

---

# DYNAMIC REPORT SYSTEM

## V1 R3.3

### Subject to License/Trade Secret

Levi, Ray & Shoup, Inc. ("LRS") has prepared this software documentation for use only by LRS personnel and its licensees. The information contained herein is the property of LRS, contains LRS trade secrets, and is licensed to you. Unless you are an employee of LRS or a licensee, you are not authorized to use this software documentation.

All warranties, if any, concerning the software are set forth in a license agreement with your company. This document creates no warranties pertaining to the software.

### Trademarks



VPS<sup>®</sup>, VPS/Secure<sup>®</sup>, LRS<sup>®</sup>, PageCenter<sup>®</sup>, PageCenter Plus<sup>®</sup>, AnyQueue<sup>®</sup>, AnyQueue/PageSorter<sup>®</sup>, AnyQueue/WebTRAC<sup>®</sup>, AnyQueue/Secure<sup>®</sup>, DRS/OutputManager<sup>™</sup>, DRS/Secure<sup>™</sup> PageCenter Plus To-Go<sup>™</sup>, VPSX<sup>™</sup>, and VPSX/OutputManager<sup>™</sup> are trademarks of Levi, Ray & Shoup, Inc. All other brand and product names are trademarks of their respective holders.

### Copyright

Copyright 2005 Levi, Ray & Shoup, Inc. All rights are reserved. As a licensed user of this documentation, to the extent allowed by your license, and only for use within your company, you may make additional copies of this manual, download it from the licensee's section of the LRS Web site ([www.lrs.com](http://www.lrs.com)) in PDF format, or order additional copies, for a charge, from LRS.

Levi, Ray & Shoup, Inc.  
2401 West Monroe Street  
Springfield, IL 62704  
Phone: 217-793-3800  
Fax: 217-787-4014  
<http://www.lrs.com>

Document Number: S020-0330-5 Revision Date: April 14, 2005

---



---

# Table of Contents

<b>Summary of Enhancements</b> .....	<b>xiii</b>
--------------------------------------	-------------

## **Section 1**

<b>Overview</b> .....	<b>1.1</b>
-----------------------	------------

DRS/API .....	1.2
Storage Estimates .....	1.2
DRS/VPI .....	1.3
Storage Estimates .....	1.3
DRS/TCPIP .....	1.4
Storage Estimates .....	1.4
DRS/STI .....	1.5
Storage Estimates .....	1.5
DRS/SAPR2 .....	1.6
Storage Estimates .....	1.6
DRS/NATURAL .....	1.7
DRS/PC .....	1.8
Advantages of using DRS .....	1.9

## **Section 2**

<b>Introduction to DRS/VPI</b> .....	<b>2.1</b>
--------------------------------------	------------

## **Section 3**

<b>DRS/VPI Installation</b> .....	<b>3.1</b>
-----------------------------------	------------

New Features in this Release .....	3.2
Migrating to DRS/VPI V1 R3.3 from R3.0, R3.1 or R3.2 .....	3.3
Installation Procedure .....	3.4
VTAM Definition Requirements .....	3.5
Defining DRS/VPI Job Control Language .....	3.7
DRS System Options Module for DRS/VPI .....	3.8
DRS/VPI Control Library Members .....	3.9
Syntax of DRS/VPI Library Members .....	3.9
System Initialization Member .....	3.10
Message Modification Member .....	3.10
Printer Activation Inclusion List Member .....	3.10
Printer Activation Exclusion List Member .....	3.11
Printer Default Member .....	3.11
Printer Definition Members .....	3.12
Output Reference Members .....	3.12
Building the System Initialization Member .....	3.13
Overview .....	3.13
System Initialization Parameters .....	3.13
Product Key Parameters .....	3.13
Logging Parameters .....	3.14
Snap Dump Parameters .....	3.14
Message Processing Parameters .....	3.14

---

User Exit Parameters . . . . .	3.14
TCP/IP Parameters . . . . .	3.15
Miscellaneous Parameters . . . . .	3.15
Building the Printer Activation Inclusion List . . . . .	3.39
Building the Printer Activation Exclusion List . . . . .	3.40
Building the Printer Default Member . . . . .	3.41
Building the Printer Definition Members . . . . .	3.42
Overview . . . . .	3.42
Spooling Attributes . . . . .	3.42
Record Format Attributes . . . . .	3.42
DASD Attributes . . . . .	3.42
TCP/IP Parameters . . . . .	3.43
Miscellaneous Parameters . . . . .	3.43
Minimum DRS/VPI Printer Definition Requirements . . . . .	3.44
Building the Output Reference Members . . . . .	3.107

**Section 4**  
**DRS/VPI With VTAM Applications . . . . . 4.1**

**Section 5**  
**DRS/VPI With DRS/PC . . . . . 5.1**

Receiving AFP records from the LAN using DRS/PC . . . . .	5.3
---	-----

**Section 6**  
**DRS/VPI With DRS/TCPIP . . . . . 6.1**

Receiving AFP Records using DRS/TCPIP . . . . .	6.3
---	-----

**Section 7**  
**DRS/VPI With DRS/STI . . . . . 7.1**

**Section 8**  
**DRS/VPI Operation . . . . . 8.1**

Introduction . . . . .	8.1
The START, STOP, and MODIFY Commands . . . . .	8.1
DRS/VPI Command Summary . . . . .	8.3
DRS/VPI Commands . . . . .	8.4

**Section 9**  
**DRS/VPI User Exits . . . . . 9.1**

Separator Routine . . . . .	9.2
DRS/VPI Exits . . . . .	9.5
Exit 01: DRS/VPI Startup Exit . . . . .	9.6
Exit 02: DRS/VPI Shutdown Exit . . . . .	9.8
Exit 03: DRS/VPI Activate Exit . . . . .	9.9
Exit 04: DRS/VPI Inactivate Exit . . . . .	9.11
Exit 05: DRS/VPI Command Exit . . . . .	9.13
Exit 06: DRS/VPI WTO Exit . . . . .	9.15

---

Exit 07: DRS/VPI Report Initialization Exit . . . . .	9.16
Exit 08: DRS/VPI Error Retry Exit . . . . .	9.18
Exit 09: DRS/VPI TCP/IP Command Exit . . . . .	9.20
Exit 10: DRS/VPI Report Termination Exit . . . . .	9.22

**Section 10**  
**Special Topics . . . . .10.1**

Introduction . . . . .	10.1
Creating a User Translation Table . . . . .	10.2
Overview . . . . .	10.2
Definitions . . . . .	10.2
Translation of Single Byte Character Sets . . . . .	10.2
Translation of Double Byte Character Sets . . . . .	10.3
Creating Translation Tables . . . . .	10.3
Creating a SBCS Translation Table . . . . .	10.4
Creating a DBCS Translation Table . . . . .	10.5
Creating a Report Termination Table . . . . .	10.6
Overview . . . . .	10.6
Definitions . . . . .	10.6
Defining the Report Termination Table . . . . .	10.6
Creating a User Report Termination Table . . . . .	10.7

**Section 11**  
**DRS/VPI Problem Determination . . . . .11.1**

Display Commands . . . . .	11.2
Displaying DRS/VPI System Status . . . . .	11.2
Displaying DRS/VPI User Exit Status . . . . .	11.2
Displaying DRS/VPI OUTREF information . . . . .	11.2
Displaying DRS/VPI Virtual Printers . . . . .	11.2
Messages . . . . .	11.4
DRS/VPI Log . . . . .	11.5
Internal Trace . . . . .	11.6
GTF Tracing . . . . .	11.7
Snap Dump . . . . .	11.8
System Dump . . . . .	11.9

**Section 12**  
**Introduction to DRS/TCPIP . . . . .12.1**

**Section 13**  
**DRS/TCPIP Installation . . . . .13.1**

New Features in this Release . . . . .	13.1
Installation Procedure . . . . .	13.2
DRS/TCPIP System Requirements . . . . .	13.3
TCP/IP for MVS Requirements . . . . .	13.3
MVS Requirements . . . . .	13.3
Adding System Keywords for DRS/TCPIP . . . . .	13.4
Building TCP/IP Print Queue Definitions . . . . .	13.5

---

Converting the IBM TCPIP LPD Configuration Data Set . . . . .	13.6
Procedure DD Statements . . . . .	13.6
Procedure Parameters . . . . .	13.6
Messages Issued by the LPD Conversion Program . . . . .	13.7
Adding TCP/IP Printers to DRS/VPI Activation List . . . . .	13.10
Installing DRS/VPI User Exit 9 for TCP/IP . . . . .	13.11

**Section 14**

<b>DRS/TCPIP Problem Determination . . . . .</b>	<b>14.1</b>
--	-------------

**Section 15**

<b>Introduction To DRS/STI . . . . .</b>	<b>15.1</b>
--	-------------

**Section 16**

<b>DRS/STI Installation . . . . .</b>	<b>16.1</b>
---------------------------------------	-------------

Installation Procedure . . . . .	16.1
Defining the DRS/STI Rules Dataset . . . . .	16.2
Allocation Guidelines . . . . .	16.4
Smart Tag Control Area . . . . .	16.4
ISPF Definition Requirements . . . . .	16.6
CICS Definition Requirements . . . . .	16.9
Populating the DRS/STI Rules Data Set . . . . .	16.10
Procedure DD Statements . . . . .	16.10
Procedure Parameters . . . . .	16.11
Messages . . . . .	16.12
Backing Up the DRS/STI Rules Dataset . . . . .	16.17
Reorganizing the DRS/STI Rules Dataset . . . . .	16.18

**Section 17**

<b>DRS/STI Administration . . . . .</b>	<b>17.1</b>
---	-------------

DRS/STI Administration Primary Menu . . . . .	17.3
DRS/STI Administration Vendor List . . . . .	17.4
DRS/STI Administration Vendor Add . . . . .	17.5
DRS/STI Administration Vendor Update . . . . .	17.6
DRS/STI Administration Vendor Delete . . . . .	17.7
DRS/STI Administration Application List . . . . .	17.8
DRS/STI Administration Application Add . . . . .	17.9
DRS/STI Administration Application Update . . . . .	17.10
DRS/STI Administration Application Delete . . . . .	17.11
DRS/STI Administration Profile List . . . . .	17.12
DRS/STI Administration Profile Add . . . . .	17.13
DRS/STI Administration Profile Update . . . . .	17.14
DRS/STI Administration Profile Delete . . . . .	17.15
DRS/STI Administration Printer Type Menu . . . . .	17.16
DRS/STI Administration Printer List . . . . .	17.17
DRS/STI Administration Printer Add . . . . .	17.18
DRS/STI Administration Printer Update . . . . .	17.19
DRS/STI Administration Printer Delete . . . . .	17.20

---

DRS/STI Administration Output List . . . . .	17.21
DRS/STI Administration Output Add Panel 1. . . . .	17.22
DRS/STI Administration Output Add Panel 2. . . . .	17.23
DRS/STI Administration Output Add Panel 3. . . . .	17.24
DRS/STI Administration Output Update Panel 1 . . . . .	17.25
DRS/STI Administration Output Update Panel 2 . . . . .	17.26
DRS/STI Administration Output Update Panel 3 . . . . .	17.27
DRS/STI Administration Output Delete Panel 1 . . . . .	17.28
DRS/STI Administration Delete Panel 2 . . . . .	17.29
DRS/STI Administration Output Delete Panel 3 . . . . .	17.30
DRS/STI Administration Printer Type List . . . . .	17.31
DRS/STI Administration Printer Type Add . . . . .	17.32
DRS/STI Administration Printer Type Update . . . . .	17.33
DRS/STI Administration Printer Type Delete. . . . .	17.34
DRS/STI Administration Userid List. . . . .	17.35
DRS/STI Administration Userid Add . . . . .	17.36
DRS/STI Administration Userid Update . . . . .	17.37
DRS/STI Administration Userid Delete . . . . .	17.38

**Section 18**

<b>DRS/STI Under CICS . . . . .</b>	<b>18.1</b>
-------------------------------------	-------------

Screen Descriptions . . . . .	18.2
DRS/STI CICS Printer Menu . . . . .	18.2
DRS/STI CICS User Profile List. . . . .	18.3
DRS/STI CICS System Printer List . . . . .	18.4
DRS/STI CICS System Output List . . . . .	18.5
DRS/STI Transaction Interface . . . . .	18.6
DRS/STI Transaction Invocation. . . . .	18.7
Messages . . . . .	18.8

**Section 19**

<b>Introduction to DRS/API . . . . .</b>	<b>19.1</b>
--	-------------

**Section 20**

<b>DRS/API Installation . . . . .</b>	<b>20.1</b>
---------------------------------------	-------------

Introduction . . . . .	20.1
New Features in this Release . . . . .	20.2
Migrating to DRS V1 R3.3 from an Earlier DRS Release . . . . .	20.3
Installation Steps . . . . .	20.4
Restoring the DRS Distribution Libraries. . . . .	20.6
Customizing the DRS System Options . . . . .	20.9
Installing the DRSSKEY Module . . . . .	20.13
Updating the CICS Tables. . . . .	20.14
Installing an Updated Version of DRS . . . . .	20.17

---

## Section 21

### DRS/API Request Calls .....21.1

Introduction .....	21.1
General Information about DRS Calls .....	21.2
CICS Mode .....	21.2
Non-CICS Mode .....	21.2
Assembler Coding .....	21.3
COBOL Coding .....	21.4
Parameters for DRS Calls .....	21.5
The INIT Call .....	21.5
The PUT Call .....	21.5
The TERM Call .....	21.6
The QURY Call .....	21.6
The CMND Call .....	21.6
The OUTP Call .....	21.6
DRS Control Blocks .....	21.7
Request Block (DRRB) .....	21.7
INIT Block (DRIB) .....	21.9
Termination Block (DRTB) .....	21.19
Query Block (DRQB) .....	21.21
Command Block (DRCB) .....	21.39
Output Block (DROB) .....	21.40

## Section 22

### DRS/API User Exits .....22.1

Introduction .....	22.1
Language .....	22.1
Reentrancy Considerations .....	22.1
Exit Types & Recovery Considerations .....	22.2
Execution Environment .....	22.2
Linkage Conventions .....	22.2
Parameters Passed .....	22.2
Return Codes .....	22.2
Exit-to-Exit Communication .....	22.2
Exit Tracing .....	22.3
Exit Selection Table .....	22.4
Initialization Exit (Exit 00) .....	22.5
Termination Exit (Exit 01) .....	22.6
Request Exit (Exit 02) .....	22.7
Add Lines Exit (Exit 03) .....	22.8
Snap Dump Exit (Exit 04) .....	22.9
WTO Exit (Exit 05) .....	22.10
OS Services Exit (Exit 06) .....	22.11

## Section 23

### DRS/API JES Considerations .....23.1

JES2 Considerations .....	23.1
JES3 Considerations .....	23.2



---

<b>Section 24</b>	
<b>DRS/API Return Codes</b> .....	<b>24.1</b>
<b>Section 25</b>	
<b>Sample DRS/API Programs</b> .....	<b>25.1</b>
Sample COBOL and Assembler Programs .....	25.2
DRS Call Utility .....	25.5
Installing the DRS Call Utility .....	25.6
<b>Section 26</b>	
<b>Introduction To DRS/SAPR2</b> .....	<b>26.1</b>
<b>Section 27</b>	
<b>DRS/SAPR2 Installation</b> .....	<b>27.1</b>
Introduction .....	27.1
Installation Steps .....	27.2
Installing the DRS/API Product .....	27.3
Building the DRS/SAPR2 Defaults Module .....	27.4
DRS/SAPR2 Defaults Module .....	27.5
Building the DRS/SAPR2 Printer Alias Table .....	27.13
DRS/SAPR2 Printer Alias Table .....	27.14
Updating the CICS Tables .....	27.15
Copying the DRS/SAPR2 Modules for CICS .....	27.17
Defining the Printer TD Queue .....	27.18
Defining the Printer to SAP R2 .....	27.19
<b>Section 28</b>	
<b>Setting SYSOUT Attributes With DRS/SAPR2</b> .....	<b>28.1</b>
DRS/SAPR2 System Defaults .....	28.2
Specifying Printer and Report SYSOUT Attributes .....	28.3
Printer Defaults .....	28.4
Report SYSOUT Attributes .....	28.5
Establishing User Defaults .....	28.7
Using the Printer Alias Table .....	28.8
Defining DRS/SAPR2 Command Strings .....	28.9
DRS/SAPR2 SYSOUT Parameters .....	28.10
<b>Section 29</b>	
<b>DRS/SAPR2 Problem Determination</b> .....	<b>29.1</b>
DRS/SAPR2 Messages .....	29.1
DRS/API Return Codes .....	29.1
DRS/API SNAP Dumps .....	29.1
Copies of SAPR2 Spool Record Data .....	29.2
SYSUDUMP from ABEND .....	29.3
<b>Section 30</b>	
<b>Introduction to DRS/NATURAL</b> .....	<b>30.1</b>

---

---

<b>Section 31</b>	
<b>DRS/NATURAL Installation.....</b>	<b>31.1</b>
Introduction .....	31.1
Installation Steps .....	31.2
Installing the DRS/API Product.....	31.3
Check Natural Product Level.....	31.4
Linkedit DRS/NATURAL Interface .....	31.5
Updating the CICS Tables .....	31.6
Copying the DRS/NATURAL Modules for CICS .....	31.7
Define DRS/NATURAL Interface to Natural.....	31.8
Define a Print and Hardcopy File Using the DRS/Natural Interface .....	31.9
Installation is Complete.....	31.10
<b>Section 32</b>	
<b>DRS/NATURAL - Controlling SYSOUT Attributes.....</b>	<b>32.1</b>
Natural Print File Attributes.....	32.1
Dynamic Control of Natural Print Attributes .....	32.3
Mapping Natural Attributes to JES SYSOUT Attributes .....	32.4
DRS/Natural Variable Substitution.....	32.7
Controlling Hardcopy Attributes.....	32.8
Building the DRS/NATURAL Output Characteristics Table .....	32.9
Output Characteristics Table Header.....	32.10
Output Characteristics Selection Table .....	32.11
Output Characteristics Table Output Options .....	32.12
Including DRS/Natural in the Natural Nucleus.....	32.20
<b>Section 33</b>	
<b>DRS/NATURAL - Problem Determination .....</b>	<b>33.1</b>
NATURAL messages .....	33.1
DRS/NATURAL Messages.....	33.1
DRS/NATURAL and DRS/API Return Codes.....	33.1
DRS/API SNAP Dumps .....	33.2
DRS/NATURAL Trace data .....	33.2
SYSUDUMP from ABEND .....	33.2
<b>Section 34</b>	
<b>DRS/NATURAL - Messages and Codes.....</b>	<b>34.1</b>
<b>Section 35</b>	
<b>Messages and Codes .....</b>	<b>35.1</b>
DRS/API Messages .....	35.1
DRS/VPI Messages .....	35.20
DRS/VPI Message Format .....	35.20
DRS/VPI Message Text.....	35.20

---

DRS/SAPR2 Messages .....	35.78
DRS/SAPR2 Message Format.....	35.78
DRS/SAPR2 Message Text.....	35.78
EDRAINED Printer Error Messages.....	35.91
DRS/STI Messages .....	35.93
DRS/STI CICS Messages .....	35.98
DRS/API Abend Codes.....	35.99
System Abend Codes.....	35.99
All Other System Abends .....	35.100
DRS/API Abend Codes.....	35.100
DRS/SAPR2 Abend Codes.....	35.101
DRS/VPI Abend Codes.....	35.102
User Abend Codes.....	35.102

**Appendix A**

<b>DRS Documentation.....</b>	<b>A.1</b>
-------------------------------	------------

<b>Index.....</b>	<b>Z.1</b>
-------------------	------------



---

## Summary of Enhancements

The following table contains the fix numbers assigned to major DRS enhancements and/or fixes. Select the link for the fix to read the description.

<b>DRS V1 R3.3 Fixes:</b>	
<a href="#">DRS V1 R3.3.029 (05/06/1999)</a>	<a href="#">page xiv</a>
<a href="#">DRS V1 R3.3.044 (01/04/2002)</a>	<a href="#">page xiv</a>
<a href="#">DRS V1 R3.3.104 (04/14/2005)</a>	<a href="#">page xiv</a>

---

**DRS V1 R3.3.029 (05/06/1999)**

- Add the ability to specify RECFM=(,M) for a printer which receives AFPDS files. DRS/VPI always sets RECFM=(V,A) for AFPDS records.

Module DV33IDRS will be zapped to set the carriage control type to "A" only if the value was blank. If the value is "A" or "M", it will not be modified.

**DRS V1 R3.3.044 (01/04/2002)**

- Enhancement to support the SCS Vertical Channel Select (VCS) control code. Vertical Channel Select moves the print position to a specific line as defined by the SVF function. Vertical channels are defined by the top margin value and the first 11 vertical tab settings that were set with the SVF function. Channel 1 is always set to the TM value, even if no vertical tabs are set. Note that the current horizontal print position is not modified.

The print line formatting control translate table in module DV33VPRT will be zapped to recognize the VCS (X'04') as a formatting control that should be processed.

Module DV33SCS0 will be source changed to process the VCS.

Because this fix includes source code changes, it is not available as a zap. Customers who need this fix should contact LRS to request updated product distribution materials.

**DRS V1 R3.3.104 (04/14/2005)**

- Add support for MGCRE macro interface in place of SVC 34.

Module DVSSUE10 will be source updated to use the MGCRE macro instead of issuing SVC 34 directly.

This fix is not available as a zap. Customers who need this fix should contact LRS to request updated product distribution materials.

---

# Section 1 Overview

The Dynamic Report System, hereafter referred to as DRS, provides an effective means of routing reports from batch jobs, online applications (i.e. CICS, IMS, IDMS, etc.), Local Area Networks, and remote TCP/IP hosts to the JES2 or JES3 spool or to a DASD file on the MVS host system.

The reports are standard SYSOUT or DASD datasets. They can be fixed, variable, or undefined format, blocked or unblocked. They can have ASA, machine, or no carriage control. SYSOUT datasets can have standard JES dataset attributes, such as class, destination, form name, writer name, FCB, UCS, hold/nohold, copies, etc. In addition, AFP attributes (e.g., PAGEDEF, FORMDEF, etc.) can be assigned to the SYSOUT reports by referencing OUTPUT JCL statements. The OUTPUT JCL statements can either be placed in the JCL or created dynamically by DRS.

DRS requires MVS/XA or MVS/ESA. MVS/370 is no longer supported.

DRS supports dates beyond December 31, 1999.

The following sections briefly describe the various components of DRS. Note that DRS/API and DRS/VPI are standard components of the DRS product. DRS/TCPIP, DRS/STI, DRS/SAPR2, DRS/NATURAL, and DRS/PC are additional products that can be purchased.

---

## DRS/API

The DRS Application Programming Interface (DRS/API) allows programs to invoke DRS directly to create reports dynamically. The programs that call the DRS/API may be running in a CICS or non-CICS environment. The reports are available for printing immediately; it is not necessary to shut down the CICS or non-CICS system to obtain the report.

### Storage Estimates

**Below 16M line:** 72K +  
(72 \* #spttasks) +  
(304 \* #actrpts) +  
((8 + BLKSIZE) \* #actrpts)

**Above 16M line:** 47K +  
(2984 \* #spttasks)

**CSA:** DRS/API does not directly allocate any storage from CSA or ECSA.

- **#spttasks** is the number of support subtasks that are currently processing DRS requests. Refer to [“Customizing the DRS System Options” on page 20.9](#) for more information on specifying the number of DRS support subtasks.
- **#actrpts** is the number of active DRS reports. “Active” means “allocated and open”.



---

## DRS/VPI

The DRS Virtual Printer Interface (DRS/VPI) intercepts output from VTAM applications such as CICS, IMS, IDMS, etc. DRS/VPI supports printer session types of LU-0, LU-1, and LU-3. For LU-0 and LU-3, the Data Stream Compatibility (DSC) buffer format is supported. For LU-1, the SNA Character Set (SCS) buffer format and Intelligent Printer Data Stream (IPDS) format are supported.

Print buffers received by DRS/VPI are transformed into print lines containing ASA carriage control. The print files received by DRS/VPI can be placed on the JES spool, in a DASD dataset, or both. Note that up to four separate SYSOUT datasets can be created, each with unique SYSOUT characteristics, for each print file received.

### Storage Estimates

**Below 16M line:** 23K +  
(624 \* #actvprt)

**Above 16M line:** 227K +  
((#maxvtam + #maxtcp + 64) \* 1792) +  
(396 \* #maxvtam) +  
(37K \* #actvprt) +  
(3840 \* #actvtam) +  
(3256 \* #actoref) +  
(4K \* #trpages)

**CSA:** DRS/VPI does not directly allocate any storage from CSA or ECSA.

- Since DRS/VPI invokes DRS/API when creating reports, you must also include the DRS/API storage estimates when calculating the amount of storage required for DRS/VPI.
- **#maxvtam** is the maximum number of VTAM virtual printers. This value is specified in the MAXPRTS keyword in the System Initialization member.
- **#maxtcp** is the maximum number of TCP/IP virtual printers. This value is specified in the MAXPRTS keyword in the System Initialization member.
- **#actvprt** is the number of active VTAM and TCP/IP virtual printers.
- **#actvtam** is the number of active VTAM virtual printers.
- **#actoref** is the number of OUTPUT JCL statements that DRS/VPI has dynamically created.
- **#trpages** is the number of trace table pages. This value is specified in the TRACE keyword in the System Initialization member.

---

## DRS/TCPIP

DRS/TCPIP allows print files to be received from remote TCP/IP hosts using LPR/LPD protocols. DRS/TCPIP executes in the DRS/VPI address space and listens for connections after connecting to the MVS TCP/IP system. As print files are received, they are placed on the JES spool, in a DASD dataset, or both, based on the options specified in the virtual printer definition. By default, DRS/TCPIP listens for connections on port 515, which is the standard 'well-known' port for a Line Printer Daemon.

### Storage Estimates

**Below 16M line:**  $(1024 * \#acttcp) +$   
 $(QBUFSIZE * \#acttcp)$

**Above 16M line:**  $768 +$   
 $(21K * \#ACTTCP)$

**CSA:** DRS/TCPIP does not directly allocate any storage from CSA or ECSA.

- Since DRS/TCPIP executes within the DRS/VPI address space, you must also include the DRS/VPI storage estimates when calculating the amount of storage required for DRS/TCPIP.
- **#acttcp** is the number of active TCP/IP virtual printers.
- **QBUFSIZE** is only used for DRS/TCPIP virtual printers defined with **COMMTYPE=(TCPIP,LPD)**. In addition, this value will only be used if the data file is received before the control file, and the control file must be used to determine how the data file should be processed.

---

## DRS/STI

The DRS Smart Tag Interface (DRS/STI) allows reports destined for the same virtual printer to be routed to the JES spool with different SYSOUT characteristics, including different AFP characteristics. DRS/STI improves the usability of DRS/VPI by dramatically reducing the number of virtual printer definitions. DRS/STI maintains a VSAM database containing information on both where and how the report is to be printed, faxed, e-mailed, etc. Vendor and legacy applications interface to DRS/STI to retrieve the appropriate SYSOUT characteristics prior to the actual sending of the report to the virtual printer.

### Storage Estimates

**Below 16M line:** 512 (system file control block)

**Above 16M line:** 11K

**CSA:** DRS/STI does not directly allocate any storage from CSA or ECSA.

- Since DRS/STI executes within the DRS/VPI address space, you must also include the DRS/VPI storage estimates when calculating the amount of storage required for DRS/STI.

---

# DRS/SAPR2

The DRS/SAPR2 interface is a direct interface from SAP R2 applications to the JES spool. It enables any existing SAP R2 application running in CICS to create a report on the JES spool without any application changes.

## Storage Estimates

**Below 16M line:** None

**Above 16M line:** (9K +  
maxlrecl) \*  
# trans

**CSA:** DRS/SAPR2 does not directly allocate any storage from CSA or ECSA.

- maxlrecl is the value specified as MAXLRECL in the DRS/SAPR2 default module (DPSSSDEF).
- trans is the number of transactions that have been defined to run DRS/SAPR2 concurrently.

All storage is obtained with CICS GETMAIN requests. Additional storage will be required when reports are opened. See [“Storage Estimates” on page 1.2.](#)

---

## **DRS/NATURAL**

DRS/NATURAL provides a direct interface from Natural applications to the JES spool. It enables any existing Natural application running in CICS or BATCH to create a report on the JES spool without any application changes.

The DRS/NATURAL interface provides complete control over the JES spool attributes assigned to a report and enables existing reports to use the printing facilities of Advanced Function Presentation (AFP) to enhance the quality of printed output from Natural applications.

---

## **DRS/PC**

DRS/PC obtains print files from Local Area Network print queues and sends those print files to DRS/VPI. DRS/PC connects to DRS/VPI using a VTAM APPC (LU6.2) connection. Print files received from DRS/PC can be placed on the JES spool or in a DASD dataset. The SYSOUT/DASD characteristics are sent to DRS/VPI from information defined to DRS/PC on the LAN.

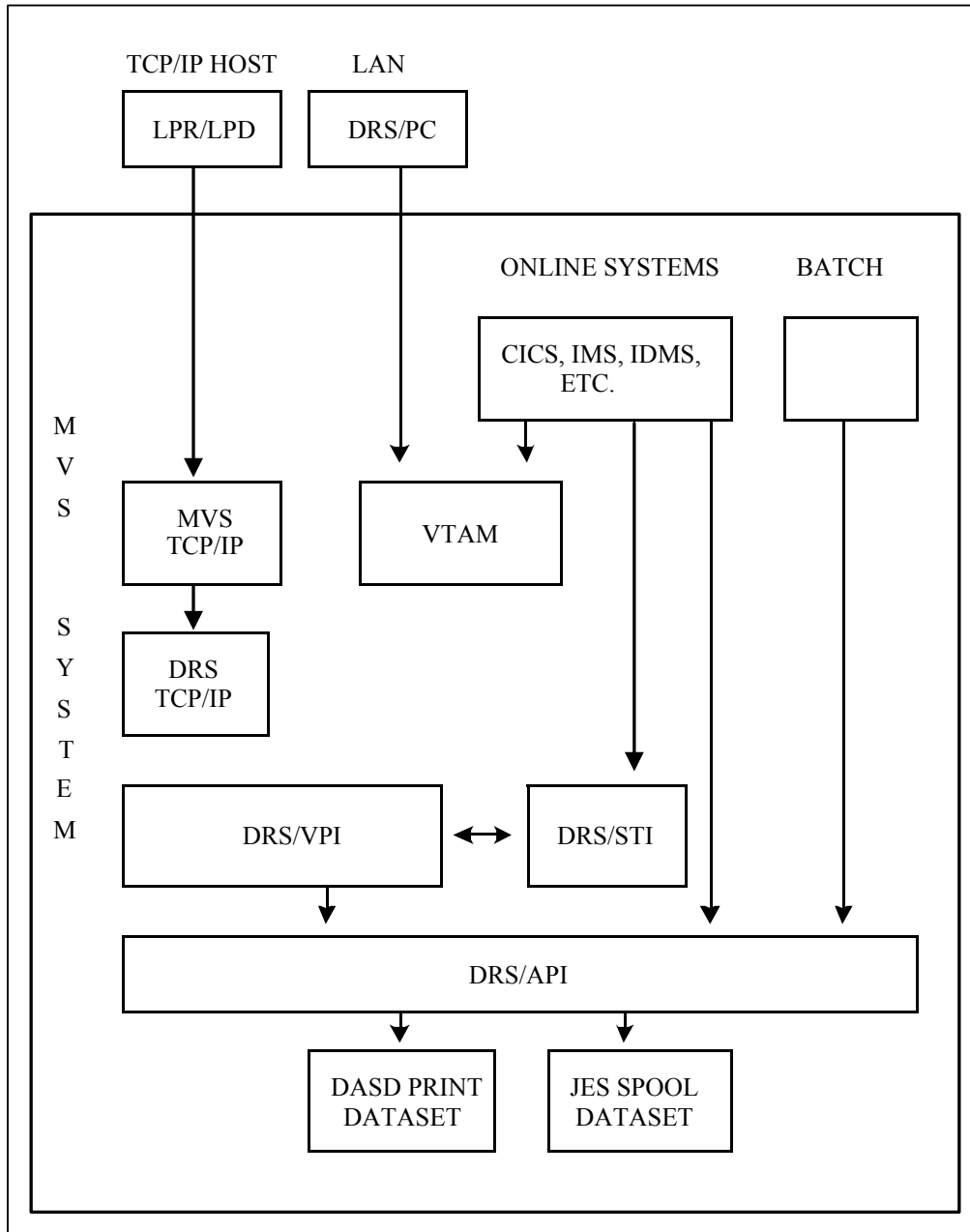
---

## Advantages of using DRS

- Reports can now be routed to any printer in the network (VTAM, TCP/IP, or Local Area Networks) without requiring changes to the application.
- Reports can be received from VTAM applications, TCP/IP remote hosts, and Local Area Networks. Once on the spool, standard printer management software, such as VPS and JES, can produce the hardcopy, within the rules applicable to all other SYSOUT in the installation.
- AFP attributes can be associated with legacy reports without requiring changes to the application.
- Multiple copies of a report can be created to allow both printing and archiving, if required.
- DRS eliminates printing problems (e.g., forms jam, out of paper, power off, etc.) from online applications (e.g., CICS, IMS, IDMS, etc.)
- DRS eliminates the problems associated with sharing printers across multiple applications.
- SYSOUT attributes of a report can be modified at report termination. For example, if an error in processing has occurred while creating the report, it can be deleted, held, or routed to a different destination or class.
- Optional user exits allow customization of DRS processing.

---

## How DRS Fits in Your System





---

## Section 2

# Introduction to DRS/VPI

The DRS Virtual Printer Interface allows receiving output from VTAM online applications or from remote TCP/IP hosts. For more information on using DRS/VPI to receive TCP/IP print files, see [“Introduction to DRS/TCPIP” on page 12.1](#).

When using DRS/VPI to intercept output from VTAM online applications, such as CICS, IMS, IDMS, etc., DRS/VPI supports printer session types of LU-0, LU-1, or LU-3. For LU-0 and LU-3, the Data Stream Compatibility (DSC) buffer format is supported. For LU-1, the SNA Character Set (SCS) buffer format and Intelligent Printer Data Stream (IPDS) format are supported.

The member name of the virtual printer must match the name of an APPL definition that is active to VTAM. For more information on defining the VTAM APPL statement, see [“VTAM Definition Requirements” on page 3.5](#). When DRS/VPI is started, the virtual printer members are read from the DRSVLIB PDS. DRS/VPI will open the ACB that matches the member name, and notify VTAM that it is ready to receive connection requests for that name.

When the online application needs to print, it will start a session with the virtual printer to send the print data. The network name used by the online application must match the member name of the virtual printer. For example, the CICS or IMS printer definition would contain the member name of the virtual printer as the VTAM network name.

As the print data is received by DRS/VPI, the print buffers are transformed into print lines. By default, DRS/VPI creates the print lines with ASA carriage control. Reports may also be created without carriage control, if desired.

The reports created by DRS/VPI are either placed on the JES spool or in a DASD file. If the reports go to the JES spool, DRS/VPI uses attributes from SYSOUT keywords, such as: CLASS, DEST, FORM, WRITER, UCS, FCB, OUTREF, etc. If the report is placed in a DASD file, the attributes of the file use DASD keywords, such as: DSN, MEMBER, DISP, SPACE, UNIT, VOLUME, etc. Up to four SYSOUT files and one DASD file can be created for each print request.

If the report is placed on the JES spool, any system that obtains output from the JES spool can access the report. For example, the report could be printed by JES, VPS or PSF or it could be acquired by a report distribution and archival product. Because OUTPUT statement information can be associated with the print data, PAGEDEF, FORMDEF, CHARS, etc., could be used for special AFP formatting.



---

## Section 3

# DRS/VPI Installation

This section provides instructions for installing the DRS Virtual Printer Interface (DRS/VPI). The details of this installation procedure are listed in [“Installation Procedure” on page 3.4](#). Before you do the actual installation, you may want to read [“New Features in this Release” on page 3.2](#) and [“Migrating to DRS/VPI V1 R3.3 from R3.0, R3.1 or R3.2” on page 3.3](#). These topics will be of interest to both the first-time user of DRS/VPI and to the user of prior DRS/VPI releases.

---

## New Features in this Release

If you have been using DRS/VPI Release 3.0, 3.1, or 3.2 at your installation, you will find Version 1 Release 3.3 will support all of the functions you are accustomed to. This release also gives you these new features.

- **Support for additional OUTPUT keywords**

Support has been added for new OUTPUT statement keywords in the OUTPUT reference member for DRS/VPI. In addition, the DEST value can now be up to 127 bytes which allows specification of the IP address in the destination field, and up to 4 NOTIFY fields may be specified.

- **DBCS translation support**

Translation of DBCS data is allowed for data received using DRS/VPI. Each virtual printer definition may specify the PRTXDBCS keyword to indicate whether DBCS data should be translated, the name of the table to use, and the disposition of the shift-out and shift-in characters.

- **New SUBSYS keyword**

The SUBSYS keyword allows sending print files to a non-JES subsystem.

- **Expanded transparency support**

The TRN keyword now allows specifying that the input data should not be modified at all, but placed into the print file as binary data. If TRN=T is specified in the printer definition, DRS/VPI will take each buffer received and place it, unchanged, into the print file.

- **Additional options for report termination**

The TERMRPT keyword has been modified to allow a table to be specified. The table is generated using the \$DRSTRPT macro and allows using a different TERMRPT option based on the VTAM LU type and the name of the session partner.

- **New JOBNAME printer keyword**

The JOBNAME keyword has been added to allow creating SYSOUT files with a name that is different from the name of the DRS/VPI job or started task. The JOBNAME can be a constant value or it can be set using symbolic parameters that represent dynamic values. Using the JOBNAME keyword requires MVS release MVS/SP 4.1.0 or later and that the DRS modules be loaded from an authorized library.

---

## Migrating to DRS/VPI V1 R3.3 from R3.0, R3.1 or R3.2

If you have been using DRS/VPI Release 3.0, 3.1, or 3.2 at your installation, you will find that Version 1 Release 3.3 will support all of the keywords you are accustomed to.

The text of many DRS/VPI messages has been changed. In some cases, the DRS/VPI message numbers are different than they were for previous releases. You should review any automatic message processing you have established for DRS/VPI messages.

The TERMRPT keyword has been changed to allow a report termination table to be specified. This table allows using a different option to terminate the report based on the session LU type as well as the name of the session partner. If you currently have multiple definitions due to report termination requirements, you may be able to consolidate those printer definitions by using the report termination table.

The default names of the DRS/VPI user exits are DVSSUExx. In DRS R3.2, the names of the exits were the same as they are now. However, in DRS R3.0 and R3.1, the default names were DV30UExx and DV31UExx. If you are using any of the sample exits provided with DRS/VPI, please check your DRS/VPI system initialization member to be certain the correct names are specified.

The DRSSYAT and DRSVSYAT macros have been changed to include the version number for DRS. Previously, the release field was a 4-byte field that contained "R3.2". This release field has been changed to 2 fields -- one is a 1-byte version number (1) and the other is a 3-byte release number (3.3). If you are currently using the 4-byte field in a user exit or separator, you can still use the old program but instead of seeing "R3.2" you will see "13.3". This change should not cause any problems unless you are doing some sort of compare on that value but it is recommended that the user exits be reassembled.

If PRTROPTS=xxx8 was specified for a printer used with DRS/VPI R3.2, the value should be changed to PRTROPTS=xxxxxx1x for this release of DRS/VPI.

---

# Installation Procedure

The steps required to install Version 1 Release 3.3 of DRS/VPI are:

1. **Read the entire installation procedure.**
2. **Restore the DRS distribution libraries** (see [“Restoring the DRS Distribution Libraries”](#) on page 20.6), **including the DRSSKEY module** (see [“Installing the DRSSKEY Module”](#) on page 20.13).
3. **Code VTAM parameters** (see [“VTAM Definition Requirements”](#) on page 3.5). This step is optional if DRS/VPI will only receive print data from TCP/IP hosts.
4. **Code the DRS/VPI Job Control Language (JCL)** (see [“Defining DRS/VPI Job Control Language”](#) on page 3.7).
5. **Assemble and linkedit the DRS System Options module (DRSSOPTS) for DRS/VPI** (see [“DRS System Options Module for DRS/VPI”](#) on page 3.8).
6. **Assemble and linkedit any of the DRS/VPI user exits that you choose to use at your installation** (optional) (see [“DRS/VPI User Exits”](#) on page 9.1).
7. **Build the DRS/VPI System Initialization member** (see [“Building the System Initialization Member”](#) on page 3.13).
8. **Build the DRS/VPI Message Modification Member** (optional) (see [“Message Modification Member”](#) on page 3.10).
9. **Build the DRS/VPI Members for Printer Definition** (see [“Building the Printer Definition Members”](#) on page 3.42).
10. **Build the DRS/VPI Member Inclusion List or Member Exclusion List** (optional) (see [“Printer Activation Inclusion List Member”](#) on page 3.10 and [“Printer Activation Exclusion List Member”](#) on page 3.11).
11. **Build the Output Reference Members** (optional) (see [“Building the Output Reference Members”](#) on page 3.107).
12. **DRS Virtual Printer Interface installation is now complete.**

The following sections provide the information required to perform the above steps.

## VTAM Definition Requirements

VTAM definitions are not required if all DRS/VPI printer definitions will receive print files from TCP/IP hosts using DRS/TCPIP.

If any DRS/VPI printer will receive output from VTAM online systems, the DRS/VPI address space will run as a VTAM application. A major node must be defined and active to VTAM that contains an APPL statement for each printer that will receive data from a VTAM online system. For example:

DRSAPPL	VBUILD	TYPE=APPL	
DRSV001	APPL	ACBNAME=DRSV001,EAS=1,VPACING=63,SESSLIM=YES, DLOGMOD=SCS,MODETAB=ISTINCLM	X
DRSV002	APPL	ACBNAME=DRSV002,EAS=1,VPACING=63,SESSLIM=YES, DLOGMOD=SCS,MODETAB=ISTINCLM	X
DRSV003	APPL	ACBNAME=DRSV003,EAS=1,VPACING=63,SESSLIM=YES, DLOGMOD=SCS,MODETAB=ISTINCLM	X
DRSV004	APPL	ACBNAME=DRSV004,EAS=1,VPACING=63,SESSLIM=YES, DLOGMOD=SCS,MODETAB=ISTINCLM	X
DRSV005	APPL	ACBNAME=DRSV005,EAS=1,VPACING=63,SESSLIM=YES, DLOGMOD=SCS,MODETAB=ISTINCLM	X
DRSV006	APPL	ACBNAME=DRSV006,EAS=1,VPACING=63,SESSLIM=YES, DLOGMOD=SCS,MODETAB=ISTINCLM	X
DRSV007	APPL	ACBNAME=DRSV007,EAS=1,VPACING=63,SESSLIM=YES, DLOGMOD=SCS,MODETAB=ISTINCLM	X
DRSV008	APPL	ACBNAME=DRSV008,EAS=1,VPACING=63,SESSLIM=YES, DLOGMOD=SCS,MODETAB=ISTINCLM	X
DRSV009	APPL	ACBNAME=DRSV009,EAS=1,VPACING=63,SESSLIM=YES, DLOGMOD=SCS,MODETAB=ISTINCLM	X
DRSV010	APPL	ACBNAME=DRSV010,EAS=1,VPACING=6, MODETAB=LRSMODE, DLOGMOD=LRSAPPC, APPC=YES, FOR DRS/PC DMINWNL=1, DMINWNR=0, DSESLIM=1, PARSESS=YES	X X X X X X X

Alternatively, a model APPL statement can be used to define VTAM applications. This process is described in manual SC31-8777 "CS: SNA Network Implementation Guide" in topic "Model Application Program Definitions". For example, to define a group of VTAM applications that have names that begin with DRSV, this APPL statement could be used:

```
DRSV*    APPL  ACBNAME=DRSV*,EAS=1,SESSLIM=YES,VPACING=63,DLOGMOD=SCS, X
          MODETAB=ISTINCLM,MODSRCH=FIRST
```

When DRS/VPI opens an ACB with a name that starts with DRSV, VTAM will activate the ACB and allow the OPEN to continue.

- 
- Note 1:** Specify ACBNAME to be the same as the member name of the virtual printer definition in the DRS/VPI control library (DRSVLIB).
- Note 2:** Specify the EAS to be the estimated number of active sessions for this printer; this value should always be 1 for DRS/VPI printer definitions.
- Note 3:** Specify VPACING to be a non-zero value, or specify non-zero PSNDPAC and SRCVPAC parameters in the session parameters (logon mode table entry) to be used for the virtual printer session.
- Note 4:** SESSLIM=YES will allow VTAM to queue pending sessions for the virtual printer if the virtual printer application is already in active session with another application. In order to use SESSLIM to queue requests and to notify the current application to release the printer (RELREQ function), one or more of the following VTAM APARS may need to be applied: OY29863, OY34258, OY48034, OY44685, OY45603, OY46775, OY48927. Without SESSLIM=YES, manual intervention may be necessary to release and acquire virtual printers from various VTAM applications.
- Note 5:** Specify a default logon mode table entry name (DLOGMOD) and the name of the logon mode table (MODETAB) to use to obtain the printer session parameters, or allow the primary application to specify the session (BIND) parameters in its printer definition.
- Note 6:** The Major Node name (the name of the member where these application definitions are stored) cannot be the same as the names used on the APPL statements.
- Note 7:** For those virtual printer definitions that allow connection to the DRS/PC product, MODETAB must be specified, and the DLOGMOD should be the name of a logon mode table entry that contains a valid set of session parameters for LU6.2 sessions. VPACING should be a non-zero value. In addition, the following must be specified:

APPC=YES,	Indicates APPC device
DMINWNL=1,	Allow DRS to win session contention
DMINWNR=0,	Allow DRS to win session contention
DSESSLIM=1,	Only one session allowed
PARSESS=YES	Must be YES for APPC

For more information, see [“DRS/VPI With DRS/PC” on page 5.1](#).



---

## Defining DRS/VPI Job Control Language

The DRS/VPI is generally run as a started task but can be run as a batch job. The JCL requirements are similar for both methods.

```
//*****  
//*   DRS VIRTUAL PRINTER INTERFACE  
//*****  
//DRSV      EXEC      PGM=DV33DRIV,PARM=DRSSTART,TIME=1440      Note 1  
//STEPLIB   DD        DSN=LRS.DRS.V1R33.LOAD,DISP=SHR          Note 2  
//DRSVLIB   DD        DSN=LRS.DRS.V1R33.CNTL,DISP=SHR          Note 3  
//DRSVLOG   DD        DSN=YOUR.DRSV.LOG,DISP=SHR              Note 4  
//DRSRULES  DD        DSN=DRS.V1R33.RULES,DISP=SHR             Note 5  
//SYSUDUMP  DD        SYSOUT=D
```

**Note 1:** The optional PARM on the execute statement specifies the member name which contains the system initialization parameters for the DRS/VPI. If no PARM is specified, a member name of DRSSTART is assumed.

**Note 2:** A STEPLIB is not required if the DRS/VPI modules are in a linklist library or SYS1.LPALIB. The DRS/VPI and DRS/API modules must be loaded from an authorized library if the JOBNAME= keyword will be used to set the JOB name for SYSOUT datasets.

**Note 3:** The DRSVLIB DD statement must be a partitioned dataset that contains the initialization member, the optional member for message modification, and the virtual printer definition members. The LRECL must be 80, and the BLKSIZE a multiple of 80.

**Note 4:** The DRSVLOG DD statement is optional and should only be used if you want the DRS/VPI log to be written to a preallocated dataset. If the DRSVLOG statement is not present, the log will be dynamically allocated using the initialization options.

If you do specify the DRSVLOG statement, you need not specify DCB attributes. DRS will choose an optimum block size based on the record length and the device type. If you specify invalid DCB attributes, DRS will override them.

**Note 5:** The DRSRULES DD statement is optional, and should only be used if you have enabled the DRS Smart Tag Interface (DRS/STI). If specified, the dataset name should be the name of your DRS/STI rules dataset. (See [“Defining the DRS/STI Rules Dataset”](#) on page 16.2 for more information.)

---

## DRS System Options Module for DRS/VPI

The DRS System Options should specify the number of support subtasks as a non-zero value; for example:

**TCBBTCH=1**

Using the default (TCBBTCH=0) will cause an initialization error with the message:

**DRSV002E DRS OPTIONS MODULE SPECIFIES UNSUPPORTED VALUE TCBBTCH=0**

For more information on setting, assembling, and linking the DRS System Options, see [“Customizing the DRS System Options” on page 20.9](#).

---

## DRS/VPI Control Library Members

The DRS/VPI reads members from the control library as the source for information about the DRS operating environment and the definition of the virtual printers supported. The dataset name of this library is specified on the DRSVLIB DD statement in the DRS/VPI started task JCL (see [“Defining DRS/VPI Job Control Language” on page 3.7](#)). This dataset must have a logical record length of 80 and a blocksize that is a multiple of 80.

This control library can contain seven types of members. They are:

- System Initialization Member
- Message Modification Member (optional)
- Printer Activation Inclusion List Member (optional)
- Printer Activation Exclusion List Member (optional)
- Printer Default Members (optional)
- Printer Definition Members - one for each virtual printer
- Output Reference Members (optional)

### Syntax of DRS/VPI Library Members

All control library members must use the following syntax:

- Control statements must begin between position 2 and 16 inclusive.
- Ending the last parameter on a control statement with a comma signifies continuation.
- As many parameters as desired can be specified on a given statement as long as they do not extend past position 71.
- Any statement containing an asterisk in position 1 is considered a comment statement and is ignored.
- Comments can also be placed on an actual statement by leaving at least one space after the control statement information.

---

## System Initialization Member

The System Initialization Member contains several keyword parameters that apply to the overall DRS/VPI system. The member name is determined by the PARM on the execute statement in the JCL. If no PARM is specified, a member name of DRSSTART is used.

The System Initialization Member is explained further in [“Building the System Initialization Member” on page 3.13](#).

## Message Modification Member

The Message Modification Member contains one or more entries that describe modifications to the “type” of the DRS/VPI message. The valid message modification types are:

- A - Action
- N - Normal
- I - Informational
- L - Enter in DRS/VPI log and SYSLOG, but do not issue WTO
- X - Do not log or issue the WTO (“no-ops” the message)

In this member, a particular message type can be modified by coding:

**MSGnnn=x**

where “nnn” is the DRS/VPI message number and “x” is the type to which the message should be changed from its normal value.

The member name for message modification is determined by the MSMODMEM keyword in the system initialization member.

## Printer Activation Inclusion List Member

The Printer Activation Inclusion List Member is a list of member names in the DRSVLIB library which contain the parameters describing the printers that are to be activated when DRS/VPI system is initialized. The member name of this member is determined by the MLISTMEM keyword in the System Initialization Member.

This member is optional and mutually exclusive with the Printer Activation Exclusion List Member. If neither an Inclusion List Member nor an Exclusion List Member is specified, no printers will be automatically activated in the DRS/VPI system at initialization. Printers can, however, be brought into the DRS/VPI System after initialization by issuing the DRS/VPI ACTIVATE command.

---

## Printer Activation Exclusion List Member

The Printer Activation Exclusion List Member is a list of member names in the DRSVLIB library which do not contain parameters describing DRS virtual printers to be activated at DRS/VPI initialization. The member name is determined by the XLISTMEM keyword in the System Initialization Member.

This member is optional and mutually exclusive with the Printer Activation Inclusion List Member. If neither an Exclusion List Member nor an Inclusion List Member is specified, no printers will be automatically active in the DRS/VPI system at initialization. Printers can, however, be brought into the DRS/VPI system after initialization by issuing the DRS/VPI ACTIVATE command.

If a Printer Activation Exclusion List Member is specified, it should include not only the member names of printers which are not to be automatically activated at DRS/VPI system initialization, but also the member names of any members in the DRSVLIB library which are not DRS virtual printer definitions. The DRS/VPI will automatically exclude the Message Modification Member (indicated by the MSMODMEM keyword in the System Initialization Member), the Printer Activation Exclusion List Member (indicated by the XLISTMEM keyword in the System Initialization Member), and the System Initialization Member itself (indicated by the PARM on the EXEC statement in the DRS/VPI job control language).

Users with a very large number of DRS virtual printers may find it less time consuming to specify a Printer Activation Exclusion List Member than to specify a Printer Activation Inclusion List Member which must be updated each time a new printer is added to DRS/VPI.

**Note:** The Exclusion List Member will not work correctly if you concatenate more than one library on the DRSVLIB JCL statement. To use an Exclusion List Member, all DRS/VPI members must exist in a single library.

## Printer Default Member

A Printer Default Member allows the DRS/VPI user to establish default values for individual printer keyword parameters. This can help limit the number of printer keyword parameters which must be specified in the Printer Definition Members, because DRS/VPI can pick up the parameter values from the appropriate Default Member if a keyword is not specified in the Printer Definition Member.

The member name of this member is determined by the DEFLTMEM keyword in either the Printer Definition Member or in the System Initialization Member. The member name specified in the DEFLTMEM keyword in the System Initialization Member can be considered the “master” default member.

DRS/VPI assigns a value to each printer keyword parameter for a printer from one of four sources. The first of these sources to specify a value for a keyword is the one that is used. The four sources and the order in which they are search are:

1. The Printer Definition Member for the printer
2. The member specified in the DEFLTMEM keyword in the Printer Definition Member (if any)
3. The member specified in the DEFLTMEM keyword in the System Initialization member (if any)
4. DRS/VPI default value for the keyword parameter.

