - RSCS Interchange privileged user [24](#)
 - RSCSAUTH administrator [24](#)
 - RSCSAUTH operator [24](#)
 - RSCSDNS operator [23](#)
 - system-authorized alternate [23](#)
- RSCS statement [274](#)
- RSCSAUTH CONFIG file [115](#)
- RSCSAUTH user ID [115](#)
- RSCSCMDS CONFIG file [117](#)
- RSCSDNS user ID [113](#)
- RSCSLINK statement [292](#)
- RSCSNAME statement [293](#)
- RSCSNET VTAMLST, example [45, 64](#)
- RSCSPRT VTAMLST, example [46](#)
- RSCSPRTN VTAMLST, example [46](#)
- RSCSRJE VTAMLST, example [48](#)
- RSCSTAB ASSEMBLE [49](#)
- RSCSTAB LKEDCTRL file [49](#)
- RSCSV2 dump format [84](#)
- rules for coding configuration file statements [129](#)

S

- SAFCLASS statement
 - category [56](#)
 - definition [69](#)
 - syntax [261](#)
- sample
 - automatic logon exec [35](#)
 - configuration file
 - adding a link [107](#)
 - creating a routing group [109, 110](#)
 - creating alternate link [108](#)
 - for printer pool [106](#)
 - for single system node [105](#)
 - multiple system nodes [107](#)
 - RSCS dynamic authorization [115](#)
 - RSCS Interchange [119](#)
 - RSCS updates for RSCS Interchange [119](#)
 - supplied with RSCS [86](#)
 - configuration file, RSCS [297](#)
 - directory definition
 - RSCS domain name server [113](#)
 - RSCS dynamic authorization server [115](#)
 - RSCS Interchange server [120](#)
 - RSCS server [31](#)
 - failure template, RSCS Interchange [119](#)
 - large network [109](#)
 - logon mode table entries [307](#)
 - network with multiple system nodes [107](#)
 - network, single system node [105](#)
 - server profile

- sample (*continued*)
 - server profile (*continued*)
 - RSCS dynamic authorization [115](#)
 - RSCS Interchange [119](#)
 - TEMPLATE file [99](#)
 - scheduling routine tasks [93](#)
 - SCIF (Single Console Image Facility) [75](#)
 - second-level addressing [89](#)
 - secondary user [75](#)
 - secure origin ID support [166](#)
 - sending mail to SMTP users [27](#)
 - SET PAGEX command [53](#)
 - SETMSG statement
 - category [56](#)
 - description [70](#)
 - SETMSG Statement [262](#)
 - setting up
 - configuration file [86](#)
 - events file [93](#)
 - NOTIFY-type link [97](#)
 - SHADOWS statement
 - category [56](#)
 - description [70](#)
 - syntax [264](#)
 - sharing printers [17](#)
 - shutting down RSCS [78](#)
 - single
 - event setup [93](#)
 - system node network [105](#)
 - Single Console Image Facility (SCIF) [75](#)
 - single system environment
 - advantages of RSCS [19](#)
 - ASCII advantages [19](#)
 - definition [17](#)
 - remote workstation advantages [19](#)
 - SLOWDOWN statement
 - category [56](#)
 - description [70](#)
 - syntax [265](#)
 - SMTP (Simple Message Transfer Protocol)
 - definition [25](#)
 - SMTP mail to z/VM [28](#)
 - SMTP, preparing for RSCS Interchange [123](#)
 - SMTPNAME statement [294](#)
 - SNA
 - initiating, session [76](#)
 - requirements, networking links [44](#)
 - RJE
 - local workstation, define to VTAM [48](#)
 - workstation requirements [47](#)
 - VTAM, connection, preparing for [44](#)
 - SNA Character Strings (SCS) [215, 220](#)
 - SNA3270P-type links
 - defining to VTAM [45](#)
 - PARM statement [217](#)
 - requirements [47](#)
 - SNANJE-type links
 - PARM statement [208](#)
 - preparing for [43](#)
 - SNARJE-type links
 - PARM statement [214](#)
 - workstation requirements [47](#)
 - special days, defining in events file [97](#)
 - spool file