---

A DRS/VPI user may wish to set up a default member for LU-0 printers, a default member for LU-1 printers, and a default member for LU-3 printers. Each of these members could specify printer keyword parameter values unique to those LU types, and specifying the DEFLTMEM keyword in the individual Printer Definition Members to point to the appropriate default would eliminate the need to specify the parameters in each Printer Definition Member.

It might also be desirable to set up a default member for a particular type of printer, such as a TCP/IP printer, and define those keywords that must have special values for this type of device.

## Printer Definition Members

The Printer Definition Member is used to define the attributes and options associated with the virtual printer being defined. A Printer Definition Member must exist for each virtual printer that is used by DRS/VPI.

For VTAM virtual printers, the member names of these members must be the same as the ACBNAME on the associated VTAM application definition statement (see [“VTAM Definition Requirements” on page 3.5](#)).

For TCP/IP virtual printers, the member name will be the print queue name specified when the print command is issued at the remote host.

## Output Reference Members

An Output Reference Member contains information that would normally be found on an OUTPUT JCL statement. The member name of the Output Reference Member can be specified on the OUTREF keyword in the Printer Definition Member.

Output Reference Members are optional. The keywords provide values that can be used to dynamically create an OUTPUT statement reference. The creation of a dynamic OUTPUT statement is supported only in an ESA environment.

---

## Building the System Initialization Member

### Overview

The function of the DRS/VPI system initialization member is to provide all parameters that relate to the overall system.

The system initialization member must reside in the DRSVLIB PDS that is defined in the DRS/VPI job control language. The member name must be “DRSSTART” unless specified otherwise by the PARM on the DRS Virtual Printer Interface execute statement.

The various DRS/VPI keyword parameter values are specified by coding the keyword followed by an EQUAL(=) followed by the keyword value. For example:

**LOG=Y**

The format of DRS/VPI control statements is defined in [“Syntax of DRS/VPI Library Members” on page 3.9.](#)

Those initialization parameters that have special relationships are listed below, followed by a description of all initialization parameters listed alphabetically.

## System Initialization Parameters

### Product Key Parameters

The DRS product and each of its optional extension products require that a unique key value be specified. The DRS key is located in the DRSSKEY module and is described in [“Installing the DRSSKEY Module” on page 20.13.](#) The optional extension products that can be loaded into the DRS/VPI address space require keys to be specified in the System Initialization member. This encrypted key value supplies information to the product concerning your license status. Without the correct value in the appropriate keyword, the product will not function.

The values for these keys are unique to each installation, and are supplied in file # 1 on the product distribution cartridge.

The product key parameters are:

**KEYDRSPC**

**KEYLPD**

**KEYSTI**

---

## Logging Parameters

The DRS/VPI can log all messages and operator commands. The following system initialization keywords are related to the logging function.

**LOG**

**LOGHOLD**

**LOGOUTP**

The log dataset can be a dynamically allocated SYSOUT dataset that can be “spun off” (made immediately available for printing) by issuing a “CLOSELOG” command. A new log SYSOUT dataset will then automatically be allocated for continued logging of DRS/VPI commands and messages.

Alternatively, the log dataset can be a preallocated DASD dataset. In this case, a “CLOSELOG” command is not considered valid. However, when the dataset is full, the DRS/VPI will close it, reopen it, and begin recording at the beginning of the dataset. A preallocated DASD dataset is specified by including a DD statement with a DDNAME of DRSVLOG in the DRS/VPI JCL. The keywords LOGOUTP and LOGHOLD do not apply to a preallocated DASD dataset.

## Snap Dump Parameters

DRS/VPI provides a snap dump facility that is used for DRS/VPI debugging. Snap dumps can be invoked by issuing the “SNAP” command, or automatically after an error occurs. The snap dataset is a dynamically allocated SYSOUT dataset that will be automatically “spun off” (made immediately available for printing) after completion of the “SNAP” command. The following parameters are applicable to the DRS/VPI snap dump facility. Snap dumps can also be automatically generated by the DRS/VPI (see the SNAP printer definition parameter).

**SNAPHOLD**

**SNAPOUTP**

## Message Processing Parameters

The following parameter is used to determine how the DRS Virtual Printer Interface will issue console messages:

**WTO**

## User Exit Parameters

At initialization, the DRS/VPI will look for user exit members of the name DVSSUEnn. If a member is found and is not overridden by the EXITnn keyword, the DRS/VPI will use the member of the name DVSSUEnn, set the status of the member to disabled and set recovery on.

If a different name, enabled/disabled status or recovery capability is required, the following keyword will be required:

**EXITnn**



---

## **TCP/IP Parameters**

For an explanation of the DRS/TCPIP product, see [“Introduction to DRS/TCPIP”](#) on page 12.1.

The TCP/IP parameters are:

**TCPIPID**  
**TCPPORT**  
**TCPTYPE**

## **Miscellaneous Parameters**

The following parameters specify information pertaining to various aspects of DRS/VPI operation:

**AUTOACT**  
**DEFLTMEM**  
**DRSDESC**  
**MAXPRTS**  
**MLISTMEM**  
**MSMODMEM**  
**SWAPABLE**  
**TRACE**  
**USEROPTS**  
**XLISTMEM**

Each of the DRS/VPI System Initialization Parameters is described on the following pages, where the parameters are listed in alphabetical order.

---

## AUTOACT

### in SYSTEM INITIALIZATION MEMBER

**AUTOACT=** Specifies, via two positional parameters, the action to be taken concerning automatic activation and definition for a virtual printer.

**Positional parameter 1 specifies whether DRS/VPI should automatically activate a printer when a connection is requested and the printer was not already active.**

This can only occur for a TCP/IP printer, because a VTAM connection can only be made if the printer was activated and the ACB that represents that printer was open and ready to receive connections. If the first parameter is set to “Y”, DRS/VPI will attempt to activate the printer which was requested on the “RECEIVE PRINT JOB” command received from the remote TCP/IP host.

**Valid Values:** Y or N

**Default:** N

**Positional parameter 2 specifies whether DRS/VPI should automatically define a printer when activation is attempted, but there is no member name in the DRS/VPI control library.**

DRS/VPI will use the values specified in the default member from DRS/VPI System Initialization to create a new virtual printer definition. Any keyword values specified in that default member as “\*” would be replaced by the printer name. Automatic definition can be performed for VTAM virtual printers which are named in the Printer Activation Inclusion List (MLISTMEM) or for TCP/IP virtual printers when a “RECEIVE PRINT JOB” command is received for a print queue name which is not found in the DRS/VPI control library.

**Valid Values:** Y or N

**Default:** N

**Example:** AUTOACT=(Y,Y)

This specification indicates that DRS/VPI should automatically activate and define any printer which is listed in the Printer Inclusion list or any TCP/IP printer which is not already activated when a printer request is received.

---

## DEFLTMEM in SYSTEM INITIALIZATION MEMBER

**DEFLTMEM=** Specifies a member within the DRS/VPI control library that describes user defined defaults. If this parameter is not specified, DRS/VPI will use the default keyword values for the printer.

**Valid Values:** Member name of a member that exists in the DRSVLIB PDS.

**Default:** None

**Example:** DEFLTMEM=DRSDEFLT

This specification signifies to DRS/VPI that member DRSDEFLT in the DRSVLIB PDS contains default values for the virtual printer keywords.

---

## **DRSDESC**

### **in SYSTEM INITIALIZATION MEMBER**

**DRSDESC=** Contains a short text description of the DRS/VPI started task for identification purposes.

**Valid Values:** Up to 48 characters of text.

**Default:** None

**Example:** DRSDESC='TEST DRS/VPI SYSTEM'

This specifies that this DRS/VPI will be identified as "TEST DRS/VPI SYSTEM" in response to the DISPLAY, SYSTEM command.

---

## EXITnn in SYSTEM INITIALIZATION MEMBER

**EXITnn=** Specifies the status, recovery capability, and member name of the load module to be used for user exit “nn”, where “nn” is a value from 01 - 10.

**Format:** (status,recovery,membernm)

**Valid Values:** Status

- ENA for enabled status
- DIS for disabled status

recovery:

- RECOV for recovery on
- NORECOV for recovery off

membernm:

- Name of a member which exists in the DRS/VPI load module library

**Default:** (DIS,RECOV,DVSSUEnn)

**Example:** EXIT01=(ENA,RECOV,MYEXIT01)

This specification would indicate that the module to be used for user exit 01 should be enabled at startup, that recovery should be on for this module, and that the member name of the module to be used is MYEXIT01.

---

## KEYDRSPC

### in SYSTEM INITIALIZATION MEMBER

**KEYDRSPC=** Specifies the trial/license code for the DRS/PC product. This keyword must be present in order to use the DRS/PC product.

**Valid Values:** 24 characters which represent 12 hex bytes of trial or license key generated by LRS. The key is supplied in file # 1 of the product distribution cartridge.

**Default:** None

**Example:** KEYDRSPC=112233445566778899AABBCC

---

## KEYLPD in SYSTEM INITIALIZATION MEMBER

**KEYLPD=** Specifies the trial/license code for the DRS/TCPIP product. This keyword must be present in order to use the DRS/TCPIP product.

**Valid Values:** 24 characters which represent 12 hex bytes of trial or license key generated by LRS. The key is supplied in file # 1 of the product distribution cartridge.

**Default:** None

**Example:** KEYLPD=112233445566778899AABBCC

---

## KEYSTI

### in SYSTEM INITIALIZATION MEMBER

**KEYSTI=** Specifies the trial/license code for the DRS/STI (Smart Tag Interface) product. This keyword must be present in order to use the DRS/STI product.

**Valid Values:** 24 characters which represent 12 hex bytes of trial or license key generated by LRS. The key is supplied in file # 1 of the product distribution cartridge.

**Default:** None

**Example:** KEYSTI=112233445566778899AABBCC



---

## LOG in SYSTEM INITIALIZATION MEMBER

**LOG=** Specifies whether the DRS/VPI logging facility is to be active.

**Valid Values:** Y or N

**Default:** Y

**Example:** LOG=Y

This specification would signify to the DRS/VPI that command and message logging should be active.

---

## LOGHOLD

### in SYSTEM INITIALIZATION MEMBER

**LOGHOLD=** Specifies whether the log SYSOUT dataset should be allocated as a held SYSOUT dataset.

**Valid Values:** Y or N

**Default:** N

**Example:** LOGHOLD=Y

This specification would signify to the DRS/VPI that the LOG SYSOUT dataset should be allocated as held.

---

## LOGOUTP in SYSTEM INITIALIZATION MEMBER

**LOGOUTP=** Specifies, via five positional parameters, characteristics to be used when allocating the DRS/VPI log SYSOUT dataset.

**Positional parameter 1 specifies the SYSOUT class to be used for the DRS/VPI log SYSOUT dataset.**

**Valid Values:** Any valid SYSOUT class (0 - 9 or A - Z)

**Default:** A

**Positional parameter 2 specifies the destination to be used for the DRS/VPI log SYSOUT dataset.**

**Valid Values:** Any valid JES destination

**Default:** LOCAL

**Positional parameter 3 specifies the form name to be used for the DRS/VPI log SYSOUT dataset.**

**Valid Values:** Any 1 - 4 character alphanumeric form name

**Default:** None

**Positional parameter 4 specifies the writer to be used for the DRS/VPI log SYSOUT dataset.**

**Valid Values:** Any 1 - 8 character alphanumeric writer name

**Default:** None

**Positional parameter 5 specifies the name of an OUTPUT JCL statement in the DRS/VPI started task JCL to be used to determine the characteristics of the DRS/VPI log dataset.**

**Valid Values:** The name of an OUTPUT JCL statement that exists in the DRS/VPI started task JCL

**Default:** None

**Note:** If you specify the name of an OUTPUT JCL statement that does not exist in the DRS started task JCL, allocation of the log dataset will fail, and DRS/VPI will operate without logging.

**Example 1:** LOGOUTP=(B,U100,STD)

This specification would signify to the DRS/VPI that the log SYSOUT dataset should have a SYSOUT class of B, a destination of U100, and a form name of STD.

**Example 2:** LOGOUTP=(,,,LOGOUT)

This specification would signify that the log SYSOUT dataset should have the characteristics specified in the OUTPUT JCL statement named LOGOUT in the DRS/VPI JCL. Destination will default to LOCAL and class will default to A.

---

## MAXPRTS

### in SYSTEM INITIALIZATION MEMBER

**MAXPRTS=** Specifies, via two positional parameters, the maximum number of virtual printers that DRS/VPI can have active at one time.

**Positional parameter 1 specifies the maximum number of VTAM virtual printer definitions that can be activated.**

**Valid Values:** 1 - 32767

**Default:** 100

**Positional parameter 2 specifies the maximum number of TCP/IP virtual printer definitions that can be activated.**

**Valid Values:** 1 - 32767

**Default:** 100

**Example:** MAXPRTS=(0,999)

This specification would signify to DRS/VPI that the maximum number of TCP/IP virtual printers that could be concurrently active is 999, but that no VTAM virtual printers could be activated.

---

## MLISTMEM

### in SYSTEM INITIALIZATION MEMBER

**MLISTMEM=** Specifies a member within the DRSVLIB PDS that defines the member names of those DRS virtual printers that are to be automatically activated in the DRS/VPI system at start-up.

**Valid Values:** Member name of a member that exists in the DRSVLIB PDS

**Default:** None

**Example:** MLISTMEM=DRSMLIST

This specification would signify to DRS that member DRSMLIST in the DRSVLIB PDS contains a list of members to be automatically activated at DRS/VPI start-up.

---

## MSMODMEM

### in SYSTEM INITIALIZATION MEMBER

**MSMODMEM=** Specifies a member within the DRSVLIB PDS which contains a list of DRS/VPI messages which are to have their type modified.

**Valid Values:** Member name of a member that exists in the DRSVLIB PDS.

**Default:** None

**Example:** MSMODMEM=VMODMSGS

This specification would indicate to the DRS/VPI that a member named VMODMSGS exists in the DRSVLIB PDS, and that this member contains information to be used to modify the types on some DRS/VPI messages.

**Note:** The message modification member will **only** affect DRS/VPI messages (DRSVxxxx messages). DRS/API messages (DRSxxx messages) will **not** be controlled by this member. When DRS/VPI is processing output, it invokes DRS/API via calls. But the DRS/API messages are outside the control of the DRS/VPI processing. To control the use of the DRS/API messages, you can use DRS/API user exit 05 (DRSSUE05). For an explanation of that see [“Exit 05: DRS/VPI Command Exit” on page 9.13.](#)

DRS/VPI messages are written to the DRSVLOG DD, or to a dynamically allocated SYSOUT dataset based on the LOGOUTP keyword in the System Initialization Member. DRS/API messages are written to the DDNAME specified on the LOGDD keyword specified in the DRSSOPTS module. The default for LOGDD is DRSLOG. Refer to [page 20.9](#) for more information about coding the DRSSOPTS module.

---

## SNAPHOLD in SYSTEM INITIALIZATION MEMBER

**SNAPHOLD=** Specifies whether the SNAP SYSOUT dataset should be allocated as a held SYSOUT dataset.

**Valid Values:** Y or N

**Default:** N

**Example:** SNAPHOLD=Y

This specification would signify to the DRS/VPI that the SNAP SYSOUT dataset should be held.

---

## SNAPOUTP

### in SYSTEM INITIALIZATION MEMBER

**SNAPOUTP=** Specifies, via five positional parameters, the characteristics to be used when allocating the DRS Virtual Printer Interface snap dump dataset.

**Positional parameter 1 specifies the SYSOUT class to be used for the DRS/VPI snap dump dataset.**

**Valid Values:** Any valid SYSOUT class (0 - 9 or A - Z)

**Default:** A

**Positional parameter 2 specifies the destination to be used for the DRS/VPI snap dump dataset.**

**Valid Values:** Any valid JES destination

**Default:** LOCAL

**Positional parameter 3 specifies the form name to be used for the DRS/VPI snap dump dataset.**

**Valid Values:** Any 1 - 4 character alphanumeric form name

**Default:** None.

**Positional parameter 4 specifies the writer to be used for the DRS/VPI snap dump dataset.**

**Valid Values:** Any 1 - 8 character alphanumeric writer name

**Default:** None.

**Positional parameter 5 specifies the name of an OUTPUT JCL statement in the DRS/VPI started task JCL to be used to determine the characteristics of the snap dump dataset.**

**Valid Values:** The name of an OUTPUT JCL statement that exists in the DRS/VPI started task JCL

**Default:** None

**Note:** If you specify the name of an OUTPUT JCL statement that does not exist in the DRS/VPI started task JCL, allocation of the snap dataset will fail.

**Example 1:** SNAPOUTP=(B,U100,STD)

This specification would signify to the DRS/VPI that the snap dump dataset should have a SYSOUT class of B, a destination of U100, and a form name of STD.

**Example 2:** SNAPOUTP=(,,,SNAPOUT)

This specification would signify that the snap dump dataset should have the characteristics specified in the OUTPUT JCL statement named SNAPOUT in the DRS/VPI JCL. Destination will default to LOCAL and class will default to A.



---

## SWAPABLE in SYSTEM INITIALIZATION MEMBER

**SWAPABLE=** Specifies whether the DRS/VPI address space should run swappable. In order to specify that DRS/VPI should run non-swappable (SWAPABLE=N), the DRS/VPI modules must be loaded from an authorized library.

**Valid Values:** Y or N

**Default:** Y

**Example:** SWAPABLE=N

This specification would signify that the DRS/VPI address space should run non-swappable.

---

## TCPIPID

### in SYSTEM INITIALIZATION MEMBER

**TCPIPID=** Identifies the address space where MVS TCP/IP is running. This name is used when DRS/VPI connects to the MVS TCP/IP address space.

**Valid Values:** 1 to 8 character name of the MVS TCP/IP system

**Default:** None

**Example:** TCPIPID=TCPIP

This parameter is required if DRS/TCPIP is installed. For IBM's MVS TCP/IP, this value should match the name of the started task or JOB name. For the MVS TCP/IP product from Interlink Computer Sciences, this parameter should match the four character subsystem ID. The default subsystem ID for ICS is ACSS.

---

## TCPPOINT

### in SYSTEM INITIALIZATION MEMBER

**TCPPOINT=** Specifies the number of the port to be used by DRS/TCPIP when it listens for connections from remote TCP/IP hosts. This value should be specified as 515 if normal LPR/LPD protocols are being used.

**Valid Values:** 1 to 65535

**Default:** 515

**Example:** TCPPOINT=516

This specification would indicate to DRS/VPI to connect to TCP/IP and listen for connects using port 516.

This keyword is used only if DRS/TCPIP is installed, and normally should be set to the default of 515.

---

## TCPTYPE

### in SYSTEM INITIALIZATION MEMBER

**TCPTYPE=** Specifies the type of application programming interface to be used by DRS/TCPIP to communicate with the MVS TCP/IP product.

**Valid Values:** IBM221, IBM310, IBM 320 or ICS, where:

IBM221 = IBM IUCV API; V2R2.1 or later

IBM310 = IBM Macro API; V3R1 or later

IBM320 = IBM HPNS; V3R2 or later.

ICS = Interlink API; V1.1 or later

**Default:** IBM221

**Example:** TCPTYPE=ICS

This specification indicates to DRS/VPI to connect to the MVS TCP/IP system using the Interlink Computer Sciences API.

**Note:** This keyword is used only if DRS/TCPIP is installed.

---

## TRACE in SYSTEM INITIALIZATION MEMBER

**TRACE=** Specifies, via three positional parameter values, the type of tracing to be in effect for DRS/VPI activities.

**Positional parameter 1 specifies which DRS/VPI activities are to be traced, in addition to those that are always traced. This parameter is specified as a two digit number, representing one hex byte. The trace type bit specifications are as follows:**

- 80** - Dispatcher/log task interrupts
- 40** - Unassigned
- 20** - Unassigned
- 10** - User exits (system related)
- 08** - Unassigned
- 04** - Unassigned
- 02** - Storage Management
- 01** - GTF tracing active

**Valid Values:** 00 - FF

**Default:** 00

**Positional parameter 2 specifies the size of the DRS/VPI trace table, in 4K pages.**

**Valid Values:** 1 - 99

**Default:** 3

**Positional parameter 3 specifies the GTF format appendage ID to be used in DRS/VPI entries in the GTF trace file, when GTF tracing is on.**

**Valid Values:** 0 - 80

**Default:** 78

**Example:** TRACE=(03,15,79)

This specification would signify to the DRS/VPI that storage management interrupts should be traced, that GTF tracing is active, that a 60K trace table should be allocated and that any DRS/VPI entries in the GTF trace file should have a format appendage ID number of 79.

---

## USEROPTS

### in SYSTEM INITIALIZATION MEMBER

**USEROPTS=** A one byte hex representation that specifies special user options to be used on a DRS/VPI system-wide basis. The user option bit specifications are as follows:

- 80** - Reject TCP/IP connection if printer is busy
- 40** - Ignore all SEND QUEUE STATE LPD commands
- 20** - Unassigned
- 10** - Unassigned
- 08** - Unassigned
- 04** - Unassigned
- 02** - Unassigned
- 01** - Unassigned

**Valid Values:** 00 - FF

**Default:** 00

**Example:** USEROPTS=80

This specification indicates to DRS/VPI that the connection attempt for a TCP/IP printer definition will be rejected if the printer is already connected. Normally, DRS/VPI would wait until the first connection is terminated to process the waiting connection.

---

## WTO

### in SYSTEM INITIALIZATION MEMBER

**WTO=** Specifies, via four positional parameter values, how the DRS/VPI is to issue console messages.

**Positional parameter 1 specifies whether or not the DRS/VPI console messages should be prefixed with the DRS/VPI started task name.**

**Valid Values:** Y or N

**Default:** Y

**Positional parameter 2 specifies whether or not the DRS/VPI action messages should be issued with a Descriptor Code of 2 (MVS action). If messages are issued with a descriptor code of 2, they will be deleted when the required action has been completed.**

**Valid Values:** Y or N

**Default:** Y

**Positional parameter 3 specifies whether or not the DRS/VPI informational messages (those with a suffix of "I") should be issued.**

**Valid Values:** Y or N

**Default:** Y

**Positional parameter 4 specifies the route code that the DRS/VPI should use when issuing unsolicited WTOs. This parameter does not apply to WTOs that are issued in response to a DRS/VPI command, since those WTOs will be issued to the console from which the command was issued.**

**Valid Values:** 1 - 16

**Default:** 1

**Example:** WTO=(Y,Y,N,5)

This specification would signify that DRS/VPI messages should be prefixed with the DRS/VPI started task name, that the DRS/VPI action messages should be issued with a Descriptor Code of 2, that no DRS/VPI informational messages are to be issued, and that any unsolicited WTOs are to use a route code of 5.

---

## **XLISTMEM**

### **in SYSTEM INITIALIZATION MEMBER**

**XLISTMEM=** Specifies a member within the DRSVLIB PDS that defines the member names of those DRS virtual printers that are not to be automatically activated in the DRS/VPI system at start-up.

**Valid Values:** Member name of a member that exists in the DRSVLIB PDS

**Default:** None

**Example:** XLISTMEM=DRSXLIST

This specification would signify to DRS that member DRSXLIST in the DRSVLIB PDS contains a list of members which are not to be automatically started by DRS at DRS/VPI start-up.



---

## Building the Printer Activation Inclusion List

The DRS/VPI Printer Activation Inclusion List specifies the member names of the DRSVLIB members defining printers that should automatically be activated when the DRS/VPI system is initialized. This member is optional, but if neither it nor the Printer Activation Exclusion List is specified, no virtual printers will be automatically activated when the DRS/VPI system is initialized. Printers can, however, be activated after the DRS/VPI system has been initialized by issuing the DRS/VPI ACTIVATE command.

The name of this member, if used, must be specified in the DRS/VPI initialization member by the MLISTMEM keyword.

Sample DRS/VPI Printer Activation Inclusion List:

```
* - - - - - *
*
*   DRSMLIST - DRS/VPI PRINTER ACTIVATION INCLUSION LIST *
*
* - - - - - *
*
*   DRSPRT01 ,
*   DRSPRT02 ,
*   DRSPRT03
```

---

## Building the Printer Activation Exclusion List

The DRS/VPI Printer Activation Exclusion list specifies the member names of the DRSVLIB members which should not automatically be activated when the DRS/VPI system is initialized. This member is optional, but if neither it nor the Printer Activation Inclusion List is specified, no printers will be automatically activated when the DRS/VPI system is initialized. Printers can, however, be activated after the DRS/VPI system has been initialized by using the DRS/VPI ACTIVATE command.

The DRS/VPI will automatically bypass the DRS/VPI System Initialization member (member name DRSSTART unless otherwise specified in the PARM on the DRS/VPI started task JCL), the XLISTMEM member itself, and the member named in the MSMODMEM parameter in the DRS/VPI System Initialization member.

The name of this member, if used, must be specified in the DRS/VPI system initialization member by the XLISTMEM keyword.

Note: The Exclusion List Member will not work correctly if you concatenate more than one library on the DRSVLIB JCL statement. To use an Exclusion List Member, all DRS/VPI-related members must exist in a single library.

Sample DRS/VPI Printer Activation Exclusion List:

```
* - - - - - *
*
* DRSXLIST - DRS/VPI PRINTER ACTIVATION EXCLUSION LIST *
*
* - - - - - *
*
*   ACTIVATE ALL DRS VIRTUAL PRINTER MEMBERS EXCEPT:
*
*   DRSTEST1,
*   DRSTEST2,
*   APGMNAME
```

---

## Building the Printer Default Member

This member is specified in the same format and has available the same keywords as the printer definition member. The purpose of this member is to establish printer default values different from the standard DRS/VPI default values. See [“Building the Printer Definition Members” on page 3.42](#) for a description of all of the DRS/VPI printer keywords.

The name of this member, if used, can be specified in the DEFLTMEM keyword in the DRSSTART System Initialization member and/or by the DEFLTMEM keyword in a Printer Definition member. See [“Printer Default Member” on page 3.11](#) for an explanation of how DRS/VPI determines keyword values for a printer.

Sample DRS/VPI Printer Default Member

```
* -----*
*
* DRSDEFLT          DRS/VPI DEFAULT MEMBER
*
* -----*
CLASS=A,           SYSOUT CLASS=A
DEST=*,           DESTINATION = MEMBER NAME
OUTREF=OUTJCL     OUTPUT JCL REFERENCE
```

---

# Building the Printer Definition Members

## Overview

One DRSVLIB member must be defined for each printer that is to be supported by the DRS/VPI. These members define the characteristics and options applicable to each virtual printer within the DRS Virtual Printer Interface system.

If AUTOACT=(,Y) is specified in the DRS/VPI initialization member, and the MLISTMEM member in the DRS/VPI control library lists the name of every VTAM virtual printer, the DRS control library does not need to include a member for each VTAM virtual printer. When DRS/VPI is started, it will use the master default printer definition in the DRS/VPI control library as a model to dynamically create a virtual printer control block in the DRS/VPI address space for each name in the MLISTMEM member. If the master default printer definition contains a keyword value such as DEST=\* or WRITER=\*, so that the virtual printer name is substituted for the value of the keyword, each virtual printer will create sysout datasets with a unique value for DEST or WRITER.

The various DRS/VPI keyword parameter values are specified by coding the keyword followed by an equal sign (=) followed by the keyword value. For example:

**DEST=U1**

The printer definition keywords are divided into these categories:

## Spooling Attributes

CLASS  
DDNAME  
DEST  
FCB  
FORM  
HOLD  
JOBNAME  
NODE  
OUTREF  
UCS  
WRITER

## Record Format Attributes

BLKSIZE  
LRECL  
RECFM

## DASD Attributes

AVGREC  
DATACLAS  
DISP  
DSN  
DSNTYPE  
EXPDT  
MEMBER  
MGMTCLAS  
RETPD  
SPACE  
STORCLAS  
UNIT  
VOLCNT  
VOLUME

---

## **TCP/IP Parameters**

QBUFSIZE  
QSPACE  
QUNIT  
QVOLUME  
TCPLIMIT  
TCPMRD  
TCPOPTS

## **Miscellaneous Parameters**

COMMTYPE  
DEFLTMEM  
DRAINED  
ERRACTN  
IPDCLASS  
IPDSOPTS  
MAXLPG  
MPP  
PRTROPTS  
PRTXDBCS  
PRTXLATE  
SEPAR  
SNAP  
SUBSYS  
TAB  
TERMRPT  
TRACE  
TRN  
UDATA

---

## Minimum DRS/VPI Printer Definition Requirements

As this section shows, there are many parameters available to customize the definition of a printer to the DRS/VPI. However, most printers will perform satisfactorily if you define the printer to the DRS Virtual Printer Interface with only the spooling attributes (for example, class and/or destination). The VTAM name of the virtual printer is the same as the member name of the virtual printer in the DRS/VPI control library (DRSVLIB).

Most virtual printer keyword parameters have default values which will give satisfactory results for a wide range of devices.

The virtual printer keyword parameters are listed in alphabetical order, starting on the next page.

---

## AVGREC in PRINTER MEMBER

**AVGREC=** Specifies, via two positional parameters, how space should be allocated for a DASD file if the first positional parameter of the SPACE keyword specifies allocation by record. AVGREC is ignored if allocation is by cylinders, tracks, or blocks.

**Positional parameter 1 specifies the average record size of records to be added to this file.**

**Valid Values:** 1 - 32760

**Default:** LRECL value

**Positional parameter 2 specifies which multiplier should be used for the primary and secondary allocation values when space is allocated.**

**Valid Values:** U, K or M, where:

U specifies a multiplier of 1

K specifies a multiplier of 1024

M specifies a multiplier of 1048576

**Default:** U

**Example:** AVGREC=(2048,K)

This specification indicates that the average record size is 2048 bytes and that the primary and secondary quantities on the SPACE keyword should be multiplied by 1024 to determine the amount of space to be allocated for the file.

---

## BLKSIZE

### in PRINTER MEMBER

**BLKSIZE=** Specifies the block size to be used when DRS/VPI creates the print file on the JES spool or on a DASD volume.

**Valid Values:** 1 - 32760

**Default:** 1024

**Example:** BLKSIZE=800

This specification indicates that DRS/VPI should create the print file with a block size of 800.

**Notes:** BLKSIZE must be equal to or larger than LRECL, and should be a multiple of LRECL for fixed length records. BLKSIZE will be adjusted if an invalid value is specified.

The default BLKSIZE value was changed from 32760 to 1024 by DRS V1 R3.3 fix 3.3.072.



---

## CLASS in PRINTER MEMBER

- CLASS=** Specifies from 1 to 4 SYSOUT classes that should be used when DRS/VPI places SYSOUT on the spool for this virtual printer.
- Valid Values:** Any valid SYSOUT class (A-Z, 0-9, \*)
- Default:** None
- Example 1:** CLASS=A
- This specification indicates that SYSOUT should be placed on the spool using SYSOUT class A.
- Example 2:** CLASS=\*
- This specification indicates that SYSOUT should be placed on the spool with the same SYSOUT class as the MSGCLASS associated with the DRS/VPI job or started task.
- CAUTION:** When a CLASS is not specified in the virtual printer or if CLASS=\* is specified for the virtual printer, the value specified for MSGCLASS will be used. If the MSGCLASS is a class which is automatically deleted by JES it will create output which will never print.
- Example 3:** CLASS=(A,C)
- This specification indicates that SYSOUT should be placed on the spool twice; once with SYSOUT class A and once with SYSOUT class C.

---

## COMMTYPE

### in PRINTER MEMBER

**COMMTYPE=** Specifies, via two positional parameters, the method of communications to be used to receive data for this virtual printer.

**Positional parameter 1 specifies the type of network communications to be used to connect for this virtual printer.**

**Valid Values:** APPC, TCPIP, or VTAM, where:  
APPC = VTAM/SNA LU 6.2  
TCPIP = TCP/IP network  
VTAM = VTAM/SNA LU 0, 1 or 3

**Default:** VTAM

**Positional parameter 2 specifies the protocol to be used when receiving data for this virtual printer.**

**Valid Values:** DRSPC, LPD or SOK where  
DRSPC = DRS/PC  
LPD = TCP/IP Line Printer Daemon  
SOK = TCP/IP Sockets

**Default:** DRSPC if first parameter is APPC  
LPD if first parameter is TCPIP  
None if first parameter is VTAM

**Example 1:** COMMTYPE=TCPIP

This specification indicates that DRS/VPI should accept connections for this printer using TCP/IP LPR/LPD protocols.

**Example 2:** COMMTYPE=(APPC,DRSPC)

This specification indicates that DRS/VPI should accept connections for this printer using APPC LU6.2 and that the sender is DRS/PC.

---

## DATACLAS in PRINTER MEMBER

**DATACLAS=** Specifies the name of the data class to be used by SMS to allocate a DASD file.

**Valid Values:** 1 to 8 character SMS data class name

**Default:** None

**Example:** DATACLAS=TEMPDATA

This specification indicates that SMS should use the TEMPDATA data class when allocating this DASD file. The data class name used must be one that has been pre-defined by the storage administrator at your installation.

---

## DDNAME

### in PRINTER MEMBER

**DDNAME=** Specifies the DDNAME of 1 to 4 spool files that DRS/VPI will create. DDNAME is ignored for a DASD print file.

**Valid Values:** Any valid DDNAME or \*

**Default:** None

**Example 1:** DDNAME=VPRT001

This specification indicates that DRS/VPI should create the print file with a DDNAME of VPRT001.

**Example 2:** DDNAME=(DD1,DD2,DD3,DD4)

This specification indicates that DRS/VPI should create four print files with DDNAMEs of DD1, DD2, DD3 and DD4. Each DDNAME must be unique. Other SYSOUT parameters must be specified for each of the four spool print files.

**Note:** DRS/VPI will use a dynamically generated DDNAME if none is specified.

Coding DDNAME=\* indicates that DRS/VPI should use the member name of the virtual printer definition as the DDNAME. Only one DDNAME can be specified as "\*", since each DDNAME must be unique.

---

## DEFLTMEM in PRINTER MEMBER

**DEFLTMEM=** Specifies a member within the DRS/VPI control library that contains user-defined defaults. If this parameter is not specified, DRS/VPI will use the values coded in the DEFLTMEM keyword from the System Initialization member. If neither is specified, DRS/VPI will use the internal default keyword values.

**Valid Values:** Member name of a member that exists in the DRSVLIB PDS

**Default:** None

**Example:** DEFLTMEM=TCPDEFLT

This specification indicates that member TCPDEFLT in the DRSVLIB PDS contains default values for this virtual printer.

---

## DEST

### in PRINTER MEMBER

**DEST=** Specifies from 1 to 4 destinations that should be used when DRS/VPI places SYSOUT datasets on the spool for this virtual printer.

**Valid Values:** Any valid JES destination

**Default:** None

**Example 1:** DEST=U99

This specification indicates that DRS/VPI should place SYSOUT datasets on the spool using destination U99.

**Example 2:** DEST=(U100,R15)

This specification indicates that DRS/VPI should place SYSOUT datasets on the spool twice; once with a destination of U100 and once with a destination of R15.

---

## DISP in PRINTER MEMBER

**DISP=** Specifies, via two positional parameters, the disposition to be used by DRS/VPI to create a DASD print file. This parameter is ignored if the DSN keyword is not specified.

**Positional parameter 1 specifies the initial disposition or status of a DASD file.**

**Valid Values:** NEW, OLD, SHR, or MOD, where:

- NEW = a new file is to be created.
- OLD = the file exists and exclusive control is required.
- SHR = the file exists and another job can share it.
- MOD = if the file already exists, it should be extended; if the file does not already exist, a new file should be created.

**Default:** NEW

**Positional parameter 2 specifies the normal termination disposition or status of a DASD file.**

**Valid Values:** CATLG, DELETE, KEEP, or UNCATLG, where:

- CATLG = a NEW file should be cataloged when it is terminated.
- DELETE = the file should be deleted when it is terminated.
- KEEP = the file should be kept when it is terminated.
- UNCATLG = the file should be uncataloged when it is terminated.

**Default:** CATLG

**Example:** DISP=(OLD,KEEP)

This specification indicates that a DASD file already exists, and that exclusive control is required. This DASD file will be kept at normal termination.

**Note:** If MOD is specified as the first parameter of DISP, and a new DASD file is created, VOLUME cannot be specified. A dynamic allocation error will occur.

---

## **DRAINED**

### **in PRINTER MEMBER**

**DRAINED=** Specifies whether this printer should be initialized in drained status. If this parameter is coded **DRAINED=Y**, a **START** command will need to be issued to this printer before it will begin receiving output.

**Valid Values:** Y or N

**Default:** N

**Example:** **DRAINED=Y**

This specification would signify to the DRS/VPI that this printer should be initialized in **DRAINED** status.



## DSN in PRINTER MEMBER

**DSN=** Specifies the dataset name of the DASD file to be created by DRS/VPI. The file can be a sequential file or a Partitioned Dataset (PDS). If the file is a PDS, the MEMBER keyword must be specified to name the member to be added by DRS/VPI.

**Valid Values:** 1 - 60 characters; after symbolic substitution, up to 44 characters of the DSN keyword will be used as the name of the DASD file. The table below is used to determine what values will be used for symbolic substitution.

**Default:** None

**Example 1:** DSN=LRS.DRS.SAMPLE.REPORT

This specification indicates that DRS/VPI should create a DASD file with a name of LRS.DRS.SAMPLE.REPORT.

**Example 2:** DSN=&JOB..&MBR..D&DATE..T&TIME..S&SEQ

This specification indicates that DRS/VPI should create a DASD file with a name based on symbolic substitution.

The first DASD file for this virtual printer would be created with a name of DRS.V009.D011096.T104522.S0000001, assuming the following:

- The DRS/VPI jobname was "DRS".
- The printer member name was "V009".
- The current date was January 10, 1996.
- The current time was 10:45:22.

Subsequent files would have updated dates and times, as well as incremented sequential numbers.

Variable names are delimited by another "&" character or a period. If a period is required following the substituted value, then two periods can be coded and only one period will be substituted in the result.

SYMBOLIC NAME	REPLACED BY	DATE	LENGTH	EXAMPLE	SEE NOTE
CLASS	SYSOUT class		1	A	1
CN	Century	CURR	2	19	2
DEST	SYSOUT destination		8	LOCAL	1
DATE	Date (MMDDYY)	CURR	6	092295	2
DAY	Day of week	CURR	3	FRI	2,4
DAYN	Day of week (numeric)	CURR	1	5	2,5
DY	Day of month	CURR	2	21	2
EDATE	Date (DDMMYY)	CURR	6	230995	2
FORM	SYSOUT form name		8	STD	1
HH	Hour of the day	CURR	2	23	2
ICN	Century	INIT	2	19	2
IDATE	Date (MMDDYY)	INIT	6	092195	2
IDAY	Day of week	INIT	3	THU	2,4

## DSN in PRINTER MEMBER

SYMBOLIC NAME	REPLACED BY	DATE	LENGTH	EXAMPLE	SEE NOTE
IDAYN	Day of week (numeric)	INIT	1	4	2,5
IDY	Day of month	INIT	2	21	2
IEDATE	Date (DDMMYY)	INIT	6	210995	2
IHH	Hour of the day	INIT	2	02	2
IJDATE	Julian date	INIT	5	95264	2
IJDAY	Julian day	INIT	3	264	2
IMM	Minutes	INIT	2	23	2
IMO	Month	INIT	2	09	2
ISS	Seconds	INIT	2	45	2
ITIME	Time (HHMMSS)	INIT	6	022345	2
IYR	Year	INIT	2	95	2
JDATE	Julian date	CURR	5	95265	2
JDAY	Julian day	CURR	3	265	2
JOB	Job name of DRS/VPI		8	DRSVPI	
MBR	Printer member name		8	VPRT001	
MM	Minutes	CURR	2	08	2
MO	Month	CURR	2	09	2
PLU	Partner LU (VTAM)		8	CICSP	3
SEQ	Sequential number		7	0000128	6
SS	Seconds	CURR	2	59	2
SYSSEQ	System-wide sequential number		7	0000345	7
TIME	Time (HHMMSS)	CURR	6	230859	2
WRITER	SYSOUT writer name		8	INTRDR	1
YR	Year	CURR	2	95	2

**Note 1:** SYSOUT values are only valid if a SYSOUT file is being created at the same time as the DASD file. Otherwise, these values will be unavailable.

**Note 2:** The date values are based on either the current (CURR) date and time or the virtual printer's initialization (INIT) date and time. All examples above use a current date and time of September 22, 1995 at 11:08:59 P.M. and a printer initialization date and time of September 21, 1995 at 2:23:45 A.M.

**Note 3:** PLU is only valid for COMMTYPE=VTAM or COMMTYPE=APPC. For TCP/IP printers, there is no partner LU name.

**Note 4:** Day of the week is a three character abbreviation. The valid values are: SUN, MON, TUE, WED, THU, FRI, SAT.

**Note 5:** The numeric day of the week is set as follows: SUN=0, MON=1, TUE=2, WED=3, THU=4, FRI=5, SAT=6.

**Note 6:** The sequential number is reset each time a virtual printer is activated. It will be incremented by 1 each time a DASD file is created.

**Note 7:** The system-wide sequential number is maintained for all printers and is incremented by 1 each time it is used. This number is not reset when any individual printer is re-activated.

---

## DSNTYPE in PRINTER MEMBER

**DSNTYPE=** Specifies the type of file to be allocated if the MEMBER keyword is specified and the file does not exist.

**Valid Values:** PDS or LIBRARY, where:  
PDS = Partitioned Dataset  
LIBRARY = PDSE Library

**Default:** None

**Example:** DSNTYPE=LIBRARY

This specification indicates that the file to be allocated is a PDSE.

---

## ERRACTN

### in PRINTER MEMBER

**ERRACTN=** Specifies the action to be performed by the DRS Virtual Printer Interface when an error occurs while placing the report on the spool.

**Valid Values:** HOLD, KEEP, or PURGE, where:

HOLD = Place report on HELD queue.

KEEP = Keep report on spool using original attributes.

PURGE = Delete report from spool.

**Default:** KEEP

**Example:** ERRACTN=PURGE

This specification would indicate to the DRS/VPI that after an error for this virtual printer task, any incomplete report should be deleted from the spool.

---

## EXPDT in PRINTER MEMBER

**EXPDT=** Specifies the expiration date for a DASD file to be created by DRS/VPI.

**Valid Values:** The Julian date in the format YYYYDDD  
Exactly seven characters must be specified.  
YYYY can be 0000 to 9999. DDD can be 001 to 366.

**Default:** None

**Example:** EXPDT=2010003  
This specification indicates that the file would expired on January 3, 2010.

**Note:** EXPDT and RETPD are mutually exclusive keywords.

---

## FCB

### in PRINTER MEMBER

**FCB=** Specifies from 1 to 4 FCB names that should be associated with reports placed on the spool for this virtual printer.

**Valid Values:** Any 1-4 character FCB name that could be specified in JCL

**Default:** None

**Example 1:** FCB=M6

This specification indicates that DRS/VPI should place SYSOUT datasets on the spool using FCB name of M6.

**Example 2:** FCB=(M6,STD8)

This specification would signify to the DRS/VPI to place SYSOUT datasets on the spool twice; once with an FCB name of M6 and once with an FCB name of STD8.

---

## FORM in PRINTER MEMBER

- FORM=** Specifies from 1 to 4 form names that should be associated with reports placed on the spool for this virtual printer.
- Valid Values:** Any 1-4 character FORM name that could be specified in JCL
- Default:** None
- Example 1:** FORM=FRM3  
This specification indicates that DRS/VPI should place SYSOUT datasets on the spool using form name of FRM3.
- Example 2:** FORM=(FRM3,CHEK)  
This specification indicates that DRS/VPI should place SYSOUT datasets on the spool twice; once using a form name of FRM3 and once using a form name of CHEK.

---

## **HOLD**

### **in PRINTER MEMBER**

**HOLD=** Specifies the HOLD/NOHOLD value for 1 to 4 datasets placed on the spool for this virtual printer.

**Valid Values:** Y or N

**Default:** N

**Example 1:** HOLD=Y

This specification indicates that the DRS/VPI should place SYSOUT datasets on the HOLD queue.

**Example 2:** HOLD=(N,Y)

This specification indicates that the DRS/VPI should place SYSOUT datasets on the spool twice; once on the output queue and once on the HOLD queue.



---

## IPDCLASS in PRINTER MEMBER

**IPDCLASS=** Specifies the SYSOUT class that should be used when DRS places IPDS output on the spool for this virtual printer.

**Valid Values:** Any valid SYSOUT class (A-Z, 0-9, \*)

**Default:** Value specified for CLASS keyword

**Example 1:** IPDCLASS=I

This specification would signify to the DRS/VPI that IPDS output should be placed on the spool using SYSOUT class I.

**Example 2:** IPDCLASS=\*

This specification would signify to the DRS/VPI that IPDS output should be placed on the spool with the same SYSOUT class as the MSGCLASS associated with the DRS/VPI job or started task.

**CAUTION:** If the MSGCLASS which is used for the DRS/VPI started task is a class which is automatically deleted by JES, specifying IPDCLASS=\* for the virtual printer or obtaining IPDCLASS=\* by default will create output which will never print.

---

## IPDSOPTS

### in PRINTER MEMBER

**IPDSOPTS=** A two byte hex representation of printer options to be in effect when processing IPDS output for this printer. The value is specified as four digits, representing two hex bytes (16 option bits). The IPDS option bit specifications are:

8000 = Remove all requests for acknowledgement

4000 = Use 18 byte acknowledgement

2000 = Undefined

1000 = Undefined

0800 = Undefined

0400 = Undefined

0200 = Undefined

0100 = Undefined

0080 = Undefined

0040 = Undefined

0020 = Undefined

0010 = Undefined

0008 = Undefined

0004 = Undefined

0002 = Undefined

0001 = Undefined

**Valid Values:** Values: 0000 - FFFF

**Default:** 0000

**Example:** IPDSOPTS=C000

This specification would indicate to the DRS/VPI that all requests for acknowledgement found in an IPDS report should be removed before placing the report on the spool, and that an 18 byte acknowledgement should be used when returning an acknowledgement to the sender of the IPDS data.

---

## JOBNAME in PRINTER MEMBER

- JOBNAME=** Specifies the name that should be used as the JOB name when a SYSOUT dataset is created.
- Valid Values:** 1 - 16 characters, including alphabetic, numeric and national characters, plus the symbolic parameters from the table below. Symbolic parameters begin with ampersand (&) and can be terminated with another symbolic parameter, with a period (.), or by the end of the keyword value.
- Default:** None
- Example 1:** JOBNAME=&PLU&SEQ  
This indicates that DRS should use the partner LU name (for a VTAM virtual printer only) and a 2-byte sequence number to create the JOBNAME. If the partner LU name was CICST, the JOBNAME would be set as CICST01, CICST02, etc.
- Example 2:** JOBNAME=Z&LPDP  
This indicates that DRS should append the first 7 bytes of the "P" LPD control record data to the letter "Z" to create the JOBNAME. If the "P" LPD control file record contained "USERABCDE", the JOBNAME would be set to ZUSERABC.
- Example 3:** JOBNAME=JOB&MM&SS.A  
This indicates that DRS should append the current time in minutes and seconds to the letters "JOB" and add the letter "A" at the end. At 11:23:14 a.m., the JOBNAME would be set to JOB2314A.
- Notes:** If JOBNAME keyword is not specified, the SYSOUT file will be created using the name of the DRS/VPI job or started task. If the JOBNAME keyword value is not valid when the printer is activated, the printer activation will fail. If the JOBNAME is not valid after symbolic substitution, a warning message will be issued and the SYSOUT file will be created using the name of the DRS/VPI job or started task.
- Using JOBNAME requires MVS/SP 4.1.0 or later and that the DRS/VPI and DRS/API modules be loaded from an authorized library.
- For TCP/IP printer members, the following symbolic parameter is not allowed:
- &PLU
- For VTAM printer members, the following symbolic parameters are not allowed:
- &LPDC  
&LPDCOMM  
&LPDH  
&LPDJ  
&LPDL  
&LPDM  
&LPDP  
&LPDT

## JOBNAME

### in PRINTER MEMBER

SYMBOLIC NAME	REPLACED BY	DATE	LENGTH	EXAMPLE	SEE NOTE
CN	Century	CURR	2	19	2
DATE	Date (MMDDYY)	CURR	6	092295	2
DAY	Day of week	CURR	3	FRI	2,4
DAYN	Day of week (numeric)	CURR	1	5	2,5
DY	Day of month	CURR	2	21	2
EDATE	Date (DDMMYY)	CURR	6	230995	2
HH	Hour of the day	CURR	2	23	2
ICN	Century	INIT	2	19	2
IDATE	Date (MMDDYY)	INIT	6	092195	2
IDAY	Day of week	INIT	3	THU	2,4
IDAYN	Day of week (numeric)	INIT	1	4	2,5
IDY	Day of month	INIT	2	21	2
IEDATE	Date (DDMMYY)	INIT	6	210995	2
IHH	Hour of the day	INIT	2	02	2
IJDATE	Julian date	INIT	5	95264	2
IJDAY	Julian day	INIT	3	264	2
IMM	Minutes	INIT	2	23	2
IMO	Month	INIT	2	09	2
ISS	Seconds	INIT	2	45	2
ITIME	Time (HHMMSS)	INIT	6	022345	2
IYR	Year	INIT	2	95	2
JDATE	Julian date	CURR	5	95265	2
JDAY	Julian day	CURR	3	265	2
JOB	Job name of DRS/VPI		8	DRSVPI	
LPDC	LPD 'C' control file record		8	CLASS	1,7
LPDCOMM	LPD '_' control file record		8	COMMENT	1,7
LPDH	LPD 'H' control file record		8	HOSTNAME	1,7
LPDJ	LPD 'J' control file record		8	JOBNAME	1,7
LPDL	LPD 'L' control file record		8	BANNER	1,7
LPDM	LPD 'M' control file record		8	MAILUSER	1,7
LPDP	LPD 'P' control file record		8	USERNAME	1,7
LPDT	LPD 'T' control file record		8	TITLE	1,7
MBR	Printer member name		8	VPRT001	
MM	Minutes	CURR	2	08	2
MO	Month	CURR	2	09	2
PLU	Partner LU (VTAM)		8	CICSP	3
SEQ	Sequential number		2	17	6
SS	Seconds	CURR	2	59	2
SYSSEQ	System-wide sequential number		2	45	8
TIME	Time (HHMMSS)	CURR	6	230859	2
YR	Year	CURR	2	95	2

---

## JOBNAME in PRINTER MEMBER

- Note 1:** If the JOBNAME keyword is not specified, the SYSOUT file will be created using the name of the DRS/VPI job or started task. If the JOBNAME keyword value is not valid at the time the printer is activated, printer activation will fail. If PRTROPTS=00000100 is not specified and the JOBNAME value is invalid after symbolic substitution is complete, a warning message will be issued, and the SYSOUT file will be created using the name of the DRS/VPI job or started task. If PRTROPTS=00000100 is specified, the printer will be EDRAINED if the JOBNAME value is invalid after symbolic substitution is complete.
- Note 2:** The date values are based on either the current (CURR) date and time or the virtual printer's initialization (INIT) date and time. All examples above use a current date and time of September 22, 1995 at 11:08:59 P.M. and a printer initialization date and time of September 21, 1995 at 2:23:45 A.M.
- Note 3:** PLU is only valid for COMMTYPE=VTAM or COMMTYPE=APPC. For TCP/IP printers, there is no partner LU name.
- Note 4:** Day of the week is a three character abbreviation. The valid values are: SUN, MON, TUE, WED, THU, FRI, SAT.
- Note 5:** The numeric day of the week is set as follows: SUN=0, MON=1, TUE=2, WED=3, THU=4, FRI=5, SAT=6.
- Note 6:** The sequential number is reset each time a virtual printer is activated. It will be incremented by 1 each time a DASD file is created.
- Note 7:** If any LPD symbolic variable is specified and TCPOPTS=80 is also specified to ignore the LPD control file, the symbolic variable will have a null value. If there are no other valid values in the JOBNAME field, printer activation will fail.
- Note 8:** The system-wide sequential number is maintained for all printers and is incremented by 1 each time it is used. This number is not reset when any individual printer is re-activated.

---

## LRECL

### in PRINTER MEMBER

**LRECL=** Specifies the logical record size to be used when DRS/VPI creates the printer file on the JES spool or on a DASD volume.

**Valid Values:** For RECFM=(U,), 1 - 32760  
For RECFM=(F,), 1 - 32760  
For RECFM=(F,A), 1 - 32760  
For RECFM=(V,), 4 - 32756  
For RECFM=(V,A), 5 - 32756

**Default:** 1020

**Example:** LRECL=80

This specification indicates that DRS/VPI should create the print file with a record size of 80.

**Note:** The default LRECL value was changed from 32756 to 1020 by DRS V1 R3.3 fix 3.3.072.

---

## MAXLPG in PRINTER MEMBER

**MAXLPG=** Specifies the maximum lines per page to be allowed when DRS/VPI is receiving print data.

**Valid Values:** 0 - 256

**Default:** 0

**Example:** MAXLPG=50

This specification indicates that a new page should be created after 50 lines of data have been received.

**Note:** MAXLPG is used only for TCP/IP print queue definitions when processing a “p” print filter, which requires that DRS/VPI add a heading to each page. If no value is specified for MAXLPG, the maximum print lines per page will be set to 55.

---

## MEMBER

### in PRINTER MEMBER

**MEMBER=** Specifies the member name of a partitioned dataset (PDS) to be created by DRS/VPI. This parameter is ignored if the DSN keyword is not specified.

**Valid Values:** 1 - 8 character name of a member of a PDS

**Default:** None

**Example:** MEMBER=VPRT009

This specification indicates that a member named VPRT009 should be added to the PDS whose name is specified in the DSN keyword for this printer definition.



---

## MGMTCLAS in PRINTER MEMBER

**MGMTCLAS=** Specifies the name of the management class to be used by SMS to allocate a DASD file.

**Valid Values:** 1 - 8 character SMS management class name

**Default:** None

**Example:** MGMTCLAS=ACCTMGMT

This specification indicates that SMS should use the ACCTMGMT management class when allocating this DASD file. The management class name used must be one that has been pre-defined by the storage administrator at your installation.

---

## MPP

### in PRINTER MEMBER

**MPP=** Specifies the maximum print position in report lines for the virtual printer. The DRS/VPI will use this value to end a print record and start a new print record when receiving output.

**Valid Values:** 1-3 digit numeric value between 0 and 255

**Default:** MPP=132

**Example:** MPP=220

This specification indicates to the DRS/VPI to create print records on the spool which do not exceed 220 characters. If no line feed or form feed characters have been encountered and 220 characters have been received in this print record, the DRS/VPI will begin a new print record on the spool.

**Note:** If the maximum print position is specified as 0, DRS/VPI will create print records based on the number of characters of data received for each print line. The end of a print line is normally indicated by line feed, new line, or form feed characters. No print record will be created which is longer than the LRECL specified.

If the LRECL is set to a value that is less than the value set for MPP, the MPP value will be adjusted to be equal to the LRECL.

---

## NODE in PRINTER MEMBER

**NODE=** Specifies from 1 to 4 node names that should be used when placing SYSOUT datasets on the spool for this virtual printer. If a node name is specified, the DEST field can contain a JES destination or a userid.

**Valid Values:** Any valid JES node name

**Default:** None

**Example 1:** NODE=N19

This specification indicates that DRS/VPI should place SYSOUT datasets on the spool using a node name of N19.

**Example 2:** NODE=(N19,N15)

This specification indicates that DRS/VPI should place SYSOUT datasets on the spool twice; once with a node name of N19 and once with a node name of N15.

---

## OUTREF

### in PRINTER MEMBER

**OUTREF=** Specifies the names of 1 to 4 output references to be associated with SYSOUT datasets on the JES spool for this virtual printer. Each name can be that of an OUTPUT JCL statement in the DRS/VPI JCL, or the name of an Output Reference member in the DRS/VPI control library. (For more information see [“Output Reference Members”](#) on page 3.12.)

Specifying multiple output references will cause the virtual printer to create multiple copies of the SYSOUT dataset. If you specify the name of an OUTPUT JCL statement that does not exist, DRS will encounter a dynamic allocation failure when attempting to allocate the SYSOUT dataset(s).

**Valid Values:** Any 1 - 8 character name of an OUTPUT JCL statement

**Default:** None

**Example:** OUTREF=(VPRTOUT1,VPRTOUT2)

This specification would signify to the DRS/VPI to place two copies of the SYSOUT datasets on the spool, using attributes from the OUTPUT JCL statement or DRS/VPI control library member named VPRTOUT1 for one copy and attributes from the OUTPUT JCL statement or DRS/VPI control library member named VPRTOUT2 for the second copy.

---

## PRTROPTS in PRINTER MEMBER

**PRTROPTS=** A four byte hex representation of special printer options to be in effect for this printer. The value is specified as eight digits, representing four hex bytes. The printer option bit specifications are:

- 80000000** = Force form feed for first carriage control in report.
- 40000000** = Set blank for first carriage control in report.
- 20000000** = Eliminate null lines in print block (LU-0 and LU-3 only).
- 10000000** = Do not force LRECL of 80 if WRITER=INTRDR.
- 08000000** = Re-block incoming AFPDS X'5A' records.
- 04000000** = Space one position after form feed (LU-0 and LU-3 only).
- 02000000** = Ignore last form feed in report.
- 01000000** = Reserved.
- 00800000** = Use first data byte in record as carriage control byte.
- 00400000** = MPP is applicable to TRN data (LU-1 only).
- 00200000** = Reserved (function replaced with PRTXLATE keyword).
- 00100000** = Eliminate 2<sup>nd</sup> form feed if no data between it and previous form feed.
- 00080000** = SCS transparent data should be individual spool records.
- 00040000** = Clear output buffer for every write command (LU-0 and LU-3 only).
- 00020000** = Reserved (replaced by TRN=T).
- 00010000** = Consider X'1A' end of file if processing AFPDS X'5A' records.
- 00008000** = Consider all input data as fixed records.
- 00004000** = Undefined.
- 00002000** = Undefined.
- 00001000** = Bypass adding page eject when Set Vertical Format command received (SCS only).
- 00000800** = Process carriage restore and line feed when line length is set by WCC (LU0 and LU3 only).
- 00000400** = Undefined.
- 00000200** = Undefined.
- 00000100** = EDRAIN printer for invalid JOBNAME after substitution.
- 00000080** = Use DRS/PC headers to update report attributes.
- 00000040** = Reserved for DRS/PC header testing.
- 00000020** = Use first 2 bytes of data record as length field (LPD only).
- 00000010** = Emulate automatic new line for DSC.
- 00000008** = Undefined.
- 00000004** = For DSC and SCS, ignore last NL in report.
- 00000002** = Treat DBCS "shift out" (X'0E') and "shift in" (X'0F') as data.
- 00000001** = Undefined.

---

## PRTROPTS in PRINTER MEMBER

**Valid Values:** 00000000 - FFFFFFFF

**Default:** 00000000

**Example:** PRTROPTS=20000000

This specification indicates that DRS/VPI should eliminate null lines when processing print data received using LU-0 or LU-3 sessions.

**Note 1:** With PRTROPTS=08000000, files which contain X'5A' in the first position will be processed as AFP records. All records must begin with X'5A' and contain a length field at the proper location. Any AFP format errors will cause processing to be terminated and the printer EDRAINED.

AFP file re-blocking allows applications running on the LAN or on a TCP/IP host to create Advanced Function Printing resources or documents which can be printed to devices attached to the MVS system.

When an application or an AFP driver creates a block of records which contain AFP structures, the blocks must be separated into individual records in order for them to be processed or printed on MVS. DRS/VPI can separate the records in these AFP blocks and place the records on the JES spool or into a DASD file. Then the AFP records can be used as resources (PAGEDEF, FORMDEF, OVERLAY, etc.), or the documents can be printed by VPS/IPDS, VPS/PCL or other AFP software.

---

## PRTXDBCS in PRINTER MEMBER

**PRTXDBCS=** Specifies, via 5 positional parameters, the type of translation that should be done for DBCS data received.

**Positional parameter 1 specifies whether or not any DBCS data should be translated.**

**Valid Values:** Y or N

**Default:** N

**Positional parameter 2 specifies the name of the translate table to be used for DBCS data.**

**Valid Values:** Any valid translate table name that exists as a member in the DRS/VPI load library.

**Default:** None

**Positional parameter 3 indicates what action should be taken for shift-out and shift-in characters.**

**Valid Values:** KEEP, DELT, REPL, ADD

KEEP = Maintains the SOSI characters in the print line.

DELT = Deletes the SOSI characters from the print line.

REPL = Replaces the SO and SI characters with the replacement SO and SI characters specified in the fourth and fifth positional parameters of the PRTXDBCS keyword.

ADD = Adds the SO and SI characters with the replacement SO and SI characters specified in the fourth and fifth positional parameters of the PRTXDBCS keyword. The SOSI characters are added based upon the DBCSID keyword used when creating the DBCS translate table. (ADD is only allowed for COMMTYPE=TCPIP.)

**Default:** KEEP

**Positional parameter 4 indicates the shift-out (SO) replacement character.**

**Valid Values:** Any 1-3 byte hex value.

**Default:** X'0E'

**Note 1:** This value is ignored unless positional parameter 3 is ADD or REPL.

---

## PRTXDBCS

### in PRINTER MEMBER

**Positional parameter 5 indicates the shift-in (SI) replacement character.**

**Valid Values:** Any 1-3 byte hex value.

**Default:** X'0F'

**Note 2:** This value is ignored unless positional parameter 3 is ADD or REPL.

**Example:** PRTXDBCS=(Y,VXTABLE,ADD,0E,0F)  
This specifies that this printer should translate DBCS data using translation table VXTABLE, that the SO/SI characters should be added before and after any DBCS data located in the input datastream.

**Note 3:** For more information on using DBCS with DRS/VPI and DRS/TCPIP, see [“Creating a User Translation Table” on page 10.2.](#)



---

## PRTXLATE in PRINTER MEMBER

**PRTXLATE=** Specifies, via five positional parameters, the type of translation to be applied to data when it is received. The last 3 parameters are only used for virtual printers with COMMTYPE=VTAM with LU-1 (SCS) sessions.

**Positional parameter 1 specifies whether or not any translation of normal data should take place for the virtual printer.**

**Valid Values:** Y or N

**Default:** N

**Positional parameter 2 specifies the name of the translate table to be used for normal data received.**

**Valid Values:** Any valid translate table name that exists as a member in the DRS/VPI load library.

**Default:** DVSSXLTE

**Positional parameter 3 specifies whether or not any translation should be done for data received as transparent data. This value is only valid for COMMTYPE=VTAM and LU-1 sessions. For more information on transparent data, see the TRN keyword.**

**Valid Values:** Y or N

**Default:** N

**Positional parameter 4 specifies the name of the translate table to be used for transparent data received.**

**Valid Values:** Any valid translate table name that exists as a member in the DRS/VPI load library.

**Default:** None

**Note:** This value is ignored unless positional parameter 3 is Y.

**Positional parameter 5 indicates what should be done with any transparency characters received.**

**Valid Values:** DELT or KEEP

**Default:** KEEP

**Example 1:** PRTXLATE=(Y,DVSSXEBC)

This specifies that DRS/VPI should translate data using a translate table named DVSSXEBC.

**Example 2:** PRTXLATE=(Y,DVSSXLTE,Y,DVSSXEBC,DELT)

This specifies that normal data received should be translated using a table named DVSSXLTE, that any transparent data should be translated using a table named DVSSXEBC, and that the transparency characters should be deleted from the data.

---

## QBUFSIZE

### in PRINTER MEMBER

**QBUFSIZE=** Specifies the buffer size that DRS/VPI will use when writing to and reading from the staging dataset.

**Valid Values:** 4096 - 32760

**Default:** 4096

**Example:** QBUFSIZE=6144

This specification indicates that DRS/VPI should use a 6144 byte buffer for building staging buffers.

**Note:** The QBUFSIZE parameter is only used for DRS/TCPIP virtual printers defined with COMMTYPE=(TCPIP,LPD). Staging is only necessary if the data file is received before the control file, and the control file must be used to determine how the data file should be processed.

No staging is done for devices with COMMTYPE=VTAM or COMMTYPE=APPC.

---

## QSPACE in PRINTER MEMBER

**QSPACE=** Specifies, via three positional parameters, the amount of DASD space to be allocated for the staging dataset. This parameter is used only for print queue definitions for TCP/IP devices using LPR/LPD protocols.

**Positional parameter 1 specifies the allocation type for the staging dataset.**

**Valid Values:** C - Allocate in cylinders  
T - Allocate in tracks

**Default:** T

**Positional parameter 2 specifies the primary allocation amount for the staging dataset.**

**Valid Values:** 1 - 999

**Default:** 30

**Positional parameter 3 specifies the secondary allocation amount for the staging dataset.**

**Valid Values:** 0 - 999

**Default:** 15

**Example:** QSPACE=(C,10,5)

This specification indicates that DRS/VPI should allocate the temporary staging dataset using cylinders. The primary allocation amount will be 10 cylinders, and the secondary amount will be 5 cylinders.

**Note:** The QSPACE parameter is only used for DRS/TCPIP virtual printers defined with COMMTYPE=(TCPIP,LPD). Staging is only necessary if the data file is received before the control file, and the control file must be used to determine how the data file should be processed.

No staging is done for devices with COMMTYPE=VTAM or COMMTYPE=APPC.

---

## QUNIT

### in PRINTER MEMBER

**QUNIT=** Specifies the unit type of the device on which DRS/VPI is to allocate the staging dataset.

**Valid Values:** Any valid DASD unit name

**Default:** SYSDA

**Example:** QUNIT=3380

This specification indicates that DRS/VPI should allocate the temporary staging dataset on a device with unit type of 3380.

**Note:** The QUNIT parameter is only used for DRS/TCPIP virtual printers defined with COMMTYPE=(TCPIP,LPD). Staging is only necessary if the data file is received before the control file, and the control file must be used to determine how the data file should be processed.

No staging is done for devices with COMMTYPE=VTAM or COMMTYPE=APPC.

---

## QVOLUME in PRINTER MEMBER

**QVOLUME=** Specifies the volume on which DRS/VPI is to allocate the staging dataset. If this parameter is not specified, DRS/VPI will allocate the staging dataset on any available public or storage volume.

**Valid Values:** Any valid DASD volume name

**Default:** None

**Example:** QVOLUME=WRK101

This specification indicates that DRS/VPI should allocate the temporary staging dataset on a volume with a name of WRK101.

**Note:** The QVOLUME parameter is only used for DRS/TCPIP virtual printers defined with COMMTYPE=(TCPIP,LPD). Staging is only necessary if the data file is received before the control file, and the control file must be used to determine how the data file should be processed.

No staging is done for devices with COMMTYPE=VTAM or COMMTYPE=APPC.

---

## RECFM

### in PRINTER MEMBER

**RECFM=** Specifies, via two positional parameter values, the record format and carriage control attributes to be used when DRS/VPI creates the print file on the JES spool or on a DASD volume.

**Positional parameter 1** specifies the record format.

**Valid Values:** F, U, or V, where:  
F = fixed record format  
U = undefined record format  
V = variable record format

**Default:** V

**Positional parameter 2** specifies the carriage control to be used when creating the print file.

**Valid Values:** A, M or blank, where  
A = ASA carriage control  
M = Machine carriage control  
Blank = No carriage control

**Default:** A

**Example 1:** RECFM=(V,A)

This specification indicates that DRS/VPI should create the print file with variable record format and ASA carriage control.

**Example 2:** RECFM=(F,)

This specification indicates that DRS/VPI should create the print file with a fixed record format and no carriage control. The second parameter value must be nullified, because its default is A.

**Note:** RECFM=(,M) is only allowed if PRTROPTS=00800000 is specified to indicate that the incoming data already has a carriage control byte in the first position of each record.

---

## RETPD in PRINTER MEMBER

**RETPD=** Specifies the retention period for a DASD file to be created by DRS/VPI.

**Valid Values:** 0 - 9999

**Default:** None

**Example:** RETPD=30

This specification indicates that the file would be retained for 30 days.

**Note:** RETPD=0 is valid and indicates that the file expires today. If no retention period is wanted, the keyword should not be specified.

EXPDT and RETPD are mutually exclusive keywords.

---

## SEPAR

### in PRINTER MEMBER

**SEPAR=** Specifies, via two positional parameter values, information about the separator pages, if any, that should be added to the print file created by DRS/VPI.

**Positional parameter 1** indicates when separator pages should be added.

**Valid Values:** S, E, B, or N, where:  
S = separator at start of dataset only  
E = separator at end of dataset only  
B = separators at both start and end of dataset  
N = no separators

**Default:** N

**Positional parameter 2** indicates the name of the separator load module to be used for formatting the separators for this virtual printer.

**Valid Values:** Any valid member name of a load module in the DRS/VPI load library

**Default:** DVSSSEPR

**Example:** SEPAR=(B,DVSSSEP4)

This specification indicates that DRS/VPI should add a separator page at the start and at the end of the print file. The name of the module that will be called to add the separator lines is DVSSSEP4.

**Note:** The DVSSSEPR separator routine that is loaded when DRS/VPI is initialized is provided in the DRS load module library on the distribution cartridge. The source for other sample separator routines can be found in the LRS.DRS.V1R33.ASM library from the distribution cartridge.

For more information about using the DRS/VPI separator routines, see [“Separator Routine” on page 9.2.](#)



---

## SNAP in PRINTER MEMBER

**SNAP=** Specifies whether the DRS/VPI should automatically take a snap dump when this virtual printer task encounters an unrecoverable error situation.

**Valid Values:** Y or N

**Default:** Y

**Example:** SNAP=N

This specification would signify that the DRS/VPI should not automatically generate snap dumps for this virtual printer task.

---

## SPACE

### in PRINTER MEMBER

**SPACE=** Specifies, via five positional parameters, the amount of space to be allocated for a DASD print file. This parameter is ignored if the DSN keyword is not specified.

**Positional parameter 1 specifies the allocation type for the DASD print file.**

**Valid Values:** B, C, R, or T, where:  
B = allocate in blocks  
C = allocate in cylinders  
R = allocate in records  
T = allocate in tracks

**Default:** T

**Positional parameter 2 specifies the primary allocation amount for the DASD print file.**

**Valid Values:** 0 - 32767

**Default:** 30

**Positional parameter 3 specifies the secondary allocation amount for the DASD print file.**

**Valid Values:** 0 - 32767

**Default:** 10

**Positional parameter 4 specifies the number of directory blocks to be allocated if the DASD file is a PDS. This parameter is required if the MEMBER keyword is specified and DISP=NEW.**

**Valid Values:** 0 - 32767

**Default:** 0

**Positional parameter 5 specifies whether unused space should be released when the file is unallocated.**

**Valid Values:** RLSE

**Default:** None

**Example:** SPACE=(C,1,10)

This specification indicates that DRS/VPI should allocate the DASD print file using cylinders allocation. The primary allocation amount will be 1 cylinder, and the secondary allocation amount will be 10 cylinders.

**Note:** If the first positional parameter of the SPACE keyword specifies "R" for record allocation, the AVGREC keyword will be used to determine the size of the average record and how the primary and secondary allocation numbers should be used to obtain DASD space.

---

## STORCLAS in PRINTER MEMBER

**STORCLAS=** Specifies the name of the storage class to be used by SMS to allocate a DASD file.

**Valid Values:** 1 to 8 character SMS storage class name

**Default:** None

**Example:** STORCLAS=MYSTOR

This specification indicates that SMS should use the MYSTOR storage class when allocating this DASD file. The storage class name used must be one that has been pre-defined by the storage administrator at your installation.

---

## SUBSYS

### in PRINTER MEMBER

**SUBSYS=** Specifies the name of the subsystem and the parameter that should be passed to that subsystem when a print file is initialized.

**Positional parameter 1 specifies the name of the subsystem.**

**Valid Values:** Any valid 1 - 4 character subsystem name.

**Default:** None

**Positional parameter 2 specifies the parameter to be passed.**

**Valid Values:** Any 1 - 16 characters.

**Default:** None

**Example:** SUBSYS=(VPC1,PARM1)

This specification indicates that the print file should be passed to a subsystem with the name VPC1 and that a parameter named PARM1 should be passed to that subsystem.

---

## TAB in PRINTER MEMBER

**TAB=** Specifies the number of blanks to be used to replace the TAB character. This keyword is only valid for TCP/IP printer definitions.

**Valid Values:** 0 - 255

**Default:** 0 (do not replace TAB character)

**Example:** TAB=5

This indicates that DRS/TCPIP should place 5 blanks into the print line to replace the TAB character.

**Note:** This keyword is ignored if COMMTYPE=VTAM or COMMTYPE=APPC.

---

## TCPLIMIT

### in PRINTER MEMBER

**TCPLIMIT=** Specifies the maximum number of bytes to be accepted from a remote TCP/IP host using the LPR/LPD protocols. If an incoming "RECEIVE PRINT FILE" command indicates that the data file will be larger than the number specified in the TCPLIMIT keyword, the file will be rejected with the error text "DATA FILE LENGTH LIMIT EXCEEDED".

**Valid Values:** 0 - 9999999

**Default:** 0 (no limit)

**Example:** TCPLIMIT=10000

This specification indicates that DRS/VPI should reject any print file which is larger than 10,000 bytes.

**Note:** The TCPLIMIT parameter is used only for DRS/TCPIP virtual printers defined with COMMTYPE=(TCPIP,LPD).

---

## TCPMRD in PRINTER MEMBER

**TCPMRD=** Specifies the time interval in minutes for DRS/VPI to wait for a TCP/IP request to complete. If the request fails to complete during that interval, the connection will be terminated.

**Valid Values:** 00 - 59

**Default:** 0 (never time out)

**Example:** TCPMRD=05

This specification indicates that DRS/VPI should terminate any connection if a TCP/IP request does not complete within five minutes.

**Note:** The TCPMRD parameter is used only for DRS/TCPIP virtual printers defined with COMMTYPE=TCPIP.

The TCPMRD keyword value must be specified as a two byte field; for example, TCPMRD=5 would not be valid.

---

## TCPOPTS

### in PRINTER MEMBER

**TCPOPTS=** A four byte hex representation of special processing options to be in effect for this TCP/IP printer. The value is specified as eight digits, representing four hex bytes. The printer option bit specifications are:

**80000000** Ignore incoming control file.

This option can be set to ignore the control file and eliminate the need to place the data file in a temporary staging area. Staging is necessary if the control file is needed to process the data file correctly, and if the control file is received after the data file. No staging is done for any virtual printer except those with COMMTYPE=(TCPIP,LPD).

If the JOBNAME keyword specifies any LPD variable and TCPOPTS=80 is specified, a null value will be substituted for that variable. If there are no other valid values in the JOBNAME field, printer activation will fail.

**40000000** Force staging of data file.

Staging is not necessary if the control file is received before the data file, or if the control file can be ignored. This option can be set to force the data file to be staged, regardless of other options. No staging is done for any virtual printer except those with COMMTYPE=(TCPIP,LPD).

**20000000** Change SYSOUT class based on CLASS record.

The first byte of the "C" class record which is received in the control file can be used for the SYSOUT class when the print file is placed on the JES spool.

**10000000** Truncate lines based on WIDTH record.

Any value specified on the "W" width record in the control file normally causes the MPP value to be set and any data beyond that point on a line to be "wrapped" to the next line. If this option is set, the line will be truncated at the WIDTH value.

**08000000** Use print queue name as printer name for DRS/STI.

The print queue name from the "RECEIVE PRINT JOB" command will be stored as the printer name in the Smart Tag Control Area. This control area is passed to the DRS/STI programs and used to obtain the SYSOUT characteristics to place the file on the JES spool.

**04000000** Use print queue name as OUTPUT name for DRS/STI.

The print queue name from the "RECEIVE PRINT JOB" command will be stored as the OUTPUT name in the Smart Tag Control Area. This control area is passed to the DRS/STI programs and used to obtain the SYSOUT characteristics to place the file on the JES spool.



---

## TCPOPTS in PRINTER MEMBER

<b>02000000</b>	Undefined
<b>01000000</b>	Bypass orderly disconnection Bypass the call to MVS TCP/IP to close the connection in an orderly manner. The connection will be abnormally terminated.
<b>00800000</b>	Undefined
<b>00400000</b>	Undefined
<b>00200000</b>	Undefined
<b>00100000</b>	Undefined
<b>00080000</b>	Undefined
<b>00040000</b>	Undefined
<b>00020000</b>	Undefined
<b>00010000</b>	Undefined
<b>00008000</b>	Request acknowledgement before disconnect DRS/VPI will send an ACK before the connection is closed.
<b>00004000</b>	Use SHUTDOWN command before disconnect DRS/VPI will send a SHUTDOWN command to the remote host before the connection is closed.
<b>00002000</b>	Undefined
<b>00001000</b>	Undefined
<b>00000800</b>	Undefined
<b>00000400</b>	Undefined
<b>00000200</b>	Undefined
<b>00000100</b>	Undefined
<b>00000080</b>	Undefined
<b>00000040</b>	Undefined
<b>00000020</b>	Undefined
<b>00000010</b>	Undefined
<b>00000008</b>	Undefined
<b>00000004</b>	Undefined
<b>00000002</b>	Undefined
<b>00000001</b>	Undefined
<b>Valid Values:</b>	00000000 - FFFFFFFF
<b>Default:</b>	00000000
<b>Example:</b>	TCPOPTS=30000000 This specification indicates that DRS/VPI should set the SYSOUT class based on the "C" class record in the control file, and truncate the data lines using the value from the "W" width record.

---

## TERMRPT

### in PRINTER MEMBER

**TERMRPT=** Specifies, via two positional parameters, under what conditions that DRS/VPI will terminate a report and make it available for printing.

**Positional parameter 1 specifies the report termination option that should be used for this printer.**

**Valid Values:** APPC, BRACKET, CHAIN, LPD, SESSION, TABLE or ss, where:

**APPC** = Terminate report at end of APPC dataset. This option is only valid for COMMTYPE=(APPC,DRSPC).

**BRACKET** = Terminate report at end of SNA bracket. This option is valid for COMMTYPE=VTAM, and is ignored for COMMTYPE=TCPIP and COMMTYPE=APPC.

**CHAIN** = Terminate report at end of SNA chain. This option is valid for COMMTYPE=VTAM, and is ignored for COMMTYPE=TCPIP and COMMTYPE=APPC.

**LPD** = Terminate report at end of print file. This option is only valid for COMMTYPE=(TCPIP,LPD).

**SESSION** = Terminate report when connection ends. This option is valid for COMMTYPE=VTAM or COMMTYPE=(TCPIP,SOK), and is ignored for COMMTYPE=(APPC,DRSPC) or COMMTYPE=(TCPIP,LPD).

**ss** = Terminate report at end of time interval where ss= the number of seconds to wait for additional data to be received. Valid values for ss are 01 – 59. This option is valid for COMMTYPE=VTAM, and is ignored for COMMTYPE=TCPIP and COMMTYPE=APPC.

**TABLE** = Terminate report using table. This option is valid for COMMTYPE=VTAM and is ignored for COMMTYPE=TCPIP and COMMTYPE=APPC. The name of the table to be used must be specified as positional parameter 2.

**Default:** **04** For COMMTYPE=VTAM, LU-0 session

**BRACKET** For COMMTYPE=VTAM, LU-1 session

**BRACKET** For COMMTYPE=VTAM, LU-3 session

**APPC** For COMMTYPE=(APPC,DRSPC)

**LPD** For COMMTYPE=(TCPIP,LPD)

**SESSION** For COMMTYPE=(TCPIP,SOK)

---

## TERMRPT in PRINTER MEMBER

**Positional parameter 2 specifies the name of the report termination table to be used if positional parameter 1 is “TABLE”.**

**Valid Values:** Any valid member name of a load module which contains a report termination table.

**Default:** None

**Example 1:** TERMRPT=05

This specification would indicate that after 5 seconds have elapsed without receiving any further data, the DRS/VPI should close the current report and make it available for printing.

**Example 2:** TERMRPT=CHAIN

This specification would indicate that after receiving data which specified that it was the last in chain, DRS/VPI should close the current report and make it available for printing.

**Example 3:** TERMRPT=BRACKET

This specification would indicate that after receiving data which specified that the bracket was ended and that the session was in a between-bracket state, DRS/VPI should close the current report and make it available for printing.

**Example 4:** TERMRPT=(TABLE,DVSSTRPT)

This specifies that DRS/VPI would use the table named DVSSTRPT to determine the report termination options.

**Note:** For COMMTYPE=(APPC,DRSPC), TERMRPT will be set to APPC, regardless of how it is specified in the virtual printer definition member.

For COMMTYPE=(TCPIP,LPD), TERMRPT will be set to LPD, regardless of how it is specified in the virtual printer definition member.

For information on creating a report termination table, see [“Creating a Report Termination Table” on page 10.6.](#)

---

## TRACE

### in PRINTER MEMBER

**TRACE=** Specifies, via two positional parameters, if tracing is to be active for this printer and the type of tracing to be performed when tracing is active.

**Positional parameter 1 specifies whether or not the DRS/VPI internal trace facility should be active for this printer. This parameter should generally be specified as Y, so that the required trace information will be available if needed for problem diagnosis. DRS/VPI internal tracing requires minimal system overhead.**

**Valid Values:** Y or N

**Default:** Y

**Positional parameter 2 is a one byte hex representation that specifies the types of printer events, in addition to those that are always traced, to be traced by the DRS/VPI. The trace type bit specifications are as follows:**

- 80 = Normal printer tracing
- 40 = Detail printer tracing
- 20 = GTF trace output and print line blocks
- 10 = User exits (printer related)
- 08 = VTAM entry/exit
- 04 = GTF trace VTAM or TCP/IP buffers
- 02 = Storage management
- 01 = GTF tracing active

**Valid Values:** 00 - FF

**Default:** 00

**Example 1:** TRACE=(Y,09)

This specification would signify to the DRS/VPI that tracing should be on for this virtual printer task, that VTAM entry and exit should be traced, and that printer trace entries should be recorded in the GTF file.

**Example 2:** TRACE=N

This specification would signify to the DRS/VPI that internal tracing of DRS Virtual Printer Interface events should not be performed for this printer.

---

## TRN in PRINTER MEMBER

**TRN=** Specifies, via three positional parameters, the transparency method to be used when processing output for this printer.

**Positional parameter 1 specifies the method of transparency to be used for this printer.**

**Valid Values:** B, C, S, or T

**B** = Buffer transparency.

Valid for COMMTYPE=VTAM, LU-1 sessions.  
Ignored for COMMTYPE=VTAM, LU-0 or LU-3 sessions. Ignored for COMMTYPE=APPC or COMMTYPE=TCPIP. Start transparency with a character and end transparency at the end of the buffer.

**C** = Character transparency.

Valid only for COMMTYPE=VTAM, LU-1 sessions.  
Ignored for COMMTYPE=VTAM, LU-0 or LU-3 sessions. Ignored for COMMTYPE=APPC or COMMTYPE=TCPIP. Start transparency with a character and end transparency after the number of characters specified in the one-byte length field immediately following the transparency character.

**S** = Start/stop transparency.

Valid only for COMMTYPE=VTAM, LU-1 sessions.  
Ignored for COMMTYPE=VTAM, LU-0 or LU-3 sessions. Ignored for COMMTYPE=APPC or COMMTYPE=TCPIP. Start transparency with a character string and end transparency when a second string is found.

**T** = Total transparency.

Valid for COMMTYPE=VTAM or COMMTYPE=TCPIP. Ignored for COMMTYPE=APPC. All data received should be treated as transparent (binary) data.

**Default:** C

**Positional parameter 2 is a 1 - 8 byte hex representation that specifies the character which will begin transparency.**

**Valid Values:** 00 - FF (for each byte)

**Default:** 35

**Positional parameter 3 is a 1 - 8 byte hex representation that specifies the character which will end transparency.**

**Valid Values:** 00 - FF (for each byte)

**Default:** None

---

## TRN in PRINTER MEMBER

- Example 1:** TRN=(C,36)  
This specification would signify to the DRS/VPI that transparency should begin when X'36' is found in the data and should continue for a length specified in the byte following the X'36'.
- Example 2:** TRN=(B,313257FF91C9C829)  
This specification would signify to the DRS/VPI that transparency should begin when X'313257FF91C9C829' is found in the data and should continue until the end of the buffer.
- Example 3:** TRN=(S,35,36)  
This specification would signify to the DRS/VPI that transparency should begin with when X'35' is found in the data and should continue until X'36' is found in the data.
- Notes:**  
If you specify the first positional parameter as 'C', the second parameter can be only one byte in length.  
If you specify the first positional parameter as 'B', the second parameter can be 1-8 bytes in length and the third parameter is not allowed.  
If you specify the first positional parameter as 'S', the second and third parameters can each be 1-8 bytes in length.  
If you specify the first positional parameter as 'T', the second and third parameters are not allowed.

---

## UCS in PRINTER MEMBER

- UCS=** Specifies from 1 to 4 UCS names that should be used when placing reports on the spool for this virtual printer.
- Valid Values:** Any 1 - 4 character valid UCS image name
- Default:** None
- Example 1:** UCS=ABCD
- This specification indicates that DRS/VPI should place reports on the spool using a UCS image name of ABCD.
- Example 2:** UCS=(ABCD,UCS5)
- This specification indicates that DRS/VPI should place reports on the spool twice; once with a UCS name of ABCD and once with a UCS name of UCS5.

---

## UDATA

### in PRINTER MEMBER

**UDATA=** Specifies 1-8 bytes of data to be associated with this virtual printer. This information will be passed to DRS/VPI user exits and separator routines to provide the ability to pass meaningful data to those routines.

**Valid Values:** Any character string from 1 to 8 bytes in length

**Default:** None

**Example:** UDATA=CICSPRTR

This specification would cause the character string 'CICSPRTR' to be passed to the DRS/VPI printer related exits. Those routines may take some action based on the value of this field.



---

## UNIT in PRINTER MEMBER

**UNIT=** Specifies, via two positional parameters, the unit type and number of the device(s) on which DRS/VPI should allocate the DASD print file. This parameter is ignored if the DSN keyword is not specified.

**Positional parameter 1 specifies the unit name of the device to be used for the DASD print file.**

**Valid Values:** Any valid DASD unit name

**Default:** SYSDA

**Positional parameter 2 specifies the number of device units to be used for the DASD print file.**

**Valid Values:** 0 - 59

**Default:** 0

**Example:** UNIT=3380

This specification indicates that DRS/VPI should allocate the DASD print file on a device with a unit type of 3380.

---

## **VOLCNT**

### **in PRINTER MEMBER**

**VOLCNT=** Specifies the volume count for volumes on which DRS/VPI should allocate the DASD print file. This parameter is ignored if the DSN keyword is not specified.

**Valid Values:** 0 - 255

**Default:** None

**Example:** VOLCNT=2

This specification indicates that DRS/VPI should use two volumes for the DASD print file.

---

## VOLUME in PRINTER MEMBER

**VOLUME=** Specifies the volume on which DRS/VPI should allocate the DASD print file. If this parameter is not specified, DRS/VPI will allocate the DASD print file on any available public or storage volume. This parameter is ignored if the DSN keyword is not specified.

**Valid Values:** Any valid DASD volume name

**Default:** None

**Example:** VOLUME=WRK101

This specification indicates that DRS/VPI should allocate the DASD print file on the volume with the name of WRK101.

**Note:** If MOD is specified as the first parameter of DISP, and a new DASD file is created, VOLUME cannot be specified. A dynamic allocation error will occur.

---

## WRITER

### in PRINTER MEMBER

- WRITER=** Specifies from 1 to 4 writer program names that should be used when placing reports on the spool for this virtual printer.
- Valid Values:** Any alphanumeric specification from one to eight characters in length
- Default:** None
- Example 1:** WRITER=MYWRITER
- This specification indicates that DRS/VPI should place reports on the spool for this virtual printer using a program writer name of MYWRITER.
- Example 2:** WRITER=(MYWRITER,DBOK)
- This specification indicates that DRS/VPI should place reports on the spool for this virtual printer twice; once using a writer name of MYWRITER and once using a writer name of DBOK.

---

## Building the Output Reference Members

One Output Reference Member can be defined for each set of OUTPUT statements that should be dynamically generated. The keyword parameters are equivalent to those specified on an OUTPUT statement in JCL. In addition, there is a comment keyword (@OUTDESC) which allows displaying up to 60 bytes of text which describe the OUTPUT statement reference.

Several OUTPUT reference members are provided in the LRS.DRS.V1R33.CNTL library on the DRS distribution cartridge. The sample OUTPUT reference members refer to PAGEDEF, FORMDEF and fonts provided with the VPS Advanced Function Printing products.

To use a particular Output Reference Member, specify the member name of that member as the value of the OUTREF keyword in a Printer Definition Member. If the Output Reference Member is modified, it is possible to reactivate the printer(s) that refer to the Output Reference Member and obtain the new characteristics.

For more information on how the Output Reference Member keyword values are used, see the section describing the OUTPUT JCL statement in the IBM Job Control Language Manual.

The keyword parameters that can be specified in an Output Reference Member are described in detail in the following pages.

---

## **@OUTDESC**

### **in OUTPUT REFERENCE MEMBER**

**@OUTDESC=** Specifies a comment to be included in a display of the Output Reference Member. The text is stored in the DROB control block which is built when the Output Reference Member is used to dynamically create the OUTPUT statement.

**Valid Values:** Any character string up to 60 bytes in length

**Default:** None

**Example:** @OUTDESC='LANDSCAPE, 1-UP, 60 LPP'

**Note:** If the value contains embedded blanks or commas, it must be enclosed in single quotes, as in the example shown above.

---

## ADDRn in OUTPUT REFERENCE MEMBER

**ADDRn=** Specifies an address field to be used for separator pages for the SYSOUT dataset, where “n” is a value from 1 to 4.

**Valid Values:** Any character string up to 60 bytes in length

**Default:** None

**Example:** ADDR1='MAIL STOP A1'  
ADDR2='TECHNICAL SUPPORT  
DEPARTMENT'  
ADDR3='2401 WEST MONROE'  
ADDR4='SPRINGFIELD, IL 62704 USA'

**Note:** If the value contains embedded blanks or commas, it must be enclosed in single quotes, as in the examples shown above.

---

## **BUILDING**

### **in OUTPUT REFERENCE MEMBER**

**BUILDING=** Specifies a building value to be used for separator pages for the SYSOUT dataset.

**Valid Values:** Any character string up to 60 bytes in length

**Default:** None

**Example:** BUILDING='BUILDING 5'

**Note:** If the value contains embedded blanks or commas, it must be enclosed in single quotes, as in the example shown above.



---

## BURST in OUTPUT REFERENCE MEMBER

**BURST=** Specifies whether the SYSOUT dataset printed on a 3800 printing subsystem is to go to the burster-trimmer-stacker, to be burst into separate sheets, or to the continuous forms stacker.

**Valid Values:** Y or N

**Default:** None

**Example:** BURST=Y

---

## CHARS

### in OUTPUT REFERENCE MEMBER

**CHARS=** Specifies the names of up to 4 character arrangement tables for printing the SYSOUT dataset.

**Valid Values:** Any 1-5 byte valid character arrangement table name

**Default:** None

**Example 1:** CHARS=K110

**Example 2:** CHARS=(GT10,GT12,GT15,GT18)

---

## CKPTLINE in OUTPUT REFERENCE MEMBER

**CKPTLINE=** Specifies the maximum number of lines on a logical page for JES to use when taking checkpoints while printing the dataset.

**Valid Values:** 0 to 32767

**Default:** None

**Example:** CKPTLINE=66

---

## **CKPTPAGE**

### **in OUTPUT REFERENCE MEMBER**

**CKPTPAGE=** Specifies the maximum number of pages to be used by JES when taking checkpoints while printing the dataset.

**Valid Values:** 1 to 32767

**Default:** None

**Example:** CKPTPAGE=10

---

## CKPTSEC in OUTPUT REFERENCE MEMBER

**CKPTSEC=** Specifies the maximum number of seconds allowed to elapse when JES is taking checkpoints while printing the dataset.

**Valid Values:** 1 to 32767

**Default:** None

**Example:** CKPTSEC=120

---

## **CLASS**

### **in OUTPUT REFERENCE MEMBER**

**CLASS=** Specifies the SYSOUT class to assign to the dataset.

**Valid Values:** Any valid SYSOUT class (A-Z, 0-9, \*)

**Default:** None

**Example:** CLASS=B

---

## COLORMAP in OUTPUT REFERENCE MEMBER

**COLORMAP=** Specifies the AFP resource for the print file which contains color translation information.

**Valid Values:** Any 1 - 8 character valid AFP resource name

**Default:** None

**Example:** COLORMAP=CMAP

---

## COMPACT

### in OUTPUT REFERENCE MEMBER

**COMPACT=** Specifies the name of a compaction table to be used by JES when sending the dataset.

**Valid Values:** Any valid compaction table name

**Default:** None

**Example:** COMPACT=TBL99



---

## COMSETUP in OUTPUT REFERENCE MEMBER

**COMSETUP=** Specifies the APF resource for the print file which contains setup information.

**Valid Values:** Any 1 - 8 character valid APF resource name

**Default:** None

**Example:** COMSETUP=CSETUP

---

## CONTROL

### in OUTPUT REFERENCE MEMBER

**CONTROL=** Specifies whether the dataset contains carriage control or if the dataset should be printed by JES with single, double or triple spacing.

**Valid Values:** PROGRAM, SINGLE, DOUBLE, or TRIPLE  
where:

- PROGRAM = Record has carriage control
- SINGLE = Force single spacing
- DOUBLE = Force double spacing
- TRIPLE = Force triple spacing

**Default:** None

**Example:** CONTROL=DOUBLE

---

## COPIES in OUTPUT REFERENCE MEMBER

**COPIES=** Specifies the number of copies to be printed.  
**Valid Values:** 1 to 255  
**Default:** None  
**Example:** COPIES=3

---

## **COPYGRP**

### **in OUTPUT REFERENCE MEMBER**

**COPYGRP=** Specifies the number of copies of each page to be printed before the next page is printed. Up to 8 values can be specified.

**Valid Values:** 1 to 255 for each subfield

**Default:** None

**Example:** COPYGRP=(4,2)

---

## DATAACK in OUTPUT REFERENCE MEMBER

**DATAACK=** Specifies whether or not print-positioning and invalid-character data check errors are to be blocked.

**Valid Values:** BLOCK, UNBLOCK, BLKCHAR, BLKPOS  
where:

- **BLOCK** = Printer is not to report any invalid-character or print-positioning errors.
- **UNBLOCK=** Printer is to report all data check exceptions.
- **BLKCHA** = Printer is not to report any invalid-character errors.
- **BLKPOS** = Printer is not to report any print-positioning errors.

**Default:** None

**Example:** DATAACK=BLKCHAR

---

## DEPT

### in OUTPUT REFERENCE MEMBER

**DEPT=** Specifies the department name to be used for separator pages.

**Valid Values:** Any character string up to 60 bytes in length

**Default:** None

**Example:** DEPT='MANUFACTURING DEPARTMENT'

**Note:** If the value contains embedded blanks or commas, it must be enclosed in single quotes, as in the example shown above.

---

## DEST in OUTPUT REFERENCE MEMBER

**DEST=** Specifies the destination to be used for SYSOUT processing.

**Valid Values:** 1 - 60 byte destination field

**Default:** None

**Example 1:** DEST=LOCAL

**Example 2:** DEST=LRS1.R1999

**Example 3:** DEST=IP:199.9.99.4

---

## **DPAGELBL**

### **in OUTPUT REFERENCE MEMBER**

**DPAGELBL=** Specifies whether the system should print the security label on each page of printed output.

**Valid Values:** Y or N

**Default:** None

**Example:** DPAGELBL=Y



---

## DUPLEX in OUTPUT REFERENCE MEMBER

**DUPLEX=** Specifies whether printing is to be done on both sides of the sheet.

**Valid Values:** NO, NORMAL, TUMBLE

**NO** = Indicates that printing is to be one only on one side of the sheet

**NORMAL** = Indicates printing on both sides of the sheet with the physical page rotated around the Y axis.

**TUMBLE** = Indicates printing on both sides of the sheet with the physical page rotated around the X axis.

**Default:** None

**Example:** DUPLEX=NORMAL

---

## **FCB**

### **in OUTPUT REFERENCE MEMBER**

**FCB=** Specifies the name of the forms control buffer (FCB) to be used to guide printing of this output.

**Valid Values:** Any valid 1-4 character FCB name

**Default:** None

**Example:** FCB=M6

---

## FLASH in OUTPUT REFERENCE MEMBER

**FLASH=** Specifies, via two positional parameters, the forms overlay and copy count.

**Positional parameter 1 specifies the name of the forms overlay to be used when printing this output.**

**Valid Values:** Any valid 1-4 character forms overlay name, NONE or STD

**Default:** None

**Positional parameter 2 specifies the number of copies on which the forms overlay should be printed.**

**Valid Values:** 0 to 255

**Default:** None

**Example:** FLASH=(LTHD,5)

---

## FORMDEF

### in OUTPUT REFERENCE MEMBER

**FORMDEF=** Specifies the name of a FORMDEF in an Advanced Function Printing FORMDEF library which should be used to print the SYSOUT dataset.

**Valid Values:** Any valid 1-6 character FORMDEF name

**Default:** None

**Example:** FORMDEF=SMPLXL

---

## FORMLEN in OUTPUT REFERENCE MEMBER

**FORMLEN=** Specifies the numeric length and unit type that will be used to change the physical paper length without reconfiguring the printer.

**Valid Values:**      nn.nnnUU (n is a digit 0-9)  
                          **UU** = Represents one of the following units:  
                          **IN** = Inches  
                          **CM** = Centimeters

**Default:**            None

**Example:**            FORMLEN=8.5IN

---

## FORMS

### in OUTPUT REFERENCE MEMBER

**FORMS=** Specifies the form name to be used when printing the SYSOUT dataset.

**Valid Values:** Any valid 1-8 character form name

**Default:** None

**Example:** FORMS=STD

---

## GROUPID in OUTPUT REFERENCE MEMBER

**GROUPID=** Specifies the name of the output reference group to which this dataset belongs.

**Valid Values:** Any valid 1-8 character name

**Default:** None

**Example:** GROUPID=OUTGRP3

---

## INDEX

### in OUTPUT REFERENCE MEMBER

**INDEX=** Specifies the left margin on a 3211 printer with the indexing feature.

**Valid Values:** 1 - 31

**Default:** None

**Example:** INDEX=5



---

## INTRAY in OUTPUT REFERENCE MEMBER

**INTRAY=** Specifies the paper source when printing AFP files.

**Valid Values:** 1 - 255

**Default:** None

**Example:** INTRAY=2

---

## LINDEX

### in OUTPUT REFERENCE MEMBER

**LINDEX=** Specifies the right margin on a 3211 printer with the indexing feature.

**Valid Values:** 1 - 31

**Default:** None

**Example:** LINDEX=7

---

## LINECT in OUTPUT REFERENCE MEMBER

**LINECT=** Specifies the maximum number of lines JES is to print on each output page.

**Valid Values:** 0 - 255

**Default:** None

**Example:** LINECT=60

---

## MODIFY

### in OUTPUT REFERENCE MEMBER

**MODIFY=** Specifies, via two positional parameters, the copy modification module name and TRC value.

**Positional parameter 1 specifies the name of the copy modification module that JES will use to print the dataset.**

**Valid Values:** 1-4 character copy modification module name

**Default:** None

**Positional parameter 2 specifies which character arrangement table name is to be used; the value is an offset into the CHARS parameters, where 0 indicates that the first CHARS subfield is to be used, 1 indicates that the second CHARS subfield is to be used, etc.**

**Valid Values:** 0 - 3

**Default:** None

**Example:** MODIFY=(MODA,2)

---

## NAME in OUTPUT REFERENCE MEMBER

**NAME=** Specifies a name to be used for separator pages for the SYSOUT dataset.

**Valid Values:** Any character string up to 60 bytes in length

**Default:** None

**Example:** NAME='MR. GEORGE WASHINGTON'

**Note:** If the value contains embedded blanks or commas, it must be enclosed in single quotes, as in the example shown above.

---

## NOTIFY

### in OUTPUT REFERENCE MEMBER

**NOTIFY=** Specifies a userid to be notified when a job completes; the user's ID can be prefixed with a valid JES node name.

**Valid Values:** 1 - 17 character userid

**Default:** None

**Example:** NOTIFY=LRS1.GWASH

---

## NOTIFY2 in OUTPUT REFERENCE MEMBER

**NOTIFY2=** Specifies the second userid to be notified when a job completes; the user's ID can be prefixed with a valid JES node name.

**Valid Values:** 1 - 17 character userid

**Default:** None

**Example:** NOTIFY2=LRS3.ALINCOLN

**Note:** If this value is specified, the NOTIFY keyword must also be specified.

---

## NOTIFY3

### in OUTPUT REFERENCE MEMBER

**NOTIFY3=** Specifies the third userid to be notified when a job completes; the user's ID can be prefixed with a valid JES node name.

**Valid Values:** 1 - 17 character userid

**Default:** None

**Example:** NOTIFY3=LRS3.SCLAUS

**Note:** If this value is specified, the NOTIFY and NOTIFY2 keywords must also be specified.



---

## NOTIFY4 in OUTPUT REFERENCE MEMBER

**NOTIFY4=** Specifies the fourth userid to be notified when a job completes; the user's ID can be prefixed with a valid JES node name.

**Valid Values:** 1 - 17 character userid

**Default:** None

**Example:** NOTIFY4=LRS3.JSMITH

**Note:** If this value is specified, the NOTIFY, NOTIFY2 and NOTIFY3 keywords must also be specified.

---

## OFFSETXB

### in OUTPUT REFERENCE MEMBER

**OFFSETXB=** Specifies the offset in the x direction from the page origin for the back side of each page of output.

**Valid Values:** mmm.nnnUU  
**m** = A digit from 0 - 9  
**n** = A digit from 0 - 9  
**UU** = Represents one of the following units:

- IN inches
- CM centimeters
- MM millimeters
- PELS
- POINTS

**Default:** None

**Example:** OFFSETXB=0.9CM

**Note:** If the UU (units) value is PELS or POINTS, the nnn value is not allowed; a whole number must be specified.

---

## OFFSETXF in OUTPUT REFERENCE MEMBER

**OFFSETXF=** Specifies the offset in the x direction from the page origin for the front side of each page of output.

**Valid Values:** mmm.nnnUU  
**m** = A digit from 0 - 9  
**n** = A digit from 0 - 9  
**UU** = Represents one of the following units:

- IN inches
- CM centimeters
- MM millimeters
- PELS
- POINTS

**Default:** None

**Example:** OFFSETXF=0.050IN

**Note:** If the UU (units) value is PELS or POINTS, the nnn value is not allowed; a whole number must be specified.

---

## OFFSETYB

### in OUTPUT REFERENCE MEMBER

**OFFSETYB=** Specifies the offset in the y direction from the page origin for the back side of each page of output.

**Valid Values:** mmm.nnnUU

**m** = A digit from 0 - 9

**n** = A digit from 0 - 9

**UU** = Represents one of the following units:

- IN inches
- CM centimeters
- MM millimeters
- PELS
- POINTS

**Default:** None

**Example:** OFFSETYB=10PELS

**Note:** If the UU (units) value is PELS or POINTS, the nnn value is not allowed; a whole number must be specified.

---

## OFFSETYF in OUTPUT REFERENCE MEMBER

**OFFSETYF=** Specifies the offset in the y direction from the page origin for the front side of each page of output.

**Valid Values:** mmm.nnnUU

- m** = A digit from 0 - 9
- n** = A digit from 0 - 9
- UU** = Represents one of the following units:
  - IN inches
  - CM centimeters
  - MM millimeters
  - PELS
  - POINTS

**Default:** None

**Example:** OFFSETYF=20POINTS

**Note:** If the UU (units) value is PELS or POINTS, the nnn value is not allowed; a whole number must be specified.

---

## **OUTBIN**

### **in OUTPUT REFERENCE MEMBER**

**OUTBIN=** Specifies the printer output bin identifier to be used for the SYSOUT dataset.

**Valid Values:** 1 to 5 decimal digits with a value from 1 to 65535

**Default:** None

**Example:** OUTBIN=2

---

## OUTDISP in OUTPUT REFERENCE MEMBER

**OUTDISP=** Specifies, via two positional parameters, the output disposition for this SYSOUT dataset.

**Positional parameter 1 specifies the normal disposition.**

**Valid Values:** WRITE, HOLD, KEEP, LEAVE, PURGE

**Default:** None

**Positional parameter 2 specifies the abnormal disposition.**

**Valid Values:** WRITE, HOLD, KEEP, LEAVE, PURGE

**Default:** None

**Example:** OUTDISP=(WRITE,PURGE)

---

## **OVERLAYB**

### **in OUTPUT REFERENCE MEMBER**

**OVERLAYB=** Specifies that the named medium overlay is to be placed on the back side of each sheet to be printed.

**Valid Values:** Any 1 - 8 character overlay name

**Default:** None

**Example:** OVERLAYB=BACK



---

## **OVERLAYF**

### **in OUTPUT REFERENCE MEMBER**

**OVERLAYF=** Specifies that the named medium overlay is to be placed on the front side of each sheet to be printed.

**Valid Values:** Any 1 - 8 character overlay name

**Default:** None

**Example:** OVERLAYF=FRONT

---

## OVFL

### in OUTPUT REFERENCE MEMBER

**OVFL=** Specifies whether or not JES3 should test for page overflow on an output printer. (JES3 only)

**Valid Values:** ON or OFF

**Default:** None

**Example:** OVFL=OFF

---

## PAGEDEF in OUTPUT REFERENCE MEMBER

**PAGEDEF=** Specifies the name of a PAGEDEF in an Advanced Function Printing PAGEDEF library which should be used to print the SYSOUT dataset.

**Valid Values:** Any valid 1-6 character PAGEDEF name

**Default:** None

**Example:** PAGEDEF=1UPL60

---

## PIMSG

### in OUTPUT REFERENCE MEMBER

**PIMSG=** Specifies, via two positional parameters, whether error messages should be printed after the dataset is printed, and an error message maximum count.

**Positional parameter 1 specifies whether error messages should be sent to the printer after the dataset has been printed.**

**Valid Values:** Y or N

**Default:** None

**Positional parameter 2 specifies the number of errors which would cause printing of the PIMSG errors to be terminated.**

**Valid Values:** 0 - 999

**Default:** None

**Example:** PIMSG=(Y,25)

---

## PORTNO in OUTPUT REFERENCE MEMBER

**PORTNO=** Specifies the TCP/IP port number at which the printing application connects to the printer.

**Valid Values:** 1 - 65535

**Default:** None

**Example:** PORTNO=515

---

## PRMODE

### in OUTPUT REFERENCE MEMBER

**PRMODE=** Specifies the process mode to be used to print the SYSOUT dataset.

**Valid Values:** LINE or PAGE or any installation-defined process mode name

**Default:** None

**Example:** PRMODE=PAGE

---

## PRTERORR in OUTPUT REFERENCE MEMBER

**PRTERORR=** Specifies how a SYSOUT dataset that has had printing terminated by a functional subsystem is to be released by JES.