- spool file (*continued*)
 - block
 - for GATEWAY-type link [175](#)
 - for LISTPROC-type link [179](#)
 - for NJE-type link [199](#)
 - for SNANJE-type link [212](#)
 - for TCPNJE-type link [232](#)
 - limits [70](#)
 - makes sharing printers possible [17](#)
 - performance considerations [53](#)
- spool, VM system
 - file tag [5](#)
- spooling, definition [3](#)
- statement
 - AUTH [132](#)
 - CHANNELS [134](#)
 - coding in configuration file [90](#)
 - comments (symbols) [59](#), [131](#)
 - CP function [56](#)
 - DEST [135](#)
 - DUMP [136](#)
 - EXIT [137](#)
 - FCB [139](#)
 - FORM [140](#)
 - HIDECHARACTER [142](#)
 - IMBED [143](#)
 - ITRACE [145](#)
 - LANGUAGE [150](#)
 - LINKDEFINE [151](#)
 - LINKTYPE [158](#)
 - LOCAL [160](#)
 - MSGNOH [161](#)
 - network definition [56](#)
 - number and sequence of, in configuration file [91](#)
 - OPFORM [162](#)
 - OPTION [163](#)
 - PARM [168](#)
 - PORT [251](#)
 - RECOVERY [252](#)
 - REROUTE [253](#)
 - RETRY [257](#)
 - ROUTE [258](#)
 - RSCS function [56](#)
 - rules for coding [129](#)
 - SAFCLASS [261](#)
 - SHADOWS [264](#)
 - SLOWDOWN [265](#)
 - summary of [90](#), [129](#)
 - TOLERANCE [266](#)
 - TRACEDEST [267](#)
 - UPARM [268](#)
- station
 - node [6](#)
- station node [6](#)
- storage
 - reserving [53](#)
 - size, defining virtual [51](#)
 - virtual, considerations [50](#)
- store-and-forward
 - node [6](#)
- structure, defining network [105](#)
- summary of configuration file statements [90](#), [129](#)
- syntax diagrams, how to read [xvii](#)
- system
 - event setup [93](#)
 - node [6](#)
 - node network
 - multiple, defining [106](#)
 - single [105](#)
 - nodes in a sample network [297](#)
 - resources, access priority [53](#)
- SYSTEM CONFIG file, sample [36](#)
- system configuration file statements
 - FORM_DEFAULT [66](#), [162](#)
 - PRINTER_TITLE [69](#), [261](#)
 - SYSTEM_IDENTIFIER [160](#)
 - SYSTEM_IDENTIFIER_DEFAULTS [160](#)
- SYSTEM NETID file [33](#)
- system spool, VM
 - file tag [5](#)
- SYSTEM_IDENTIFIER_DEFAULTS, system configuration statement [160](#)
- SYSTEM_IDENTIFIER, system configuration statement [160](#)
- system-authorized alternate operator [23](#), [132](#)
- system, peer [5](#)

T

- tag
 - slot [5](#)
 - to complete transfer of files [11](#)
 - to manage files [4](#)
- tag shadow elements, reserving [70](#), [264](#)
- tasks
 - automating complex [96](#)
- TCP/IP Machine Configuration file [41](#), [42](#)
- TCP/IP networking support
 - defining IP address [39](#)
 - IUCV requirements [39](#)
 - required information [39](#)
 - sample network [88](#)
- TCPASCII-type links
 - exits, used for customizing [72](#)
 - PARM statement [224](#)
 - requirements [41](#)
- TCPNJE-type links
 - PARM statement [227](#)
 - requirements [41](#)
 - sample network [88](#)
- TCU, defining a 37xx [37](#)
- TELL statement [275](#)
- TEMP_KEEP statement [295](#)
- TEMPLATE files, defining and sample [98](#)
- temporary event setup [93](#)
- title lines, inserting additional [261](#)
- TN3270E-type link [234](#)
- TOLERANCE statement
 - category [56](#)
 - definition [71](#)
 - example [71](#)
 - syntax [266](#)
- toleration, RACF [33](#)
- TRACEDEST statement
 - category [56](#)
 - description [71](#)
 - syntax [267](#)
- TRACERSCSAUTH statement [276](#)

tracing facility used for diagnosis [83](#)
tracing internal events [62](#)
transition link, creating [107](#)
translation repositories [101](#)
transmission algorithms
 customizing RSCS using [72](#)
 used in exit routine [102](#)
TRANSPPOOL statement [277](#)
two-level addressing [89](#)

U

UFT-type link PARM statement [240](#)
UFTD statement
 UFTD statement [243](#)
UFTD-type link PARM statement [243](#)
UNKNOWN TEMPSAMP file, sample [99](#)
UPARM statement
 category [56](#)
 description [71](#)
 syntax [268](#)
user
 defined exits [101](#)
 event setup [93](#)
 ID naming convention [57](#)
 secondary [75](#)
user exchange
 electronic mail [20](#)
 how RSCS handles messages [20](#)
using second-level routing to a printer pool [108](#)

V

variables, NOTIFY message template [98](#)
virtual machine group [3](#)
virtual storage considerations [50](#)
VM RSCS Data Interchange Manager
 illustration [25](#)
 overview [25](#)
VM system pool
 file tag [5](#)
VM-to-SMTP mail [27](#)
VMFLKED command [49](#)
VSCS (VTAM SNA Console Support) [46](#)
VTAM
 communicating with [8](#)
 cross-domain support [48](#)
 define
 local non-SNA 3270 printer to [46](#)
 local SNA 3270 printer to [46](#)
 local SNA RJE workstation to [48](#)
 RSCS as a, application [45](#)
 SNA 3270 printer to [45](#)
 logon mode table entries
 sample RSCSTAB ASSEMBLE file [307](#)
 supplying [49](#)
 pacing facility [45](#), [308](#)
 preparing, for SNA [44](#)

W

warning

warning (*continued*)
 customizing RSCS [102](#)
 using LINKTYPE statement [63](#), [158](#)
work hours, defining automatically [96](#)
workstation
 local SNA, define to VTAM [48](#)
 remote, tending to [81](#)
 requirements, remote non-SNA
 38
 RJE, SNA requirements [47](#)

X

XCHANGE CONFIG file [119](#)
XCHANGE user ID [120](#)

Z

z/VM
 defining the RSCS [31](#)
 group, typical [34](#)
 preparing for RSCS [31](#)
 resource considerations [52](#)
 sample virtual machine definition [31](#)



Product Number: 5741-A09

Printed in USA

SC24-6320-73