**Valid Values:** DEFAULT, QUIT, HOLD

**DEFAULT** = Indicates the default functional subsystem action is to be taken.

**QUIT** = Indicates the functional subsystem should release the dataset even if a terminating error occurred during printing.

**HOLD** = Indicates the functional subsystem should request the dataset to be held on the JES spool for possible corrective action if a terminating error occurred.

**Default:** None

**Example:** PRTERORR=QUIT

---

## **PRTOPTNS**

### **in OUTPUT REFERENCE MEMBER**

**PRTOPTNS=** Specifies the named entity that contains additional print options for an IP-destined dataset that is being sent by a functional subsystem.

**Valid Values:** Any 1 - 16 character name of a printer options entity.

**Default:** None

**Example:** PRTOPTNS=OPT1



---

## PRTQUEUE in OUTPUT REFERENCE MEMBER

**PRTQUEUE=** Specifies the print queue name used when printing the IP-destined dataset.

**Valid Values:** Any 1 - 60 character name of a TCP/IP print queue

**Default:** None

**Example:** PRTQUEUE=LPT1

---

## **PRTY**

### **in OUTPUT REFERENCE MEMBER**

**PRTY=** Specifies the initial priority at which the SYSOUT dataset enters the output queue in JES.

**Valid Values:** 0 - 255

**Default:** None

**Example:** PRTY=10

---

## RESFMT in OUTPUT REFERENCE MEMBER

**RESFMT=** Specifies the resolution used to format the print dataset.

**Valid Values:** P240, P300

**P240** = Specifies 240 PELS per inch

**P300** = Specifies 300 PELS per inch

**Default:** None

**Example:** RESFMT=P240

---

## **RETAINF**

### **in OUTPUT REFERENCE MEMBER**

**RETAINF=** Specifies how long a functional subsystem will retain an IP-destined dataset after a failed transmission.

**Valid Values:** Any 1 - 10 character value supported by the FSS

**Default:** None

**Example:** RETAINF=02:30

---

## RETAINS in OUTPUT REFERENCE MEMBER

**RETAINS=** Specifies how long a functional subsystem will retain an IP-destined dataset after a successful transmission.

**Valid Values:** Any 1 - 10 character value supported by the FSS

**Default:** None

**Example:** RETAINS=01:15

---

## RETRYL

### in OUTPUT REFERENCE MEMBER

**RETRYL=** Specifies the number of attempts an FSS will try for transmission of an IP-destined dataset.

**Valid Values:** 1 -32767

**Default:** None

**Example:** RETRYL=3

---

## **RETRYT**

### **in OUTPUT REFERENCE MEMBER**

**RETRYT=** Specifies how much time a functional subsystem will wait between retries of transmission attempts of a dataset.

**Valid Values:** Any 1 - 10 characters supported by the FSS

**Default:** None

**Example:** RETRYT=00:15

---

## ROOM

### in OUTPUT REFERENCE MEMBER

**ROOM=** Specifies a room identification to be used for separator pages for the SYSOUT dataset.

**Valid Values:** Any character string up to 60 bytes in length

**Default:** None

**Example:** ROOM='LAST ROOM ON THE RIGHT'

**Note:** If the value contains embedded blanks or commas, it must be enclosed in single quotes, as in the example shown above.



---

## SYSAREA in OUTPUT REFERENCE MEMBER

**SYSAREA=** Specifies whether or not the system should reserve an area on the security label on each page of printed output.

**Valid Values:** Y or N

**Default:** None

**Example:** SYSAREA=Y

---

## TITLE

### in OUTPUT REFERENCE MEMBER

**TITLE=** Specifies a title identification to be used for separator pages for the SYSOUT dataset.

**Valid Values:** Any character string up to 60 bytes in length

**Default:** None

**Example:** TITLE='VICE-PRESIDENT OF MARKETING'

**Note:** If the value contains embedded blanks or commas, it must be enclosed in single quotes, as in the example shown above.

---

## TRC in OUTPUT REFERENCE MEMBER

**TRC=** Specifies whether or not the SYSOUT dataset should have a TRC character in the second character of each record.

**Valid Values:** Y or N

**Default:** None

**Example:** TRC=Y

---

## UCS

### in OUTPUT REFERENCE MEMBER

**UCS=** Specifies the name of a universal character set, print train, or character arrangement table.

**Valid Values:** Any valid 1-4 character UCS name

**Default:** None

**Example:** UCS=PN

---

## USRDTAnn in OUTPUT REFERENCE MEMBER

**USRDTAnn=** Specifies a user data field to be used for this dataset, where “nn” is a value from 01 to 16.

**Valid Values:** Any character string up to 60 bytes in length

**Default:** None

**Example 1:** USRDTA01='MY DATA'

**Example 2:** USRDTA14='SOME MORE DATA'

**Note:** If the value contains embedded blanks or commas, it must be enclosed in single quotes, as in the example shown above.

---

## USRLIBnn

### in OUTPUT REFERENCE MEMBER

**USRLIBnn=** Specifies the name of a user library to be used when printing the SYSOUT dataset, where “nn” is a value from 01 to 08.

**Valid Values:** Any valid 1-44 character dataset name

**Default:** None

**Example 1:** USRLIB01=LRS.VPS.R70.PDEFLIB

**Example 2:** USRLIB05=SYS1.SAMPLE.PDEFLIB

---

## WRITER in OUTPUT REFERENCE MEMBER

**WRITER=** Specifies the writer name to be associated with the SYSOUT dataset.

**Valid Values:** Any valid 1-8 character writer name

**Default:** None

**Example:** WRITER=MYWTR





---

## Section 4

# DRS/VPI With VTAM Applications

DRS/VPI can intercept output from VTAM online applications such as CICS, IMS, IDMS, etc. DRS/VPI supports printer session types of LU-0, LU-1 or LU-3. For LU-0 and LU-3, the Data Stream Compatibility (DSC) buffer format is supported. For LU-1, the SNA Character Set (SCS) buffer format and Intelligent Printer Data Stream (IPDS) format are supported.

The member name of the virtual printer must match the name of an APPL definition which is active to VTAM. (See [“VTAM Definition Requirements” on page 3.5](#)). When DRS/VPI is started, the virtual printer members are read from the DRSVLIB PDS. DRS/VPI opens the ACB that matches the member name, and notifies VTAM that it is ready to receive connection requests for that name.

When the online application needs to print, it will start a session with the virtual printer to send the print data. The network name used by the online application must match the member name of the virtual printer. For example, the CICS or IMS printer definition would contain the member name of the virtual printer as the VTAM network name.

As the print data is received by DRS/VPI, the print buffers are transformed into print lines. The reports created by DRS/VPI are either placed on the JES spool or in a DASD file. If the reports go to the JES spool, DRS/VPI uses attributes from SYSOUT keywords. If the report is placed in a DASD file, DRS/VPI uses attributes from DASD keywords. Up to four SYSOUT files and one DASD file can be created for each print request.

If the report is placed on the JES spool, any system which obtains output from the JES spool can access the report. For example, the report could be printed by JES, VPS, or PSF, or it could be acquired by a report distribution and archival product. Because OUTPUT statement information can be associated with the print data, PAGEDEF, FORMDEF, CHARS, etc., could be used for special formatting.

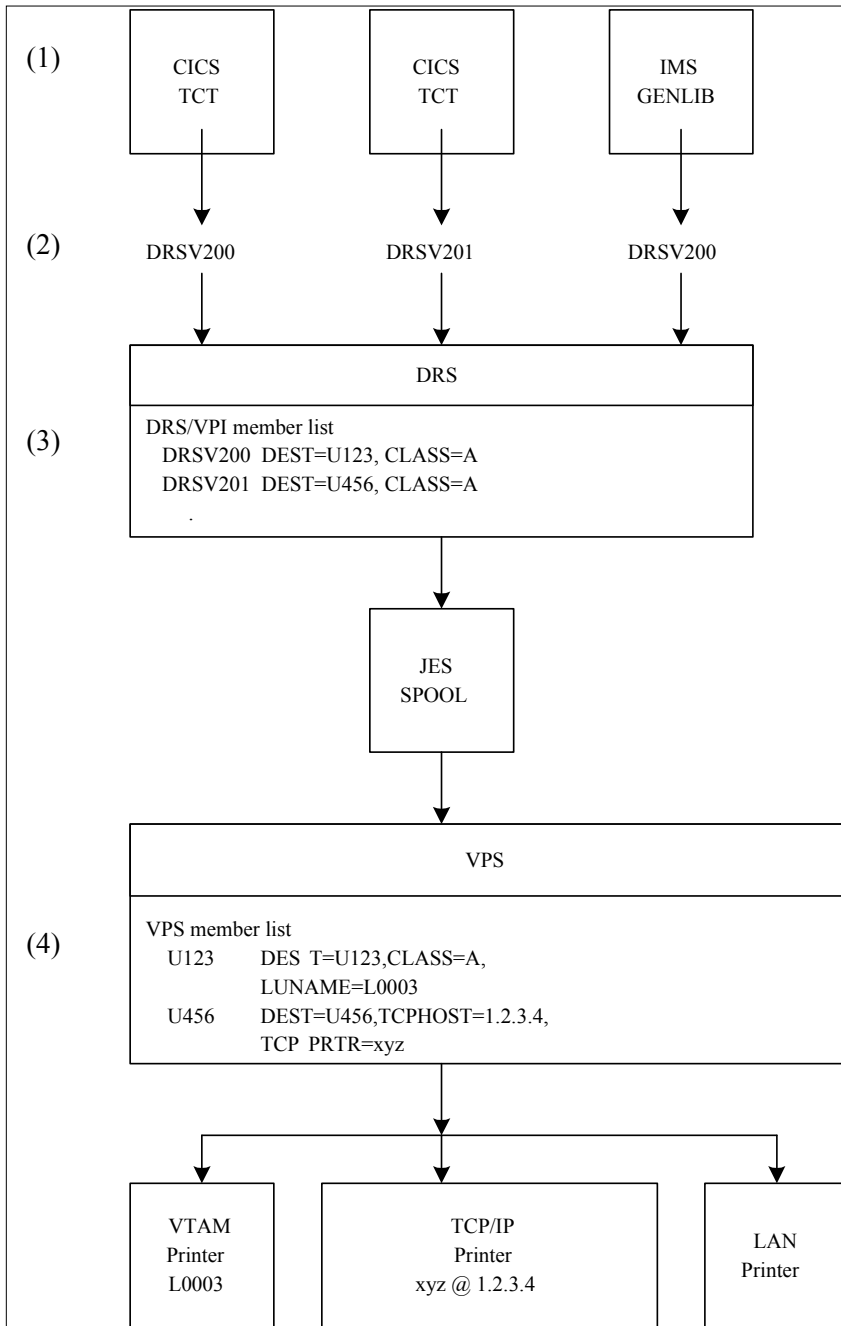
An example of a printer definition member for a VTAM virtual printer is shown here:

```
* -----*
* MEMBER NAME = VPRT001 *
* -----*
CLASS=A,                SYSOUT CLASS
COMMTYPE=VTAM,          VTAM DEFINITION
DEST=R987,              DESTINATION
FORM=STD                 FORM NAME
```

This example indicates that the communication type is VTAM, which is the default. The output received for the VPRT001 printer will be placed on the JES spool with destination of R987, SYSOUT class of A and form name of STD. For more information on printer definition keywords, see [“Building the Printer Definition Members” on page 3.42](#).

In a typical online environment, a single online application owns a VTAM printer at any point in time. An application wanting to “acquire” a printer must wait until it is “released” by the application currently using the printer. Operational problems, such as forms jams, out of forms, or power off, can cause online transaction delays or abends. Also, this environment can typically route only to devices that are defined to VTAM. DRS/VPI supplies the functions needed to improve the online environment. The next diagram describes how the originating application regions are changed to map printer names to “virtual” printers rather than “real” printers.

- Applications (1) are mapped to point to the LU names of “virtual” VTAM printers (2) rather than “real” VTAM printers.
- When DRS/VPI receives data for a particular virtual printer, it creates a dataset with the attributes found in DRS/VPI’s printer definition for that printer. The resulting dataset is placed on the JES spool. (3)
- The VPS address space requests SYSOUT datasets from the JES spool for printers defined in VPS’ printer list. As the datasets are selected from the JES spool, they can be sent to VTAM printers, TCP/IP printers, or LAN printers. (4)



---

## Section 5

# DRS/VPI With DRS/PC

The DRS Virtual Printer Interface allows receiving print data from DRS/PC. DRS/PC obtains print files from users in a LAN environment and sends those print files to DRS/VPI. Installation steps for DRS/PC are listed in the DRS/PC manual.

In order to activate a DRS/PC virtual printer definition, the KEYDRSPC keyword in the DRS/VPI system initialization member must specify a valid trial or license key for the DRS/PC product. Any virtual printer definition which is to be used with DRS/PC must specify COMMTYPE=(APPC,DRSPC).

DRS/PC connects to DRS/VPI using a VTAM APPC (LU6.2) connection. The member name of the virtual printer which will receive from DRS/PC must match the name of an APPL definition which is active to VTAM. The APPL statement should specify a DLOGMOD and MODETAB which contain LU6.2 parameters. The following example shows the logon mode table parameters which are recommended:

```
LRSMODE  MODETAB
LRSAAPP  MODEENT LOGMODE=LRSAAPP,                X
          TYPE=0,          NEGOTIABLE BIND        X
          FMPROF=X'13',    REQUIRED FOR LU6         X
          TSPROF=X'07',    REQUIRED FOR LU6         X
          PRIPROT=X'B0',   REQUIRED FOR LU6         X
          SECPROT=X'B0',   REQUIRED FOR LU6         X
          COMPROT=X'D0B1', REQUIRED FOR LU6         X
          RUSIZES=X'8787', 1024/1024              X
          SSNDPAC=X'3F'    NON-ZERO FORCES USE OF VPACING X
          PSNDPAC=X'3F'    NOT USED BY DRS/PC       X
          SRCVPAC=X'3F'    NOT USED BY DRS/PC       X
          PSERVIC=X'0602000000000000000000000000'
          MODEEND
          END
```

The VTAM APPL statement which describes the virtual printer must include keywords which indicate that it is capable of supporting LU6.2 sessions. The following keywords should be specified:

```
MODETAB=lrsmode,    Logon mode table name
DLOGMOD=lrsappc,    Logon mode entry which contains LU6.2 parms
APPC=YES,           Allow DRS to issue APPCCMD macros
VPACING=n,          Non-zero pacing value
PARSESS=YES,        Must be YES, even though session limit is 1
DMINWNL=1,          DRS will win if session contention
DMINWNR=0,          DRS/PC will lose if session contention
DSESLIM=1           Session limit of 1
```

For more information on defining the VTAM APPL statements to be used for the virtual printers, see [“VTAM Definition Requirements” on page 3.5](#).

When DRS/PC sends output to DRS/VPI, a special record is sent before the data buffers. This special record describes whether the print file should be placed on the JES spool or in a DASD file. If the file is destined for the JES spool, the SYSOUT attributes are sent, such as CLASS, DEST, FORM, WRITER. If the print data should be stored in a DASD file, DASD keyword values are sent, such as DSN, DISP, SPACE, UNIT, VOLUME. If allocation fails using the attributes passed in the header record, DRS/VPI will use the SYSOUT and/or DASD keywords specified in the DRS virtual printer definition. Up to four SYSOUT files and one DASD file can be created for each print request.

If the report is placed on the JES spool, any system which obtains output from the JES spool can access the report. For example, the report could be printed by JES, VPS or PSF, or it could be acquired by a report distribution and archival product. Because OUTPUT statement information can be associated with the print data, PAGEDEF, FORMDEF, CHARS, etc., could be used for special formatting.

An example of a virtual printer definition member for a DRS/PC printer is shown here:

```

* - - - - - *
*  MEMBER NAME = DRSPC01 *
* - - - - - *
CLASS=A,                SYSOUT CLASS
COMMTYPE=(APPC,DRSPC),  APPC DEFINITION FOR DRSPC
DEST=LOCAL,             DESTINATION
HOLD=YES,               PLACE IN HOLD QUEUE
MPP=0                   NO MAXIMUM LINE LENGTH

```

This example indicates the communication type is APPC. The second subfield specifies that the sender is DRS/PC, which is the default if the first subfield is APPC. The output received for the DRSPC01 printer will be processed using the information received from DRS/PC header records. If that allocation fails, an informational message will displayed, and the print file will placed on the JES spool with destination of LOCAL, SYSOUT class of A and will be held. MPP=0 allows lines up to 32756 to be received.

For more information on printer definition keywords, see [“Building the Printer Definition Members” on page 3.42.](#)

---

## Receiving AFP records from the LAN using DRS/PC

AFP file re-blocking allows applications running on the LAN to create Advanced Function Printing resources or documents which can be printed to devices attached to the MVS system.

If an application or an AFP driver on the LAN creates a block of records which contain AFP structures, the blocks must be separated into individual records in order for them to be processed or printed on MVS. DRS/VPI can separate the records in these AFP blocks and place the records on the JES spool or into a DASD file. Then the AFP records can be used as resources (PAGEDEF, FORMDEF, OVERLAY, etc.) or the documents can be printed by VPS/IPDS, VPS/PCL or other AFP software.

DRS/PC can be used to send the AFP blocks to DRS/VPI on MVS. There are two methods to indicate to DRS/VPI to re-block these incoming AFP buffers of data. One is to specify PRTOPTS=0800 in the virtual printer definition for DRS/PC. The other is to define a print queue as one which would send AFP records by setting the AFPDS=Y flag in the .DRS initialization file for DRS/PC. For more information on setting the AFPDS flag, see the DRS/PC installation manual. For more information on specifying PRTOPTS, see [“Building the Printer Definition Members” on page 3.42](#).

If either method is used, files which contain X'5A' in the first position will be processed as AFP records. All records must begin with X'5A' and contain a length field at the proper location. Any AFP format errors will cause processing to be terminated and the printer EDRAINED.



---

## Section 6

# DRS/VPI With DRS/TCPIP

The DRS Virtual Printer Interface allows receiving output from TCP/IP hosts using LPR/LPD protocols. DRS/TCPIP connects to the MVS TCP/IP system and listens for connections on the PORT number specified in the TCPPOINT keyword. The TCPPOINT defaults to be 515, which is the standard 'well-known PORT' for a Line Printer Daemon.

When a connection is received from the remote TCP/IP host, the first command expected is the 'receive print job' command. This command contains a print queue name. The name must match the member name of the virtual printer. Therefore, it is restricted to a maximum length of 8 characters and should begin with an alphabetic or national character. No special characters are allowed.

The connection is then passed to the virtual printer that matches the print queue name. If no member is active with that name, DRS/VPI will use the AUTOACT keyword values to accept or reject the connection. If AUTOACT=(N,N) was specified, DRS/VPI will refuse the connection. If AUTOACT=(Y,N) was specified, DRS/VPI will attempt to activate a member with the name specified. If none exists, the connection will be rejected. If AUTOACT=(Y,Y) was specified and no matching printer definition is found, DRS/VPI will define a printer using the print queue name as the member name and keyword values from the default member specified at DRS/VPI system initialization.

DRS/VPI will receive a control file and a data file from the TCP/IP host. The control file has records which identify the originator of the print file and records which indicate how the file should be printed. The data file is ASCII print data. DRS/VPI can use or ignore the control file based on the TCPOPTS virtual printer keyword. The print data can be translated from ASCII to EBCDIC based on the PRTXLATE keyword.

The reports created by DRS/VPI are either placed on the JES spool or in a DASD file. If the reports go to the JES spool, DRS/VPI uses attributes from SYSOUT keywords, such as: CLASS, DEST, FORM, WRITER, UCS, FCB, OUTREF, etc. If the report is placed in a DASD file, the attributes of the file use DASD keywords, such as: DSN, MEMBER, DISP, SPACE, UNIT, VOLUME, etc. Up to four SYSOUT files and one DASD file can be created for each print request.

If the report is placed on the JES spool, any system which obtains output from the JES spool can access the report. For example, the report could be printed by JES, VPS or PSF or it could be acquired by a report distribution and archival product. Because OUTPUT statement information can be associated with the print data, PAGEDEF, FORMDEF, CHARS, etc., could be used for special formatting.

---

In order to activate a TCP/IP virtual printer, the KEYLPD keyword in the DRS/VPI system initialization member must specify a valid trial or license key for the DRS/TCPIP product. Any virtual printer which is to be used with DRS/TCPIP must specify COMMTYPE=TCPIP. If dynamic definition is specified by setting AUTOACT=(Y,Y), the DEFLTMEM keyword in the system initialization member must specify a default printer definition member which includes a keyword value of COMMTYPE=TCPIP.

An example of a virtual printer definition member for a TCP/IP virtual printer is shown here:

```
* -----*
* MEMBER NAME = U9988 *
* -----*
CLASS=A,                SYSOUT CLASS
COMMTYPE=(TCPIP,LPD),   TCP/IP LPD DEFINITION
DEST=U9988,             DESTINATION
PRTXLATE=(Y,DVSSXEBC)  ASCII TO EBCDIC TRANSLATION
```

This example indicates that the communication type is TCP/IP. The second subfield of COMMTYPE indicates LPR/LPD protocol, which is the default if the first subfield is TCPIP. The print data received for this print queue will be translated from ASCII to EBCDIC using the DVSSXEBC translate table module and will be placed on the JES SPOOL with destination of U9988 and SYSOUT class of A.

For more information on using DRS/TCPIP see [“Introduction to DRS/TCPIP”](#) on page 12.1.



---

## Receiving AFP Records using DRS/TCPIP

AFP file re-blocking allows applications running on a TCP/IP host to create Advanced Function Printing resources or documents which can be printed to devices attached to the MVS system.

If an application or an AFP driver on a TCP/IP host creates a block of records which contain AFP structures, the blocks must be separated into individual records in order for them to be processed or printed on MVS. DRS/VPI can separate the records in these AFP blocks and place the records on the JES spool or into a DASD file. Then the AFP records can be used as resources (PAGEDEF, FORMDEF, OVERLAY, etc.) or the documents can be printed by VPS/IPDS, VPS/PCL or other AFP software.

To indicate that DRS/VPI should re-block these incoming AFP buffers of data, specify PRTROPTS=0800 in the virtual printer definition for the print queue. For more information on using PRTROPTS, see [“Building the Printer Definition Members” on page 3.42](#).

With PRTROPTS=0800, files which contain X'5A' in the first position will be processed as AFP records. All records must begin with X'5A' and contain a length field at the proper location. Any AFP format errors will cause processing to be terminated and the printer EDRAINED.



---

## Section 7

# DRS/VPI With DRS/STI

The DRS Virtual Printer Interface allows receiving output containing DRS Smart Tag Interface (DRS/STI) buffers. The DRS/STI buffers are used to override the SYSOUT characteristics associated with the virtual printer. This allows a single virtual printer to route output to the JES spool with different SYSOUT attributes.

DRS/VPI scans each buffer received, looking for the DRS/STI buffer. Note that the scan is only performed if a valid KEYSTI= keyword was specified during DRS/VPI initialization. Within the Smart Tag buffer is information that is used to extract the report's SYSOUT characteristics from the DRS/STI Smart Tag rules database. The rules database is a VSAM KSDS dataset containing information on where and how a report is to be printed.

Upon detecting a Smart Tag buffer, DRS/VPI will close and unallocate the current report, regardless of the TERMRPT= keyword option, and immediately allocate and open a new report with the SYSOUT characteristics retrieved from the Smart Tag rules database. After the new report has been successfully initialized, DRS/VPI discards the Smart Tag buffer and proceeds with normal processing.

If a Smart Tag buffer is received and a valid KEYSTI= keyword was not specified during DRS/VPI initialization, then DRS/VPI will treat the Smart Tag buffer as data and simply write it to the currently allocated report. No error message will be generated.



---

# Section 8

## DRS/VPI Operation

### Introduction

There are many situations in which an operator or person at the “help” desk may wish to communicate with the DRS/VPI system. For example, an operator in the computer room may wish to start or stop the DRS/VPI task. “Help” desk personnel may wish to inquire about the status of a DRS virtual printer or to diagnose a problem with a particular virtual printer.

These functions can be accomplished via the OS/MVS START, STOP and MODIFY commands.

### The START, STOP, and MODIFY Commands

The START (abbreviated as S) command is used only for starting the DRS/VPI system. Its format is:

```
S stcname
```

where stcname is the name of the DRS/VPI started task. The START command is not used if the DRS/VPI system is executed as a batch job.

The STOP (abbreviated as P) command is used only for stopping the DRS/VPI address space. Its format is:

```
P stcname
```

where stcname is the name of the DRS/VPI started task (or the jobname of the DRS/VPI batch job). A STOP command is used for DRS/VPI “fast” termination. SYSOUT datasets that are currently being received, if any, will be terminated, and kept or deleted according to the ERRACTN specified in the virtual printer definition.

---

The MODIFY (abbreviated as F) command can be used for all other DRS/VPI operator communications. Its format is:

**F stcname,cmd{,prtrid}{,parml,parm2,parmn}**

where:

- stcname:** The name of the DRS/VPI started task (all examples in this section assume a started task name of DRSV).
- cmd:** The DRS/VPI command name (only sufficient characters to make the command unique need be specified).
- prtrid:** The name of the virtual printer(s) to which the command applies; “prtrid” is one of the following:
- The full member name of the virtual printer in DRSVLIB.
  - A generic or partial member name (for example, ABC\*).
  - S=status where status is:
    - A = For active printers (those currently busy doing work).
    - I = For idle printers (those waiting for work).
    - F = For virtual printer tasks whose sessions have been terminated by the FORCE command.
    - E = For error drained printers (those that are drained due to an unrecoverable error).
  - SYSTEM can be used in place of prtrid to indicate that the command applies to the DRS/VPI system as a whole rather than an individual printer. The commands which allow SYSTEM to be specified are:
    - DISPLAY**
    - SNAP**
  - OUTREF can be used in place of prtrid to indicate that the command applies to OUTPUT reference members. To display all OUTREF members, specify “\*” for the next display parameter. To display a specific or a group of OUTREF members, use the full or partial name of the OUTREF member(s).
- parms:** Consist of variable data, depending upon the requested command.

**Examples of using the prtrid in DRS/VPI commands are listed below.**

F DRSV,DIS,*	Displays all activated printers.
F DRSV,DIS,ABC*	Displays all printers whose names begin with ‘ABC’.
F DRSV,DIS,SYSTEM	Displays current DRS/VPI system status.
F DRSV,DIS,OUTREF,*	Displays a list of OUTPUT reference members.
F DRSV,DIS,ABCD,ATTR	Displays report attributes for printer ‘ABCD’.
F DRSV,DIS,ABCD,OUT	Displays OUTPUT references for printer ‘ABCD’.
F DRSV,DIS,S=E	Displays all printers which are EDRAINED.
F DRSV,ACT,XYZ	Activates a virtual printer named ‘XYZ’.
F DRSV,FOR,ABCD	Forces session termination and drains printer ‘ABCD’.
F DRSV,REA,ABCD	Inactivates and re-activates virtual printer ‘ABCD’.

For more information about each command, see the following pages.

---

## DRS/VPI Command Summary

COMMAND NAME	FUNCTION
<u>ABEND</u>	Abnormally terminate DRS/VPI.
<u>ACTIVATE</u>	Add a printer to the DRS/VPI system.
<u>CLOSELOG</u>	Close and re-open the DRS/VPI log dataset.
<u>DISPLAY</u>	Display DRS/VPI status and option information.
<u>END</u>	Normally terminate DRS/VPI.
<u>FORCE</u>	Force termination of the session for specified printer.
<u>INACTIVATE</u>	Remove a printer from the DRS/VPI system.
<u>LOG</u>	Put a message in the DRS/VPI log dataset.
<u>REACTIVATE</u>	Remove a printer from the DRS/VPI system and immediately add the printer to the DRS/VPI system.
<u>SET</u>	Change current DRS/VPI printer parameters.
<u>SNAP</u>	SNAP dump DRS/VPI control blocks.
<u>SSET</u>	Change current DRS/VPI system parameters.
<u>START</u>	Start (remove drained status from) specified printer.

---

# DRS/VPI Commands

## ABEND

**Function:** Abnormally terminate DRS/VPI.  
**Description:** This command will abnormally terminate the DRS Virtual Printer Interface with a U001 abend.  
**Format:** **F DRSV,ABEND**  
**Comments:** This command has no operands or prtrid specification.  
**Example:** **F DRSV,ABEND**  
Abnormally terminate DRS/VPI.

## ACTIVATE

**Function:** Add a printer to the DRS/VPI System.  
**Description:** This command allows the addition of a printer to the DRS/VPI System without having to terminate and restart the DRS/VPI address space.  
**Format:** **F DRSV,ACTIVATE,membernm**  
**Comments:** membernm is the DRSVLIB PDS member name defining the printer to be added.  
**Example:** **F DRSV,ACTIVATE,DRSPRT99**  
Add the printer that is described in the DRSVLIB member DRSPRT99 to the DRS/VPI System.

## CLOSELOG

**Function:** Close and re-open the DRS/VPI log dataset.  
**Description:** This command provides the facility to close and re-open the DRS/VPI log dataset. If the DRS/VPI log dataset is a pre-allocated DASD dataset, the CLOSELOG command will be rejected. If the DRS/VPI log dataset is a SYSOUT dataset, the log dataset will be available for printing, and a new SYSOUT dataset will be allocated and opened for continued logging.  
**Format:** **F DRSV,CLOSELOG**  
**Comments:** This command has no operands or prtrid specification.  
**Example:** **F DRSV,CLOSELOG**  
Close the DRS/VPI log dataset.



---

## DISPLAY

- Function:** Display DRS/VPI status and option information.
- Description:** This command provides the facility to display option and status information pertinent to the individual virtual printers and/or the DRS VPI system.
- Format:** **F DRSV,DISPLAY,SYSTEM**  
Display the DRS/VPI system-wide options and status.
- F DRSV,DISPLAY,EXITS**  
Display status of DRS user exits (this can be coded as EXITS to see the status of all user exits, as EXITnn to display the status of an individual user exit, or as EXITnn-nn to display the status of a range of user exits.)
- F DRSV,DISPLAY,OUTREF,\***  
Display all OUTPUT reference members currently in use.
- F DRSV,DISPLAY,OUTREF,O99\***  
Display all OUTPUT reference members currently in use with names that begin with "O99".
- F DRSV,DISPLAY,prtrid,disopt1{,disopt2.....,disoptn}**  
Display options in a particular printer member, where disoptn specifies the requested display option(s).  
Only sufficient characters to make the option unique need be specified.
- |                          |                                      |
|--------------------------|--------------------------------------|
| <b><u>A</u>TTRIBUTES</b> | Printer spooling or DASD attributes  |
| <b><u>D</u>IAG</b>       | Printer diagnostic information       |
| <b><u>M</u>ISC</b>       | Printer miscellaneous options        |
| <b><u>O</u>UTREF</b>     | Printer OUTPUT reference information |
| <b><u>P</u>ROCESSING</b> | Printer processing statistics        |
| <b><u>Q</u>UEUE</b>      | Printer staging information          |
| <b><u>S</u>TATUS</b>     | Printer status                       |
| <b><u>T</u>CP/IP</b>     | TCP/IP related information           |
| <b><u>V</u>TAM</b>       | VTAM related information             |
| *                        | All the above except DIAG and OUTREF |
- Comments:** STATUS is the default display option for a printer display if no options are specified.
- Example:** **F DRSV,DISPLAY,DRSPRT99,S,V**  
Display status information and VTAM information for printer DRSPRT99.

---

## **END**

- Function:** Normally terminate DRS/VPI.
- Description:** This command will notify DRS/VPI to initiate normal termination processing.
- Format:** **F DRSV,END**
- Comments:** This command has no operands or prtrid specification. The DRS/VPI system will not terminate until all printers that are currently busy complete the report that they are processing.
- Example:** **F DRSV,END**  
Normally terminate DRS/VPI.

## **FORCE**

- Function:** Terminate a virtual printer session immediately.
- Description:** This command will terminate the session between a virtual printer and its partner application. The virtual printer will be placed in a "FORCED" status, and a START command will be required to allow new sessions to be established.
- Format:** **F DRSV,FORCE,prtrid**
- Comments:** This command can be used to stop a virtual printer immediately.
- Example:** **F DRSV,FORCE,DRSPRT99**  
Terminate the session for DRSPRT99 and place the virtual printer in a "FORCED" status.

## **INACTIVATE**

- Function:** Remove a printer from the DRS/VPI System.
- Description:** This command allows the removal of a printer from the DRS/VPI System without having to terminate and restart the DRS/VPI address space. If the printer is currently busy, the inactivation will be scheduled and completed after the current job being printed is complete.
- Format:** **F DRSV,INACTIVATE,prtrid**
- Comments:** Printer parameters can be dynamically changed without terminating the DRS/VPI address space by changing the parameters in the printer definition member, issuing an inactivate command for the printer, and issuing an activate command for the printer. At activation, the values specified in the printer definition member are brought into the DRS/VPI system.
- Example:** **F DRSV,INACTIVATE,DRSPRT99**  
Remove printer with a printer id of the DRSPRT99 from the DRS/VPI System.

---

## LOG

- Function:** Put a message in the DRS/VPI log dataset.
- Description:** This command allows a message to be placed in the DRS/VPI log dataset.
- Format:** **F DRSV,LOG,message**
- Comments:** The message must not be greater than 50 characters.
- Example:** **F DRSV,LOG,THIS IS A MESSAGE**  
Place "THIS IS A MESSAGE" in the DRS/VPI log dataset.

## REACTIVATE

- Function:** Remove a printer from the DRS/VPI System and immediately add the printer to the DRS/VPI system.
- Description:** This command allows the definition of a DRS virtual printer to be "refreshed" from the Printer Definition Member in the DRS/VPI Control Library.
- Format:** **F DRSV,REACTIVATE,prtrid**
- Comments:** The most common use for this command would be to bring into the DRS/VPI an updated printer definition member for a printer already activated to the DRS/VPI.
- Example:** **F DRSV,REACTIVATE,DRSPRT99**  
Remove the printer with a printer id of DRSPRT99 from the DRS/VPI system and immediately add it to the DRS/VPI system, using the printer definition in the DRS/VPI Control Library.

## SET

- Function:** Change current printer options.
- Description:** This command provides the facility to change printer options.
- Format:** **F DRSV,SET,prtrid,optn=(value1,value2)**  
**optn** = The option to be changed  
**value** = The new value to be assigned to the option
- Comments:** The SET command must be issued for a specific printer; it provides the facility to change the following DRS/VPI printer parameters:  
TRACE (2 positional subparameters)  
To retain the original value for a particular positional parameter value, use a dash. To remove a value, use a null.
- Example 1:** **F DRSV,SET,DRSPRT99,TRACE=(Y,FF)**  
Activate internal tracing for printer DRSPRT99 and to set the trace options to X'FF'.
- Example 2:** **F DRSV,SET,DRSPRT11,TRACE=N**  
Inactivate internal tracing for printer DRSPRT11.
- Example 3:** **F DRSV,SET,DRSPRT11,TRACE=(-,03)**  
Set the trace options to X'03'.

---

## SNAP

- Function:** SNAP dump DRS/VPI control blocks.
- Description:** This command provides the facility to snap dump DRS/VPI blocks for a specified printer or for the DRS/VPI system.
- Option 1:** **F DRSV,SNAP,prtrid**
- Comments:** Command must be issued for a specific virtual printer.  
The SNAP command has no parameters.  
The output of the SNAP command will be queued to the class, destination, form, and writer specified in the DRSSTART member via the SNAPOUTP parameter.
- Example:** **F DRSV,SNAP,DRSPRT99**  
Snap the control blocks for printer DRSPRT99.
- Option 2:** **F DRSV,SNAP,SYSTEM**  
Snap the control blocks for the DRS Virtual Printer Interface address space to obtain system diagnostic information.
- Note:** Use the SYSTEM option only by request of the DRS technical support staff to diagnose a particular problem. This option consumes a large amount of CPU time.

## SSET

- Function:** Change current DRS/VPI system parameters.
- Description:** This command provides the facility to modify the DRS/VPI System parameters.
- Format:** **F DRSV,SSET,sopt=value**  
**optn** = The option to be changed  
**value** = The new value to be assigned to the option
- Comments:** The SSET command provides the facility to change the following the DRS/VPI System parameters:
- EXITnn=E|D** To enable (E) or disable (D) DRS/VPI user exit nn. The exit number can be specified as a 1 or 2 digit number; i.e., EXIT01 and EXIT1 are both acceptable.
- LOG=Y|N** To enable (Y) or disable (N) DRS/VPI logging.
- MAXVTAM=nnnn** To specify the maximum number of VTAM virtual printers that can be concurrently active. This value can be set from 0 to the maximum VTAM printers specified at DRS/VPI initialization.
- MAXTCPIP=nnnn** To specify the maximum number of TCP/IP virtual printers that can be concurrently active. This value can be set from 0 to the maximum TCP/IP printers specified at DRS/VPI initialization.
- TRTYPES=** To specify the types of DRS/VPI system events to be traced.
- Example:** **F DRSV,SSET,LOG=N**  
Inactivate the DRS/VPI logging.

---

## **START**

- Function:** Start (remove drained status from) specified printer.
- Description:** This command provides the ability to notify a printer to allow session establishment.
- Format:** **F DRSV,START,prtrid**
- Comments:** The START command will be rejected if DRS/VPI system termination is in process.
- Example:** **F DRSV,START,DRSPRT99**  
Notify printer DRSPRT99 that SYSOUT processing can resume.



---

## Section 9

# DRS/VPI User Exits

This section describes the DRS/VPI modules which you can modify to fit the needs of your installation or to activate particular functions of DRS/VPI.

The DRS/VPI Separator Routine is described first. Although it is not really an “exit,” it can be modified to suit your needs, so it is included in this section. Several versions of the separator (banner) page are supplied to you on the distribution cartridge in both source and load module format. You can also create your own separator modules.

The User Exits allow you to write specialized code for your installation which will be invoked at certain key points in DRS/VPI processing. The distribution cartridge will contain sample source code for these exits.

---

## Separator Routine

**Function:** The DRS/VPI separator routine is responsible for adding separator (banner) pages at the start and end of each print file. Several sample separator routines are supplied on the distribution cartridge, or you can write your own separator routine. The separator routines should be linked into the DRS/VPI load module library, along with the other DRS/VPI modules and user exits. The SEPAR= printer keyword determines the name of the load module that will be called to create separator pages for a virtual printer; each virtual printer can have its own separator page routine.

**Supplied Separator Routines:** Six separator routines (both source and load modules) are supplied with DRS/VPI. They are:

Source Name	Load Name	Description	Width
DVSSSEPR	DVSSSEPR	Block letters - standard	132
DVSSSEP2	DVSSSEP2	Block letters - standard	80
DVSSSEP3	DVSSSEP3	DRS/PC separator	80
DVSSSEP4	DVSSSEP4	TCP/IP separator	132
DVSSSEP5	DVSSSEP5	TCP/IP separator	132
DVSSSEP6	DVSSSEP6	TCP/IP separator to print control file	132

**Table 9.1 - Supplied Separator Routines**

**Language:** The separator routine must be written in Assembler Language.

**Reentrancy Considerations:** The separator page routine must be reentrant.

**Execution Environment:** The separator routine is given control in problem state, storage protect key 8. The \$SYSAUTH flag in the System Attributes will indicate if DRS/VPI is authorized.

**Linkage Conventions:** When the separator page routine receives control, register 15 contains the entry point of the separator page routine, register 14 contains the address to which the separator page routine must return control, and register 13 points to an 18-word save area. When a separator page routine receives control, it must save the caller's registers according to standard MVS linkage conventions.

When the separator page routine returns control, it must restore the caller's registers, except register 15, which must be set to an appropriate return code.

**Point of Processing:** If the virtual printer keyword SEPAR=S or SEPAR=B is specified, the separator routine is given control after the INIT call has been made to initialize the report, but before the first PUT call to add data lines to the report. If SEPAR=N or SEPAR=E is specified, the separator routine will not be given control for start dataset processing.

If the virtual printer keyword SEPAR=E or SEPAR=B is specified, the separator routine is given control before the TERM call terminates the report. If SEPAR=N or SEPAR=S is specified, the separator routine will not be given control for end dataset processing.



---

**Recovery:**

- If the separator page routine abends, the virtual printer will be EDRAINED.

**Register Contents on Entry:**

<b>R0</b>	Request code 00 - Call is for a start separator 04 - Call is for an end separator 08 - Call is for a cleanup request (The separator routine will be called with this request code only if an event has occurred that will prevent further calls to the separator routine)
<b>R1</b>	Address of parameter list in the following format: Word 0 (+00) Address of DRS/VPI System Attributes (DRSVSYAT) Word 1 (+04) Address of 2-Word User Work Area Word 2 (+08) Address of Printer Attributes (DRSVPRAT) Word 3 (+12) Address of Spooling Attributes (DRSDRIB) Word 4 (+16) Address of DRS/PC Lan Attributes (DRSLANA), if available Word 5 (+20) Address of TCP/IP Control Area (#LRTCCNT), if available Word 6 (+24) Address of area for separator record The high-order bit of the last parameter will be set to 1.
<b>R2-R12</b>	N/A
<b>R13</b>	Address of register save area
<b>R14</b>	Return Address
<b>R15</b>	Entry Address

**Register Contents on Exit:**

<b>R0-R14</b>	Same as upon entry
<b>R15</b>	Return code

**Return Codes:**

<b>00</b>	Separator record available to print and return control
<b>04</b>	Nothing to print and do not return control
<b>08</b>	Reserved
<b>12</b>	Reserved
<b>16</b>	Disastrous error, nothing to print and do not return control

---

### **Programming Considerations:**

1. If the separator routine is being called for a cleanup request (request code = 08), no more separator lines can be added. The only purpose of this type of call is to give the separator routine an opportunity to do any necessary cleanup processing. For example, if your separator routine had issued a GETMAIN during start separator processing and intended to issue a FREEMAIN during end separator processing, the FREEMAIN would need to be issued when the cleanup call is made.
2. All print lines added by the separator routine should be in VBA format. All lines will be re-formatted by DRS/VPI to match the record format specified for the print data file.
  - The separator line should have a 4-byte RDW. The first 2 bytes of the RDW should contain the length of the print record, including the 4-byte RDW. The second 2 bytes of the RDW should be zero.
  - The first byte of the separator line should contain an ASA carriage control character.
3. If the separator is being called for a start or end request (i.e., other than a cleanup request), and you wish to add separator records to the report:
  - Move the separator print line to the area whose address is provided in the parameter list.
  - Set the return code to 00 to indicate there is a print line to be added to the report.
  - When all lines have been added, set return code to 04 to indicate that the separator routine does not need to be called again.
4. If the separator is being called for a start or end request (i.e., other than a cleanup request), and you do not wish to add any separator records to the report, set the return code to 04 on the initial call and exit.

---

## DRS/VPI Exits

The following user exits are supplied with DRS/VPI and can be installed as required:

1. **DRS/VPI Startup Exit:**  
Receives control during DRS/VPI initialization. When it is called, DRS/VPI has already allocated the virtual printer control blocks and the trace table and has loaded all the user exits.
2. **DRS/VPI Shutdown Exit:**  
Receives control after DRS/VPI termination processing is complete.
3. **DRS/VPI Activation Exit:**  
Receives control prior to activating a virtual printer to the DRS/VPI system.
4. **DRS/VPI Inactivation Exit:**  
Receives control prior to inactivating a virtual printer (removing it from the DRS/VPI system).
5. **DRS/VPI Command Exit:**  
Receives control prior to the execution of every DRS/VPI command. The exit can modify the command text or request that the command not be executed.
6. **DRS/VPI WTO Exit:**  
Receives control prior to the issuing of any DRS/VPI message. The exit can modify the WTO or request that the WTO not be issued.
7. **DRS/VPI Report Initialization Exit:**  
Receives control before the DRS INIT call for each dataset. The exit can modify any parameter in the SYSOUT attributes (DRIB) for that dataset.
8. **DRS/VPI Error Retry Exit**  
Receives control after any error that would EDRAIN the virtual printer. The exit can decide to restart the printer, leave the printer EDRAINED, or inactivate the printer.
9. **DRS/VPI TCP/IP Command Exit**  
Receives control as print commands and print files are received from the TCP/IP remote host. The exit can modify the control file received or reject the print data file.
10. **DRS/VPI Report Termination Exit**  
Receives control after the DRS TERM call for each dataset.

The source to the sample user exits is distributed as members in library LRS.DRS.V1R33.ASM on the distribution cartridge. Sample JCL to assemble and linkedit the user exits is provided as member EXITASMV in file LRS.DRS.V1R33.CNTL.

The following pages give more information about using the DRS/VPI user exits.

---

## Exit 01: DRS/VPI Startup Exit

**Function:** This exit is called during DRS/VPI initialization. When it is called, DRS/VPI has already allocated its printer control blocks and trace table and has loaded the user exits. After this exit is invoked, DRS/VPI will complete initialization, including activating any virtual printers listed for automatic activation via an Inclusion List Member or Exclusion List Member and initializing the DRS/VPI VTAM connection (OPEN ACB and SETLOGON START) for each active printer.

**Exit Type:** System  
**Exit TCB:** System  
**Point of Processing:** This exit is taken during DRS/VPI initialization, before any printers have been activated.  
**Related Exits:** N/A

### Recovery:

- If recovery is off and this exit abends, the DRS/VPI address space will be terminated.
- If recovery is on and this exit abends, the exit will automatically be disabled and a storage dump will be taken.

Since the DRS/VPI address space will be terminated if this exit abends and recovery is off, it is recommended that recovery always be on for this exit.

### Register Contents on Entry:

**R0** Request Code (always 00)  
**R1** Address of parameter list in the following format:  
Word 0 (+00) Address of DRS/VPI System Attributes (DRSVSYAT)  
The high-order bit of the last parameter will be set to 1.  
**R2-R12** N/A  
**R13** Address of register save area  
**R14** Return Address  
**R15** Entry Address

### Register Contents on Exit:

**R0-R14** Same as upon entry  
**R15** Return code

### Return Codes:

**00** Must be zero

### Exit Dependent Trace Data:

**On Entry** User words (SYAUSRWD) from System Attribute DSECT (DRSVSYAT)  
**On Exit** User words (SYAUSRWD) from System Attribute DSECT (DRSVSYAT)

---

**Programming Considerations:**

1. This exit can store information in the user words of the System Attribute DSECT (DRSVSYAT). This information would then be available to all other DRS/VPI exits invoked after this point in processing. If the DRS/VPI WTO Exit is in effect (User Exit 06), the WTO Exit would be invoked before this exit for any DRS/VPI messages issued during earlier stages of initialization.

**Sample Program:** Member DRS DVSSUE01 in the DRS distribution file LRS.DRS.V1R33.ASM is an example of the DRS/VPI Startup Exit. This example merely issues a message via WTO which includes the DRS/VPI job name.

---

## Exit 02: DRS/VPI Shutdown Exit

<b>Function:</b>	This exit is called after DRS/VPI termination processing has been completed.
<b>Exit Type:</b>	System
<b>Exit TCB:</b>	System
<b>Point of Processing:</b>	This exit is taken after DRS/VPI termination processing has been completed.
<b>Related Exits:</b>	N/A

### Recovery:

- If recovery is off and this exit abends, the DRS/VPI address space will be terminated.
- If recovery is on and this exit abends, the exit will automatically be disabled and a storage dump will be taken.

Since the DRS/VPI address space will be terminated if this exit abends and recovery is off, it is recommended that recovery always be on for this exit.

### Register Contents on Entry:

<b>R0</b>	Request Code (always 00)
<b>R1</b>	Address of parameter list in the following format: Word 0 (+00) Address of DRS/VPI System Attributes (DRSVSYAT) The high-order bit of the last parameter will be set to 1.
<b>R2-R12</b>	N/A
<b>R13</b>	Address of register save area
<b>R14</b>	Return Address
<b>R15</b>	Entry Address

### Register Contents on Exit:

<b>R0-R14</b>	Same as upon entry
<b>R15</b>	R15 Return code

### Return Codes:

<b>00</b>	Must be zero
-----------	--------------

### Exit Dependent Trace Data:

<b>On Entry</b>	User words (SYAUSRWD) from System Attribute DSECT (DRSVSYAT)
<b>On Exit</b>	User words (SYAUSRWD) from System Attribute DSECT (DRSVSYAT)

### Programming Considerations:

<b>Sample Program:</b>	Member DVSSUE02 in the DRS distribution file LRS.DRS.V1R33.ASM is an example of the DRS/VPI Shutdown Exit. This example issues a message via WTO which includes the DRS/VPI job name, the highest number of active printers, the maximum virtual storage allocated, and the total number of lines and datasets printed.
------------------------	---

---

## Exit 03: DRS/VPI Activate Exit

<b>Function:</b>	This exit is called prior to activating a printer to the DRS/VPI System.
<b>Exit Type:</b>	Printer
<b>Exit TCB:</b>	System
<b>Point of Processing:</b>	This exit is taken prior to activating a virtual printer to the DRS/VPI System.
<b>Related Exits:</b>	N/A

### Recovery:

- If recovery is off and this exit abends, the DRS/VPI address space will be terminated.
- If recovery is on and this exit abends, the exit will automatically be disabled and a storage dump will be taken.

Since the DRS/VPI address space will be terminated if this exit abends and recovery is off, it is recommended that recovery always be on for this exit.

### Register Contents on Entry:

<b>R0</b>	Request Code 00 - Activate request at DRS/VPI Startup 04 - Activate request by Command
<b>R1</b>	Address of parameter list in the following format: Word 0 (+00) Address of DRS/VPI System Attributes (DRSVSYAT) Word 1 (+04) Address of 2-Word User Work Area Word 2 (+08) Address of Printer Attributes (DRSVPRAT) The high-order bit of the last parameter will be set to 1.
<b>R2-R12</b>	N/A
<b>R13</b>	Address of register save area
<b>R14</b>	Return Address
<b>R15</b>	Entry Address

### Register Contents on Exit:

<b>R0-R14</b>	Same as upon entry
<b>R15</b>	Return code

### Return Codes:

<b>00</b>	Must be zero
-----------	--------------

### Exit Dependent Trace Data:

<b>On Entry</b>	8-Byte - Printer member name 2-Byte - Printer sequence number 1-Byte - Printer transparency flag 4-Byte - Printer option flags
<b>On Exit</b>	8-Byte - Printer member name (LUNAME) 2-Byte - Printer sequence number 1-Byte - Printer transparency flag 4-Byte - Printer option flags

---

**Programming Considerations:**

**Sample Program:** Member DVSSUE03 in the DRS distribution file LRS.DRS.V1R33.ASM is an example of the DRS/VPI Activate Exit. This example issues a WTO which includes the DRS/VPI job name, the printer member name, and spooling attributes defined for the printer (CLASS, DEST, USERID, FORM and WRITER).



---

## Exit 04: DRS/VPI Inactivate Exit

<b>Function:</b>	This exit is called prior to inactivating a virtual printer (removing it from the DRS/VPI System).
<b>Exit Type:</b>	Printer
<b>Exit TCB:</b>	System
<b>Point of Processing:</b>	This exit is taken prior to inactivating a virtual printer (removing it from the DRS/VPI System).
<b>Related Exits:</b>	N/A

### Recovery:

- If recovery is off and this exit abends, the DRS/VPI address space will be terminated.
- If recovery is on and this exit abends, the exit will automatically be disabled and a storage dump will be taken.

Since the DRS/VPI address space will be terminated if this exit abends and recovery is off, it is recommended that recovery always be on for this exit.

### Register Contents on Entry:

<b>R0</b>	Request Code (always 00)
<b>R1</b>	Address of parameter list in the following format: Word 0 (+00) Address of DRS/VPI System Attributes (DRSVSYAT) Word 1 (+04) Address of 2-Word User Work Area Word 2 (+08) Address of Printer Attributes (DRSVPRAT) The high-order bit of the last parameter will be set to 1.
<b>R2-R12</b>	N/A
<b>R13</b>	Address of register save area
<b>R14</b>	Return Address
<b>R15</b>	Entry Address

### Register Contents on Exit:

<b>R0-R14</b>	Same as upon entry
<b>R15</b>	Return code

### Return Codes:

<b>00</b>	Must be zero
-----------	--------------

### Exit Dependent Trace Data:

<b>On Entry</b>	8-Byte - Printer member name 2-Byte - Printer sequence number 1-Byte - Printer transparency flag 4-Byte - Printer option flags
<b>On Exit</b>	8-Byte - Printer member name 2-Byte - Printer sequence number 1-Byte - Printer transparency flag 4-Byte - Printer option flags

---

**Programming Considerations:**

**Sample Program:** Member DVSSUE04 in the DRS distribution file LRS.DRS.V1R33.ASM is an example of the DRS/VPI Inactivate Exit. This example issues a WTO which includes the DRS/VPI job name, the printer member name, and the number of lines and number of datasets received and spooled.

---

## Exit 05: DRS/VPI Command Exit

<b>Function:</b>	This exit is called prior to processing each command sent to DRS/VPI. It can modify the command, or it can request that the command be ignored.
<b>Exit Type:</b>	System
<b>Exit TCB:</b>	System
<b>Point of Processing:</b>	This exit is taken prior to processing of every command that is being sent to DRS/VPI.
<b>Related Exits:</b>	N/A

### Recovery:

- If recovery is off and this exit abends, the DRS/VPI address space will be terminated.
- If recovery is on and this exit abends, the exit will automatically be disabled and a storage dump will be taken.

Since the DRS/VPI address space will be terminated if this exit abends and recovery is off, it is recommended that recovery always be on for this exit.

### Register Contents on Entry:

<b>R0</b>	Request Code (always 00)
<b>R1</b>	Address of parameter list in the following format: Word 0 (+00) Address of DRS/VPI System Attributes (DRSVSYAT) Word 1 (+04) Address of Command Buffer (see Programming Considerations for format) The high-order bit of the last parameter will be set to 1.
<b>R2-R12</b>	N/A
<b>R13</b>	Address of register save area
<b>R14</b>	Return Address
<b>R15</b>	Entry Address

### Register Contents on Exit:

<b>R0-R14</b>	Same as upon entry
<b>R15</b>	Return code

### Return Codes:

<b>00</b>	Process the command
<b>04</b>	Ignore the command

### Exit Dependent Trace Data:

<b>On Entry</b>	First 16 bytes of the command buffer
<b>On Exit</b>	First 16 bytes of the command buffer

---

**Programming Considerations:**

1. The second word of the input parameter list will contain the address of the command buffer for all commands except "P DRSV". If the command is "P DRSV", the second word will contain binary zeroes, except that the high-order (X'80') bit will be set.
2. The command buffer format is:
  - 2 byte length (does not include length of length field itself).
  - 120 bytes command text
3. Both the command buffer length and data fields can be altered.

**Sample Program:** Member DVSSUE05 in the DRS distribution file LRS.DRS.V1R33.ASM is an example of the Command Exit. This example indicates that DRS/VPI should ignore a DRS/VPI ABEND command.

---

## Exit 06: DRS/VPI WTO Exit

<b>Function:</b>	This exit is called prior to the issuing of all WTOs. The exit can modify the WTO, or it can request that the WTO not be issued.
<b>Exit Type:</b>	System
<b>Exit TCB:</b>	System
<b>Point of Processing:</b>	This exit is taken prior to issuing every WTO.
<b>Related Exits:</b>	N/A

### Recovery:

- If recovery is off and this exit abends, the DRS/VPI address space will be terminated.
- If recovery is on and this exit abends, the exit will automatically be disabled and a storage dump will be taken.

Since the DRS/VPI address space will be terminated if this exit abends and recovery is off, it is recommended that recovery always be on for this exit.

### Register Contents on Entry:

<b>R0</b>	Request Code (always 00)
<b>R1</b>	Address of parameter list in the following format: Word 0 (+00) Address of DRS/VPI System Attributes (DRSVSYAT) Word 1 (+04) Address of WTO Attributes (DRSVWTAT) The high-order bit of the last parameter will be set to 1.
<b>R2-R12</b>	N/A
<b>R13</b>	Address of register save area
<b>R14</b>	Return Address
<b>R15</b>	Entry Address

### Register Contents on Exit:

<b>R0-R14</b>	Same as upon entry
<b>R15</b>	Return code

### Return Codes:

<b>00</b>	Proceed with the WTO.
<b>04</b>	Do not issue the WTO.
<b>08</b>	Do not issue the WTO and do not log the WTO.

### Exit Dependent Trace Data:

<b>On Entry</b>	First 16 bytes of the WTO DSECT (DRSVWTAT format)
<b>On Exit</b>	First 16 bytes of the WTO DSECT (DRSVWTAT format)

### Programming Considerations:

1. The WTO attributes can be altered.
2. The WTO text and length can be altered.

**Sample Program:** Member DVSSUE06 in the DRS distribution file LRS.DRS.V1R33.ASM is an example of the WTO Exit.

This example changes the attribute of the initialization message (DRSV000N) from normal to action.

---

## Exit 07: DRS/VPI Report Initialization Exit

<b>Function:</b>	This exit is called prior to calling DRS for the INIT call to create a new SYSOUT dataset on the JES spool. The exit can modify any of the SYSOUT attributes (stored in DRSDRIB control block) before the dataset is initialized.
<b>Exit Type:</b>	Printer
<b>Exit TCB:</b>	Printer
<b>Point of Processing:</b>	This exit is taken prior to initializing each report.
<b>Related Exits:</b>	N/A

### Recovery:

- If recovery is off and this exit abends, the DRS virtual printer will be EDRAINED.
- If recovery is on and this exit abends, the exit will automatically be disabled and a storage dump will be taken.

### Register Contents on Entry:

<b>R0</b>	Request Code (always 00)
<b>R1</b>	Address of parameter list in the following format: Word 0 (+00) Address of DRS/VPI System Attributes (DRSVSYAT) Word 1 (+04) Address of 2-word user work area Word 2 (+08) Address of Printer Attributes (DRSVPRAT) Word 3 (+12) Address of Spooling Attributes (DRSDRIB) Word 4 (+16) Address of DRS/PC LAN attributes (DRSLANA) Word 5 (+20) Address of TCP/IP Control Area (#LRTCCNT) Word 6 (+24) Address of first print record The high-order bit of the last parameter will be set to 1.
<b>R2-R12</b>	N/A
<b>R13</b>	Address of register save area
<b>R14</b>	Return Address
<b>R15</b>	Entry Address

### Register Contents on Exit:

<b>R0-R14</b>	Same as upon entry
<b>R15</b>	Return code

### Return Codes:

<b>00</b>	Only return code expected. Any other return code will be ignored.
-----------	---

---

**Exit Dependent Trace Data:**

- On Entry** 8 bytes - Printer member name  
2 bytes - Printer sequence number  
1 bytes - Printer transparency option  
4 bytes - Printer options flags
- On Exit** 8 bytes - Printer member name  
2 bytes - Printer sequence number  
1 bytes - Printer transparency option  
4 bytes - Printer options flags

**Programming Considerations:**

1. Any spooling attributes located in the DRSDRIB control block can be updated by the exit.
2. The exit can decide to change the spooling attributes based on the current session information (partner application, for example), printer attributes, or the text in the first record of the report.
3. The first print record of the report is preceded by a 4 byte RDW. The first 2 bytes of the RDW contain the length of the print record, including the RDW. The second 2 bytes of the RDW are X'0000'.
4. The text of the first line of the report should not be altered. However, if the text is altered, the LENGTH of the text **MUST NOT** be changed.
5. The parameter list for DRS/VPI Exit 07 has been changed from that used in DRS release 3.1. The order of the parameters passed to the exit was changed in DRS release 3.2, and additional parameters are now being passed. Please review any versions of this exit you may have modified to accommodate these changes.

**Sample Program:** Member DVSSUE07 in the DRS distribution file LRS.DRS.V1R33.ASM is an example of the Report Initialization Exit.

This example changes the class associated with the SYSOUT to "V" if the partner application for the printer has a VTAM APPL name which begins with the characters "VPS". In addition, the exit issues a WTO to show the CLASS, DEST, FORM and WRITER associated with the current report as well as the virtual printer member name.

---

## Exit 08: DRS/VPI Error Retry Exit

<b>Function:</b>	This exit is called after an error causes the DRS virtual printer to be EDRAINED. The exit can allow the printer to remain EDRAINED, attempt to restart the printer after the error, or inactivate the printer.
<b>Exit Type:</b>	Printer
<b>Exit TCB:</b>	System
<b>Point of Processing:</b>	This exit is called after a session is terminated for a printer error that caused the printer to be EDRAINED.
<b>Related Exits:</b>	N/A

### Recovery:

- If recovery is off and this exit abends, the DRS/VPI address space will be terminated.
- If recovery is on and this exit abends, the exit will automatically be disabled and a storage dump will be taken.

Since the DRS/VPI address space will be terminated if this exit abends and recovery is off, it is recommended that recovery always be on for this exit.

### Register Contents on Entry:

<b>R0</b>	Request Code (always 00)
<b>R1</b>	Address of parameter list in the following format: Word 0 (+00) Address of DRS/VPI System Attributes (DRSVSYAT) Word 1 (+04) Address of 2-word user work area Word 2 (+08) Address of Printer Attributes (DRSVPRAT) Word 3 (+12) Address of Printer Error Information (DRSVPRER) The high-order bit of the last parameter will be set to 1.
<b>R2-R12</b>	N/A
<b>R13</b>	Address of register save area
<b>R14</b>	Return Address
<b>R15</b>	Entry Address

### Register Contents on Exit:

<b>R0-R14</b>	Same as upon entry
<b>R15</b>	Return code

### Return Codes:

<b>00</b>	Retry the error.
<b>04</b>	Don't retry the error.
<b>08</b>	No more retries allowed.
<b>12</b>	Inactivate the printer.

### Exit Dependent Trace Data:

<b>On Entry</b>	16 bytes of error information
<b>On Exit</b>	16 bytes of error information



---

**Programming Considerations:**

1. Since this exit runs under a major DRS/VPI system TCB, it should not perform any function which would suspend the TCB.
2. The error retry count field (PRECOUNT) of the Printer Error Information DSECT (DRSVPRER) is incremented by 1 each time this exit returns to DRS/VPI with a return code of 00. The count is set to zero after DRS/VPI has successfully created a print file. Therefore, the PRECOUNT field is the total number of errors that have occurred for the file currently being processed.

**Sample Program:** Member DVSSUE08 in the DRS distribution file LRS.DRS.V1R33.ASM is an example of the Error Retry Exit.

This example contains a table of possible error codes that could cause the virtual printer to be EDRAINED. For all error codes, the exit sets return code to 04 to cause the printer to remain EDRAINED. For any error code that retries should be allowed, the table can be changed to enable the retry. A maximum retry count can be set in the error table.

---

## Exit 09: DRS/VPI TCP/IP Command Exit

<b>Function:</b>	This exit is called when TCP/IP LPR/LPD commands are received from the TCP/IP remote host. The exit can modify the control file received or reject the print data file.
<b>Exit Type:</b>	Printer
<b>Exit TCB:</b>	Printer
<b>Point of Processing:</b>	This exit is called at four times in the TCP/IP LPR/LPD processing. Each time, a request code is used to indicate which command or file was received from the remote host. The exit can deny access to the print queue at any of these points.
<b>Related Exits:</b>	N/A

### Recovery:

- If recovery is off and this exit abends, the DRS virtual printer will be EDRAINED.
- If recovery is on and this exit abends, the exit will automatically be disabled and a storage dump will be taken.

### Register Contents on Entry:

<b>R0</b>	Request Code 00 - RECEIVE PRINT JOB subcommand 04 - RECEIVE CONTROL FILE subcommand 08 - RECEIVE DATA FILE subcommand 12 - Validate control file
<b>R1</b>	Address of parameter list in the following format: Word 0 (+00) Address of DRS/VPI System Attributes (DRSVSYAT) - for all request codes Word 1 (+04) Address of 2-word user work area - for request codes 04, 08, 12 Word 2 (+08) Address of Printer Attributes (DRSVPRAT) - for request codes 04, 08, 12 Word 3 (+12) Address of Spooling Attributes (DRSDRIB) - for request codes 04, 08, 12 Word 4 (+16) Address of TCP/IP Control Area (#LRTCCNT) - for all request codes Word 5 (+20) Address of Smart Tag Control Area (DRSVSTCA) - for request codes 04, 08, 12 Word 6 (+24) Address of print queue name - for all request codes The high-order bit of the last parameter will be set to 1.
<b>R2-R12</b>	N/A
<b>R13</b>	Address of register save area
<b>R14</b>	Return Address
<b>R15</b>	Entry Address

---

**Register Contents on Exit:**

**R0-R14** Same as upon entry  
**R15** Return code

**Return Codes:**

**00** Continue processing print file.  
**04** Send negative acknowledgement and deny print file.  
**08** Processing error - continue processing print file.

**Exit Dependent Trace Data:**

**On Entry** 8 bytes - Printer member name  
2 bytes - Printer sequence number  
1 byte - Printer transparency option  
4 bytes - Printer options flags  
**On Exit** 8 bytes - Printer member name  
2 bytes - Printer sequence number  
1 byte - Printer transparency option  
4 bytes - Printer options flags

**Programming Considerations:**

1. This exit can modify the print queue name. The print queue name received in the "RECEIVE PRINT JOB" subcommand is placed in a 32-byte work area and passed to this exit with request code 00. All characters of the name will be translated to uppercase by DRS. The first eight bytes of the name returned to DRS will be used as the virtual printer member name.
2. This exit can modify the control file to change how DRS/TCPIP processes the print data file. For example, by setting a binary filter, translation and line control processing could be avoided.
3. This exit can reject the print file based on any combination of the following criteria:
  - Virtual Printer characteristics: member name, UDATA, etc.
  - SYSOUT characteristics: CLASS, DEST, FORM, WRITER, etc.
  - DASD characteristics: DSN, MEMBER, DISP, VOLUME, etc.
  - Source information: IP address, host name, user name, etc.
  - Size of data file or control file.

**Sample Program:** Member DVSSUE09 in the DRS distribution file LRS.DRS.V1R33.ASM is an example of the TCP/IP Command Exit.

This example compares a table of remote TCP/IP host names and MVS spool destinations and denies access to certain print destinations, based on the source TCP/IP system.

---

## Exit 10: DRS/VPI Report Termination Exit

**Function:** This exit is called after the DRS TERM call to terminate a print file.  
**Exit Type:** Printer  
**Exit TCB:** Printer  
**Point of Processing:** This exit is called after terminating each report.  
**Related Exits:** N/A

### Recovery:

- If recovery is off and this exit abends, the DRS virtual printer will be EDRAINED.
- If recovery is on and this exit abends, the exit will automatically be disabled and a storage dump will be taken.

### Register Contents on Entry:

**R0** Request Code (always 00)  
**R1** Address of parameter list in the following format:  
Word 0 (+00) Address of DRS/VPI System Attributes (DRSVSYAT)  
Word 1 (+04) Address of 2-word user work area  
Word 2 (+08) Address of Printer Attributes (DRSVPRAT)  
Word 3 (+12) Address of Spooling Attributes (DRSDRIB)  
Word 4 (+16) Address of DRS/PC LAN Attributes (DRSLANA)  
Word 5 (+20) Address of TCP/IP Control Area (#LRTCCNT)  
Word 6 (+24) Address of DRS Termination Block (DRSDRTB)  
The high-order bit of the last parameter will be set to 1.  
**R2-R12** N/A  
**R13** Address of register save area  
**R14** Return Address  
**R15** Entry Address

### Register Contents on Exit:

**R0-R14** Same as upon entry  
**R15** Return code

### Return Codes:

**00** Only return code expected. Any other return code will be ignored.

---

**Exit Dependent Trace Data:**

- On Entry** 8 bytes - Printer member name  
2 bytes - Printer sequence number  
1 byte - Printer transparency option  
4 bytes - Printer options flags
- On Exit** 8 bytes - Printer member name  
2 bytes - Printer sequence number  
1 byte - Printer transparency option  
4 bytes - Printer options flags

**Programming Considerations:**

1. This exit can issue a command to notify another address space that a print file has been delivered. If SVC34 is used to issue a command, the DRS/VPI modules, including this exit, must be loaded from an authorized load library.

**Sample Program:** Member DVSSUE10 in the DRS distribution file  
LRS.DRS.V1R33.ASM is an example of the Report Termination Exit.

This example issues SVC34 to send a "POST" command to notify VPS that a print file has been created on the JES spool. The POST command will cause VPS to search for work immediately. Because it is necessary to be authorized to issue SVC34, the exit tests a flag in the System Attributes to see if DRS/VPI is authorized.



---

# Section 10

## Special Topics

### Introduction

Because this section covers a wide range of topics, all of it may not be applicable to your environment.

We will not attempt to cover error conditions here. That information is included in [“Messages and Codes” on page 35.1](#).

As always, feel free to call DRS Support any time you need information about DRS. We would also be interested in your suggestions for topics to be added to this section.

---

# Creating a User Translation Table

## Overview

DRS/VPI allows for the translation of both single-byte character sets (SBCS) and double-byte character sets (DBCS) in print lines.

The character mappings used for translation are contained in translation look-up tables. These tables are specified using the printer keyword `PRTXLATE` for SBCS and `PRTXDBCS` for DBCS. The translation tables are in two formats, one for SBCS and one for DBCS. The user can compile these tables using macros supplied in `LRS.DRS.V1R33.MACLIB`, and several examples have been supplied in `LRS.DRS.V1R33.ASM`.

## Definitions

A “single-byte character set” (SBCS) uses one byte to encode each character. EBCDIC and ASCII are both single-byte character sets.

A “double-byte character set” (DBCS) uses two bytes to encode each character. Double-byte character sets are necessary for those languages such as Japanese requiring more than the 256 code points (distinguishable characters) a single-byte character set can hold.

“Shift-out” (SO) and “shift-in” (SI) character codes signal the beginning and end of a run of DBCS data within a mixed SBCS/DBCS dataset. SO has the value of `X'0E'`; SI has the value of `X'0F'`.

## Translation of Single Byte Character Sets

The default for SBCS translation is `PRTXLATE=N` which indicates that DRS/VPI should not translate any incoming SBCS data. An example of a useful SBCS translation is the removal of data characters not printable on a given printer. If the first positional parameter of the `PRTXLATE` keyword is “Y”, then DRS/VPI will perform SBCS translation.

The translation table which DRS/VPI uses for the SBCS translation process is identified in the second positional parameter value of the `PRTXLATE` keyword. The default translation table name is `DRSSXLTE`. You receive the source version of this translation table on the distribution cartridge as member `DVSSXLTE` in file `LRS.DRS.V1R33.ASM` and the load module version as member `DVSSXLTE` in file `LRS.DRS.V1R33.LOAD`.

To put your translation table into effect for a printer, enter its load module name as the second positional parameter of the `PRTXLATE` keyword. For example, if you linked the load module with a member name of `MYTABLE`, you would code `PRTXLATE=(Y,MYTABLE)` in the printer definition for each printer which should use that SBCS translation table.



---

## Translation of Double Byte Character Sets

DBCS translation tables are specified using the PRTXDBCS printer keyword, which takes five positional parameters.

The first positional parameter indicates whether any DBCS translation at all should be performed for the specified printer. The default is “N”, specifying no translation.

The second positional parameter identifies the DBCS translate table. Unlike the case for SBCS, there is no default DBCS translate table, and you must specify it explicitly in the PRTXDBCS parameter list. To put your translation table into effect for a printer, enter its load module name as the second positional parameter value of the PRTXDBCS keyword.

The third through fifth positional parameters specify the treatment of SOSI codes during translation of mixed SBCS/DBCS datasets. One of four possible behaviors can be specified using the third positional parameter. If this parameter value is “KEEP”, the SOSI codes are passed through unchanged. If the value is “DELT”, the SOSI codes are deleted. If the value is “REPL”, the codes are replaced with new values as follows: the SO and SI codes are replaced with the fourth and fifth positional parameter values respectively. If the value is “ADD”, the SOSI codes are added using the fourth and fifth positional parameters.

For example, to specify that a printer should translate DBCS data using the translate table DBTABLE, and that the SOSI codes in mixed SBCS/DBCS datasets should be passed through unchanged, you would code PRTXDBCS=(Y,DBTABLE,KEEP) in the printer definition for each printer which should use the translation table.

## Creating Translation Tables

When you have created the source code for your translation table, assemble it and link it into the same library as the other DRS/VPI load modules. *Be sure to link it as reentrant.*

There is one additional consideration when you create your own load module. Do not give it a name identical to any DRS load module which is already in existence. For Version 1 Release 3.3 of DRS, you should be very careful not to name one of your own load modules with a name that starts with “DV33”, “DVSS”, “DRSS” or “DPSS”. If you do so, you run the risk of replacing one of DRS’ primary load modules with your own module, and results will be unpredictable.

---

## Creating a SBCS Translation Table

The source code for the SBCS translation table (minus comments) is reproduced here:

```
DVSSXLTE  CSECT
          DC  X'40404040404040404040404040404040'  (HEX00-0F)
          DC  X'40404040404040404040404040404040'  (HEX10-1F)
          DC  X'40404040404040404040404040404040'  (HEX20-2F)
          DC  X'40404040404040404040404040404040'  (HEX30-3F)
          DC  X'40404040404040404040404A4B4C4D4E4F'  (HEX40-4F)
          DC  X'5040404040404040404040405A5B5C5D5E5F'  (HEX50-5F)
          DC  X'6061404040404040404040406A6B6C6D6E6F'  (HEX60-6F)
          DC  X'4040404040404040404040797A7B7C7D7E7F'  (HEX70-7F)
          DC  X'4081828384858687888990404040404040'  (HEX80-8F)
          DC  X'4091929394959697989990404040404040'  (HEX90-9F)
          DC  X'40A1A2A3A4A5A6A7A8A940404040404040'  (HEXA0-AF)
          DC  X'40404040404040404040404040404040'  (HEXB0-BF)
          DC  X'C0C1C2C3C4C5C6C7C8C940404040404040'  (HEXC0-CF)
          DC  X'D0D1D2D3D4D5D6D7D8D940404040404040'  (HEXD0-DF)
          DC  X'E040E2E3E4E5E6E7E8E940404040404040'  (HEXE0-EF)
          DC  X'F0F1F2F3F4F5F6F7F8F940404040404040'  (HEXF0-FF)
          END
```

The translation table consists of 256 values, each of which represents a character which should be substituted for a character of the equivalent hexadecimal value. For example, if translation table DVSSXLTE shown above is used, print line characters with a value from X'00' through X'49' (among others) will be translated to X'40' by DRS.

A translation table for translating ASCII to EBCDIC is supplied as member DVSSXEBC in file LRS.DRS.V1R33.ASM and as member DVSSXEBC in file LRS.DRS.V1R33.LOAD. This translation table is normally used for printers controlled by DRS/TCPIP.

---

## Creating a DBCS Translation Table

DBCS translation tables are built using the macro LRDBCS in LRS.R10.MACLIB. The LRDBCS macro is written as follows:

```
LRDBCS TYPE=INITIAL,  
      DESC='table description',  
      DEFAULT=0000  
  
LRDBCS TYPE=DBCSSID, ID=( 81-FF )  
LRDBCS TYPE=DBCS, XLATE=( in, out )  
LRDBCS TYPE=DBCS, XLATE=( in, out )  
LRDBCS TYPE=DBCS, XLATE=( in, out )  
  
LRDBCS TYPE=FINAL
```

TYPE=INITIAL must be the first LRDBCS macro coded, and TYPE=FINAL must be the last. DESC= specifies a 40-byte table description, and only applies to TYPE=INITIAL. DEFAULT= specifies a two-byte hex value to be used if an invalid DBCS character is encountered, and only applies to TYPE=INITIAL. Default value for DEFAULT= is 0000.

Each LRDBCS TYPE=DBCS macro within the table specifies an input value, and its corresponding translated output value. Both are two-byte hex values.

The LRDBCS TYPE=DBCSSID macro identifies the values of the first byte of DBCS pairs if no “shift-out” or “shift-in” characters will be found in the data.

LRS has provided two sample DBCS translate tables, LRSDBCS1 and LRSDBCS2, that may be used as models in constructing additional DBCS translate tables. LRSDBCS1 translates from Japanese EBCDIC (code page 300) to Japanese ASCII (code page 301), and LRSDBCS2 translates from Japanese ASCII to Japanese EBCDIC. Both tables were generated from information contained in the IBM Character Data Representation Architecture Level 1 Registry (SC09-1391-00). These tables are supplied as members LRSDBCS1 and LRSDBCS2 of distribution file LRS.DRS.V1R33.ASM.

**Note:** If a DBCS translation table will be created, restore from the DRS distribution libraries the file indicated in the table below.

FILE SEQUENCE NUMBER (with VPS R7.0)	DSNAME	CONTENTS
#03	LRS.R10.MACLIB	Mapping macros for Common Product Interfaces

---

# Creating a Report Termination Table

## Overview

DRS/VPI allows a report termination table to be associated with any VTAM virtual printer. This table specifies different report termination options that should be used when the virtual printer is connected to certain session partners or using certain session LU types. To specify that the report termination table should be used, the first positional parameter of the TERMRPT keyword should be set to TABLE and the second positional parameter should be the name of the table to be used. TERMRPT=TABLE is considered valid only for COMMTYPE=VTAM, and if specified for a printer with COMMTYPE=TCPIP or COMMTYPE=APPC, an error message will be issued and printer activation will fail.

## Definitions

A partner LU (PLU) is the name of the VTAM application which is connected to the virtual printer, such as CICS or IMS. The partner LU is normally the primary LU for the VTAM session.

The LUTYPE is the type of session that has been established between the virtual printer LU and the PLU. The LU type can be 0, 1 or 3.

The TERMRPT option specifies how the report should be terminated. The report can be terminated when the end-of-chain flag is received by DRS/VPI, when the bracket is ended, when the session is ended, or after a wait interval has expired.

## Defining the Report Termination Table

The report termination table is created using the \$DRSTRPT macro in LRS.DRS.V1R33.MACLIB. The \$DRSTRPT macro is written as follows:

```
$DRSTRPT TYPE=INITIAL  
$DRSTRPT  
TYPE=ENTRY,LUTYPE=a,PLUNAME=bbbbbbbb,TERMRPT=ccccccc  
$DRSTRPT TYPE=ENTRY,LUTYPE=d,PLUNAME=eee,TERMRPT=ffffff  
$DRSTRPT TYPE=FINAL
```

TYPE=INITIAL must be the first \$DRSTRPT macro coded, and the TYPE=FINAL must be the last. The TYPE=ENTRY macros specify the report termination options to be used.

LUTYPE specifies the type of session established. Valid values are 0, 1, 3, or \*. If LUTYPE is specified as '\*', then all session types will be considered to match.

PLUNAME specifies the name of the session partner. Values can be 1-8 alphanumeric or national characters, \* or ?. An asterisk (\*) can be specified to create a generic prefix or suffix for the value. A question mark (?) can be used to indicate any character can be a match at that location. The asterisk (\*) and the question mark (?) cannot both be specified in the same PLUNAME value.

If PLUNAME=ABC\* is specified, any partner LU name that begins with ABC would be considered a match. If PLUNAME=\*XYZ is specified, any partner LU name that ends with XYZ would be considered a match. If PLUNAME=\* or PLUNAME=\*\* is specified, all partner LU names would match.

If PLUNAME=A?C?E is specified, partner LU names that have A in the first position, C in the third position and E in the fifth position would be considered a match. For example, ABCDE would be considered a match, but ABCDF or ABCDEFG would not.

---

TERMRPT specifies the report termination option. Valid values are BRACKET, CHAIN, SESSION or ss, where ss is a two-byte number that represents a time interval. TERMRPT=BRACKET indicates the report should be terminated when the bracket state is ended. TERMRPT=CHAIN indicates the report should be terminated whenever the end-of-chain indicator is received. TERMRPT=SESSION indicates the report should be terminated whenever the session is ended. TERMRPT=ss indicates the report should be terminated after that number of seconds has expired with no additional data received.

**NOTE:** Since the report termination option will be chosen when the first PLUNAME and LUTYPE values in the table are matched to the current session partner name and the session LU type, it is important to arrange the table with the most specific values before those which are more generic. If no matching entries are located in the table, the default report termination option will be chosen based on the session LU type. For more information on the default report termination values, see [“TERMRPT” on page 3.96](#).

The source for a sample report termination table is distributed as DVSSSTRPT in the LRS.DRS.V1R33.ASM and the load module is distributed as member DVSSSTRPT in LRS.DRS.V1R33.LOAD. This sample table provides the same report termination as would be chosen if no TERMRPT value were specified for the printer.

```
$DRSTRPT TYPE=INITIAL
```

```
$DRSTRPT TYPE=ENTRY,LUTYPE=0,PLUNAME=*,TERMRPT=04
```

```
$DRSTRPT TYPE=ENTRY,LUTYPE=1,PLUNAME=*,TERMRPT=BRACKET
```

```
$DRSTRPT TYPE=ENTRY,LUTYPE=3,PLUNAME=*,TERMRPT=BRACKET
```

```
$DRSTRPT TYPE=FINAL
```

## Creating a User Report Termination Table

To create a user report termination table, develop the source for the table using the \$DRSTRPT macro as described above. Assemble and linkedit the table into the load library used for DRS/VPI. Sample JCL is distributed in members TRPTASMH and TRPTASML in LRS.DRS.V1R33.CNTL. Then, modify the TERMRPT keyword to specify TABLE for the first positional parameter and the name of your table for the second positional parameter. The next time you activate the virtual printer definition, you will obtain the new TERMRPT values.

For more information on report termination, see [“TERMRPT” on page 3.96](#).



---

# Section 11

## DRS/VPI Problem Determination

This section contains hints about diagnosing problems with DRS/VPI. DRS/VPI provides many service aids which assist in problem determination.

- DRS/VPI display commands provide information about system and printer status.
- DRS/VPI messages are issued for any significant event, including system and printer errors.
- The DRS/VPI log provides an audit trail of all DRS/VPI messages issued.
- DRS/VPI has an internal trace table that records events as requested in the TRACE keywords in the system initialization member and in the printer definition members.
- GTF tracing can be activated to allow tracing system or printer events, as well as input and output buffers.
- SNAP dumps can be created automatically when errors occur or upon request, without disturbing the DRS/VPI system. SNAP dumps contain system and/or printer control blocks, the values of system or printer keywords, and the text for each error message.
- Any ABEND of the DRS/VPI address space will create a storage dump in a SYSUDUMP dataset, if allocated.

Many types of problems may occur in a system as complex as DRS/VPI. The most common problem is incorrect output. By tracing the buffers of data received and the print lines which are being created from those buffers, it is possible to determine the cause of most errors.

If you find you need assistance to resolve any problem, the DRS support staff is always ready to help you.

---

# Display Commands

The DRS/VPI display command is a tool for gathering information about a printer or the DRS/VPI system. The display command is a good starting point for most types of problem determination. In all the examples below, 'drsv' should be replaced with the name of the DRS/VPI system (job name or started task name), 'prtr-id' with the member name of the printer, and 'out-id' with the member name of the OUTPUT statement reference member.

## Displaying DRS/VPI System Status

To obtain the DRS/VPI system status, including storage used and the values of keywords specified at system initialization, issue the following command:

```
F drsv,DISPLAY,SYSTEM
```

Messages DRSV940R - DRSV949R will be issued in response to the system display command. The messages show the current version, release and fix level of DRS. The DRSV947R message indicates if DRS/VPI is connected to the MVS TCP/IP system to allow receiving data using DRS/TCPIP.

## Displaying DRS/VPI User Exit Status

To obtain status information about DRS/VPI user exits, issue the following command:

```
F drsv,DISPLAY,EXITS
```

Message DRSV955R will be displayed for each user exit. It is also possible to issue the same command for a specific user exit, such as EXIT07. The status messages show the name of the user exit module and whether the user exit is currently enabled or disabled.

## Displaying DRS/VPI OUTREF information

To determine which OUTPUT statement reference members are in use in the DRS/VPI system, issue the following display command:

```
F drsv,DISPLAY,OUTREF,*
```

Message DRSV960R will be displayed for each OUTREF in use. The status will indicate whether the OUTREF is in use (ACTIVE) and how many printers are currently referring to this OUTPUT statement reference.

It is also possible to display all the OUTPUT characteristics defined by a specific OUTREF member with the following command, where out-id is the name of the member which contains the OUTPUT statement keywords:

```
F drsv,DISPLAY,OUTREF,out-id
```

Messages DRSV960R - DRSV977R will be issued to display all the OUTREF keyword values.

## Displaying DRS/VPI Virtual Printers

To determine the status of a specific printer in the DRS/VPI system, issue the following display command:

```
F drsv,DISPLAY,prtr-id
```



---

DRSV980R will indicate if the printer is IDLE, whether it is connected to another system, or if it is in a DRAINED or EDRAINED status. If the printer is EDRAINED, the error message text displayed will indicate the type of error which caused the printer to be stopped. For more information on error text messages, see [“EDRAINED Printer Error Messages” on page 35.91](#). The original message which reported the error will be stored in the DRS/VPI log dataset. See [“DRS/VPI Log” on page 11.5](#) for more information about message logging.

For a VTAM printer, including a DRS/PC printer definition which uses a VTAM APPC connection, it may also be helpful to issue the following command to display the VTAM status:

```
F drsv,DISPLAY,prtr-id,VTAM
```

Messages DRSV997R and DRSV998R will be issued to indicate whether the VTAM ACB has been opened and the SETLOGON start was successful. If the message indicates SLU(DISABLED), then a START command should be issued to make the printer available for sessions. If the printer is currently in session, the session partner name and the bind parameters used for the session will be displayed.

For a TCP/IP printer, the following display will show the status of a connection to the remote host:

```
F drsv,DISPLAY,prtr-id,TCPIP
```

Messages DRSV995R and DRSV996R will be issued showing if a connection is active. If there is a connection, the remote host IP address and port number will be displayed.

In order to determine if the printer has successfully received reports and placed them in DASD or JES spool files, issue the following command:

```
F drsv,DISPLAY,prtr-id,PROCESSING
```

The DRSV981R message will indicate how many print records and how many print datasets have been created since the printer was activated.

In addition to DRS/VPI display commands, it may be helpful to use VTAM display commands to display VTAM virtual printers. Because it is necessary to activate an APPL statement which matches the member name of the VTAM virtual printer, the following display could help determine the VTAM status of the definition:

```
D NET,E,ID=prtr-id
```

The VTAM virtual printer should be ACTIVE and indicate the JOBNAME of the DRS/VPI address space that has opened the ACB.

---

## Messages

When DRS/VPI detects an exception condition, it will issue a message to the operator. This message will contain information about what DRS/VPI was attempting to do at the time. The message may be an informational message, a warning or an error message. Any return codes or other information obtained at the time of the error will be displayed in the message, including error codes from VTAM or TCP/IP.

If the message was for a virtual printer and is an informational or warning message, the printer will continue to receive print data and process the report. If the message is an error message, the printer will be EDRAINED, and the connection will be terminated with the remote system.

When a message is received, look up the message in [“Messages and Codes” on page 35.1](#). The error and the return codes will be explained, as well as possible action that could eliminate the error.

In some cases, the MVS system log (SYSLOG) may provide additional informational messages that were issued at the time of the error. For example, a VTAM message may be issued when an operator inactivated a virtual printer APPL statement; this would cause an error in the DRS/VPI system, and a message would also be issued by DRS/VPI.

---

## DRS/VPI Log

The DRS/VPI log contains an audit trail of all commands and messages processed by DRS/VPI. The log contains date and time values to help correlate an event and the messages associated with the event. All DRS/VPI messages are entered into the DRS/VPI log, regardless of parameters which can be coded to prevent certain messages from appearing on the operator console. We recommend that logging always be active in the DRS/VPI system.

If the DRS/VPI log is defined as a SYSOUT dataset (rather than a pre-allocated DASD dataset), a “CLOSELOG” command can be issued to make the DRS/VPI log immediately available for viewing or printing. This allows problem determination without shutting down DRS/VPI.

For more information on the CLOSELOG command, see [“DRS/VPI Operation” on page 8.1](#). For more information about defining the LOG parameters, see [“Building the System Initialization Member” on page 3.13](#).

---

## Internal Trace

The DRS/VPI system writes internal trace table records for events based on the TRACE keywords in the system initialization member and the printer definition members. The trace records are recorded in “wrap-around” mode so the trace table will not fill up and terminate tracing. The size of the trace table is specified in the system initialization member as a number of 4K pages of MVS storage.

When an event occurs, DRS/VPI records a 4-byte event type, a time stamp, and other significant data.

Certain system-related or printer-related events are always traced. This basic set of trace table entries provides a good representation of DRS/VPI processing, but it is not detailed. If more detailed tracing is needed, additional events can be traced via the TRACE keyword. For more system events to be traced, modify the TRACE keyword in the system initialization member. For more printer events, modify the TRACE keyword in the appropriate printer definition member. The first subfield of the TRACE printer keyword should always be set to ‘Y’ to do internal tracing.

You can also request that the internal DRS/VPI trace entries be written in the GTF trace file using the TRACE keyword. For more information on GTF tracing, see [“GTF Tracing” on page 11.7](#). For more information on the TRACE system and printer keywords, see [“Building the System Initialization Member” on page 3.13](#) and [“Building the Printer Definition Members” on page 3.42](#).

---

## GTF Tracing

DRS/VPI allows tracing to the MVS GTF trace file. This allows tracing for data that cannot be written to the internal trace table, as well as the internal trace table entries.

The internal trace entries can be written to the GTF trace file by specifying a value for the TRACE keyword at system initialization or in the TRACE keyword in a printer definition.

Buffers of data can be written to the GTF trace file which show data received and the print lines created by DRS/VPI. All the parameters which are passed to user exits can be traced.

For more information on specifying the TRACE keyword in the system initialization member or the printer definition member, see [“Building the System Initialization Member” on page 3.13](#) or [“Building the Printer Definition Members” on page 3.42](#).

In order to write the trace records to GTF, GTF must be started with USR type tracing active. When the trace data has been created, GTF should be stopped. To print the trace records, use program LRSPRGTF supplied in the LRS.DRS.V1R33.LOAD library on the distribution cartridge. Sample JCL is supplied in member PRINTGTF in the LRS.DRS.V1R33.CNTL library on the DRS distribution cartridge.

If buffer records are to be printed, the LUNAME must be specified in the parameters for the LRSPRGTF program. The LUNAME should be set to the member name of the virtual printer that was traced. If LUNAME=\* is specified, no buffer entries will be formatted. If the buffers received are in ASCII (when using DRS/TCPIP, for example), specifying OPTION=ASCII will allow formatting the printable characters based on how they would print on a ASCII device.

GTF will only record trace entries in the primary extent allocated for the GTF trace file. When the primary extent is full, GTF begins to write over the records already recorded in the primary extent. If the trace file “wraps” in this fashion, it will be necessary to reallocate the GTF trace file with a larger primary extent.

For more information on the LRSPRGTF program, see the documentation in the sample JCL supplied in member PRINTGTF in the LRS.DRS.V1R33.CNTL library.

---

## Snap Dump

The snap dump facility of DRS/VPI captures a “picture” of certain DRS/VPI control blocks at the time the SNAP command was issued or at the time of an error. The title of the SNAP dump indicates whether the dump was taken for an error condition or was issued because of a SNAP command.

Each snap dump will dynamically create a SYSOUT data set on the JES spool. Upon completion of the snap dump, the requested information is available for printing or viewing. This allows problem determination to begin immediately. DRS/VPI does not have to be shut down to print or view the dumps.

If SNAP=Y is specified in the virtual printer definition, DRS/VPI will automatically create a snap dump for certain error conditions. It is also possible to obtain a snap dump at any time by issuing a SNAP command. The SNAP command can be issued for the whole DRS/VPI system with the following command:

```
F drsv,SNAP,SYSTEM
```

To obtain information related to a specific printer, issue the following command:

```
F drsv,SNAP,prtr-id
```

For more information on how the SYSOUT dataset will be created, see the SNAP keywords in the system initialization member as described in [“Building the System Initialization Member” on page 3.13](#). For more information about the SNAP command, see [“DRS/VPI Operation” on page 8.1](#).

---

## System Dump

The system dump is an operating system service aid. The dumps are automatically created by the system when a task within the DRS/VPI address space has abended. This dump is extremely valuable for resolving problems where a subtask has abended.

It is also possible to request that DRS/VPI create a system dump by issuing the ABEND command. This type of dump may be helpful if the DRS/VPI system seems to be hung up. If the ABEND command is not successful in stopping DRS/VPI, then a system command can be issued to cancel the DRS/VPI address space. If the cancel command is used, a dump should be requested.

When you have a system dump from DRS/VPI, you can call DRS support for assistance. Some problems of this type can be resolved in a phone conversation. In most cases, you will be asked to mail the dump to DRS support for diagnosis. You can choose to send the dump on a tape (1600 or 6250 bpi) or on a cartridge (3480, non-compacted) or in hardcopy format. It may also be possible to upload the dump to our Web site. Please call DRS support before sending any dump.





---

## Section 12

# Introduction to DRS/TCPIP

DRS/TCPIP allows receiving print files from remote TCP/IP hosts and placing those print files on the JES spool to be printed at a printer attached to the MVS system. The same DRS/VPI address space can be used to receive print data from VTAM online applications (CICS, IMS, IDMS, etc.) and from TCP/IP hosts. All print queues can be displayed and controlled using DRS/VPI console commands.

DRS/TCPIP provides the functions of a Line Printer Daemon (LPD) print server on MVS. DRS/TCPIP listens for connections on the PORT specified at DRS/VPI system initialization; the PORT should be set to 515 to receive print data using LPR/LPD protocols.

The LPR/LPD protocol defines a 'receive print job' command which provides a print queue name. DRS/TCPIP uses the print queue name to locate a virtual printer. Because the print queue name is also the name of a member in the DRS/VPI control library, it must be 1 to 8-characters, beginning with an alphabetic or a national character. If any part of the name is received in lower case, it will be translated to upper case. The virtual printer definition contains the SYSOUT characteristics (DEST, CLASS, FORM, WRITER, OUTREF, etc.) to be associated with the print data on the JES spool.

After the print queue name has been received, DRS/TCPIP will receive a control file and a data file. The data file contains the data to be placed on the JES spool to be printed. The control file contains records which indicate how the data should be printed. DRS/VPI User Exit 09 allows modifying the control file, which may cause DRS to process the file differently. It is also possible to indicate that the control file should be ignored for a print queue definition.

If the definition of the print queue indicates that the control file should be used to determine how to process the print data, but the control file is received after the data file, the data file must be kept in a temporary staging file until the control file can be processed. If the control file is delivered first, or if the control file can be ignored, no temporary staging is necessary.

---

The filter record of the control file indicates how the data file will be processed. For normal print filters, translation can be done from ASCII to EBCDIC based on the PRTXLATE keyword. For print queues with TRN=T or if a binary filter is received, no translation or line control processing will be done. The following table indicates which filters are considered valid and how DRS/TCPIP will process the data. DRS/VPI User Exit 09 can be used to change the value of the filter to change the manner in which DRS/TCPIP processes a file.

<b>FILTER</b>	<b>DESCRIPTION</b>	<b>DRS/TCPIP SUPPORT</b>
c	Plot CIF file	Stored as binary data
d	Plot DVI file	Stored as binary data
f	Print formatted file	Text with ASA carriage control
g	Plot file (UNIX)	Stored as binary data
l	Print file leaving carriage control	Text with ASA carriage control
n	Print DITROFF file	Stored as binary data
o	Print PostScript file	Stored as binary data
p	Print with PR format	Text w/page headings added and ASA carriage control
r	Print using FORTRAN carriage control	Text with ASA carriage control
t	Print phototypesetter	Stored as binary data
v	Print raster file	Stored as binary data

The print data is placed on the JES spool using the SYSOUT attributes defined for the print queue, including OUTPUT JCL characteristics. The print file is then ready to be delivered to the printer.

---

# Section 13

## DRS/TCPIP Installation

### New Features in this Release

If you have been using DRS/TCPIP Release 3.2 at your installation, you will find Version 1 Release 3.3 will support all of the functions you are accustomed to. This release also gives you these new features.

- **Support for additional OUTPUT keywords**

Support has been added for new OUTPUT statement keywords in the OUTPUT reference member. In addition, the DEST value can now be up to 127 bytes which allows specification of the IP address in the destination field, and up to 4 NOTIFY values may be specified.

- **DBCS translation support**

Translation of DBCS data is allowed for data received using DRS/TCPIP. Each print queue definition may specify the PRTXDBCS keyword to indicate whether DBCS data should be translated, the name of the table to use, and the disposition of the shift-out and shift-in characters.

- **Multiple data file support for LPD**

DRS V1 R3.3 allows receiving multiple LPD data files with one control file. This allows a remote TCP/IP host to send up to 256 print files in one connection.

- **Support for IBM's HPNS**

DRS/TCPIP can now use IBM's High Performance Native Sockets application programming interface which is available with IBM TCP/IP for MVS V3R2. HPNS provides increased performance and additional functions that were not provided with earlier programming interfaces.

- **SUBSYS keyword**

The SUBSYS keyword allows sending print files to a non-JES subsystem.

- **TAB keyword**

The TAB keyword in the print queue definition allows replacing the TAB character received from a remote TCP/IP host with 1 or more blanks.

- **JOBNAME printer keyword**

The JOBNAME keyword has been added to allow creating SYSOUT files with a name that is different from the name of the DRS/VPI job or started task. The JOBNAME can be a constant value or it can be set using symbolic parameters that represent dynamic values. Using the JOBNAME keyword requires MVS/SP 4.1.0 or later and that the DRS modules be loaded from an authorized library.

---

# Installation Procedure

The steps required to install the DRS/TCPIP product are:

1. **Install the MVS TCP/IP Software** (see [“DRS/TCPIP System Requirements”](#) on page 13.3).
2. **Install DRS/VPI** (see [“Installation Procedure”](#) on page 3.4).
3. **Add DRS/TCPIP keywords to the System Initialization Member for the DRS/VPI address space** (see [“Adding System Keywords for DRS/TCPIP”](#) on page 13.4).
4. **Build the DRS/TCPIP Printer Definition Members for the TCP/IP print queues** (see [“Building TCP/IP Print Queue Definitions”](#) on page 13.5 and [“Converting the IBM TCPIP LPD Configuration Data Set”](#) on page 13.6).
5. **Add the DRS TCP/IP printer definitions to the Printer Activation List** (optional) (see [“Adding TCP/IP Printers to DRS/VPI Activation List”](#) on page 13.10).
6. **Implement DRS/VPI User Exit 09 to modify the TCP/IP control file or to reject print files** (optional) (see [“Installing DRS/VPI User Exit 9 for TCP/IP”](#) on page 13.11).

---

# DRS/TCPIP System Requirements

## TCP/IP for MVS Requirements

Either TCP/IP for MVS from IBM or SNS/TCPassess for MVS from Interlink Computer Sciences must be installed on the MVS system where DRS/TCPIP will run.

For the TCP/IP for MVS product from IBM, the 31-bit version is required. The releases that are supported are V2R2.1 or later. If TCPTYPE=IBM221 is specified, the following IBM module must be available to be loaded during DRS/VPI system initialization; it can reside in a linklist library, LPALIB, or in the DRS/VPI STEPLIB.

<u>Module Name</u>	<u>Distribution Library</u>
IUCVMULT	SYS1.tcpip.SEZALINK

For the TCP/IP for MVS product (SNS/TCPassess) from Interlink Computer Sciences, all releases after V1.1 are supported. The following parameters in your configuration member for API (APICFGxx) may need to be reviewed:

- MAXADDR** Set as large as the number of DRS/VPI systems using DRS/TCPIP.
- MAXUSER** Set to the sum of the maximum number of DRS/TCPIP print queues (second subfield of MAXPRTRS keyword) plus the number needed for other products which use the API (for example, VPS/TCPIP).
- MAXEPA** Set to the sum of the maximum number of DRS/TCPIP print queues (second subfield of MAXPRTRS keyword) plus the number needed for other products which use the API (for example, VPS/TCPIP).

## MVS Requirements

To use the new JOBNAME= keyword, the DRS modules must be loaded from an authorized library and the system must be at level MVS/SP 4.1.0 or above.

---

## Adding System Keywords for DRS/TCPIP

The following DRS/TCPIP keywords can be added to the System Initialization Member for DRS/VPI:

**KEYLPD**  
**TCPIPID**  
**TCPPORT**  
**TCPTYPE**

The **KEYLPD** keyword specifies the trial/license code for the DRS/TCPIP product. This keyword **MUST** be present to use the DRS/TCPIP product. This key is generated by LRS and is supplied in file 1 of the product distribution cartridge.

The **TCPIPID** keyword is required. If TCPIPID is not specified, the DRS/VPI system will not attempt to connect to the MVS TCP/IP address space, and no connections will be accepted from remote TCP/IP hosts. If using TCPTYPE=IBM221, TCPTYPE=IBM310 or TCPTYPE=IBM320, the TCPIPID should match the jobname or started task of the IBM TCP/IP address space. For TCPTYPE=ICS, the TCPIPID should match the 4-character subsystem ID of Interlink's SNS/TCPaccess address space. The default subsystem ID for Interlink is ACSS.

The **TCPPORT** is optional; it specifies the PORT to be used when DRS/TCPIP listens for connections. The default is 515, which is the PORT used for the LPD print server.

It is possible that port 515 has already been reserved for a particular task name in the TCP/IP configuration file. If so, one of the following should be done:

1. Remove the entry for port 515 from the reserved ports list.
2. Change the entry for port 515 in the reserved ports list to specify the name of the DRS/VPI address space.
3. Change the name of the DRS/VPI address space to match the name specified in the reserved ports list entry for port 515.

The **TCPTYPE** keyword is optional; the default is TCPTYPE=IBM221. The TCPTYPE keyword indicates the method by which DRS/TCPIP will connect to the MVS TCP/IP product.

<b>TCPTYPE=IBM221</b>	<b>Use IBM IUCV interface (V2R2.1 or later)</b>
<b>TCPTYPE=IBM310</b>	<b>Use IBM Macro API interface (V3R1 or later)</b>
<b>TCPTYPE=IBM320</b>	<b>Use IBM HPNS interface (V3R2 or later)</b>
<b>TCPTYPE=ICS</b>	<b>Use Interlink Computer Sciences API (V1.1 or later)</b>

Some standard DRS/VPI system initialization keywords may be used for DRS/TCPIP processing. For example, the MAXPRTRS keyword specifies the maximum number of VTAM and TCP/IP printers that can be activated. The second subfield of the MAXPRTRS keyword should never be set to zero if TCP/IP printers are to be activated. In addition, if automatic activation or dynamic definition are required, the AUTOACT keyword will be necessary. The DEFLTMEM keyword should be specified for any parameters required for those printers automatically activated.

For more information on the DRS/VPI system keywords and the values that are allowed, see ["System Initialization Member" on page 3.10](#).

---

## Building TCP/IP Print Queue Definitions

The following keywords are used only for DRS/TCPIP print queues:

- TCPLIMIT=** Specifies the maximum size, in number of bytes, of any print file that this print queue will accept. Any requests to deliver a print file larger than this number will be denied.
- TCPMRD=** Specifies the time interval in minutes for DRS/VPI to wait for a TCP/IP request to complete. If the request does not complete within this interval, the connection will be terminated, and the printer will be EDRAINED. The default is 00, which means that DRS/VPI will never terminate the connection due to a time-out.
- TCPOPTS=** A hex representation of special printer options to be in effect for this TCP/IP print queue. The value is specified as eight digits, representing four hex bytes.

If the control file is delivered first, or if the control file can be ignored, no staging is necessary. If staging of the data file is necessary, the following parameters may be specified:

- QBUFSIZE=** Specifies the size of the buffer DRS/TCPIP will use when writing to and reading from the staging dataset.
- QSPACE=** Specifies, via three positional parameters, the amount of DASD space to be allocated to the staging dataset.
- QUNIT=** Specifies the unit type of the device on which the staging dataset is to be allocated.
- QVOLUME=** Specifies the volume on which the staging dataset is to be allocated.

Some standard DRS/VPI keywords may be necessary in order to process print files for TCP/IP print queues; for example:

- COMMTYPE=** The communications type that will be used to accept a connection for this print queue. The default for COMMTYPE is VTAM. For TCP/IP printers using LPR/LPD protocols, COMMTYPE=(TCPIP,LPD) should be specified.
- PRTXLATE=** In order to translate ASCII data to EBCDIC, the PRTXLATE keyword must be specified. The first subfield should be 'Y' to request translation, and the second subfield should specify a table name.

For more information on all the DRS/VPI printer keywords and the values that are allowed, see ["Building the System Initialization Member" on page 3.13](#).

A sample DRS/TCPIP printer definition for a print queue named "LOCAL" may contain the following:

```
* -----*
*   DEFINITION OF PRINT QUEUE FOR LOCAL PRINTER   *
*   -----*
CLASS=A,                SYSOUT CLASS = A
COMMTYPE=( TCPIP ,LPD) , TCPIP LPR/LPD PROTOCOL
DEST=LOCAL,            JES DESTINATION = LOCAL
PRTXLATE=( Y ,DVSSXEBC) , TRANSLATE ASCII TO EBCDIC
TCPLIMIT=99999         REFUSE IF LARGER THAN 99,999 BYTES
```

---

# Converting the IBM TCPIP LPD Configuration Data Set

DRS V1 R3.3 provides a facility to convert your IBM TCPIP LPD configuration SERVICE definitions into DRS/VPI virtual printer definitions.

Sample JCL to convert your LPD configuration SERVICE definitions is contained in member CONVLPD in data set LRS.DRS.V1R33.CNTL. That JCL is reproduced here:

```
//JOBNAME JOB (YOUR JOB CARD INFORMATION)
//*
//CONVLPD EXEC PGM=DV33CLPD,REGION=2M,PARM='REPLACE'
//STEPLIB DD DSN=LRS.DRS.V1R33.LOAD,DISP=SHR
//SYSIN DD DSN=TCPIP.LPD.CONFIG,DISP=SHR
//SYSOUT DD SYSOUT=*
//DRSVLIB DD DSN=LRS.DRS.V1R33.CNTL,DISP=SHR
//SYSUDUMP DD SYSOUT=*
//*
//
```

## Procedure DD Statements

The SYSIN DD statement identifies your IBM TCPIP LPD configuration data set.

The SYSOUT DD statement is used to write a log of the results of the procedure. You should review the output written to this file to identify any errors that may occur during the process. For any severe errors, a message is also written to the console. Note that if you choose to direct the output of this file to a disk or tape data set, rather than the spool, the DCB attributes must be LRECL=128,BLKSIZE=132,RECFM=VB.

The DRSVLIB DD statement identifies your DRS/VPI Version 1 Release 3.3 control library.

## Procedure Parameters

The procedure which converts your IBM TCPIP LPD configuration SERVICE definitions supports only keyword parameters. Currently, the only supported keyword parameter is REPLACE, which indicates that the procedure is to replace existing virtual printer definitions in the DRS/VPI control library with the corresponding SERVICE definitions from the LPD configuration data set. If this parameter is omitted, any existing entries in the DRS/VPI control library will be retained, but new entries will still be created using the definitions from the IBM TCPIP LPD configuration data set for which there is no matching entry in the DRS/VPI control library.



---

## Messages Issued by the LPD Conversion Program

The following messages can result from using program DV33CLPD to convert IBM TCPIP LPD configuration SERVICE definitions to DRS/VPI virtual printer definition members.

- CLPD920I** DEFINITION action FOR PRINTER printer  
action: Indicates the action taken against the named printer definition (CREATED, REPLACED, or RETAINED).  
printer: Indicates the name of the DRS virtual printer definition.  
**Message Meaning:** The LPD configuration conversion program took the action shown in the message for the virtual printer definition shown in the message.  
**System Action:** None.  
**Required Action:** None.
- CLPD924I** count PRINTER DEFINITIONS action  
count: Indicates the number of printer definitions for which the indicated action was taken.  
action: Indicates the action taken against the printer definitions (CREATED, REPLACED, or RETAINED).  
**Message Meaning:** The LPD configuration conversion program took the action shown in the message for the number of virtual printer definitions shown in the message.  
**System Action:** None.  
**Required Action:** None.
- CLPD960W** EXEC STATEMENT PARAMETER parm IS NOT RECOGNIZED  
parm: Indicates the JCL EXEC statement parameter that is not recognized by the LPD configuration conversion program.  
**Message Meaning:** A JCL EXEC statement parameter was specified that is not recognized by the LPD configuration conversion program.  
**System Action:** The conversion processing continues, ignoring the unrecognized parameter.  
**Required Action:** None.

---

<b>CLPD970W</b>	MEMBER member IS IN USE AND HAS BEEN RETAINED
	member: Indicates the DRS control library member name that is in use.
	<b>Message Meaning:</b> The DRS control library member shown in the message is in use by another job or user. Most likely, the member is being edited by an ISPF/PDF user.
	<b>System Action:</b> The conversion processing continues with the next LPD service definition, while retaining the DRS virtual printer definition for the member shown in the message.
	<b>Required Action:</b> Rerun the LPD configuration conversion program when the member shown in the message is accessible.
<b>CLPD980E</b>	OPEN FAILED FOR FILE ddname
	ddname: The JCL DDNAME of the file for which the OPEN failed.
	<b>Message Meaning:</b> The LPD configuration conversion program attempted to OPEN the file shown in the message, but the attempt failed.
	<b>System Action:</b> The conversion processing terminates.
	<b>Required Action:</b> Correct the LPD configuration conversion process JCL.
<b>CLPD974E</b>	SERVICE NAME ssss IS NOT CONVERTIBLE
	sss: The LPD configuration file service name.
	<b>Message Meaning:</b> The LPD configuration conversion program encountered a service definition with a name that is not in the form of a valid partitioned dataset member name. As a result, the conversion program was unable to convert the named service definition.
	<b>System Action:</b> The conversion processing continues with the next LPD service definition.
	<b>Required Action:</b> None.

---

<b>CLPD982E</b>	STOW FAILED FOR MEMBER member, RETCODE=X'retcode', RSNCODE=X'rsncode'
member:	The DRS control library member for which the STOW request failed.
retcode:	The STOW return code.
rsncode:	The STOW reason code.
<b>Message Meaning:</b>	The LPD configuration conversion program attempted to update the DRS control library directory to reflect the changes made to the member shown in the message, but the attempt failed with the return code and reason code shown in the message.
<b>System Action:</b>	The conversion processing continues with the next LPD service definition, while retaining the DRS virtual printer definition for the member shown in the message (if it previously existed).
<b>Required Action:</b>	Contact DRS technical support.
<b>CLPD984E</b>	RDJFCB FAILED FOR FILE ddname, RETCODE=X'retcode'
ddname:	The JCL DDNAME of the file for which the RDJFCB failed.
retcode:	The RDJFCB return code.
<b>Message Meaning:</b>	The LPD configuration conversion program attempted to read the Job File Control Block for the file shown in the message, but the attempt failed.
<b>System Action:</b>	The conversion processing terminates.
<b>Required Action:</b>	Contact DRS technical support.
<b>CLPD988E</b>	DRSVLIB ALLOCATION IS INVALID—desc
desc:	The description of the allocation attributes that are inconsistent with those of a valid DRS control library.
<b>Message Meaning:</b>	The data set allocated to the JCL DDNAME DRSVLIB is not a valid DRS control library, or is incorrectly specified.
<b>System Action:</b>	The conversion processing terminates.
<b>Required Action:</b>	Correct the LPD configuration conversion process JCL.

---

## Adding TCP/IP Printers to DRS/VPI Activation List

If you are using the Printer Inclusion List to automatically activate DRS/VPI printer definitions, you should add the member names of the DRS/TCPIP printers to that list. For more information, see [“Building the Printer Activation Inclusion List” on page 3.39](#).

---

## Installing DRS/VPI User Exit 9 for TCP/IP

DRS/VPI User Exit 9 is an optional user exit that will be called only for DRS/TCPIP print queues. The exit allows modification of the LPR/LPD control file received from the remote TCP/IP host, which may change the processing of that file by DRS/TCPIP. In addition, the print file can be rejected at several points by DRS/VPI User Exit 9, based on the sender's host name or address, based on the user name or other information in the control file, or based upon the print queue name requested.

If the print file is rejected by DRS/VPI User Exit 9, a negative acknowledgement is sent to the remote host with "USER EXIT REFUSED PRINT FILE" as the text.

All control records received from the remote host are placed in a control line workarea which is passed to User Exit 9. The macro used to map the control line workarea is shipped in LRS.DRS.V1R33.MACLIB on the distribution cartridge as #LRTCCNT.

The prefix of the control lines area contains information about all the control records, such as total length of the control block, the number of control lines available, the number of control lines used, and a copy count. The prefix area should not be modified.

The control records follow the prefix. Each record has a field to indicate the length of the data used in that entry and status flags to indicate if the control record is the filter record or if the control record can be bypassed by setting the bypass flag.

Each control record has a 'type' field followed by up to 256 bytes of additional data. The last byte of the data must be the ASCII new line (X'0A'). If the data in the control record is changed, the length of the data must be updated to include the 'type' field, the data, and the new line character.

The valid types of print filters are documented in the #LRTCCNT macro. If multiple filter records are received from the remote host, the data file names must be the same. The copy count is set based on the number of filters and is stored in the prefix to the control line workarea. The copy count will be used to set the number of copies to be requested when the print data is placed on the JES spool.

When the exit returns to DRS/VPI, the control records will be scanned. Any records which contain errors will be flagged and bypassed. If the records meet the minimum requirements, they will be used for any processing that DRS will do with the print data file. For example, by changing the filter from a normal print filter to one that would be treated as a 'binary' filter, translation could be avoided. In addition, a banner page could be eliminated by bypassing the "L" control record.

For more information about implementing DRS/VPI User Exit 9, see ["DRS/VPI User Exits" on page 9.1](#).



---

## Section 14

# DRS/TCPIP Problem Determination

DRS/VPI provides facilities to aid in problem determination; these are described in [“DRS/VPI Problem Determination” on page 11.1](#). In addition, the messages issued by DRS/API and DRS/VPI are described in [“Messages and Codes” on page 35.1](#). The possible meanings of the message text, along with suggested actions appear with the messages.

DRS/VPI will issue message DRSV380I when DRS/TCPIP connects to the MVS TCP/IP address space. The message includes the local host name for the MVS system and the PORT that DRS/TCPIP will use to listen for connections.

If the MVS system is terminated for any reason, an error message will be issued to indicate that the TCP/IP connections have been lost. To determine if the DRS system is connected to TCP/IP, a system display command can be issued:

```
F DRSVPI,DIS,SYSTEM
```

The results of the display will include message DRSV947R, which will include the current status of the connection, followed by the local host name, the TCPIPID specified, the PORT number, and the TCPTYPE.

In order to re-connect to the MVS TCP/IP address space and wait for new connections, the following command can be issued:

```
F DRSVPI,SSET,TCPIP=E
```

To display the connection information for any print queue, issue the following command:

```
F DRSVPI,DIS,prtrid,TCPIP
```

DRSV995R and DRSV996R will indicate the current status of the TCP/IP connection, as well as the values of the TCP/IP keywords specified for the print queue.

All the DRS/VPI trace and snap dump facilities are available for the DRS/TCPIP devices. GTF tracing allows capturing all commands and data sent and received from TCP/IP. For more information on these and other DRS/VPI diagnostic facilities, see [“DRS/VPI Problem Determination” on page 11.1](#).





---

# Section 15

## Introduction To DRS/STI

The DRS Smart Tag Interface (hereafter referred to as DRS/STI) is a new feature of the DRS/VPI that allows print output destined to a single virtual printer to be placed on the JES spool with different attributes. The attributes that may be changed for each print output include not only the JES class, destination, form, and writer, but also all of the AFP attributes (orientation, etc.).

The DRS/STI provides a means for print output to be routed to multiple physical printers with different AFP attributes, without requiring a separate virtual printer definition for each combination of physical printer and AFP attributes in use at your installation.

The DRS/STI will improve the usability of DRS/VPI and simplify an enterprise's overall printing environment. More specifically, the DRS/STI will provide the following benefits:

- Greatly reduce the number of DRS/VPI virtual printer definitions.
- Greatly reduce the maintenance required to support DRS/VPI virtual printers in a typical CICS or IMS environment.
- Simplify the installation and maintenance of the DRS/VPI.
- Provide all legacy and third-party applications with a simple interface to the VPS print distribution system from an online environment.
- Facilitate the distribution of online print output to other media, such as fax and electronic mail.

In the CICS transaction processing environment, the DRS/STI provides a common interface that allows the CICS terminal user to select a specific DRS/STI printer entry and/or DRS/STI output reference entry for use in assigning the attributes to the print output.



---

# Section 16

## DRS/STI Installation

### Installation Procedure

1. **Read the entire installation procedure.**
2. **Restore the DRS distribution libraries** (see [“Restoring the DRS Distribution Libraries”](#) on page 20.6).
3. **Define the DRS/STI Rules Data Set** (see [“Defining the DRS/STI Rules Dataset”](#) on page 16.2).
4. **Complete the ISPF definition requirements** (see [“ISPF Definition Requirements”](#) on page 16.6).
5. **Complete the CICS definition requirements** (optional) (see [“CICS Definition Requirements”](#) on page 16.9).
6. **Refer to [“Updating the CICS Tables”](#) on page 20.14, and follow the procedures for installation of DRS/STI in a CICS environment.** (optional)
7. **Modify the DRS/VPI Job Control Language (JCL) to include the DD statement for the DRS/STI rules data set** (see [“Defining DRS/VPI Job Control Language”](#) on page 3.7).
8. **Modify the DRS/VPI System Initialization member to include the KEYSTI parameter** (see [“Building the System Initialization Member”](#) on page 3.13).
9. **Populate the DRS/STI rules data set** (see [“Populating the DRS/STI Rules Data Set”](#) on page 16.10).
10. **Establish backup procedures for the DRS/STI rules data set** (see [“Backing Up the DRS/STI Rules Dataset”](#) on page 16.17).
11. **Establish reorganization procedures for the DRS/STI rules data set** (see [“Reorganizing the DRS/STI Rules Dataset”](#) on page 16.18).
12. **DRS/STI installation is now complete.**

---

## Defining the DRS/STI Rules Dataset

The DRS/STI rules dataset is a variable-length record VSAM KSDS that is updated by DRS/STI and by the installation's DRS system administrator. The data set contains the following types of entries:

- Vendor entries which define the software vendors for the application software installed at your installation. The vendor entries are created and maintained by the DRS system administrator, but are optional. By defining them, along with the application and profile entries, you may define default printer and output references to be used if a specific printer and/or output reference is not requested.
- Application entries which define the application software installed at your installation. The application entries are created and maintained by the DRS system administrator, but are optional. By defining them, along with the vendor and profile entries, you may define default printer and output references to be used if a specific printer and/or output reference is not requested.
- Profile entries which allow you to define a specific combination of printer and output reference. The profile entries are created and maintained by the DRS system administrator, but are optional. By defining them, along with the vendor and application entries, you may define default printer and output references to be used if a specific printer and/or output reference is not requested.
- Printer type entries which define the attributes of each printer type (PRINTER, FAX, EMAIL, etc.). A base set of printer type entries is created when the DRS/STI rules data set is defined. Other entries will be created and maintained by the DRS system administrator, as required by the application software installed at your installation.
- Printer entries which define the attributes of each printer. These include the CLASS, DEST, FORM, and WRITER with which the output directed to the printer will be placed on the JES spool. The printer entries are created and maintained by the DRS system administrator and are required.
- Note that included with DRS V1 R3.3 is a program that will populate the DRS/STI rules data set with printer entries using your VPS R7.0 printer definitions. For more information, see [“Populating the DRS/STI Rules Data Set” on page 16.10](#).
- Output reference entries which define the AFP attributes of the output to be placed on the JES spool. The output reference entries are created and maintained by the DRS system administrator and are optional.
- Included with DRS V1 R3.3 is a program that will populate the DRS/STI rules data set with output reference entries using your DRS V1 R3.3 control library output reference definitions. For more information, see [“Populating the DRS/STI Rules Data Set” on page 16.10](#).
- User entries which define the preferred printers and output references for your installation's users. The user entries are created and maintained by the DRS/STI, and can also be added by your DRS/STI administrator.

---

Sample JCL to define the DRS/STI rules data set is contained in member DEFRULES in data set LRS.DRS.V1R33.CNTL. That JCL is reproduced here:

```
//JOBNAME JOB (YOUR JOB CARD INFORMATION)
//*
//DEFINE EXEC PGM=IDCAMS,REGION=1M
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DEFINE CLUSTER(
    NAME(DRS.V1R33.RULES)
    VOLUMES(XXXXXX)
    INDEXED
    KEYS(48 0)
    RECORDSIZE(96 4086)
    CYLINDERS(5 1)
    SHAREOPTIONS(4 3)
  )
  DATA(
    NAME(DRS.V1R33.RULES.DATA)
    CONTROLINTERVALSIZE(4096)
  )
  INDEX(
    NAME(DRS.V1R33.RULES.INDEX)
    CONTROLINTERVALSIZE(3584)
  )
/*
//*
//LOAD EXEC PGM=DS33LOAD,REGION=2M,COND=(0,LT)
//STEPLIB DD DSN=LRS.DRS.V1R33.LOAD,DISP=SHR
//SYSOUT DD SYSOUT=*
//DRSRULES DD DSN=DRS.V1R33.RULES,DISP=SHR
//SYSUDUMP DD SYSOUT=*
//*
//
```

---

## Allocation Guidelines

- You must specify a valid volume serial for the Access Method Services VOLUMES keyword of the DEFINE CLUSTER statement.
- The direct access space required for the DRS/STI rules data set depends primarily on the number of entries (vendors, printers, etc.) defined in the data set. On the average, the data set requires approximately 1 cylinder on a 3380 device for each 500 entries defined.
- The direct access space must be allocated in cylinders.
- You may optionally modify the cluster, data, and index component names.
- All other keywords not mentioned above should be coded exactly as shown.

## Smart Tag Control Area

The print data stream written to the virtual printer must be prefixed with the smart tag control area. This is a 128-byte buffer prefix that identifies the print output vendor, application, and profile; and may directly identify the printer and output statement to use in spooling the data stream.

Following is the assembly language DSECT of the smart tag control area. This DSECT is supplied as member DRVSSTCA in file LRS.DRS.V1R33.MACLIB.

```
*      *
*      *
*      DYNAMIC REPORT SYSTEM - VERSION 1 RELEASE 3.3*
*      *
*      © COPYRIGHT - 1998*
*      LEVI, RAY & SHOUP, INC.*
*      2401 WEST MONROE*
*      SPRINGFIELD, IL 62704*
*      *
*      *
*      *
*      S M A R T   T A G   C O N T R O L   A R E A*
*      *
*      *
STCADSCT DSECT
STCAPRFX DC    CL08'$LRSDRS$'          CONTROL BLOCK PREFIX.
STCAVERS DC    CL01'0'                 CONTROL BLOCK VERSION ID.
          DC    CL03' \                 RESERVED - ALIGNMENT.
STCAVNDN DC    CL08' \                 VENDOR NAME.
STCAAPLN DC    CL08' \                 APPLICATION NAME.
STCAPRFN DC    CL16' \                 PROFILE NAME.
STCAPRTN DC    CL08' \                 PRINTER NAME.
STCAOUTN DC    CL08' \                 OUTPUT STATEMENT REFERENCE.
STCAUID  DC    CL08' \                 CREATOR USERID.
          DC    CL52' \                 RESERVED - EXPANSION.
STCASUFN DC    CL08'$LRSDRS$'          CONTROL BLOCK SUFFIX.
STCADSLN EQU   *-STCADSCT              LENGTH OF DSECT.
```

---

The smart tag control area consists of the following data elements:

- STCAPRFX is an 8-byte control area prefix that must contain the literal constant '\$LRSDRS\$'. This control area prefix is used to identify the data as a smart tag control area.
- STCAVERS is a 1-byte control area version indicator. This data element should be set to character '0' (X'F0'), which is currently the only supported version of the control area.
- STCAVNDN is the 8-byte vendor identifier. This data element is used in conjunction with the application identifier and profile identifier to determine the printer and output statement to use in spooling the data stream, if the printer and/or output statement were not explicitly provided.
- STCAAPLN is the 8-byte application identifier. This data element is used in conjunction with the vendor identifier and profile identifier to determine the printer and output statement to use in spooling the data stream, if the printer and/or output statement were not explicitly provided.
- STCAPRFN is the 16-byte profile identifier. This data element is used in conjunction with the vendor identifier and application identifier to determine the printer and output statement to use in spooling the data stream, if the printer and/or output statement were not explicitly provided.
- STCAPRTN is the 8-byte printer identifier to use in spooling the data stream.
- STCAOUTN is an 8-byte output statement identifier to use in spooling the data stream.
- STCAUID is an 8-byte userid of the user that created the output data stream. This data element is not currently used by the smart tag interface. It should, however, be provided by the application building the data stream whenever possible.
- STCASUFX is an 8-byte control area suffix that must contain the literal constant '\$LRSDRS\$'. This control area suffix is used to identify the data as a smart tag control area.

Note that all reserved areas of the smart tag control area should be initialized to spaces (X'40').

The data elements of the smart tag control area are used to determine the printer and output statement to use in spooling the print data stream. If the printer identifier and output statement identifier are explicitly provided in the smart tag control area, then the attributes of the printer and output statement provided are determined from the smart tag rules data set and used to spool the data stream.

If either the printer identifier or output statement identifier is not provided in the smart tag control area, and the vendor identifier and application identifier are provided, then an attempt is made to determine the default printer and/or output statement. The default printer and/or output statement is obtained from the smart tag rules data set profile entry, using the profile identifier provided in the smart tag control area. If a profile identifier was not provided, the default profile identifier (if any) defined in the smart tag rules data set application entry is used.

---

## ISPF Definition Requirements

Included with the DRS/STI is an ISPF application that allows the DRS system administrator to maintain the entries in the DRS/STI rules data set. Using this application, the DRS system administrator maintains the definitions for the DRS/STI printers and output references, and may optionally maintain definitions for the other entries in the DRS/STI rules data set.

The ISPF administration application is initiated via a TSO/E CLIST and must be initiated from within a valid ISPF environment. The PROC statement of the TSO/E CLIST may require some modification, depending upon the data set names chosen at your installation for the DRS control library, load library, and DRS/STI rules data set. That PROC statement is reproduced here:

```
PROC 0 DSSTCLIB(LRS.DRS.V1R33.CNTL) DSSTLLIB(LRS.DRS.V1R33.LOAD) +
      DSSTRLIB(DRS.V1R33.RULES)
```

The keyword parameters named DSSTCLIB, DSSTLLIB, and DSSTRLIB should be the data set names of your DRS control library, load library, and DRS/STI rules data set, respectively. If the names of these data sets at your installation are different from those listed above, you must modify the keyword parameter values in the TSO/E CLIST PROC statement.

Once the TSO/E CLIST has been customized, you may initiate the DRS/STI administration application using the TSO/E EXEC command from within ISPF. For example, type the following from the COMMAND line of any ISPF panel:

```
TSO EXEC `LRS.DRS.V1R33.CNTL(DS33STRT)`
```



You may add the DRS/STI Administration option to the ISR@PRIM panel, or if you prefer, an ISPF submenu. When doing so, you may use any unique combination of letters and numbers for the DRS/STI Administration menu selection characters. The remainder of this section assumes that you have chosen SA as the menu selection characters. A sample ISR@PRIM panel is shown below.

```

% - - - - - ISPF/PDF PRIMARY OPTION MENU - - - - -
%OPTION  ==>_ZCMD
%
%                                +USERID   - &ZUSER
% 0 +ISPF PARMS - Specify terminal and user parameters +TIME     - &ZTIME
% 1 +BROWSE     - Display source data or output listings+TERMINAL - &ZTERM
% 2 +EDIT       - Create or change source data          +PF KEYS  - &ZKEYS
% 3 +UTILITIES  - Perform utility functions
% 4 +FOREGROUND - Invoke language processors in foreground
% 5 +BATCH      - Submit job for language processing
% 6 +COMMAND    - Enter TSO command or CLIST
% 7 +DIALOG TEST - Perform dialog testing
% 8 +LM UTILITIES- Perform library administrator utility functions
% 9 +IBM PRODUCTS- Additional IBM program development products
% C +CHANGES   - Display summary of changes for this release
% T +TUTORIAL   - Display information about ISPF/PDF
% SA +SMARTTAG  - DRS Smart Tag Interface Administration
% X +EXIT       - Terminate ISPF using log and list defaults
%
+Enter%END+command to terminate ISPF.
%
)INIT
  .HELP = ISR0003
  &ZPRIM = YES          /* ALWAYS A PRIMARY OPTION MENU */
  &ZHTOP = ISR0003     /* TUTORIAL TABLE OF CONTENTS */
  &ZHINDEX = ISR91000 /* TUTORIAL INDEX - 1ST PAGE */
  VPUT (ZHTOP,ZHINDEX) PROFILE
)PROC
&ZQ = &Z
IF (&ZCMD = ' ')
  &ZQ = TRUNC(&ZCMD, '.')
IF (&ZQ = ' ')
  .MSG = ISRU000
&ZSEL = TRANS( &ZQ
               0, 'PANEL(ISPOPTA)'
               1, 'PGM(ISRBRO) PARM(ISRBRO01)'
               2, 'PGM(ISREDIT) PARM(P,ISREDM01)'
               3, 'PANEL(ISRUTIL)'
               4, 'PANEL(ISRFPA)'
               5, 'PGM(ISRJBL) PARM(ISRJPA) NOCHECK'
               6, 'PGM(ISRPTC)'
               7, 'PGM(ISRYXDR) NOCHECK'
               8, 'PANEL(ISRLPRIM)'
               9, 'PANEL(ISRDIIS)'
               C, 'PGM(ISPTUTOR) PARM(ISR00005)'
               T, 'PGM(ISPTUTOR) PARM(ISR00000)'
               SA, 'CMD(EX ' 'LRS.DRS.V1R33.CNTL(DS33STRT)') NOCHECK MODE(FSCR)'
               ' ', ' '
               X, 'EXIT'
               *, '?' )
&ZTRAIL = .TRAIL
)END

```

---

The statement used to initiate the DRS/STI Administration application is:

```
SA, 'CMD(EX ` `LRS.DRS.V1R33.CNTL(DS33STRT)') NOCHECK MODE(FSCR)'
```

Change SA to the letters or numbers you have chosen, if applicable, and change LRS.DRS.V1R33.CNTL to the name of your DRS control library.

You must also code the following statement on the ISPF panel you choose, and place it after the &ZSEL variable:

```
&ZTRAIL = .TRAIL
```

Note that you must exit and reenter ISPF for the newly added options to be usable.

---

## CICS Definition Requirements

In the CICS transaction processing environment, the DRS/STI provides a common interface that allows the CICS terminal user to select a specific DRS/STI printer entry and/or DRS/STI output reference entry for use in assigning the attributes to the print output. This interface requires that certain table entries be added to your CICS system.

Refer to [“Updating the CICS Tables” on page 20.14](#) for more information about this process.

---

## Populating the DRS/STI Rules Data Set

DRS V1 R3.3 provides a facility to populate the DRS/STI rules data set with printer and output reference entries using your VPS R7.0 printer definitions and your DRS V1 R3.3 output reference definitions.

Sample JCL to populate the DRS/STI rules data set is contained in member POPRULES in data set LRS.DRS.V1R33.CNTL. That JCL is reproduced here:

```
//JOBNAME JOB (YOUR JOB CARD INFORMATION)
//*
//LOAD EXEC PGM=DS33LOAD,REGION=2M,PARM=' , ,REPLACE'
//STEPLIB DD DSN=LRS.DRS.V1R33.LOAD,DISP=SHR
//SYSOUT DD SYSOUT=*
//VPSLIB DD DSN=LRS.VPS.R70.CNTL,DISP=SHR
//DRSVLIB DD DSN=LRS.DRS.V1R33.CNTL,DISP=SHR
//DRSRULES DD DSN=DRS.V1R33.RULES,DISP=SHR
//SYSUDUMP DD SYSOUT=*
//*
//
```

### Procedure DD Statements

The SYSOUT DD statement is used to write a log of the results of the procedure. You should review the output written to this file to identify any errors that may occur during the process. For any severe errors, a message is also written to the console. Note that if you choose to direct the output of this file to a disk or tape data set, rather than the spool, the DCB attributes must be LRECL=128,BLKSIZE=132,RECFM=VB.

If your installation does not have VPS R7.0 installed, or chooses not to create/update DRS/STI printer entries using the VPS printer definitions, you may omit the VPSLIB DD statement from the JCL provided. Similarly, if your installation chooses not to create/update DRS/STI output reference entries using the DRS output reference definitions, you may omit the DRSVLIB DD statement from the JCL provided.

---

## Procedure Parameters

The procedure which populates the DRS/STI rules data set supports both positional and keyword parameters. The first two parameters specified in the JCL EXEC statement PARM= keyword are positional parameters and all remaining parameters are keyword parameters. Note that if the positional parameters are not specified, at least 2 commas must precede the keyword parameters.

The first positional parameter is the member name in the VPS R7.0 control library that contains the VPS system initialization parameters. If this parameter is omitted, a member name of VPSSTART is assumed.

The second positional parameter is the member name in the DRS V1 R3.3 control library that contains the DRS VPI system initialization parameters. If this parameter is omitted, a member name of DRSSTART is assumed.

Currently, the only supported keyword parameter is REPLACE, which indicates that the procedure is to replace existing printer and/or output reference entries in the DRS/STI rules data set with the corresponding definitions from the VPS and DRS control libraries. If this parameter is omitted, any existing entries in the DRS/STI rules data set will be retained, but new entries will still be created using the definitions from the VPS and DRS control libraries for which there is no matching entry in the DRS/STI rules data set.

---

## Messages

The following messages can result from using program DS33LOAD to populate the DRS/STI rules dataset.

<b>DSST010E</b>	type REQUEST FAILED FOR FILE DSSTRLIB, RC=X'retcode' type: The file access request type. retcode: The file access request return code. <b>Message Meaning:</b> The rules file access request shown in the message failed with the return code shown in the message. <b>System Action:</b> Processing terminates. <b>Required Action:</b> Ensure that the DRS/STI rules dataset name is properly defined in the CLIST. If unable to resolve the problem, notify DRS technical support.
<b>DSST020E</b>	type REQUEST FAILED FOR FILE DSSTRLIB, RETCODE=X'rc', ERRCODE=X'ec', INFCODE=X'ic' type: The file allocation request type. rc: Dynamic allocation return code. ec: Dynamic allocation error code. ic: Dynamic allocation info code. <b>Message Meaning:</b> The rules file dynamic allocation request shown in the message failed with the error feedback information shown in the message. <b>System Action:</b> Processing terminates. <b>Required Action:</b> Ensure that the DRS/STI rules dataset name is properly defined in the CLIST. If unable to resolve the problem, notify DRS technical support.
<b>DSST900I</b>	INITIALIZING DRS PRINT RULES DATA SET <b>Message Meaning:</b> The DRS/STI rules data set has been opened in VSAM load mode and is being loaded with the initial control data. <b>System Action:</b> None. <b>Required Action:</b> None.

---

**DSST904I** ddname STATEMENT NOT PRESENT—PRINT RULES type DATA BYPASSED

ddname: The JCL DDNAME of the file which is not present. This will be either VPSLIB or DRSVLIB.

type: The type of print rules data for which processing was bypassed. This will be either PRINTER or OUTPUT.

**Message Meaning:** The DD statement shown in the message is not present in the job control language. As a result, processing for the print rules data type shown in the message will be bypassed.

**System Action:** None.

**Required Action:** None.

**DSST908I** UPDATING PRINT RULES type DATA USING SYSTEM INITIALIZATION MEMBER member

type: The type of print rules data for which processing is being performed. This will be either PRINTER or OUTPUT.

member: The name of the VPS or DRS system initialization member used to determine the printer and output statement definitions to process.

**Message Meaning:** Processing for the print rules data type shown in the message is beginning using the system initialization member shown in the message.

**System Action:** None.

**Required Action:** None.

**DSST920I** PRINT RULES action FOR PRINTER printer

action: Indicates the action taken against the named printer definition (CREATED, REPLACED, or RETAINED).

printer: Indicates the name of the print rules printer definition.

**Message Meaning:** Print rules processing took the action shown in the message for the printer shown definition in the message.

**System Action:** None.

**Required Action:** None.

---

<b>DSST924I</b>	<p>PRINT RULES action FOR OUTPUT STATEMENT outref</p> <p>action: Indicates the action taken against the named output definition (CREATED, REPLACED, or RETAINED).</p> <p>printer: Indicates the name of the print rules output definition.</p> <p><b>Message Meaning:</b> Print rules processing took the action shown in the message for the output definition shown in the message.</p> <p><b>System Action:</b> None.</p> <p><b>Required Action:</b> None.</p>
<b>DSST960W</b>	<p>EXEC STATEMENT PARAMETER parm IS NOT RECOGNIZED</p> <p>parm: Indicates the JCL EXEC statement parameter that is not recognized.</p> <p><b>Message Meaning:</b> A JCL EXEC statement parameter was specified that is not recognized.</p> <p><b>System Action:</b> Processing continues, ignoring the unrecognized parameter.</p> <p><b>Required Action:</b> None.</p>
<b>DSST970E</b>	<p>SYNTAX ERROR IN STMT stmt OF MEMBER member—desc</p> <p>stmt: Indicates the statement number of the VPS or DRS control library member that contains the syntax error.</p> <p>member: Indicates the VPS or DRS control library member name that contains the syntax error.</p> <p>desc: The description of the syntax error.</p> <p><b>Message Meaning:</b> A syntax error was identified in the VPS or DRS control library member shown in the message.</p> <p><b>System Action:</b> If the syntax error was identified in the system initialization member or the inclusion/exclusion list member, processing terminates. Otherwise, processing continues with the next printer or output statement definition.</p> <p><b>Required Action:</b> Correct the syntax error and retry the process.</p>



---

<b>DSST972E</b>	<p>SYNTAX ERROR IN KEYWORD keyword OF MEMBER member—desc</p> <p>keyword: Indicates the keyword statement of the VPS or DRS control library member that contains the syntax error.</p> <p>member: Indicates the VPS or DRS control library member name that contains the syntax error.</p> <p>desc: The description of the syntax error.</p> <p><b>Message Meaning:</b> A syntax error was identified in the VPS or DRS control library member shown in the message.</p> <p><b>System Action:</b> If the syntax error was identified in the system initialization member or the inclusion/exclusion list member, processing terminates. Otherwise, processing continues with the next printer or output statement definition.</p> <p><b>Required Action:</b> Correct the syntax error and retry the process.</p>
<b>DSST976E</b>	<p>MEMBER member IS NOT A VALID PRINTER DEFINITION</p> <p>member: Indicates the VPS control library member name.</p> <p><b>Message Meaning:</b> The member name shown in the message was identified by the member inclusion or exclusion list as a VPS printer definition. However, the member does contain any statement specifying the printer CLASS, DEST, FORM, or WRITER.</p> <p><b>System Action:</b> Processing continues with the next printer definition.</p> <p><b>Required Action:</b> None.</p>
<b>DSST978E</b>	<p>MLISTMEM/XLISTMEM NOT SPECIFIED IN SYSTEM INITIALIZATION MEMBER member</p> <p>member: Indicates the VPS or DRS control library system initialization member name.</p> <p><b>Message Meaning:</b> The system initialization member shown in the message does not specify either a member inclusion list (MLISTMEM) or a member exclusion list (XLISTMEM).</p> <p><b>System Action:</b> Processing terminates.</p> <p><b>Required Action:</b> Correct the system initialization member and retry the process.</p>

---

<b>DSST980E</b>	<p>OPEN FAILED FOR FILE ddname</p> <p>ddname:               The JCL DDNAME of the file for which the OPEN failed.</p> <p><b>Message Meaning:</b> The attempt to OPEN the file shown in the message failed.</p> <p><b>System Action:</b>     Processing terminates.</p> <p><b>Required Action:</b>    Correct the JCL and retry the process.</p>
<b>DSST984E</b>	<p>request REQUEST FAILED FOR FILE DRSRULES, RETCODE=X'retcode'</p> <p>request:               The file access request type.</p> <p>retcode:               The file access request return code.</p> <p><b>Message Meaning:</b> The print rules file access request shown in the message failed with the return code shown in the message.</p> <p><b>System Action:</b>     Processing terminates.</p> <p><b>Required Action:</b>    Ensure that the DRS/STI print rules data set is properly defined in the job control language. If unable to resolve the problem, notify DRS technical support.</p>
<b>DSST988E</b>	<p>FIND FOR LIBRARY MEMBER member FAILED, RETCODE=X'retcode', RETCODE=X'retcode', RSNCODE=X'rsncode'</p> <p>member:                The VPS or DRS control library member for which the FIND request failed.</p> <p>retcode:                The FIND return code.</p> <p>rsncode:                The FIND reason code.</p> <p><b>Message Meaning:</b> The attempt to locate the VPS or DRS control library member shown in the message failed with the return code and reason code shown in the message.</p> <p><b>System Action:</b>     If the locate failure occurred while processing the system initialization member or the inclusion/exclusion list member, processing terminates. Otherwise, processing continues with the next printer or output statement definition.</p> <p><b>Required Action:</b>    Correct the VPS or DRS control library definitions and retry the process.</p>

---

## Backing Up the DRS/STI Rules Dataset

It is recommended that the DRS/STI rules dataset usage be monitored by site DASD administration personnel and put on a regular backup schedule.

To back up the DRS/STI rules data set, use a standard Access Method Services REPRO to copy the DRS/STI rules data set to a sequential file. The sequential file should have the following DCB attributes:

```
RECFM=VB
LRECL=4090
BLKSIZE=4094 (or greater)
```

Sample JCL to back up the DRS/STI rules data set is contained in member BKPRULES in data set LRS.DRS.V1R33.CNTL. That JCL is reproduced here:

```
//JOBNAME JOB (YOUR JOB CARD INFORMATION)
//*
//BACKUP EXEC PGM=IDCAMS,REGION=1M
//BKPRULES DD DSN=DRS.V1R33.BACKUP.RULES,DISP=(NEW,CATLG,DELETE),
//          UNIT=3380,VOL=SER=XXXXXX,SPACE=(CYL,(5,1),RLSE),
//          DCB=(LRECL=4090,BLKSIZE=23476,RECFM=VB)
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
        REPRO INDATASET(DRS.V1R33.RULES)
OUTFILE(BKPRULES)
/*
//*
//
```

Using the sample JCL provided, you must specify a valid unit and volume serial for your installation. In addition, the block size and space allocation parameters should be reviewed to ensure that they are appropriate for the unit chosen and the current allocations of the DRS/STI rules data set.

To restore the data to the DRS/STI rules data set, again use Access Method Services with the REUSE option on the REPRO statement.

---

## Reorganizing the DRS/STI Rules Dataset

It is recommended that the DRS/STI rules dataset usage be monitored by site DASD administration personnel and put on a regular reorganization schedule.

To reorganize the DRS/STI rules dataset, use the Access Method Services IMPORT and EXPORT commands.

Sample JCL to reorganize the DRS/STI rules data set is contained in member REORULES in data set LRS.DRS.V1R33.CNTL. That JCL is reproduced here:

```
//JOBNAME JOB (YOUR JOB CARD INFORMATION)
//*
//EXPORT EXEC PGM=IDCAMS,REGION=1M
//EXPRULES DD DSN=DRS.V1R33.EXPORTED.RULES,DISP=(NEW,CATLG,DELETE),
//          UNIT=3380,VOL=SER=XXXXXX,SPACE=(CYL,(5,1),RLSE),
//          DCB=BLKSIZE=23476
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
EXPORT DRS.V1R33.RULES -
OUTFILE(EXPRULES) -
TEMPORARY
/*
/**
//IMPORT EXEC PGM=IDCAMS,REGION=1M,COND=(0,LT)
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
IMPORT INDATASET(DRS.V1R33.EXPORTED.RULES) -
OUTDATASET(DRS.V1R33.RULES)
/*
/**
//
```

Using the sample JCL provided, you must specify a valid unit and volume serial for your installation. In addition, the block size and space allocation parameters should be reviewed to ensure that they are appropriate for the unit chosen and the current allocations of the DRS/STI rules data set.

---

# Section 17

## DRS/STI Administration

The DRS/STI Administration processing consists of a series of ISPF panels that are used to maintain data in a VSAM dataset. The data is entered by an administrator and is used by the DRS/VPI address space to place data on the spool. The panels use standard ISPF PF key conventions.

The administrator can use these panels to create and maintain the following record types in the DRS/STI rules dataset.

- Vendor** This record is the root of a hierarchy of records that allows a default profile to be associated with a vendor within application. The vendor name is unique within the dataset.
- Application** This record forms the next level in the hierarchy below the vendor. It describes the application and indicates the printer types this application can access from the STI CICS transaction. A default profile name can also be supplied. This profile can be passed in the first STI buffer from a CICS application and will be used by the DRS/VPI to retrieve the printer and its attributes and place the data on the spool. The application record is unique within vendor.
- Profile** This record is the last level in the hierarchy below vendor and application. This record will contain a printer name and optionally an associated output name. The printer and output attributes will be used during allocation of the SYSOUT dataset. The profile name can be passed in the first STI buffer from a CICS application and will be used by the DRS/VPI to retrieve the printer and output attributes. The profile is unique within application.
- Printer** This record describes a printer name and its associated attributes. The administrator supplies the CLASS, DESTINATION, FORM, and/or WRITER, or a program supplied by LRS can create these records, using the VPS control library printer members as input. The printer is unique within the dataset.
- Output** This record describes all the output statement attributes that can be dynamically associated with a SYSOUT dataset. The output name can be supplied by the STI CICS transaction or from the output name field of a profile record. The administrator can supply these values, or a program supplied by LRS can create these records, using the DRS control library output members as input. The output is unique within the dataset.
- Printer Type** This record describes a printer menu item on the STI CICS printer menu and the administration printer menu accessed from Primary Option 2. The type is used to mask the printer list panel to only the selected type.
- User** One or more of these records are used to associate printers and optionally outputs for a user. There will be one record for each printer type the user has created associations for.

---

The following is a list of the data field names that will appear on the administration panels, with their associated meanings.

<b>VENDOR</b>	User-supplied 8-byte name for a vendor.
<b>APPLICATION</b>	User-supplied 8-byte name for an application.
<b>PROFILE</b>	User-supplied 16-byte name for a profile that associates a printer and output entry.
<b>DESCRIPTION</b>	User-supplied 60-byte description of a entry in the rules dataset. (VENDOR, APPLICATION, etc.)
<b>ADDRESS</b>	Three user-supplied 60-byte lines that supply a vendor address.
<b>CITY</b>	User-supplied 40-byte name of a vendor's city.
<b>STATE</b>	User-supplied 2-byte state code for a vendor.
<b>ZIP</b>	User-supplied 9-byte field defining a vendor's zip code.
<b>COUNTRY CODE</b>	User-supplied 4-byte field defining a vendor's country code.
<b>PROVINCE</b>	User-supplied 10-byte field defining a vendor's province, if applicable.
<b>CONTACT NAME</b>	User-supplied 40 byte field defining a contact name with a vendor.
<b>CONTACT PHONE</b>	User-supplied 10-byte field defining a vendor contact's phone number.
<b>DEFAULT PROFILE</b>	User-supplied 16-byte field used to supply a printer name and optionally associate an output description with it.
<b>STA</b>	This column on the application maintenance panels indicates whether the printer type entry is enabled for this application. If enabled, this entry will appear as a menu pick on the STI CICS printer type menu.
<b>NBR</b>	This column on the application maintenance panels shows the printer code associated with this printer type.
<b>TYPE</b>	This column on the application maintenance panels and the printer maintenance panels shows the printer type name associated with this printer type. On the printer type maintenance panels it defines an 8-byte field where the type name can be supplied.
<b>PRINTER</b>	User-supplied 8-byte field that names a printer entry in the rules dataset.
<b>OUTPUT</b>	User-supplied 8-byte field that names an output entry in the rules dataset.
<b>CLASS</b>	User-supplied 1-byte field that defines a class on a printer entry.
<b>DESTINATION</b>	User-supplied 8-byte field that defines a destination on a printer entry.
<b>FORM</b>	User-supplied 8-byte field that defines a form on a printer entry.
<b>WRITER</b>	User-supplied 8-byte field that defines a writer on a printer entry.
<b>OUTPUT FIELDS</b>	User supplied values for the various JCL OUTPUT statement keywords. See MVS JCL OUTPUT statement documentation for a full description of the possible values.
<b>USERID</b>	User-supplied 8-byte value that defines a user profile record of printer and output associations.

---

## DRS/STI Administration Primary Menu

The DRS/STI Administration Primary Menu is available through the administration application. This is the first panel displayed when the administration application is started. An option is selected by entering a number or letter representing the option in the OPTION field and pressing the ENTER key.

```
----- DRS/STI Primary Menu -----  
OPTION ===>  
  
  1 Vendor      - Process vendor information  
  2 Printer     - Process printer information  
  3 Output      - Process output information  
  4 Printer Types - Process printer type information  
  5 Users       - Process userid information  
  
  X EXIT        - Terminate  
  
Enter END command to terminate.
```

---

## DRS/STI Administration Vendor List

The DRS/STI Administration Vendor List is available by selecting option 1 from the Primary option menu. This scrollable list displays all vendors that have been defined. The list entries can be maintained through the use of the select command from the command line or by entering one of the listed line commands in the select field in front of the desired entry. If a select command is entered from the command line and a name is supplied, but the name does not exist as a vendor entry, the vendor add panel will be displayed. If the name does exist, the vendor update panel will be displayed. The (L)ist line command displays the application list panel, which represents application entries defined for this vendor. The (D)elete line command allows this vendor to be deleted, including any application and profile entries defined for this vendor.

```
-----DRS/STI Vendor List -----
COMMAND ===> - - - - - SCROLL ===> CSR

Line Commands - (D)elete, (L)ist, (S)elect

Vendor      Description
-----
_ IBM       International Business Machines Corporation
_ LRS       Levi, Ray and Shoup Inc.
***** BOTTOM OF DATA *****
```



---

## DRS/STI Administration Vendor Add

The DRS/STI Administration Vendor Add is available by entering the (S)elect command line command from the vendor list panel. This panel allows vendor information to be saved in the rules dataset. All fields on this panel are for informational purposes, except the vendor name, which is required. Entering the END or RETURN command line command will add the vendor entry to the rules dataset. To return to the vendor list without adding the vendor, enter the (CAN)cel command line command.

```
- - - - - DRS/STI Vendor Add- - - - -  
COMMAND ===>  
  
Vendor           : HRVENDOR  
Description      : Human Resources Vendor  
Address          : One Human Resources Way  
                 :  
                 :  
City             :  
State            :  
Zip              :  
Country Code    :  
Province        :  
Contact Name    :  
                Phone :  
Contact Name    :  
                Phone :  
Enter CANCEL to terminate.  
Enter END to perform add.
```

---

## DRS/STI Administration Vendor Update

The DRS/STI Administration Vendor Update is available by entering the (S)elect command line command with the name of an existing vendor entry or by keying the (S)elect line command in the select field of an entry on the vendor list panel. This panel allows all the vendor information to be modified except the vendor name. All fields on this panel are for informational purposes, except the vendor name. Entering the END or RETURN command line command will add the vendor entry to the rules dataset. To return to the vendor list without updating the vendor, enter the (CAN)cel command line command.

```
----- DRS/STI Vendor Update -----  
COMMAND ===>  
  
Vendor           : LRS  
Description      : Levi, Ray and Shoup Inc.  
Address          : 2401 West Monroe Street  
                 :  
                 :  
City             : Springfield  
State            : IL  
Zip              : 62704  
Country Code    : USA  
Province        :  
Contact Name    :  
                Phone :  
Contact Name    :  
                Phone :
```

Enter CANCEL to terminate.  
Enter END to perform update.

---

## DRS/STI Administration Vendor Delete

The DRS/STI Administration Vendor Delete is available by entering the (D)delete line command in the select field of a vendor list entry and pressing ENTER. This panel displays the current vendor information as confirmation prior to deleting the vendor entry from the rules dataset. If the delete is confirmed by pressing the ENTER key, the vendor entry and any application and profile entries for this vendor will also be deleted. To return to the vendor list without deleting the vendor, enter the END or RETURN command line command.

```
----- DRS/STI Vendor Delete -----  
COMMAND ===>  
  
Vendor           : LRS  
Description      : Levi, Ray and Shoup Inc.  
Address         : 2401 West Monroe Street  
                :  
                :  
City            : Springfield  
State           : IL  
Zip             : 62704  
Country Code    : USA  
Province        :  
Contact Name    :  
                :  
                :  
Contact Name    :  
                :  
                :  
                :  
  
Warning: All application and profile records for this vendor  
will also be deleted.  
  
Press ENTER to confirm delete request.  
Enter END to cancel delete request.
```

---

## DRS/STI Administration Application List

The DRS/STI Administration Application List is available by entering a (L)ist line command in the select field for the desired vendor. This scrollable list displays all applications defined for this vendor. The list entries can be maintained through the use of the (S)elect command from the command line or by entering one of the listed line commands in the select field in front of the desired entry. If a (S)elect command is entered from the command line and a name is supplied, but the name does not exist as an application entry, the application add panel will be displayed. If the name does exist, the application update panel will be displayed. The (L)ist line command displays the profile list panel, which represents predefined associations between printers and outputs. The (D)elele line command allows this application and any profiles defined for this application to be deleted.

```
----- DRS/STI Application List -----
COMMAND ==>                                SCROLL ==> CSR

Line Commands - (D)elele, (L)ist, (S)elect

HRVENDOR      Human Resources Vendor

Appl          Description
-----
_ PAYROLL     Payroll Application
_ PERSONEL    Personnel Application
*****BOTTOM OF DATA*****
```

---

## DRS/STI Administration Application Add

The DRS/STI Administration Application Add is available by entering the (S)elect command line command from the application list panel. This panel allows application information for a specific vendor to be saved in the rules dataset. The only required field is the application name. The scrollable list represents all the available printer types, and the 'Sta' column represents the enabled or disabled status of each type for this application. If the status is enabled, then this printer type becomes available to CICS users through the STI transaction. The types are enabled or disabled by placing either a 'E' or 'D' in the select field of the desired list entry. Entering the END or RETURN command line command will add the application entry to the rules dataset. To return to the application list without adding the application, enter the (CAN)cel command line command.

```
----- DRS/STI Application Add -----
COMMAND ==>                                SCROLL ==> CSR

Vendor           : HRVENDOR
Description      : HUMAN RESOURCES VENDOR
Application      : NEWAPP
Description      : New Application
Default Profile  :

   Sta  Nbr   Type      Description
   ---  ---  - - - - -  - - - - -
- ENA  1     Printer  - Select From Printer Types
- ENA  2     Fax      - Select From Fax Types
- ENA  3     Email    - Select From Email Types
- ENA  4     Archive  - Select From Archive Types
***** BOTTOM OF DATA*****

Enter CANCEL to terminate. Enter UP command to scroll backward.
Enter END to perform add. Enter DOWN command to scroll forward.
```

**Default Profile:** Defines a profile entry that identifies a printer and optionally an output that is to be used to place output on the spool if no printer or output information is supplied.

---

## DRS/STI Administration Application Update

The DRS/STI Administration Application Update is available by entering the (S)elect command line command with the name of an existing application entry or by keying the (S)elect line command in the select field of a application list entry on the application list panel. This panel allows all application information except application name to be modified and saved in the rules dataset. Entering the END or RETURN command line command will update the application entry in the rules dataset. To return to the application list without updating the application, enter the (CAN)cel command line command.

```
----- DRS/STI Application Update -----
COMMAND ===>                                SCROLL ===> CSR

Vendor           :   HRVENDOR
Description      :   HUMAN RESOURCES VENDOR
Application      :   PAYROLL
Description      :   Payroll Application
Default Profile  :

   Sta  Nbr    Type      Description
   ---  ---  - - - - -  - - - - -
- ENA   1    Printer    - Select From Printer Types
- ENA   2     Fax       - Select From Fax Types
- ENA   3    Email     - Select From Email Types
- ENA   4    Archive   - Select From Archive Types
***** BOTTOM OF DATA *****

Enter CANCEL to terminate.   Enter UP command to scroll backward.
Enter END to perform update.  Enter DOWN command to scroll forward.
```

**Default Profile:** Defines a profile entry that identifies a printer and optionally an output that is to be used to place output on the spool if no printer or output information is supplied.

---

## DRS/STI Administration Application Delete

The DRS/STI Administration Application Delete is available by entering the (D)delete line command in the select field of an application list entry and pressing ENTER. This panel displays the current application information as confirmation prior to deleting the application entry from the rules dataset. If the delete is confirmed by pressing the ENTER key, the application entry and any profile entries for this application will also be deleted. To return to the application list without deleting the application entry, enter the END or RETURN command line command.

```
----- DRS/STI Application Delete -----
COMMAND ===>                                SCROLL ===> CSR

Vendor           : HRVENDOR
Description      : HUMAN RESOURCES VENDOR
Application      : PAYROLL
Description      : Payroll Application
Default Profile  :

   Sta  Nbr  Type      Description
   ---  ---  ---      -----
- ENA   1   Printer - Select From Printer Types
- ENA   2   Fax      - Select From Fax Types
- ENA   3   Email    - Select From Email Types
- ENA   4   Archive  - Select From Archive Types
***** BOTTOM OF DATA *****

Warning: All profile records for this vendor will also be
deleted.

Enter END to terminate.      Enter UP  command to scroll backward.
Press ENTER to perform delete. Enter DOWN command to scroll forward.
```

---

## DRS/STI Administration Profile List

The DRS/STI Administration Profile List is available by entering a (L)ist line command in the select field for the desired application. This scrollable list displays all profiles defined for this application. The list entries can be maintained through the use of the (S)elect command from the command line or by entering one of the listed line commands in the select field in front of the desired entry. If a (S)elect command is entered from the command line, and a name is supplied but the name does not exist as a profile entry, the profile add panel will be displayed. If the name does exist, the profile update panel will be displayed. The (D)elete line command allows the selected profile to be deleted.

```
----- DRS/STI Profile List -----
COMMAND ===>                                SCROLL ===> CSR

Line Commands - (D)elete, (S)elect

HRVENDOR      Human Resources Vendor
PAYROLL       Payroll Application

Profile       Description
-----

_ PAYCHECK    Payroll Checks
*****
***** BOTTOM OF DATA *****
```



---

## DRS/STI Administration Profile Add

The DRS/STI Administration Profile Add is available by entering the (S)elect command line command from the profile list panel. This panel allows profile information for a specific vendor/application to be saved in the rules dataset. The profile name and either the printer name or output name is required. The profile represents a predefined printer and output association. The printer name corresponds to an entry maintained by the primary option menu 2 selection, and the output corresponds to an entry maintained by the primary option menu 3 selection. Entering the END or RETURN command line command will add the profile entry to the rules dataset. To return to the profile list without adding the profile, enter the (CAN)cel command line command.

```
----- DRS/STI Profile Add -----  
COMMAND ===>  
  
Vendor      : HRVENDOR  
Description  : Human Resources Vendor  
Application  : PAYROLL  
Description  : Payroll Application  
  
Profile     : BENEFIT  
Description  : Benefit Statement  
  
Printer     : PRT1  
Output      :  
  
Enter CANCEL to terminate.  
Enter END    to perform add.
```

**Printer:** Identifies a defined printer whose attributes will be used to place a dataset on the JES spool.

**Output:** Identifies a defined output description that will be used to dynamically define an output statement to be associated with a SYSOUT dataset.

---

## DRS/STI Administration Profile Update

The DRS/STI Administration Profile Update is available by entering the (S)elect command line command with the name of an existing profile entry or by keying the (S)elect line command in the select field of an entry on the profile list panel. This panel allows all profile information except profile name to be modified and saved in the rules dataset. Entering the END or RETURN command line command will update the profile entry in the rules dataset. To return to the profile list without updating the profile, enter the (CAN)cel command line command.

```
----- DRS/STI Profile Update -----  
COMMAND ===>  
  
Vendor      : HRVENDOR  
Description  : Human Resources Vendor  
Application  : PAYROLL  
Description  : Payroll Application  
  
Profile     : BENEFIT  
Description  : Benefit Statement  
  
Printer     : PRT1  
Output      :  
  
Enter CANCEL to terminate.  
Enter END    to perform update.
```

**Printer:** Identifies a defined printer whose attributes will be used to place a dataset on the JES spool.

**Output:** Identifies a defined output description that will be SYSOUT to dynamically define an output statement to be associated with a SYSOUT dataset.

---

## DRS/STI Administration Profile Delete

The DRS/STI Administration Profile Delete is available by entering the (D)delete line command in the select field of a profile list entry and pressing ENTER. This panel displays the current profile information as confirmation prior to deleting the profile entry from the rules dataset. The delete is confirmed by pressing the ENTER key. To return to the profile list without deleting the profile, enter the END or RETURN command line command.

```
----- DRS/STI Profile Delete -----  
COMMAND ===>  
  
Vendor      : HRVENDOR  
Description : Human Resources Vendor  
Application : PAYROLL  
Description : Payroll Application  
  
Profile     : BENEFIT  
Description : Benefit Statement  
  
Printer     : PRT1  
Output      :  
  
Press ENTER to confirm delete request.  
Enter END   to cancel delete request.
```

---

## DRS/STI Administration Printer Type Menu

The DRS/STI Administration Printer Type Menu is available by entering option 2 or 5 from the command line on the Primary Option Menu and pressing ENTER. This menu lists the defined printer types and is used to mask the Printer List panel or the Userid List panel to the selected type. A type is selected by keying the number of the type in the command line and pressing ENTER.

```
----- DRS/STI Printer Type Menu -----  
OPTION ==>>                                SCROLL => PAGE  
  
1   Printer   -   Select From Printer Types  
2   Fax       -   Select From Fax Types  
3   Email     -   Select From Email Types  
4   Archive   -   Select From Archive Types
```

---

## DRS/STI Administration Printer List

The DRS/STI Administration Printer List is available by entering the desired option number from the command line on the Printer Type Menu and pressing ENTER. This scrollable list displays all the defined printers of the selected type.

When invoked as a result of initially selecting Primary Option 2, these list entries can be maintained through the use of the (S)elect command from the command line, or by entering one of the listed line commands in the select field of the desired entry. If a (S)elect command is entered from the command line, and a name is supplied but the name does not exist as a printer entry, the printer add panel will be displayed. If the name does exist, the printer update panel will be displayed. The (D)elele line command allows the selected printer to be deleted.

When invoked from the Userid Update or Add panels, the (D)elele line command is unavailable, and the (S)elect line command selects the printer entry and places the name under the printer column on the Userid Update or Add panels.

```
----- DRS/STI Printer List -----
COMMAND ===>                                SCROLL ===> PAGE

  Line Commands - (D)elele, (S)elect

  Printer      Description
  -----
_ PCL PCL Printer Description
***** BOTTOM OF DATA *****
```

---

## DRS/STI Administration Printer Add

The DRS/STI Administration Printer Add is available by entering the (S)elect command line command from the printer list panel. This panel allows printer information necessary to allocate a SYSOUT dataset on the JES spool to be defined and saved in the rules dataset. The printer name and at least one of the SYSOUT attributes (Class, Destination, Form, Writer) are required. Entering the END or RETURN command line command will add the printer entry to the rules dataset. To return to the printer list without adding the printer, enter the (CAN)cel command line command.

```
----- DRS/STI Printer Add -----  
COMMAND ===>  
  
Type           : Printer  
Printer        : NEWPRT  
Description    : Printer located in building 3  
  
Class          : A  
Destination   : U100  
Form          :  
Writer        :  
  
Enter CANCEL   to terminate.  
Enter END      to perform add.
```

**Class:** Defines the SYSOUT class to be assigned to the SYSOUT dataset.

**Destination:** Defines the destination to be assigned to the SYSOUT dataset.

**Form:** Defines the form to be assigned to the SYSOUT dataset.

**Writer:** Defines the writer to be assigned to the SYSOUT dataset.

---

## DRS/STI Administration Printer Update

The DRS/STI Administration Printer Update is available by entering the (S)elect command line command with the name of an existing printer entry or by keying the (S)elect line command in the select field of a printer list entry on the printer list panel. This panel allows all printer information except printer name to be modified and saved in the rules dataset. At least one of the attribute fields (Class, Destination, Form, Writer) is required. Entering the END or RETURN command line command will update the printer entry in the rules dataset. To return to the printer list without updating the printer, enter the (CAN)cel command line command.

```
----- DRS/STI Printer Update -----  
COMMAND ===>  
  
Type          : Printer  
Printer       : NEWPRT  
Description   : Printer located in building 3  
  
Class        : A  
Destination  : U100  
Form         :  
Writer       :  
  
Enter CANCEL  to terminate.  
Enter END     to perform update.
```

**Class:** Defines the SYSOUT class to be assigned to the SYSOUT dataset.

**Destination:** Defines the destination to be assigned to the SYSOUT dataset.

**Form:** Defines the form to be assigned to the SYSOUT dataset.

**Writer:** Defines the writer to be assigned to the SYSOUT dataset.

---

## DRS/STI Administration Printer Delete

The DRS/STI Administration Printer Delete is available by entering the (D)delete line command in the select field of a printer list entry and pressing ENTER. This panel displays the current printer information as confirmation prior to deleting the printer entry from the rules dataset. The delete is confirmed by pressing the ENTER key. To return to the printer list without deleting the printer, enter the END or RETURN command line command.

```
----- DRS/STI Printer Delete -----  
COMMAND ===>  
  
Type       : Printer  
Printer    : NEWPRT  
Description : Printer located in building 3  
  
Class      : A  
Destination : U100  
Form       :  
Writer     :  
  
Press ENTER to confirm delete request.  
Enter END   to cancel delete request.
```



---

## DRS/STI Administration Output List

The DRS/STI Administration Output List is available by entering option number 3 in the command line of the primary option menu and pressing ENTER. This scrollable list displays all the output definitions. The list entries can be maintained through the use of the (S)elect command from the command line or by entering one of the listed line commands in the select field of the desired entry. If a (S)elect command is entered from the command line and a name is supplied, but the name does not exist as an output entry, the output add panel will be displayed. If the name does exist, the output update panel will be displayed. The (D)elele line command allows the selected output to be deleted.

When invoked from the Userid Update or Add panels, the (D)elele line command is unavailable, and the (S)elect line command selects the output entry and places the name under the output column on the Userid Update or Add panels.

```
----- DRS/STI Output List -----
COMMAND ==>                                SCROLL ==> PAGE

  Line Commands - (D)elele, (S)elect

  Output          Description
  -----
_ OUT1           output 1 description
_ OUT2           output 2 description
***** BOTTOM OF DATA *****
```

---

## DRS/STI Administration Output Add Panel 1

The DRS/STI Administration Output Add Panels are available by entering the (S)elect command line command from the output list panel. These scrollable panels allow output information that can be dynamically created and associated with a SYSOUT dataset to be saved in the rules dataset. The output name is the only required entry. Entering the END or RETURN command line command will add the output entry to the rules dataset. To return to the output list without adding the output, enter the (CAN)cel command line command.

```
-----DRS/STI Output Add -----Screen 1 of 3
COMMAND ===>

Output name  :
Description  :

CHARS       :          CLASS       :
COPIES      :          COPY GROUPS :
DATAACK     :          DESTINATION  :
FCB         :          FORMDEF      :
FORMS       :          NOTIFY      :
OUTDISP     :          PAGEDEF      :
UCS         :          WRITER       :
BURST       :          CKPTLINE     :          CKPTPAGE     :
CKPTSEC     :          COMPACT      :          CONTROL     :
DPAGELBL    :          FLASH        :          FLASH COUNT  :
GROUP ID    :          INDEX        :          LINDEX      :
LINECT      :          MODIFY       :          OUTBIN      :
PIMSG       :          PIMSG COUNT    :          PRMODE      :
PRTY        :          SYSAREA      :          TRC         :
```

Enter CANCEL to terminate.  
Enter END to perform add.

**NOTE:** See the Output JCL statement for a description of data that can be entered on an output statement.

---

## DRS/STI Administration Output Add Panel 2

The DRS/STI Administration Output Add Panels are available by entering the (S)elect command line command from the output list panel. These scrollable panels allow output information that can be dynamically created and associated with a SYSOUT dataset to be saved in the rules dataset. The output name is the only required entry. Entering the END or RETURN command line command will add the output entry to the rules dataset. To return to the output list without adding the output, enter the (CAN)cel command line command.

```
-----DRS/STI Output Add -----Screen 2 of 3
COMMAND ===>

Output name      :
NAME             :
TITLE           :
ADDRESS         :
                :
                :
                :
DEPARTMENT      :
BUILDING        :
ROOM            :
USERLIB         :
                :
                :
                :
                :
                :
Enter CANCEL    to terminate.
Enter END       to perform add.
```

**NOTE:** See the Output JCL statement for a description of data that can be entered on an output statement.

---

## DRS/STI Administration Output Add Panel 3

The DRS/STI Administration Output Add Panels are available by entering the (S)elect command line command from the output list panel. These scrollable panels allow output information that can be dynamically created and associated with a SYSOUT dataset to be saved in the rules dataset. The output name is the only required entry. Entering the END or RETURN command line command will add the output entry to the rules dataset. To return to the output list without adding the output, enter the (CAN)cel command line command.

```
-----DRS/STI Output Add -----Screen 3 of 3
COMMAND ===>

Output name  :
USERDATA    :
           :
           :
           :
           :
           :
           :
           :
           :
           :
           :
           :
           :
           :
           :

Enter CANCEL   to terminate.
Enter END      to perform add.
```

**NOTE:** See the Output JCL statement for a description of data that can be entered on an output statement.

---

# DRS/STI Administration Output Update Panel 1

The DRS/STI Administration Output Update Panels are available by entering the (S)elect command line command with the name of an existing output entry or by keying the (S)elect line command in the select field of a output list entry on the output list panel. This panel allows all output information except output name to be modified and saved in the rules dataset. This information is then used to dynamically create output statements that can be associated with a SYSOUT dataset on the JES spool. Entering the END or RETURN command line command will update the output entry in the rules dataset. To return to the output list without updating the output, enter the (CAN)cel command line command.

```
-----DRS/STI Output Update ----- Screen 1 of 3
COMMAND ===>

Output name  :
Description  :

CHARS       :          CLASS       :
COPIES      :          COPY GROUPS  :
DATAACK     :          DESTINATION  :
FCB         :          FORMDEF      :
FORMS       :          NOTIFY       :
OUTDISP     :          PAGEDEF      :
UCS         :          WRITER       :
BURST       :          CKPTLINE     :          CKPTPAGE      :
CKPTSEC     :          COMPACT       :          CONTROL      :
DPAGELBL   :          FLASH         :          FLASH COUNT  :
GROUP ID    :          INDEX         :          LINDEX       :
LINECT     :          MODIFY         :          OUTBIN       :
PIMSG      :          PIMSG COUNT     :          PRMODE       :
PRTY       :          SYSAREA      :          TRC          :
```

Enter CANCEL to terminate.  
Enter END to perform update.

**NOTE:** See the Output JCL statement for a description of data that can be entered on an output statement.

---

## DRS/STI Administration Output Update Panel 2

The DRS/STI Administration Output Update Panels are available by entering the (S)elect command line command with the name of an existing output entry or by keying the (S)elect line command in the select field of a output list entry on the output list panel. This panel allows all output information except output name to be modified and saved in the rules dataset. This information is then used to dynamically create output statements that can be associated with a SYSOUT dataset on the JES spool. Entering the END or RETURN command line command will update the output entry in the rules dataset. To return to the output list without updating the output, enter the (CAN)cel command line command.

```
-----DRS/STI Output Update -----Screen 2 of 3
COMMAND ===>

Output name      :
NAME             :
TITLE            :
ADDRESS          :
                 :
                 :
                 :
DEPARTMENT       :
BUILDING         :
ROOM             :
USERLIB          :
                 :
                 :
                 :
                 :
                 :
                 :
                 :
                 :
Enter CANCEL     to terminate.
Enter END        to perform update.
```

**NOTE:** See the Output JCL statement for a description of data that can be entered on an output statement.

---

## DRS/STI Administration Output Update Panel 3

The DRS/STI Administration Output Update Panels are available by entering the (S)elect command line command with the name of an existing output entry or by keying the (S)elect line command in the select field of an entry on the output list panel. This panel allows all output information except output name to be modified and saved in the rules dataset. This information is then used to dynamically create output statements that can be associated with a SYSOUT dataset on the JES spool. Entering the END or RETURN command line command will update the output entry in the rules dataset. To return to the output list without updating the output, enter the (CAN)cel command line command.

```

-----DRS/STI Output Update -----Screen 3 of 3
COMMAND ===>

Output name   :
USERDATA     :
              :
              :
              :
              :
              :
              :
              :
              :
              :
              :
              :
              :
              :
              :
              :

Enter CANCEL  to terminate.
Enter END     to perform update.
```

**NOTE:** See the Output JCL statement for a description of data that can be entered on an output statement.

---

# DRS/STI Administration Output Delete Panel 1

The DRS/STI Administration Output Delete Panels are available by entering the (D)elele line command in the select field of an output list entry and pressing ENTER. This series of panels displays the current output information as confirmation prior to deleting the output entry from the rules dataset. The delete is confirmed by pressing the ENTER key. To return to the output list without deleting the output, enter the END or RETURN command line command.

```
-----DRS/STI Output Delete -----Screen 1 of 3
COMMAND ==>

Output name  :
Description  :

CHARS       :          CLASS       :
COPIES      :          COPY GROUPS  :
DATAACK     :          DESTINATION  :
FCB         :          FORMDEF       :
FORMS       :          NOTIFY        :
OUTDISP     :          PAGEDEF        :
UCS         :          WRITER          :
BURST       :          CKPTLINE       :          CKPTPAGE       :
CKPTSEC     :          COMPACT        :          CONTROL       :
DPAGELBL   :          FLASH          :          FLASH COUNT  :
GROUP ID    :          INDEX          :          LINDEX        :
LINECT     :          MODIFY         :          OUTBIN        :
PIMSG       :          PIMSG COUNT    :          PRMODE       :
PRTY       :          SYSAREA       :          TRC          :
```

Press ENTER to confirm delete request.  
Enter END to cancel delete request.



---

## DRS/STI Administration Delete Panel 2

The DRS/STI Administration Output Delete Panels are available by entering the (D)delete line command in the select field of an output list entry and pressing ENTER. This series of panels displays the current output information as confirmation prior to deleting the output entry from the rules dataset. The delete is confirmed by pressing the ENTER key. To return to the output list without deleting the output, enter the END or RETURN command line command.

```
-----DRS/STI Output Delete -----Screen 2 of 3
COMMAND ===>

Output name  :
NAME        :
TITLE       :
ADDRESS     :
           :
           :
           :
DEPARTMENT  :
BUILDING    :
ROOM        :
USERLIB     :
           :
           :
           :
           :
           :
           :
Press ENTER   to confirm delete request
Enter END     to cancel delete request.
```

---

## DRS/STI Administration Output Delete Panel 3

The DRS/STI Administration Output Delete Panels are available by entering the (D)elele line command in the select field of an output list entry and pressing ENTER. This series of panels displays the current output information as confirmation prior to deleting the output entry from the rules dataset. The delete is confirmed by pressing the ENTER key. To return to the output list without deleting the output, enter the END or RETURN command line command.

```
-----DRS/STI Output Delete -----Screen 3 of 3
COMMAND ===>

Output name   :
USERDATA      :
              :
              :
              :
              :
              :
              :
              :
              :
              :
              :
              :
              :
              :

Press ENTER   to confirm delete request.
Enter END     to cancel delete request.
```

---

## DRS/STI Administration Printer Type List

The DRS/STI Administration Printer Type List is available by entering option number 4 in the command line of the primary option menu and pressing ENTER. This scrollable list displays all the defined printer types. The list entries can be maintained through the use of the (S)elect command from the command line or by entering one of the listed line commands in the select field in front of the desired entry. If a (S)elect command is entered from the command line and a name is supplied, but the name does not exist as a printer type entry, the printer type add panel will be displayed. If the name does exist, the printer type update panel will be displayed. The (D)eleate line command allows the selected printer type to be deleted.

```
----- DRS/STI Printer Type List -----
COMMAND ===>                                SCROLL ===> PAGE

Line Commands - (D)eleate, (S)elect

  Nbr   Type           Description
  ---   -
_  1    Printer - Select From Printer Types
_  2    Fax          - Select From Fax Types
_  3    Email        - Select From Email Types
_  4    Archive      - Select From Archive Types
***** BOTTOM OF DATA *****
```

---

## DRS/STI Administration Printer Type Add

The DRS/STI Administration Printer Type Add panel is available by entering the (S)elect command line command from the Printer Type List panel. This panel allows a printer type menu entry to be created and saved in the rules dataset. This type will then appear on the printer type menu and will allow the printer list to be masked to that type. All fields are required. Entering the END or RETURN command line command will add the printer type entry to the rules dataset. To return to the printer type list without adding the printer type, enter the (CAN)cel command line command.

```
-----DRS/STI Printer Type Add -----  
COMMAND ===>  
  
Number      : 5  
Type        : Newtype  
Description  : - New type of printer  
  
Enter CANCEL to terminate.  
Enter END    to perform add.
```

- Number:** Defines the printer type number as it appears in the printer type entries.
- Type:** Defines the printer type name as it will appear on the printer list screen as the printer name column heading.
- Description:** Defines the menu text for the printer type menu.

---

## DRS/STI Administration Printer Type Update

The DRS/STI Administration Printer Type Update is available by entering the (S)elect command line command with the name of an existing Printer type entry or by keying the (S)elect line command in the select field of a printer type list entry on the printer type list panel. This panel allows printer type information except the type number to be modified and saved in the rules dataset. This modified information is then available from the administration printer selection menu and the CICS printer selection menu if the application is enabled for this printer type. Entering the END or RETURN command line command will update the printer type in the rules dataset. To return to the printer type list without updating the printer type, enter the (CAN)cel command line command.

```
----- DRS/STI Printer Type Update -----  
COMMAND ===>  
  
      Number      : 5  
      Type        : Newtype  
      Description  : - New type of printer  
  
Enter CANCEL to terminate.  
Enter END      to perform update.
```

- Number:** Defines the printer type number as it appears in the printer type entries.
- Type:** Defines the printer type name as it will appear on the printer list screen as the printer name column heading.
- Description:** Defines the menu text for the printer type menu.

---

## DRS/STI Administration Printer Type Delete

The DRS/STI Administration Printer Type Delete is available by entering the (D)delete line command in the select field of a printer type list entry and pressing ENTER. This panel displays the current printer type information as confirmation prior to deleting the printer type entry from the rules dataset. The delete is confirmed by pressing the ENTER key. To return to the printer type list without deleting the printer type, enter the END or RETURN command line command.

```
----- DRS/STI Printer Type Delete -----  
COMMAND ===>  
  
Number      : 5  
Type        : Newtype  
Description  : - New type of printer  
  
Enter END   to terminate.  
Press ENTER to perform delete.
```

---

## DRS/STI Administration Userid List

The DRS/STI Administration Userid List is available by selecting option 5 from the primary option menu and then selecting an option from the Printer type menu. This scrollable list displays all userids that have been defined for the specified printer type. This list is maintained through the use of the (S)elect command from the command line or by entering one of the listed line commands in the select field in front of the desired entry. If a select command is entered from the command line, and a name is supplied, but the name does not exist as a userid record, the userid add panel will be displayed. If the name does exist, the userid update panel will be displayed. The (D)eleate line command will display the userid delete panel in order to confirm the userid prior to deletion.

```
----- DRS/STI Userid List -----  
COMMAND ==>                                SCROLL ==> CSR  
  
Line Commands - (D)eleate, (S)elect  
  
Userid  
-----  
_ LRS00  
_ LRS01  
***** BOTTOM OF DATA *****
```

---

## DRS/STI Administration Userid Add

The DRS/STI Administration Userid Add is available by entering the (S)elect command line command with no name or with the name of a new userid entry. This panel allows the userid profile information to be created and saved in the rules dataset. This panel contains 16 entries masked by the selected printer type that can be tailored by the user. Each entry can define a printer and, optionally, an output that will be used for printing if the entry is selected from the CICS interface. The entries are tailored by entering the (P)rinters line command in the select field of the entry to be changed and pressing ENTER to display the System Printer List. An entry can be selected from the System Printer List and will be placed in the printer column of the userid update panel. Optionally, if an output is to be associated with this printer, it can be selected by keying the (O)utputs line command in the select field of the entry and pressing ENTER. The System Output List will then be displayed, from which an entry can be selected. The selected entry will be placed in the output column of the entry on the userid update panel. Entering the END or RETURN command line command will add the userid record to the rules dataset. To return to the userid list without adding the userid, enter the (CAN)cel command line command.

```
----- DRS/STI Userid Add -----  
COMMAND ===>  
  
Userid:  LRS000  Line Commands - (O)utputs, (P)rinters  
Printer  Output  Description  
-----  
-  
-  
-  
-  
-  
-  
-  
-  
-  
-  
-  
-  
-  
-  
-  
-  
-  
-  
-  
-  
-  
  
Enter CANCEL to terminate.  Enter END to perform add.
```





---

## DRS/STI Administration Userid Delete

The DRS/STI Administration Userid Delete is available by entering the (D)delete line command in the select field of a userid list entry and pressing ENTER. This panel displays the current userid information as confirmation prior to deleting the userid record from the rules dataset. If the delete is confirmed by pressing the ENTER key, the userid record will be deleted. To return to the userid list without deleting the userid, enter the END or RETURN command line command.

```
----- DRS/STI Userid Delete -----  
COMMAND ===>  
  
  Userid:  LRS000  
  Printer  Output  Description  
-----  
_ PCL      OUT1    PCL Printer Description  
_ NEWPRT   OUT2    printer located in building 3  
  
_  
_  
_  
_  
_  
_  
_  
_  
_  
_  
_  
_  
_  
_  
_  
_  
_  
_ Enter CANCEL to terminate.   Enter END to perform add.
```

---

## Section 18

# DRS/STI Under CICS

The CICS transaction interface is a series of screens that allows a user to select a printer and, optionally, an output description, to associate with the current print request. The user can create printer and output associations unique to his user ID, and these combinations are saved in a profile for later use. Printers are categorized into types, such as normal printers types and fax types, that mask the user profile list screen to these types. The user profile list contains 16 entries where printer and output associations can be created. After creating one or more entries in the list, an entry can be selected in order to supply attributes for the current print request.

The STI screens run as a CICS started transaction that passes a commarea. This commarea is described by a DSECT (DSSTSTI) that is supplied in file LRS.DRS.V1R33.MACLIB. The calling transaction should supply information in the commarea, including a return transaction ID, and then start the STI transaction, passing the commarea followed by a normal return to CICS. The STI transaction takes control of the terminal until the user chooses a printer to use for this request or returns without making a selection.

After the user has made a selection, the STI transaction will start the transaction passed in the commarea. The original updated commarea will be returned to the transaction and can be obtained using a CICS RETRIEVE command. The commarea will contain return codes that indicate the action taken by the user or any errors that may have occurred. The caller of the STI transaction should supply a vendor and application name, which is used to access an application record. The application record is used to tailor the initial printer menu to the types allowed by the application. The user name should also be supplied, and is used as a key for accessing the user's profiled associations for this printer type.

Space is also provided in the commarea for two control blocks that can be used with the DRS call interface. The control blocks are a DRIB (DRSDRIB) and a DROB (DRSDROB) which represent a printer and output description. These areas can be passed into the DRS call interface to allocate a SYSOUT dataset and, optionally, associate a dynamic output description with it. The calling transaction could then invoke DRS/API, using PUT requests to place data in the SYSOUT dataset that was allocated. Alternatively, if DRS virtual printers are being used, the transaction can pass the printer and output name in the first buffer of a CICS SEND request in a special format that can be recognized by the DRS/VPI address space. The format of this buffer is described in [“Smart Tag Control Area” on page 16.4](#). The data can then be used to dynamically alter the SYSOUT characteristics for this virtual printer for this report. This method allows a few virtual printers to create reports for many VPS printers.

---

## Screen Descriptions

This section describes each of the DRS/STI CICS screens. These screens can be invoked from another CICS application to allow end users to dynamically select printers for this application's reporting.

### DRS/STI CICS Printer Menu

The DRS/STI CICS Printer Menu provides the user the ability to mask printer selections to a specific type of printer. These types can be application vendor or installation defined. The types that appear on this menu are the same types that the administrator enables on either the application add or application update administrator panels. One of the types can be selected by either keying the (S)elect line command in the select field in front of the entry or by pressing enter with the cursor positioned in the select field of the entry.

DRS/STI Printer Menu			Profile:	LRS000
			Termid:	A000
_	1	Printer - Select From Printer Types		
_	2	Fax - Select From Fax Types		
_	3	Email - Select From Email Types		
_	4	Archive - Select From Archive Types		

PF1 = Help PF3 = Terminate

---

## DRS/STI CICS User Profile List

The DRS/STI CICS User Profile List is available by selecting one of the available types from the Printer Menu and pressing ENTER. This list consists of sixteen entries masked by the selected printer type that can be tailored by the user. Each entry can define a printer and optionally an output that will be used for printing if the entry is selected. The entries are tailored by positioning the cursor in the select field of the entry to be changed and pressing the PF5 key to display the System Printer List. An entry can be selected from the System Printer List and will be placed in the entry at the cursor position. If an output is to be associated with this printer, it can be selected by pressing the PF6 key while positioned to the same entry, and the System Output List will be displayed. If an entry is selected from this list, it will be placed in the output column of the entry at the cursor position. The tailored entries will be saved for the user before exiting and are available in future sessions. After tailoring an entry, it can be selected for use in printing the current report by placing the (S)elect line command in the select field in front of the desired entry. If no selection is desired, press the PF3 key to return to the Printer Menu.

DRS/STI User Profile List			Profile: LRS000
Printer	Output	Description	Termid: A000
__ PCL	OUT1	PCL Printer Description	
—			
—			
—			
—			
—			
—			
—			
—			
—			
—			
—			
—			
—			
—			
—			

PF1 = Help   PF3 = Return   PF5 = Printer List   PF6 = Output List

---

## DRS/STI CICS System Printer List

The DRS/STI CICS System Printer List is available by positioning the cursor on one of the sixteen entries on the User Profile List screen and pressing the PF5 key. This scrollable list displays all the defined printers for the selected type. The list can be positioned by pressing the PF7 key to scroll UP and pressing the PF8 key to scroll DOWN. The list can also be positioned by keying a partial or complete printer name in the entry field under the printer type heading and pressing ENTER. After locating an entry it can be selected by keying a (S)elect line command in the entry field in front of the desired entry. The selected entry will be returned to the User Profile List screen and will be placed in the entry at the cursor position under the printer type heading. To return without selecting an entry, press the PF3 key, which will erase the entry on the User Profile List at the cursor position.

DRS/STI System Printer List	
Printer	DESCRIPTION
	Profile: LRS000
	Termid: A000
_ NEWPRT	Printer located in building 6
_ PCL	PCL Printer Description

PF1 = Help PF3 = Return

---

## DRS/STI CICS System Output List

The DRS/STI CICS System Output List is available by positioning the cursor on one of the sixteen entries on the User Profile List screen and pressing the PF6 key. This scrollable list displays all the defined outputs. The list can be positioned by pressing the PF7 key to scroll UP and pressing the PF8 key to scroll DOWN. The list can also be positioned by keying a partial or complete output name in the entry field under the output heading and pressing ENTER. After locating an entry it can be selected by keying a (S)elect line command in the entry field in front of the desired entry. The selected entry will be returned to the User Profile List screen and will be placed in the entry at the cursor position under the output heading. To return without selecting an entry, press the PF3 key, which will erase the output name from the entry on the User Profile List at the cursor position.

DRS/STI System Output List		
Output	Description	Profile: LRS000 Termid: A000
_OUT1	Output 1 description	
_OUT2	Output 2 description	

PF1 = Help PF3 = Return

## DRS/STI Transaction Interface

This table describes the interface control block an application will provide in order to invoke the DRS/STI CICS transaction interface. The DSECT describing the control block is defined in the DRS maclib as member DSSTSTI. The following table describes the fields the application should supply and the fields that can be returned.

Label	Data Type	Description
STIRETRN	DC A(0)	Return code
		0 = Request successful
		4 = No printer selected
		8 = DRS error
		12 = VSAM error
		16 = CICS error
STIREASN	DC A(0)	Reason code
		For return code 8 = DRRRC (DRS return code)
		For return code 12 = VSAM Feedback codes
STITRNID	CL4' '	Supplied CICS transaction ID to start on return from the DRS/STI transaction
STIVENDR	CL8' '	Supplied vendor name
STIAPPLN	CL8' '	Supplied application name
STIPROFN	CL16' '	Supplied profile
STIPRNTR	CL8' '	Returned printer name
STIOUTPN	CL8' '	Returned output name (optional)
STIPRNTT	AL2(0)	Returned printer type code
STIRLDDN	CL8' '	Supplied DDNAME for DRS rules dataset
STIUSRID	CL8' '	Supplied name for this user
STIVNDWK	XL256'00'	Area for vendor use
STICCMN	XL256'00'	Area for LRS use
STIDRIBL	AL2(0)	Actual length of returned DRIB
STIDROBL	AL2(0)	Actual length of returned DROB
STIDRIB	XL512	Area containing attributes for selected printer
STIDROB	XL4096	Area containing output attributes for selected printer if associated with an output description

### Notes:

- The area defined with label STIVNDWK is an area for the vendor to use to save any information needed in order to restore his environment upon return from the DRS/STI transaction.
- The area reserved at label STIDROB will be initialized by the DRS/STI transaction if the selected printer was associated with an output description. The area STIDROBL will be zeros if the DROB was not returned.



---

## DRS/STI Transaction Invocation

As described earlier the DRS/STI interface runs as a started CICS transaction. The application calling the interface does not perform the CICS START command, but performs a CICS LINK command passing a commarea and length. The called module (DSSTSTI) then searches the defined CICS transactions for a program by the same name and if found starts the corresponding transaction. The interface will then return to the caller who must perform a return to CICS allowing the started transaction to take control of the terminal. The following is an example of the link command for starting the DRS/STI transaction:

```
EXEC CICS LINK PROGRAM('DSSTSTI') X
        COMMAREA(STIAREA) X
        LENGTH(=AL2(STIDLEN))
```

Before invoking the link the caller should have supplied a transaction id that can be started in order to return the resulting selection to the caller. After the return transaction is started it should perform a CICS RETRIEVE command in order to obtain the commarea from the DRS/STI transaction. The following is an example of the command:

```
EXEC CICS RETRIEVE SET(R1) LENGTH(STILENG)
```

---

## Messages

<b>DSSC005E</b>	TRANSACTION INVOCATION ERROR – COMMAREA LENGTH INVALID
<b>Message Meaning:</b>	The commarea length is invalid.
<b>System Action:</b>	Processing terminates.
<b>Required Action:</b>	Correct the programming error and retry the process.
<b>DSSC010E</b>	TRANSACTION INVOCATION ERROR – MUST BE LINKED TO FROM A PROGRAM PASSING A COMMEAREA
<b>Message Meaning:</b>	A transaction invocation error has occurred as indicated in the message.
<b>System Action:</b>	Processing terminates.
<b>Required Action:</b>	Correct the programming error and retry the process.

---

# Section 19

## Introduction to DRS/API

The Dynamic Report System Application Programming Interface (DRS/API) allows programs to invoke DRS to create reports dynamically. The programs which call the DRS/API may be running in a CICS or non-CICS environment. The reports are available for printing immediately; it is not necessary to shut down the CICS or non-CICS system to obtain the report.

The DRS/API was referred to in earlier releases as the DRS “CALL” interface. Any programs written to invoke DRS in earlier releases should function equally well with DRS Version 1 Release 3.3.

The reports that are created using the DRS/API can be standard SYSOUT datasets on the JES spool or DASD print files. They can be fixed, variable, or undefined format, blocked or unblocked. They can have ASA, machine or no carriage control. If they are SYSOUT files, they can have standard JES dataset attributes, such as destination, writer name, form name, FCB, UCS, class, hold/no hold, and number of copies. They can refer to OUTPUT statements for parameters that would appear on the OUTPUT JCL statement, such as PAGEDEF, FORMDEF, CHARS, BUILDING, ADDRESS, etc.

If the report is placed on the JES spool, any system which obtains output from the JES spool can access the report. For example, the report could be printed by JES, VPS or PSF or it could be acquired by a report distribution and archival product. Because OUTPUT statement information can be associated with the print data, PAGEDEF, FORMDEF, CHARS, etc., could be used for special formatting.

With DRS/API, requests are made by passing parameters to the DRS/API interface module. A single “call” to DRS/API can initiate a report, add lines to a report, or terminate a report. There is also a call which can be used to inquire about active reports, a call to issue commands, and, for MVS/ESA users, a call to dynamically define OUTPUT JCL statements.

The two modes of DRS/API execution are CICS mode and non-CICS mode. Both modes of processing use the same parameters. The differences between the two modes are the format of the linkage to DRS/API and the internal processing.

In CICS mode, a program control link is made to the DRS/API CICS interface module (DRSSINTC). In non-CICS mode, a program CALL statement is used to call the DRS/API non-CICS interface module (DRSSINTB).

More than one CICS transaction can participate in the creation of a report. For example, one transaction could initiate the report. Then any transaction which has access to the report identifier could add lines to or terminate the report.

In an MVS/ESA environment, DRS/API can invoke Dynamic Output, which allows the application to dynamically add or delete an OUTPUT JCL statement. This allows the application to vary the SYSOUT processing options at execution time.



---

# Section 20

## DRS/API Installation

### Introduction

This section gives you the information you need to install DRS/API on your system. The details of this installation procedure are listed in [“Installation Steps” on page 20.4](#). Before you do the actual installation, you may want to read [“New Features in this Release” on page 20.2](#) and [“Migrating to DRS V1 R3.3 from an Earlier DRS Release” on page 20.3](#). These topics will be of interest to both the first-time user of DRS/API and to the user of prior DRS releases.

---

## New Features in this Release

If you have been using DRS/API Release 1.0, 2.0, 3.0, 3.1 or 3.2 at your installation, you will find Version 1 Release 3.3 will support all of the functions you are accustomed to. This release also gives you these new features.

- DRS/API Validation Option

DRS V1 R3.3 allows setting an option flag to indicate that the control blocks for a call should be validated, but that the call should not be processed. This allows validation of the DRIB for an INIT request or the DROB for an OUTPUT ADD request without opening and closing a print file.

- Support for additional OUTPUT keywords

Support has been added for new OUTPUT statement keywords. In addition, the DEST value can now be up to 127 bytes which allows specification of the IP address in the destination field, and up to 4 NOTIFY values may be specified. The new OUTPUT fields are:

**COLORMAP**

**COMSETUP**

**DUPLEX**

**FORMLEN**

**INTRAY**

**OFFSETXB**

**OFFSETXF**

**OFFSETYB**

**OFFSETYF**

**OVERLAYB**

**OVERLAYF**

**OVFL**

**PORTNO**

**PRERROR**

**PRTOPTNS**

**PRTQUEUE**

**RESFMT**

**RETRYL**

**RETRYT**

- Support for JOBNAME changes

DRS V1R3.3 allows the job name associated with a SYSOUT dataset to be set at report initialization time.

Note that DRS/API modules must be loaded from an authorized library to set the job name associated with a SYSOUT dataset. Additionally, MVS/SP 4.1.0 or greater must be running and the HBB4410 extensions to MVS must be present, or the job name change will be rejected. CICS application programs that call DRS/API will not be able to use this feature, due to APF authorization restrictions in the CICS environment.

---

## Migrating to DRS V1 R3.3 from an Earlier DRS Release

The DRS interface modules supplied to you on the distribution cartridge have aliases which are identical to the names of these modules in all prior releases of DRS. For example, the CICS interface module for DRS Version 1 Release 3.3 is named DRSSINTC, with an alias of DRS1INTC. The non-CICS interface module for DRS Version 1 Release 3.3 is named DRSSINTB, with an alias of DRS1INTB. This means that you can move the Version 1 Release 3.3 versions of the DRS modules into the appropriate load module libraries and run your existing programs which call “DRS1INTC” or “DRS1INTB” without changes, so long as those programs do not have the DRS interface modules link-edited with them.

If you linked the DRS interface modules with your programs that called DRS, you will need to relink those programs, so they can pick up the new version of the DRS interface modules. (You may want to refer to [“General Information about DRS Calls” on page 21.2](#) to see how to avoid having to go through this re-link procedure every time you get a new version of the DRS interface modules).

If you have an existing program which uses the query function of DRS, and you are migrating to DRS V1 R3.3 from DRS R1.0 or DRS R2.0, you will need to assemble/compile and link the program, using the new DRS libraries. Some fields are in different positions in the query block in Version 1 Release 3.3 of DRS, and some new fields have been added to the query block. Picking up the new description of the query block from the Version 1 Release 3.3 version of the DRS macro library at assembly/compile time will generate the correct code for the query call.

If you use DRS in CICS, and you were previously running DRS R1.0, you **MUST** replace PLT entry DRS1SHUT with DRSSSHTC. Both programs perform the same function. The difference is that DRS1SHUT is a CICS macro level program and DRSSSHTC is a CICS command level program. The macro level version will not work with DRS Version 1 Release 3.3. This change was made because IBM has dropped support of the CICS macro level interface. An SA03 abend will occur at CICS termination if you are not using DRSSSHTC and all DRS subtasks have not been detached.

If you use DRS in a non-CICS TP monitor environment, the version of module DRSSSHTB distributed with prior DRS releases will still function with DRS Version 1 Release 3.3. You can, however, replace it with the version supplied with DRS Version 1 Release 3.3, just to keep all load modules at the same level.

The DRS parameter options module now allows specification of the number of support subtasks to be used for applications running under CICS to be different from the number of support subtasks to be used for applications that execute in non-CICS environments. This allows the same DRS parameter options module to be used in multiple environments. SPTTCBS= in the parameter options module should be replaced with TCBCICS for CICS applications and TCBBTCH for non-CICS applications.

DRS/API now requires the presence of a load module (DRSSKEY) which contains the DRS/API product key. [“Installing the DRSSKEY Module” on page 20.13](#) describes how this module can be installed and updated.

---

# Installation Steps

The steps required to install Version 1 Release 3.3 of DRS/API are:

1. **Read the entire installation procedure.**
2. **Restore the DRS distribution libraries** (see [“Restoring the DRS Distribution Libraries”](#) on page 20.6).
3. **Customize the DRS System Options** (see [“Customizing the DRS System Options”](#) on page 20.9).
4. **Assemble and linkedit the DRS System Options into a STEPLIB or LINKLIST library** (see [“Customizing the DRS System Options”](#) on page 20.9).
5. **Assemble and linkedit any DRS user exit which you choose to use at your site into a STEPLIB or LINKLIST library.** All user exits are optional. Information on assembling and linking these modules is given in [“DRS/API User Exits”](#) on page 22.1).
6. **Install the DRSSKEY module** (see [“Installing the DRSSKEY Module”](#) on page 20.13).
7. **Copy the following modules from LRS.DRS.V1R33.LOAD into a STEPLIB or LINKLIST library:**
  - DRSSATCH
  - DRSSDTCH
  - DRSSEXIT
  - DRS1INTB (alias of DRSSINTB)
  - DRSSGTRC
  - DRSSINTB
  - DRSSKEY
  - DRSSLOG
  - DRSSMAIN
  - DRSSMSGH
  - DRSSQRQE
  - DRSSQSM
  - DRSSHTB (used only in a non-CICS TP monitor environment)
  - DRSTRCE
  - DRSSVSIO
  - DRSSVSM
  - DRSSVSST



- 
8. **Update the CICS tables to define the DRS/API programs and transactions.** (Only necessary if executing DRS within CICS. See [“Updating the CICS Tables”](#) on page 20.14).
  9. **Copy the following modules from LRS.DRS.V1R33.LOAD into one of the CICS DFHRPL libraries** (only necessary if executing DRS within CICS):
    - DMSTSTI (for DRS/STI only)
    - DRSSINTC
    - DRS1INTC (alias of DRSSINTC)
    - DRSSSHTC
    - DRS1SHUT (alias of DRSSSHTC)
    - DSSTSTI (for DRS/STI only)
  10. **DRS/API installation is now complete.**

The remainder of this section provides the information required to perform the above steps.

---

## Restoring the DRS Distribution Libraries

| DRS is distributed on a 3480 or 3490 cartridge in IEBCOPY unloaded format with the following attributes:

- Standard labels
- VOL=SER=VPSR70, if distributed with VPS R7.0

**The DRS files are:**

<b>FILE SEQUENCE NUMBER</b> (with VPS R7.0)	<b>DSNAME</b>	<b>CONTENTS</b>
#26	LRS.DRS.V1R33.ASM	Source for sample exits and sample programs
#27	LRS.DRS.V1R33.MACLIB	Mapping macros
#28	LRS.DRS.V1R33.CNTL	Sample JCL
#29	LRS.DRS.V1R33.LOAD	DRS load modules for user exits

Sample JCL to restore the files on the distribution cartridge is supplied in file 1, which is named "LRS.PRODUCT.JCL". (This is a sequential file of 80-byte records. You may wish to use IEBGENER to restore it to a sequential file or to a member of a PDS).

The portion of that JCL which pertains to VPS is reproduced here.

```
//JOBNAME JOB ACCT,PROGRAMMER,TIME=25,REGION=1024K
/**
//ALLOC PROC PREFIX=, ** HIGH LEVEL QUALIFIER
// VOLSER=XXXXXX, ** DISK VOLUME SERIAL
// ASM=23440, ** ASM BLKSIZE <== Any multiple of 80
// MACLIB=23440, ** MACLIB BLKSIZE <== Same as SYS1.MACLIB
// CNTL=23440, ** CNTL BLKSIZE <== Any multiple of 80
// LOAD=4096 ** LOAD BLKSIZE <== Any valid value for the
device
/** type
//ALLOC EXEC PGM=IEFBR14
/**
//DISTASM DD DSN=&PREFIX..LRS.DRS.V1R33.ASM,
// DISP=(,CATLG),UNIT=SYSALLDA,VOL=SER=&VOLSER,
// DCB=(BLKSIZE=&ASM,LRECL=80,RECFM=FB),
// SPACE=(&ASM,(140,35,25),RLSE)
/**
//DISTMAC DD DSN=&PREFIX..LRS.DRS.V1R33.MACLIB,
// DISP=(,CATLG),UNIT=SYSALLDA,VOL=SER=&VOLSER,
// DCB=(BLKSIZE=&MACLIB,LRECL=80,RECFM=FB),
// SPACE=(&MACLIB,(35,5,25),RLSE)
/**
//DISTCNTL DD DSN=&PREFIX..LRS.DRS.V1R33.CNTL,
// DISP=(,CATLG),UNIT=SYSALLDA,VOL=SER=&VOLSER,
// DCB=(BLKSIZE=&CNTL,LRECL=80,RECFM=FB),
// SPACE=(&CNTL,(35,5,50),RLSE)
/**
//DISTLOAD DD DSN=&PREFIX..LRS.DRS.V1R33.LOAD,
// DISP=(,CATLG),UNIT=SYSALLDA,VOL=SER=&VOLSER,
// DCB=(BLKSIZE=&LOAD,RECFM=U),
// SPACE=(&LOAD,(450,25,50),RLSE)

/**
// PEND
/**
//RESTORE PROC PREFIX=, ** HIGH LEVEL QUALIFIER
// TAPE=CART ** TAPE UNIT NAME
/**
//RESTORE EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=*
/**
//TAPEASM DD DSN=LRS.DRS.V1R33.ASM,DISP=OLD,
// VOL=(,RETAIN,SER=VPSxxx),LABEL=(nnn,SL),UNIT=&TAPE
//DISTASM DD DSN=&PREFIX..LRS.DRS.V1R33.ASM,DISP=OLD
/**
//TAPEMAC DD DSN=LRS.DRS.V1R33.MACLIB,DISP=OLD,
// VOL=(,RETAIN,SER=VPSxxx),LABEL=(nnn,SL),UNIT=&TAPE
//DISTMAC DD DSN=&PREFIX..LRS.DRS.V1R33.MACLIB,DISP=OLD
/**
```

Continued on next page.

---

```

//TAPECNTL DD DSN=LRS.DRS.V1R33.CNTL,DISP=OLD,
//
//          VOL=( ,RETAIN,SER=VPSxxx),LABEL=(nnn,SL),UNIT=&TAPE
//DISTCNTL DD DSN=&PREFIX..LRS.DRS.V1R33.CNTL,DISP=OLD
//*
//TAPELOAD DD DSN=LRS.DRS.V1R33.LOAD,DISP=OLD,
//
//          VOL=( ,RETAIN,SER=VPSxxx),LABEL=(nnn,SL),UNIT=&TAPE
//DISTLOAD DD DSN=&PREFIX..LRS.DRS.V1R33.LOAD,DISP=OLD
//*
//          PEND
//*
//ALLOC EXEC ALLOC
//*
//RESTORE EXEC RESTORE
//SYSIN DD *
COPY INDD=((TAPEASM,R)),OUTDD=DISTASM
COPY INDD=((TAPEMAC,R)),OUTDD=DISTMAC
COPY INDD=((TAPECNTL,R)),OUTDD=DISTCNTL
COPY INDD=((TAPELOAD,R)),OUTDD=DISTLOAD
/*

```

**Note:** In the TAPEASM, TAPEMAC, TAPECNTL and TAPELOAD DD statements, the volume serial number and the file sequence number will vary. Please refer to the previous page for the correct VOL SER and file number.

---

## Customizing the DRS System Options

DRS looks for a DRS System Options module named DRSSOPTS. This module specifies values to be used for certain DRS system parameters. You generate the DRS System Options module during DRS installation by performing the following two steps:

1. Modify member DRSSOPTS in library LRS.DRS.V1R33.ASM, specifying the values you want to use for each of the System Options parameters. This member consists of the \$DRSOPTS macro, with default values specified. You can change any of the parameter values to any valid value.
2. Create the load module named DRSSOPTS by assembling and linking your modified version of DRSSOPTS. Sample JCL to assemble and link the DRS System Options is contained in member OPTSASML in dataset LRS.DRS.V1R33.CNTL. That JCL is reproduced here:

```
//JOBNAME JOB (YOUR JOB CARD INFORMATION)
//ASMH EXEC PGM=IEV90,PARM='OBJECT,NODECK,RENT,XREF(SHORT)'
//SYSLIB DD DSN=SYS1.MACLIB,DISP=SHR
// DD DSN=SYS1.AMODGEN,DISP=SHR
// DD DSN=LRS.DRS.V1R33.MACLIB,DISP=SHR
// DD DSN=LRS.DRS.V1R33.ASM,DISP=SHR
//SYSUT1 DD UNIT=VIO,SPACE=(CYL,(5,2))
//SYSUT2 DD UNIT=VIO,SPACE=(CYL,(5,2))
//SYSUT3 DD UNIT=VIO,SPACE=(CYL,(5,2))
//SYSPUNCH DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSIN DD DSN=LRS.DRS.V1R33.ASM(DRSSOPTS),DISP=SHR
//SYSLIN DD DSN=&&OBJ,DISP=(,PASS),SPACE=(TRK,(6,3)),
// UNIT=VIO,DCB=BLKSIZE=1600
//*
//LKED EXEC PGM=IEWL,PARM='XREF,LET,LIST,RENT,REUS,REFR,MAP,AC=0',
// COND=(5,LT,ASMH)
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD UNIT=VIO,SPACE=(CYL,(1,1))
//SYSLOAD DD DSN=LRS.DRS.V1R33.LOAD(DRSSOPTS),DISP=SHR
//SYSLIN DD DSN=&&OBJ,DISP=(OLD,DELETE,DELETE)
/*
```

If DRS is unable to locate the DRSSOPTS module, the default values will be used.

You can share one DRSSOPTS module for DRS in both CICS and non-CICS mode, or you can use a separate DRSSOPTS module for each environment.

The \$DRSOPTS macro is coded as follows:

---

```

name    module name.  Must begin in column 1.

b       One or more blanks must precede $DRSOPTS

$DRSOPTS
  {,GTFFID=}
  {,LOG=}
  {,LOGDD=}
  {,LOGCLAS=}
  {,LOGDEST=}
  {,LOGFORM=}
  {,LOGWTR=}
  {,LOGFCB=}
  {,LOGUCS=}
  {,LOGHOLD=}
  {,SNACLAS=}
  {,SNADEST=}
  {,SNAFORM=}
  {,SNAWTR=}
  {,SNAFCB=}
  {,SNAUCS=}
  {,SNAHOLD=}
  {,STATS=}
  {,SYSLOG=}
  {,TCBBTCH=}
  {,TCBCICS=}
  {,TRPAGES=}
  {,TRTYPES=}

```

Brackets, {}, are used to enclose optional parameters, which may or may not be specified. If a parameter is not coded, the default for that parameter will be used.

The parameters of the \$DRSOPTS macro are explained below:

- GTFFID=** Specifies the format appendage that is to be invoked by the edit function of PRDMP. Valid values range from 1-80. Default is 20.
- LOG=** Specifies whether or not DRS logging is to be activated. Valid values are YES and NO. Default is NO. Note that a value of YES should be used only when executing DRS in an environment which allows execution of program DRSSHTC or DRSSHTB at shutdown.
- LOGDD=** Specifies the DDNAME to be used for the DRS log dataset. Valid values consist of any valid DDNAME. Default is DRSLOG.
- LOGCLAS=** Specifies the SYSOUT class of the DRS log dataset. Valid values consist of A-Z and 0-9. Default is A.
- LOGDEST=** Specifies the destination of the DRS log dataset. Valid values consist of any valid JES destination. Default is LOCAL.
- LOGFORM=** Specifies the form name of the DRS log dataset. Valid values consist of any valid form name. If this keyword is not specified, then form name will not be used when allocating the DRS log dataset.
- LOGWTR=** Specifies the writer name of the DRS log dataset. Valid values consist of any valid writer name. If this keyword is not specified, then writer name will not be used when allocating the DRS log dataset.
- LOGFCB=** Specifies the FCB name of the DRS log dataset. Valid values consist of any valid FCB name. If this keyword is not specified, then FCB name will not be used when allocating the DRS log dataset.

---

<b>LOGUCS=</b>	Specifies the UCS name of the DRS log dataset. Valid values consist of any valid UCS name. If this keyword is not specified, then UCS name will not be used when allocating the DRS log dataset.
<b>LOGHOLD=</b>	Specifies whether or not the DRS log dataset will be held. Valid values consist of YES and NO. Default is YES.
<b>SNACLAS=</b>	Specifies the SYSOUT class of the DRS snap dataset. Valid values consist of A-Z and 0-9. Default is A.
<b>SNADEST=</b>	Specifies the destination of the DRS snap dataset. Valid values consist of any valid JES destination. Default is LOCAL.
<b>SNAFORM=</b>	Specifies the form name of the DRS snap dataset. Valid values consist of any valid form name. If this keyword is not specified, then form name will not be used when allocating the DRS snap dataset.
<b>SNAWTR=</b>	Specifies the writer name of the DRS snap dataset. Valid values consist of any valid writer name. If this keyword is not specified, then writer name will not be used when allocating the DRS snap dataset.
<b>SNAFCB=</b>	Specifies the FCB name of the DRS snap dataset. Valid values consist of any valid FCB name. If this keyword is not specified, then FCB name will not be used when allocating the DRS snap dataset.
<b>SNAUCS=</b>	Specifies the UCS name of the DRS snap dataset. Valid values consist of any valid UCS name. If this keyword is not specified, then UCS name will not be used when allocating the DRS snap dataset.
<b>SNAHOLD=</b>	Specifies whether or not the DRS snap dataset will be held. Valid values consist of YES and NO. Default is YES.
<b>STATS=</b>	Specifies whether or not DRS statistics gathering is enabled. Valid values consist of YES and NO. Default is NO. (A value of NO decreases the amount of CPU time used by DRS).
<b>SYSLOG=</b>	Specifies if DRS will write its messages to SYSLOG. Valid values consist of YES and NO. Default is NO.
<b>TCBBTCH=</b>	Specifies the number of support subtasks that DRS will OS attach to process the DRS requests in a non-CICS environment. Valid values range from 0-64. Default is 0. Note that a value other than 0 should only be used when executing DRS in an environment which allows execution of program DRSSHTC or DRSSHTB at shutdown to detach the support subtasks.
<b>TCBCICS=</b>	Specifies the number of support subtasks that DRS will OS attach to process the DRS requests in a CICS environment. Valid values range from 0-64. Default is 1. Note that a value other than 0 should only be used when executing DRS in an environment which allows execution of program DRSSHTC or DRSSHTB at shutdown to detach the support subtasks.
<b>TRPAGES=</b>	Specifies the number of 4K pages that DRS will allocate for its internal trace table. Valid values range from 0-10. Default is 0. (A value of 0 decreases the amount of CPU time used by DRS).

---

**TRTYPES=** Specifies the types of events DRS will trace. Valid values consist of any valid hex value. Default is 00. Following is a description of the trace types:

- 80** = Trace GETMAIN/FREEMAIN requests
- 40** = Trace ENQ/DEQ requests
- 20** = Trace log requests
- 10** = Trace user exits
- 08** = Trace detail (PUT requests)
- 04** = (Not defined)
- 02** = CICS tracing is active
- 01** = GTF tracing is active



---

## Installing the DRSSKEY Module

This release of DRS requires a unique “key” for the DRS base product to function. This key is supplied in a load module named DRSSKEY. This module is LOADED during DRS initialization. The key value is unique for each installation.

The DRSSKEY module MUST be present.

Load module DRSSKEY is supplied in the DRS load module library (LRS.DRS.V1R33.LOAD). You can copy this module to a library that is accessible to DRS. If you are using more than one of the products that require this module, you might want to put the module in a LINKLIST library, so all products have access to the same module.

If you are using a trial version of one of these products, you may be given a new key value to extend the trial period. You will also be given a new key value when you purchase the product. Any time that you receive a distribution cartridge, the DRSSKEY module will be included, and the applicable key value will also be supplied in the first file of the distribution tape. This file has a dataset name of LRS.PRODUCT.JCL, and is a sequential file with a record length of 80, a block size of 23440, and a record format of FB. You can use IEBGENER to restore this file to a sequential file or to print it.

If you should receive a new key value over the phone, you can update member DRSSKEY in file LRS.DRS.V1R33.ASM, and assemble and linkedit it to put the new value into effect. Sample JCL to assemble and linkedit the DRSSKEY module is provided as member KEYASML in file LRS.DRS.V1R33.CNTL. Once the DRSSKEY module has been linkedited, it should be copied into a library accessible to the appropriate product.

The DRSSKEY macro is used to generate the DRSSKEY module.

The DRSSKEY macro is written as follows:

<b>b</b>	<b>One or more blanks must precede DRSSKEY.</b>
<b>DRSSKEY</b>	
<b>b</b>	<b>One or more blanks must follow DRSSKEY.</b>
<b>,KEYDRS=</b>	<b>KEYDRS must be specified</b>
<b>{,KEYSAPR2=}</b>	<b>KEYSAPR2 is optional</b>
<b>{,KEYNAT=}</b>	<b>KEYNAT is optional</b>

The parameters of the DRSSKEY macro are explained below:

- KEYDRS=** Specifies a 24 character key, as supplied in the first file of the distribution tape, that enables the operation of DRS/API.
- KEYSAPR2=** Specifies a 24 character key, as supplied in the first file of the distribution tape, that enables the operation of DRS/SAPR2.
- KEYNAT=** Specifies a 24 character key, as supplied in the first file of the distribution tape, that enables the operation of the DRS/NATURAL product.

### Notes:

- Once the DRSSKEY module has been created, it will only be necessary to reinstall the module as products move from trial status to license status, or when a trial product is extended.
- A common DRSSKEY module can be shared across different products, and can be shared across different releases of the DRS products, beginning with DRS R3.2.

---

## Updating the CICS Tables

CICS programs that invoke DRS/API, as well as the DRS/STI CICS programs, require that entries be added to the CICS PPT, PLT, and PCT. You can add the entries using RDO, or, for CICS/MVS, you can use macro definitions to add the entries. If you choose to use RDO, sample JCL is supplied as member CICSRDO in file LRS.DRS.V1R33.CNTL. The following RDO or macro definitions are required.

For CICS programs that invoke DRS/API:

Add to the PPT using RDO:

```
DEFINE PROGRAM(DRSSINTC) GROUP(gggg) LANG(ASSEMBLER)
DEFINE PROGRAM(DRS1INTC) GROUP(gggg) LANG(ASSEMBLER)
DEFINE PROGRAM(DRSSSHTC) GROUP(gggg) LANG(ASSEMBLER)
DEFINE PROGRAM(DRS1SHUT) GROUP(gggg) LANG(ASSEMBLER)
```

or, add to the PPT using macro definitions:

```
DFHPPT TYPE=ENTRY,          DRS/CICS INTERFACE      *
    PROGRAM=DRSSINTC,        *
    PGMLANG=ASSEMBLER
DFHPPT TYPE=ENTRY,          DRS/CICS INTERFACE      *
    PROGRAM=DRS1INTC,        (alias of DRSSINTC)    *
    PGMLANG=ASSEMBLER
DFHPPT TYPE=ENTRY,          DRS/CICS SHUTDOWN           *
    PROGRAM=DRSSSHTC,        *
    PGMLANG=ASSEMBLER
DFHPPT TYPE=ENTRY,          DRS/CICS SHUTDOWN           *
    PROGRAM=DRS1SHUT,        (alias of DRSSSHTC)          *
    PGMLANG=ASSEMBLER
```

Add to the PLT (PLTSD) using macro definitions:

```
DFHPLT TYPE=ENTRY,          *
    PROGRAM=DRSSSHTC
```

If you are using RDO and you have created a new group for the DRS programs, you can add the programs to an active CICS system by entering the following CEDDA command:

```
INSTALL GROUP(gggg)
```

Alternatively, add group “gggg” to your CICS startup GRPLIST.

**Note:** If you were using DRS Release 1.0, this PLT entry is a replacement for DRS Release 1.0 program DRS1SHUT. Both programs perform the same function. However, DRS1SHUT is a CICS macro level program, and DRSSSHTC is a CICS command level program. The change was made because IBM will be dropping support of the CICS macro level interface in future releases of CICS. The macro level version of this program will not function with Version 1 Release 3.3 of DRS.

If you are running CICS 3.3 or higher, you should specify EXECKEY(CICS) for programs DRSSINTC and DRSSSHTC.

---

## For DRS/STI CICS programs:

Add to the FCT using RDO:

```
DEFINE FILE DRSRULES GROUP(gggg) DSNAME(DRS.V1R33.RULES)
DISPOSITION(SHARE)
```

or, add to the FCT using macro definitions:

```
DRSRULES DFHFCT TYPE=DATASET,DATASET=DRSRULES, *
DSNAME=DRS.V1R33.RULES,DISP=SHR,ACCMETH=VSAM, *
FILSTAT=(ENABLED,CLOSED)
```

Add to the PCT using RDO:

```
DEFINE TRANSACTION(xxxx) GROUP(gggg) PROGRAM(DSSTSTI)
```

or, add to the PCT using macro definitions:

```
DFHPCT TYPE=ENTRY, *
TRANSID=xxxx,...any transaction ID you choose... *
PROGRAM=DSSTSTI
```

Add to the PPT using RDO:

```
DEFINE PROGRAM(DRSSINTC) GROUP(gggg) LANG(ASSEMBLER)
DEFINE PROGRAM(DRSSSHTC) GROUP(gggg) LANG(ASSEMBLER)
DEFINE PROGRAM(DSSTSTI) GROUP(gggg) LANG(ASSEMBLER)
DEFINE MAPSET(DMSTSTI) GROUP(gggg)
```

**Note:** If DRSSINTC and DRSSSHTC have already been defined for use with CICS programs that call DRS/API, those two definitions will not be required again.

or, add to the PPT using macro definitions:

```
DFHPPT TYPE=ENTRY, DRS/CICS INTERFACE (if not *
PROGRAM=DRSSINTC, already added) *
PGMLANG=ASSEMBLER

DFHPPT TYPE=ENTRY, DRS/CICS SHUTDOWN (if not *
PROGRAM=DRSSSHTC, already added) *
PGMLANG=ASSEMBLER

DFHPPT TYPE=ENTRY, DRS/STI INTERFACE *
PROGRAM=DSSTSTI, *
PGMLANG=ASSEMBLER

DFHPPT TYPE=ENTRY, DRS/STI MAPSET *
PROGRAM=DMSTSTI, *
PGMLANG=ASSEMBLER
```

---

Add to the PLT (PLTSD) using macro definitions (if not already added for CICS programs that call DRS/API):

```
DFHPLT TYPE=ENTRY, *  
PROGRAM=DRSSHTC
```

If you are using RDO and you have created a new group for the DRS programs, you can add the programs to an active CICS system by entering the following CEDA command:

```
INSTALL GROUP(gggg)
```

Alternatively, add group “gggg” to your CICS startup GRPLIST.

**Running DRS in a QOR (Queue Owning Region) so that TS queues are shared across all CICS regions:**

Normally, CICS temporary storage queues are local to each CICS system and are automatically deleted at CICS startup unless you specifically define the Queue name as a recoverable resource.

DRS stores information in a TS Queue called DR32xxxx (where xxxx is the CICS system ID) which is deleted at normal CICS termination by program DRSSHTC. But if CICS abends or is terminated abnormally then DRSSHTC doesn't get executed. This normally doesn't cause any problems because CICS will delete the TS queue during startup.

But if you are running with a Queue Owning Region which shares TS queues between multiple CICS regions, then the TS queues are not automatically deleted when the AOR that created them restarts.

To avoid this problem, define our TS queue as a local, non shared, queue. It will then work as before and be purged when CICS restarts.

Add the following definition to the CICS Temporary Storage Table.

```
DFHTST TYPE=LOCAL, *  
DATAID=(DR32) DEFINE DR32XXXX AS A LOCAL TS QUEUE
```

---

## Installing an Updated Version of DRS

If you find that you need a “fix” for some problem or an enhancement to DRS that is not available in the version that you have installed, DRS support will send you a distribution cartridge that contains a version of DRS that includes the fix or enhancement that you need, as well as any other fixes and enhancements created for other DRS users since you received your previous installation cartridge.

The cartridge that you receive will be in the format described in the installation section of this manual. Installation of a new version of DRS Version 1 Release 3.3 should include:

- Replacing existing DRS load modules with the load modules in file LRS.DRS.V1R33.LOAD on the distribution cartridge (most fixes are applied as “zaps” to the load modules).
- Replacing sample user exit program source code and sample DRS program source code with the members in file LRS.DRS.V1R33.ASM on the distribution cartridge (remember, if you have made any modifications to user exits for your installation, you will want to carry those modifications forward into the newest version of the source code).
- Replacing sample JCL for assembling DRS user exits and options with the members in file LRS.DRS.V1R33.CNTL (remember, if you have made any modifications to this sample JCL unique to your installation, you will want to carry those modifications forward into the newest version of the sample JCL).
- Replacing macros used by DRS with the members in file LRS.DRS.V1R33.MACLIB.

If the problem you were experiencing or the enhancement which you requested was related to a DRS User Exit or to the DRS Systems Options module, you should perform the appropriate assemble-and-link job after you have refreshed your libraries from the distribution cartridge.



---

# Section 21

## DRS/API Request Calls

### Introduction

There are three types of calls used by the application programmer to create a report. They are the **INIT**, **PUT**, and **TERM** calls.

The **INIT** call establishes the attributes of the report. This group of attributes is similar to parameters that you would code in **SYSOUT** JCL statements. There can be from 1 to 1,024 attribute groups specified in an **INIT** control block. One **SYSOUT** or **DASD** dataset is created for each attribute group. By using more than one attribute group, a report can be broadcast to many destinations, classes, writers, etc. Once the **INIT** call is successful, the report is considered active until a **TERM** call is issued. There is no **DRS** limit on the number of reports that can be active concurrently.

The **PUT** call adds lines to the report. From 1 to 32,767 lines of data can be added to a report with a single **PUT** call.

The **TERM** call ends the report. Once the **TERM** call is successful, the report is available for printing from the **JES** spool. The report is no longer considered active in **DRS**.

There are three additional calls which may be useful for certain applications.

The **QURY** call allows inquiry into **DRS**. A **QURY** call can be general (querying all active reports), detail (querying a specific report), or system (querying the status of **DRS**). A **QURY** call can also be used to inquire about dynamically defined **OUTPUT** JCL statements, and can be a general query, returning information about all existing dynamically defined **OUTPUT** JCL statements, or a detail query, returning information about a specific dynamically defined **OUTPUT** JCL statement.

The **CMND** call allows you to issue commands to **DRS**. Commands to **DRS** can request dumps of **DRS** control blocks (**SNAP**), close the **DRS** log dataset (**CLOSELOG**), reset logging and tracing values (**SSET**), instruct **DRS** to reject initiation of new reports (**QUIESCE**), or instruct **DRS** to accept initiation of new reports (**RESTART**).

The **OUTP** call allows you, under **MVS/ESA**, to dynamically add or delete an **OUTPUT** JCL statement. This allows the application to vary the **SYSOUT** characteristics at execution time.

---

## General Information about DRS Calls

A call is made to DRS by linking to an interface module. For CICS programs, the interface module is DRSSINTC. For non-CICS programs, the interface module is DRSSINTB. A parameter list is passed on the call to provide the appropriate control block and data line addresses.

### CICS Mode

The calls made to DRS to create a report can be from one or more CICS tasks. A field in the request block known as the REPORT ID identifies the report to which the call applies.

The CICS mode call is made with a CICS program control link. The parameters are passed in the CICS COMMAREA. The DRS Request Block (DRRB) must be first in the COMMAREA. Any other parameters must be passed immediately following the DRRB in the COMMAREA.

Although CICS requires that you specify the length of the COMMAREA for the program control link, DRS ignores this length field.

The CICS user ID and CICS terminal ID associated with the DRS call are retrieved and passed to DRS user exits. The user ID is retrieved via:

```
EXEC CICS ASSIGN(USERID)
```

and the CICS terminal ID is retrieved from the EIB (field EIBTRMID).

### Non-CICS Mode

In non-CICS mode, standard linkage is used to call DRS. The parameter list format and control blocks are identical to the CICS mode.



---

## Assembler Coding

There are three techniques available to call DRS from an assembler language program. All three techniques use the same parameters and require that register 13 point to an 18-word save area.

**Technique 1** - Use a LOAD macro to load the interface. Save the address of the loaded module. Then use a branch and link to call the module. This technique has the advantage of both good performance (the module is loaded only once) and ease of maintenance (a new version of the interface module would not require any program changes or re-linkeds). *This is the recommended technique.*

Coding for this technique:

### TO LOAD THE MODULE:

```
LOAD    EP=DRSSINTB  LOAD MODULE DRSSINTB.
ST      R0,INTBADDR  SAVE ADDRESS IN INTBADDR.
```

### ON EACH CALL:

```
LA      R1,PARMLIST  POINT R1 TO PARAMETER LIST.
L       R15,INTBADDR LOAD R15 WITH MODULE ADDRESS.
BALR   R14,R15      BRANCH TO MODULE.
```

```
PARMLIST DC  A(PARM1),X'80',AL3(PARM2)
INTBADDR DS  F
```

**Technique 2** - Use a LINK macro. This will cause the interface module to be loaded on the first call. It will then stay in virtual storage. The advantage to this method is that the interface module can be changed easily. However, the disadvantage is the overhead of doing many links.

Coding for this technique:

```
LA      R1,PARMLIST  POINT R1 TO PARAMETER LIST.
LINK    EP=DRSSINTB  LINK TO MODULE.
```

```
PARMLIST DC  A(PARM1),X'80',AL3(PARM2)
```

**Technique 3** - Use a CALL macro. This will cause the interface module to be linkedited with the application program. This way is very good from a performance standpoint. However, whenever the interface module changes for a new release, all application programs using DRS would have to be re-linked. This technique is not recommended.

Coding for this technique:

```
LA      R1,PARMLIST  POINT R1 TO PARAMETER LIST.
CALL    DRSSINTB     CALL MODULE.
```

```
PARMLIST DC  A(PARM1),X'80',AL3(PARM2)
```

---

## COBOL Coding

There are two techniques in COBOL to call DRS. The COBOL source code is the same in either case. The difference is in the specification of compiler option DYNAM/NODYNAM. DYNAM is recommended, because it indicates that the interface module should be loaded at execution time. Specifying NODYNAM will cause the interface to be linked with the application program, which means that any program using DRS would have to be re-linked when a new release of the interface module is distributed.

### Example of COBOL Coding:

```
.  
.   
WORKING-STORAGE SECTION.  
.   
.   
01 WS-DRRB.  
COPY DRSDRRBC.  
.   
01 WS-DRIB.  
COPY DRSDRIBC.  
.   
.   
PROCEDURE DIVISION.  
.   
.   
CALL 'DRSSINTB' USING WS-DRRB  
                                WS-DRIB.  
.   
.
```

---

## Parameters for DRS Calls

The parameter list for a DRS call consists of one or more DRS control blocks. Every call requires that the REQUEST BLOCK be passed as the first parameter in the parameter list. Other parameters vary by the type of call being requested.

All of the DRS control blocks, as well as the data lines passed to DRS on a PUT call, can be allocated above or below the 16M line.

All parameters passed to DRS are edited prior to use. Parameters can be defined as alpha (A - Z), numeric (0 - 9), alphanumeric (A - Z and 0 - 9), or national (@, \$ and #). Values in character type parameters should be left justified and blank filled. A character type field containing a blank or X'00' in the first byte is considered to be not specified.

Any areas in DRS control blocks which are described as “reserved” or “filler” should be initialized to X'00' (LOW-VALUES in COBOL) to maintain compatibility with future releases.

The individual fields which make up each of the DRS control blocks are described in detail later in this section.

### The INIT Call

The INIT call initiates a report. This call requires that two parameters be passed. The first is the REQUEST BLOCK and the second is the INIT BLOCK. The INIT BLOCK may be anywhere in storage. DRS references it, but does not modify it.

(In CICS mode, the REQUEST BLOCK and the INIT BLOCK must exist contiguously in the COMMAREA, with the REQUEST BLOCK first).

### The PUT Call

The PUT call adds print lines to a report. This call requires that two parameters be passed. The first is the REQUEST BLOCK and the second is the print line(s) to be added to the report.

If the dataset is fixed format, the second parameter should point directly to the first byte of data (the carriage control byte, if the dataset has carriage control, or the first byte of print data, if it does not have carriage control). If the dataset is variable format, the second parameter should point to the 4-byte Record Descriptor Word (RDW) which precedes the print line. DRS will use the record length in the first 2 bytes of the RDW to determine the length of the line. If the dataset is undefined format, the second parameter should point to a 4-byte field preceding the print line. DRS will use the record length in the first 2 bytes of this field to determine the length of the line (just as it would for variable format datasets), and will remove the 4-byte record length field before it adds the print line to the report.

You can add more than one print line to a report with one PUT call by indicating the number of lines in the DATA LINE COUNT field of the REQUEST BLOCK. The lines to be added must exist contiguously in storage, and the second parameter in the parameter list should point to the first of the print lines to be added. The maximum number of print lines which can be added to a report with one PUT call is 32,767.

(In CICS mode, the REQUEST BLOCK and any print lines being passed on the PUT call must exist contiguously in the COMMAREA, with the REQUEST BLOCK first).

---

## The TERM Call

The TERM call terminates the report. This call requires only that the REQUEST BLOCK be passed. However, you may optionally pass the TERMINATION BLOCK as the second parameter on a TERM call. The TERMINATION BLOCK allows you to change attributes of a dataset or delete a dataset at termination of the report.

(In CICS mode the REQUEST BLOCK and the TERMINATION BLOCK, if supplied, must exist contiguously in the COMMAREA, with the REQUEST BLOCK first).

## The QUERY Call

The QUERY (query) call is used to inquire about the status of all of the reports which are active in DRS, about a specific report, or about the DRS system in general. This request requires that the REQUEST BLOCK and the QUERY BLOCK be passed.

(In CICS mode, the REQUEST BLOCK and the QUERY BLOCK must exist contiguously in the COMMAREA, with the REQUEST BLOCK first).

## The CMND Call

The CMND (command) call allows commands to be issued to DRS. This call requires that the REQUEST BLOCK and the COMMAND BLOCK be passed.

(In CICS mode, the REQUEST BLOCK and the COMMAND BLOCK must exist contiguously in the COMMAREA, with the REQUEST BLOCK first).

## The OUTP Call

The OUTP (dynamic output call) allows users of MVS/ESA to add or delete OUTPUT JCL statements. This call requires that the REQUEST BLOCK and the OUTPUT BLOCK be passed.

(In CICS mode, the REQUEST BLOCK and the OUTPUT BLOCK must exist contiguously in the COMMAREA, with the REQUEST BLOCK first).

---

# DRS Control Blocks

## Request Block (DRRB)

The REQUEST BLOCK contains information needed for all types of calls. It identifies the report, the type of call, and the options for the call, and also provides an area for DRS to return information to the caller. The REQUEST BLOCK must contain the characters “DRRB” in the first 4 bytes. If these characters are not found, DRS will abend the task or the program. The REQUEST BLOCK must be in modifiable storage.

The individual fields which make up the REQUEST BLOCK are:

<b>‘DRRB’</b>	The control block identifier. <b>Assembler name:</b> DRRID <b>COBOL name:</b> DRS-DRRB-ID
<b>FUNCTION</b>	The type of call being requested. Valid values are “INIT”, “PUT”, “TERM”, “QURY”, “CMND”, and “OUTP”. <b>Assembler name:</b> DRRFUNC <b>COBOL name:</b> DRS-DRRB-FUNCTION
<b>REPORT ID</b>	The identifier of the report. If this field is blank on an INIT call, DRS will assign a report id and will place it in this field, to be used on all subsequent calls for the same report. If you wish to supply a report id, it should be supplied for the INIT call and all subsequent calls for the same report, and must consist of 1 to 8 alphanumeric or national characters, with an alpha character required in the first position. This field does not apply to a QURY or CMND call. <b>Assembler name:</b> DRRRID <b>COBOL name:</b> DRS-DRRB-REPORT-ID
<b>OPTIONS</b>	Four option bytes to allow for special requirements. Option field 1 (DRROPT1) may contain “V” to indicate that the call should be validated, but not processed. This allows control block validation without creating a print file. (In a prior release, option field 1 could contain a “C” to indicate that conditional GETMAINs should be performed. This is not required for this release of DRS, but a “C” in the DRROPT1 field is still considered a valid value for compatibility purposes.) Option fields 2, 3 and 4 are currently unused and should contain blanks or binary zeroes (LOW-VALUES in COBOL). <b>Assembler name:</b> DRROPTS <b>COBOL name:</b> DRS-DRRB-OPTIONS
<b>RETURN CODE</b>	A four-byte binary field, filled in by DRS, indicating the results of the call. A zero value indicates that the call was successful. A non-zero value indicates that an error occurred. The meanings of non-zero return codes are listed in <a href="#">“DRS/API Return Codes” on page 24.1</a> . <b>Assembler name:</b> DRRRC <b>COBOL name:</b> DRS-DRRB-RETURN-CODE

---

<b>OS RETURN CODE</b>	<p>A four-byte binary field, filled in by DRS, containing the operating system return code received by DRS. This is present only for some DRS return codes. The OS return code field format varies with the type of failure.</p> <p><b>Assembler name:</b> DRRRCS</p> <p><b>COBOL name:</b> DRS-DRRB-OS-RETURN-CODE</p>
<b>ATTR NUMBER IN ERROR</b>	<p>If the DRS return code indicates a failure related to attribute processing, DRS will fill this two-byte binary field with the number (relative to 1) of the attribute group which caused the failure.</p> <p>This field does not apply to a QURY or CMND call.</p> <p><b>Assembler name:</b> DRRATRE</p> <p><b>COBOL name:</b> DRS-DRRB-ATTR-IN-ERROR</p>
<b>DATA LINE COUNT</b>	<p>On a PUT call, this field indicates to DRS how many lines are being passed to DRS for addition to the report. For compatibility with the prior release, a value of either blanks or binary zeroes will be assumed to represent one line.</p> <p><b>Assembler name:</b> DRRLINCT</p> <p><b>COBOL name:</b> DRS-DRRB-DATA-LINE-CNT</p>
<b>DRDC ADDRESS</b>	<p>This field contains the address of an area which is initialized by DRS on the first call made and used on subsequent calls.</p> <p><b>Assembler name:</b> DRRDRDC</p> <p><b>COBOL name:</b> DRS-DRRB-DRDC-ADDRESS</p>
<b>REPORT LINE COUNT</b>	<p>A four-byte binary field where DRS stores a count of lines "PUT" to the report. It is returned by DRS on PUT and TERM calls.</p> <p><b>Assembler name:</b> DRRLINE</p> <p><b>COBOL name:</b> DRS-DRRB-REPORT-LINE-CNT</p>

The macro "DRSDRRB" in file LRS.DRS.V1R33.MACLIB is supplied to generate a DSECT of the REQUEST BLOCK. The copy member "DRSDRRBC" in file LRS.DRS.V1R33.MACLIB is supplied to generate a COBOL description of the REQUEST BLOCK.

---

## INIT Block (DRIB)

The INIT block is used by the caller to describe the attributes of a particular report. The INIT BLOCK can be anywhere in storage. DRS references it, but does not modify it.

The individual fields which make up the INIT BLOCK are:

<b>'DRIB'</b>	The control block identifier. <b>Assembler name:</b> DRIID <b>COBOL name:</b> DRS-DRIBC-ID or DRS-DRIBX-ID
<b>ATTRIBUTE COUNT</b>	A two-byte binary field specifying the number of attribute groups in this DRIB. There can be from 1 to 1,024 attribute groups in one DRIB. <b>Assembler name:</b> DRINATTR <b>COBOL name:</b> DRS-DRIBC-NBR-ATTR-GRPS or DRS-DRIBX-NBR-ATTR-GRPS
<b>EXTENSION INDICATOR</b>	An indicator of whether extended attribute groups are specified in this DRIB. Values of "Y" or "N" are valid. <b>Assembler name:</b> DRIXAGRP <b>COBOL name:</b> DRS-DRIBC-EXTENDED-ATTR or DRS-DRIBX-EXTENDED-ATTR
<b>DELETION INDICATOR</b>	An indicator of whether incomplete reports described by this DRIB (reports which have not had a TERM call) should be deleted when DRS or the application using DRS terminates. Values of "Y" or "N" are valid. <b>Assembler name:</b> DRIDELT <b>COBOL name:</b> DRS-DRIBC-DELETE-INCMPL or DRS-DRIBX-DELETE-INCMPL
<b>LAN ATTRIBUTES</b>	The address of LAN attributes received from DRS/PC. These attributes identify the source of the print file and can be used for creation of separators (banners). <b>Assembler name:</b> DRILANA <b>COBOL name:</b> DRS-DRIBC-LAN-ATTRIBUTE or DRS-DRIBX-LAN-ATTRIBUTE
<b>RECORD FORMAT</b>	The record format of the report. Valid values are "F" for fixed format, "V" for variable format, and "U" for undefined format. <b>Assembler name:</b> DRIRFT <b>COBOL name:</b> DRS-DRIBC-RFT or DRS-DRIBX-RFT
<b>CARRIAGE CONTROL</b>	The carriage control type of the report. Valid values are "A" for ASA, "M" for machine, or blank (X'40') for no carriage control. <b>Assembler name:</b> DRIRFC <b>COBOL name:</b> DRS-DRIBC-RFC or DRS-DRIBX-RFC

---

**BLOCKING INDICATOR**

A one-byte field indicating if the records in this report should be blocked ("B") or unblocked (blank).

**Assembler name:** DRIRFB

**COBOL name:** DRS-DRIBC-RFB or  
DRS-DRIBX-RFB

**RECORD LENGTH**

A two-byte binary field specifying the record length of records in this report. This length must include the carriage control byte, if carriage control is specified as "A" or "M". (This field is ignored for a record format of "U").

For fixed format reports, the minimum record length is 1 and the maximum record length is 32,760. For variable format reports, the minimum record length is 4 (the size of the four-byte RDW) for a null record without carriage control or 5 (the sum of the four-byte RDW and one byte of carriage control) for a null line with carriage control. The maximum record length is 32,756 for variable format reports.

**Assembler name:** DRILREC

**COBOL name:** DRS-DRIBC-LRECL or  
DRS-DRIBX-LRECL

**BLOCK SIZE**

A two-byte binary field specifying the block size of this report. For fixed format reports, the minimum is the record length and the maximum is 32,760. (However, block size must be a multiple of record length for fixed format reports). For variable format reports, the minimum is the record length plus 4, and the maximum is 32,760. For undefined format, the minimum is 1 and the maximum is 32,760.

**Assembler name:** DRIBLKS

**COBOL name:** DRS-DRIBC-BLKSIZE or  
DRS-DRIBX-BLKSIZE

**Note:** Although blocking is allowed, it is recommended that blocking not be used. Blocking increases the amount of storage used and gains no performance in JES.



---

The following fields are specified for each individual attribute group.

<b>DESTINATION/ NODE</b>	<p>The destination to which the dataset represented by this attribute group should be routed. If this field is not specified, a destination of LOCAL is assumed. If destination is specified, it must be a destination defined to JES, such as a remote number (Rnnnn), a special local number (Unnnn), or a name specified in a DESTID card in the JES initialization deck. The destination must be alphanumeric or national characters and begin with an alpha character. From 1 to 8 characters can be specified.</p> <p>An alternate meaning for this field is a JES node name. If this field contains a node name, then the extended version of the DRIB must be used, and the SYSOUT USERID field in the extended area must contain the destination.</p> <p><b>Assembler name:</b> DRIDEST <b>COBOL name:</b> DRS-DRIBC-DEST or DRS-DRIBX-DEST</p>
<b>WRITER NAME</b>	<p>Specifies the special writer name for the dataset. This field is optional. If writer name is specified, it must be alphanumeric or national characters. From 1 to 8 characters can be specified.</p> <p><b>Assembler name:</b> DRIWRTR <b>COBOL name:</b> DRS-DRIBC-WRITER or DRS-DRIBX-WRITER</p>
<b>FORM</b>	<p>Specifies the form name for the dataset. The form name is optional. If form name is specified, it must be alphanumeric or national characters. From 1 to 4 characters can be specified.</p> <p><b>Assembler name:</b> DRIFORM <b>COBOL name:</b> DRS-DRIBC-FORM or DRS-DRIBX-FORM</p>
<b>FCB</b>	<p>Specifies the FCB image to be used to print the dataset. The FCB is optional. If no FCB is specified, the JES default will be used. If FCB is specified, it must be alphanumeric or national characters. From 1 to 4 characters can be specified.</p> <p><b>Assembler name:</b> DRIFCB <b>COBOL name:</b> DRS-DRIBC-FCB or DRS-DRIBX-FCB</p>
<b>UCS</b>	<p>Specifies the UCS image to be used to print the dataset. The UCS is optional. If no UCS is specified, the JES default will be used. If UCS is specified, it must be alphanumeric characters. From 1 to 4 characters can be specified.</p> <p><b>Assembler name:</b> DRIUCS <b>COBOL name:</b> DRS-DRIBC-UCS or DRS-DRIBX-UCS</p>

---

<b>CLASS</b>	<p>Specifies the SYSOUT class of the dataset. This field is optional. It is one character, and if it is specified, it must be alphanumeric or an asterisk. When an asterisk is specified, the MSGCLASS of the job or started task is used. If CLASS is not specified, the dataset will receive the class specified on a referenced OUTPUT JCL statement, if any. Otherwise, the dataset will receive the MSGCLASS associated with the job or started task.</p> <p><b>Assembler name:</b> DRICLAS</p> <p><b>COBOL name:</b> DRS-DRIBC-CLASS or DRS-DRIBX-CLASS</p>
<b>HOLD</b>	<p>Specifies whether or not the dataset should be in “HOLD” status. Valid values are “Y” and “N”. This field is required.</p> <p><b>Assembler name:</b> DRIHOLD</p> <p><b>COBOL name:</b> DRS-DRIBC-HOLD or DRS-DRIBX-HOLD</p>
<b>COPY COUNT</b>	<p>A two-byte binary field specifying how many copies of the dataset are requested. Values can range from 1 to 255. This field is required.</p> <p><b>Assembler name:</b> DRICOPY</p> <p><b>COBOL name:</b> DRS-DRIBC-COPIES or DRS-DRIBX-COPIES</p>
<b>DDNAME</b>	<p>An eight-byte character field identifying a DDNAME to be used when DRS allocates the report. If it is not specified, DRS will assign a DDNAME.</p> <p><b>Assembler name:</b> DRIDDN</p> <p><b>COBOL name:</b> DRS-DRIBC-DDNAME or DRS-DRIBX-DDNAME</p>

---

The following fields are part of the “extended” DRIB. They are part of the DRIB only if a value of “Y” is specified in the EXTENSION INDICATOR field of the DRIB.

<b>EXTENDED FORM NAME</b>	<p>Specifies a 8-position form name for the dataset. This field is optional. If specified, it will be used instead of the value specified in the 4-position form name field in the non-extended portion of the DRIB. If specified, it must be alphanumeric or national characters. Up to eight characters can be specified. (Some versions of the operating system do not support 8-position form names on dynamically allocated datasets. If you specify an 8-position form name and the operating system which you run does not support it, allocation will fail).</p> <p><b>Assembler name:</b> DRIFORM <b>COBOL name:</b> DRS-DRIBX-EXTENDED-FORM</p>
<b>SYSOUT USERID</b>	<p>For a DRIB where the DESTINATION field contains a node name, this field contains the destination.</p> <p><b>Assembler name:</b> DRIUSRID <b>COBOL name:</b> DRS-DRIBX-USERID</p>
<b>OUTPUT STMT STEPNAME</b>	<p>Specifies the stepname to be used to associate this dataset with an OUTPUT JCL statement (if any).</p> <p><b>Assembler name:</b> DRIOUTST <b>COBOL name:</b> DRS-DRIBX-OUTPUT-STMT-STEP</p>
<b>OUTPUT STMT PROCEDURE</b>	<p>Specifies the name of the procedure step to be used to associate this dataset with an OUTPUT JCL statement (if any).</p> <p><b>Assembler name:</b> DRIOUTPR <b>COBOL name:</b> DRS-DRIBX-OUTPUT-STMT-PROC</p>
<b>OUTPUT STMT NAME</b>	<p>Specifies the name of the OUTPUT JCL statement to be associated with this dataset (if any).</p> <p><b>Assembler name:</b> DRIOUTNM <b>COBOL name:</b> DRS-DRIBX-OUTPUT-STMT-NAME</p>
<b>UCS VERIFICATION</b>	<p>Specifies whether or not UCS verification is requested. Valid values are “Y” or “N”.</p> <p><b>Assembler name:</b> DRIUCSVR <b>COBOL name:</b> DRS-DRIBX-UCS-VERIF</p>
<b>FOLD</b>	<p>Specifies whether or not the chain or train for the universal character set should be loaded in fold mode. (Fold mode is described in “IBM 2821 Control Unit Component Description”). Valid values are “Y” or “N”.</p> <p><b>Assembler name:</b> DRIFOLD <b>COBOL name:</b> DRS-DRIBX-FOLD</p>
<b>BURST</b>	<p>Specifies, for a dataset to be printed on a 3800 Printing Subsystem, whether forms should be sent to the burster-trimmer-stacker to be burst into separate sheets (value of “B”) or whether they should be sent to the continuous form stacker (value of “C”).</p> <p><b>Assembler name:</b> DRIBURST <b>COBOL name:</b> DRS-DRIBX-BURST</p>

---

<b>COPY GROUP(S)</b>	<p>Specifies, for a dataset to be printed on a 3800 Printing Subsystem, how many copies of each page are to be printed before the next page is printed. Each of the 8 fields is a 1-byte binary field representing a value from 1 to 255. (The sum of the 8 copy groups cannot exceed 255).</p> <p><b>Assembler name:</b> DRICOPYG  <b>COBOL name:</b> DRS-DRIBX-COPY-GROUPS</p>
<b>CHARACTER ARRANGEMENT TABLE(S)</b>	<p>Specifies, for a dataset to be printed on a 3800 Printing Subsystem, the name of a character-arrangement table to be used for printing this dataset. Each of the 4 character arrangement table fields is a 4-byte alphanumeric or national field. For more information on character-arrangement tables, see "3800 Printing Subsystem Programmer's Guide".</p> <p><b>Assembler name:</b> DRICHARS  <b>COBOL name:</b> DRS-DRIBX-3800-CHAR-ARR</p>
<b>FLASH</b>	<p>Specifies, for a dataset to be printed on a 3800 Printing Subsystem, the name of the forms overlay to be used in printing this dataset. The name is 1 to 4 alphanumeric or national characters.</p> <p><b>Assembler name:</b> DRIFLASH  <b>COBOL name:</b> DRS-DRIBX-FLASH-OVERLAY</p>
<b>COPY MODIFICATION MODULE</b>	<p>Specifies, for a dataset to be printed on a 3800 Printing Subsystem, the name of a copy-modification module that tells JES how to print this SYSOUT dataset. For more information on copy-modification, see "Data Administration: Utilities". The module name is 1 to 4 alphanumeric or national characters.</p> <p><b>Assembler name:</b> DRICMOD  <b>COBOL name:</b> DRS-DRIBX-3800-COPY-MODULE</p>
<b>FLASH FORMS OVERLAY COUNT</b>	<p>Specifies, for a dataset to be printed on a 3800 Printing Subsystem, the number of copies on which the forms overlay (specified in the FLASH field) is to be printed. This is a 1-byte binary field representing a count from 0 to 255. (A count of 0 indicates that all copies should be flashed).</p> <p><b>Assembler name:</b> DRIFLSCT  <b>COBOL name:</b> DRS-DRIBX-3800-FLASH-COUNT</p>
<b>COPY MODULE TABLE REFERENCE</b>	<p>Specifies, for a dataset to be printed on a 3800 Printing Subsystem, which table name in the character-arrangement table should be used, when a copy-modification module is specified. This is a 1-byte character field where a value of 0 represents the first table name, a value of 1 represents the second table name, etc.</p> <p><b>Assembler name:</b> DRICMTRC  <b>COBOL name:</b> DRS-DRIBX-COPY-MOD-TAB</p>
<b>SUBSYSTEM NAME</b>	<p>Specifies the name of the subsystem that is to process the allocation request, when that subsystem is not the default subsystem. If specified, this 4-byte field must begin with an alphabetic or national character. All other characters must be alphanumeric or national.</p> <p><b>Assembler name:</b> DRISSNM  <b>COBOL name:</b> DRS-DRIBX-SUBSYS-NAME</p>

---

**SUBSYSTEM  
PARAMETERS**

Specifies parameters to be passed to the subsystem specified in the SUBSYSTEM NAME field. There are no restrictions on the format of the contents of this 16-byte field. This field is ignored if it contains either all binary zeroes or all blanks.

**Assembler name:** DRISSPRM

**COBOL name:** DRS-DRIBX-SUBSYS-PARM

**JOBNAME**

Specifies the jobname that DRS will use to create this SYSOUT dataset on the spool. This field is optional. If specified, the value must contain alphanumeric or national characters, and the first character must be alphabetic or national. From 1 to 8 characters can be specified. Additionally, DRS/API modules must be loaded from an authorized library, MVS/SP 4.1.0 or greater must be running and the HBB4410 extensions to MVS must be present in order to specify a value for jobname. CICS application programs that call DRS/API will not be able to use this feature, due to APF authorization restrictions in the CICS environment.

**Assembler name:** DRIJOBNM

**COBOL name:** DRS-DRIBX-ASSIGNED-JOBNAME

**USER AREA**

This field supplies the user with 32 bytes in which to store data unique to this dataset. This user-supplied data is passed to the "Add Lines" exit. (This field could be used for some information to be printed on banner pages created by the "Add Lines" exit). There are no restrictions on the format of this data.

**Assembler name:** DRIUSER

**COBOL name:** DRS-DRIBX-USER-BYTES

---

The following fields are also part of the “extended” DRIB. If the report is to be created as a DASD file rather than a SYSOUT dataset on the JES spool, these labels should be used to supply the attributes of the DASD file.

<b>MEMBER NAME</b>	An 8-byte character field which supplies the member name of a PDS to be created by DRS. <b>Assembler name:</b> DRIDMEMB <b>COBOL name:</b> DRS-DRIBX-DASD-MEMBER
<b>UNIT NAME</b>	An 8-byte character field which supplies the unit type upon which a DASD print file should be allocated. <b>Assembler name:</b> DRIDUNIT <b>COBOL name:</b> DRS-DRIBX-DASD-UNIT
<b>INITIAL DISPOSITION</b>	A 1-byte character field which supplies the initial disposition of a DASD print file. Valid values are “N” (NEW), “O” (OLD), “S” (SHR), and “M” (MOD). The default, if not specified, is “N”. <b>Assembler name:</b> DRIDSTAT <b>COBOL name:</b> DRS-DRIBX-DASD-ISTATUS
<b>NORMAL DISPOSITION</b>	A 1-byte character field which supplies the normal disposition of a DASD print file. Valid values are “C” (CATLG), “U” (UNCATLG), “K” (KEEP), and “D” (DELETE). The default, if not specified, is “K”. <b>Assembler name:</b> DRIDNDSP <b>COBOL name:</b> DRS-DRIBX-DASD-NORMAL-DISP
<b>RETENTION PERIOD IND</b>	A 1-byte character field which indicates whether the retention period has been specified. Valid values are “Y” and “N”. The default, if not specified, is “N”. <b>Assembler name:</b> DRIDRTPS <b>COBOL name:</b> DRS-DRIBX-DASD-RETPTS
<b>RETENTION PERIOD</b>	A 2-byte binary field which supplies the retention period of a DASD print file. Valid values are in the range of 0 to 9999, inclusive. This field is not used by DRS unless DRS actually creates the print file dataset and the retention period indicator field is set to “Y”. <b>Assembler name:</b> DRIDRTPD <b>COBOL name:</b> DRS-DRIBX-DASD-RETPD
<b>AVERAGE RECORD SIZE</b>	A 2-byte binary field which supplies the average record length, in bytes, of the data. Using the average record length, the system computes the block size and the number of tracks to allocate. Valid values are in the range of 1 to 65535. This field is not used by DRS unless DRS actually creates the print file dataset. SMS must be active for this field to be effective. <b>Assembler name:</b> DRIDAVRS <b>COBOL name:</b> DRS-DRIBX-DASD-AVGREC-SIZE

---

<b>AVERAGE RECORD UNIT</b>	<p>A 1-byte character field which supplies the average record unit of allocation of a DASD print file. Valid values are “U” (UNITS), “K” (THOUSANDS), and “M” (MILLIONS). This field is not used by DRS unless DRS actually creates the print file dataset. SMS must be active for this field to be effective.</p> <p><b>Assembler name:</b> DRIDAVRU  <b>COBOL name:</b> DRS-DRIBX-DASD-AVGREC-UNIT</p>
<b>VOLUME NAME</b>	<p>A 6-byte character field which supplies the volume on which a DASD print file should be allocated.</p> <p><b>Assembler name:</b> DRIDVOL  <b>COBOL name:</b> DRS-DRIBX-DASD-VOLUME</p>
<b>EXPIRATION DATE</b>	<p>A 7-byte character field which supplies the Julian expiration date of a DASD print file. The format of the field is “yyyyddd”, where “yyyy” is the 4-digit year of the expiration date and “ddd” is the 3-digit Julian day of the expiration date. This field is not used by DRS unless DRS actually creates the print file dataset.</p> <p><b>Assembler name:</b> DRIDEXPD  <b>COBOL name:</b> DRS-DRIBX-DASD-EXPDT</p>
<b>DATA SET TYPE</b>	<p>A 1-byte character field which supplies the dataset type of a DASD print file. Valid values are “L” (LIBRARY or PDSE) and “P” (PDS). This field is not used by DRS unless DRS actually creates the print file dataset. SMS must be active for this field to be effective.</p> <p><b>Assembler name:</b> DRIDDSTY  <b>COBOL name:</b> DRS-DRIBX-DASD-DSNTYPE</p>
<b>RELEASE UNUSED SPACE</b>	<p>A 1-byte character field which indicates whether unused space should be released from a DASD print file when the file is unallocated. Valid values are “Y” and “N”. This field is not used by DRS unless DRS actually creates the print file dataset. The default, if not specified, is “N”.</p> <p><b>Assembler name:</b> DRIDRLSE  <b>COBOL name:</b> DRS-DRIBX-DASD-RLSE</p>
<b>DASD INDICATOR FLAG</b>	<p>A 1-byte character field indicating whether or not this report is to be allocated as a DASD file (rather than a SYSOUT file). Valid values are “Y” and “N”.</p> <p><b>Assembler name:</b> DRIDDASD  <b>COBOL name:</b> DRS-DRIBX-DASD-INDICATOR</p>
<b>DASD ALLOCATION TYPE</b>	<p>A 1-byte character field which supplies the allocation type for a DASD print file. Valid values are “B” (blocks), “C” (cylinders) and “T” (tracks).</p> <p><b>Assembler name:</b> DRIDATYP  <b>COBOL name:</b> DRS-DRIBX-DASD-ALLOC-TYPE</p>
<b>DASD PRIMARY ALLOCATION</b>	<p>A 2-byte binary field which supplies the amount for the primary allocation of a DASD print file.</p> <p><b>Assembler name:</b> DRIDPRI  <b>COBOL name:</b> DRS-DRIBX-DASD-PRI-ALLOC</p>

---

---

**DASD  
SECONDARY  
ALLOCATION**

A 2-byte binary field which supplies the amount for the secondary allocation of a DASD print file.

**Assembler name:** DRIDSEC

**COBOL name:** DRS-DRIBX-DASD-SEC-ALLOC

**DASD  
DIRECTORY  
BLOCKS**

A 2-byte binary field which supplies the number of directory blocks to be used for allocation of a DASD print file that is a PDS with a disposition of "NEW".

**Assembler name:** DRIDDIR

**COBOL name:** DRS-DRIBX-DASD-DIR-BLOCKS

**DATASET NAME**

A 44-byte character field which supplies the dataset name of a DASD print file.

**Assembler name:** DRIDDSNM

**COBOL name:** DRS-DRIBX-DASD-DSNAME

**VOLUME COUNT**

A 2-byte binary field which supplies the volume count for volumes on which a DASD print file is to be allocated.

**Assembler name:** DRIDVLCT

**COBOL name:** DRS-DRIBX-DASD-VOLUME-CNT

**UNIT COUNT**

A 2-byte binary field which supplies the number of device units to be used for a DASD print file.

**Assembler name:** DRIDUNCT

**COBOL name:** DRS-DRIBX-DASD-UNIT-CNT

**STORAGE CLASS**

An 8-byte character field which supplies the SMS storage class of a DASD print file. This field is not used by DRS unless DRS actually creates the print file dataset. SMS must be active for this field to be effective.

**Assembler name:** DRIDSTCL

**COBOL name:** DRS-DRIBX-DASD-STORCLAS

**MANAGEMENT  
CLASS**

An 8-byte character field which supplies the SMS management class of a DASD print file. This field is not used by DRS unless DRS actually creates the print file dataset. SMS must be active for this field to be effective.

**Assembler name:** DRIDMGCL

**COBOL name:** DRS-DRIBX-DASD-MGMTCLAS

**DATA CLASS**

An 8-byte character field which supplies the SMS data class of a DASD print file. This field is not used by DRS unless DRS actually creates the print file dataset. SMS must be active for this field to be effective.

**Assembler name:** DRIDDACL

**COBOL name:** DRS-DRIBX-DASD-DATACLAS

The macro "DRSDRIB" in file LRS.DRS.V1R33.MACLIB is supplied to generate a DSECT of the INIT BLOCK. The copy member "DRSDRIBC" in file LRS.DRS.V1R33.MACLIB is supplied to generate a COBOL description of the INIT BLOCK without extensions. Member "DRSDRIBX" generates a COBOL description of the INIT BLOCK with extensions.



---

## Termination Block (DRTB)

The TERMINATION block can be used on a TERM call to modify some attributes of a report or to delete a report. If this control block is supplied on a TERM call, DRS uses the information in the control block, but does not modify it.

The individual fields which make up the TERMINATION BLOCK are:

<b>'DRTB'</b>	The control block identifier. <b>Assembler name:</b> DRTID <b>COBOL name:</b> DRS-DRTB-ID
<b>ATTRIBUTE COUNT</b>	A two-byte binary field specifying the number of attribute groups in this DRTB. There can be from 1 to 1,024 attribute groups in one DRTB. <b>Assembler name:</b> DRTNATTR <b>COBOL name:</b> DRS-DRTB-NBR-ATTR-GRPS

The following fields are specified for each individual attribute group which is to be modified.

<b>ATTRIBUTE INDEX</b>	A two-byte binary field specifying the number of the attribute group which is to be modified by this occurrence. <b>Assembler name:</b> DRTATRIX <b>COBOL name:</b> DRS-DRTB-ATTR-INDX
<b>OVERRIDING DISPOSITION</b>	Specifies if this dataset is to be deleted at termination rather than kept on the JES spool for printing. A value of "D" indicates that the dataset should be deleted. <b>Assembler name:</b> DRTDISP <b>COBOL name:</b> DRS-DRTB-DISP
<b>OVERRIDING SYSOUT CLASS</b>	Specifies the SYSOUT class of the dataset described by this attribute group, which is to replace the CLASS specified in the DRIB when this report was initiated. It is one character and must be alphanumeric or an asterisk. When an asterisk is specified, the MSGCLASS of the job or started task is assumed. <b>Assembler name:</b> DRTCLASS <b>COBOL name:</b> DRS-DRTB-CLASS
<b>OVERRIDING HOLD</b>	Specifies, via a value of "Y", that a dataset which was not marked as "held" in the DRIB should be held. <b>Assembler name:</b> DRTHOLD <b>COBOL name:</b> DRS-DRTB-HOLD
<b>OVERRIDING NOHOLD</b>	Specifies, via a value of "Y", that a dataset which was marked as "held" in the DRIB should not be held. <b>Assembler name:</b> DRTNHOLD <b>COBOL name:</b> DRS-DRTB-NHOLD

---

**OVERRIDING  
DESTINATION**

The alphanumeric or national characters representing the destination to which the dataset represented by this attribute group should be routed, rather than to the DESTINATION specified in the DRIB when this report was initiated. If this field is specified, it must be a destination defined to JES, such as a remote number (Rnnnn), a special local number (Unnnn), or a name specified in a DESTID card in the JES initialization deck. The destination must be alphanumeric or national characters and begin with an alpha character. Up to eight characters may be specified.

**Assembler name:** DRTDEST

**COBOL name:** DRS-DRTB-DEST

The macro "DRSDRTB" in file LRS.DRS.V1R33.MACLIB is supplied to generate a DSECT of the TERMINATION BLOCK. The copy member "DRSDRTBC" in file LRS.DRS.V1R33.MACLIB is supplied to generate a COBOL description of the TERMINATION BLOCK.

---

## Query Block (DRQB)

The QUERY block is used on a QURY call to inquire about some facet of DRS. A query can be general (about all of the active reports in DRS), detail (about a specific report), system (about the functioning of DRS), general output statement (about all dynamically defined OUTPUT JCL statements), or detail output statement (about a specific dynamically defined OUTPUT JCL statement).

The individual fields which make up the QUERY BLOCK are:

<b>'DRQB'</b>	The control block identifier. <b>Assembler name:</b> DRQID <b>COBOL name:</b> DRS-DRQBG-ID, DRS-DRQBD-ID, DRS-DRQBS-ID, DRS-DRQBO-ID, or DRS-DRQBL-ID
<b>QUERY TYPE</b>	Specifies which type of query is being requested. The possible values are "GENL" for a general query, "DETL" for a detail query, "SYS" for a system query, "OUTG" for a general dynamically defined OUTPUT JCL statement query, or "OUTD" for a detail dynamically defined OUTPUT JCL statement query. <b>Assembler name:</b> DRQTYP <b>COBOL name:</b> DRS-DRQBG-QUERY-TYPE, DRS-DRQBD-QUERY-TYPE, DRS-DRQBS-QUERY-TYPE, DRS-DRQBO-QUERY-TYPE, or DRS-DRQBL-QUERY-TYPE
<b>REPORT ID</b>	For a general query (query type of "GENL"), this specifies the "starting point" for returning information about reports. If it is left blank, DRS will return information starting with the first active report. For a detail query (query type of "DETL"), this is the report id of the report to which the query applies. For a general dynamically defined OUTPUT JCL statement query (query type of "OUTG"), this specifies the "starting point" for returning information about dynamically defined OUTPUT JCL statements. If it is left blank, DRS will return information starting with the first dynamically defined OUTPUT JCL statement. For a detail dynamically defined OUTPUT JCL statement query (query type of "OUTD"), this specifies the name of the dynamically defined OUTPUT JCL statement to which the query applies. This field does not apply to a system query. <b>Assembler name:</b> DRQRID or DRQOUTPN <b>COBOL name:</b> DRS-DRQBG-REPORT-ID, DRS-DRQBD-REPORT-ID, DRS-DRQBO-OUTPUT-NAME, or DRS-DRQBL-OUTPUT-NAME

---

**NUMBER OF SLOTS AVAILABLE**

For a general query, a two-byte binary field specifying the number of “slots” available in the DRQB for DRS to return information about active DRS reports.

This field does not apply to a detail query or a system query.

**Assembler name:** DRQNSLT

**COBOL name:** DRS-DRQBG-NBR-SLOTS-AVAIL or DRS-DRQGO-NBR-SLOTS-AVAIL

**NUMBER OF SLOTS USED**

For a general query, DRS fills in this two-byte binary field with a value representing the number of slots that it filled with information about active DRS reports.

This field does not apply to a detail query or a system query.

**Assembler name:** DRQUSLT

**COBOL name:** DRS-DRQBG-NBR-SLOTS-USED or DRS-DRQGO-NBR-SLOTS-USED

**EXTENSION SIZE**

This two-byte binary field is required for all QURY calls. It must specify the size, in bytes, of the extension in the DRQB being used for the call. For a general query, the size must be equal to the size of one slot times the number of slots specified in the NUMBER OF SLOTS AVAILABLE FIELD. For a detail query, the size will be equal to either the size of the “basic” attribute group information, or to the sum of the size of the “basic” attribute group information and the size of the “extended” attribute group information. For a system query, the size must be equal to the size of the system query extension.

**Assembler name:** DRQESIZ

**COBOL name:** DRS-DRQBG-EXTENSION-SIZE, DRS-DRQBD-EXTENSION-SIZE, DRS-DRQBS-EXTENSION-SIZE, DRS-DRQBO-EXTENSION-SIZE, or DRS-DRQBL-EXTENSION-SIZE

**REQUESTED ATTR**

For a detail query, this two-byte binary field must specify the number of the particular attribute group of the report which is being queried.

This field does not apply to a general query or a system query.

**Assembler name:** DRQATR

**COBOL name:** DRS-DRQBD-ATTR-GRP-REQUESTED

The remaining fields in the DRQB are filled in by DRS and vary by the type of query requested.

---

For a general query, the extension consists of one or more “slots”. The following fields exist in each slot.

<b>REPORT ID</b>	An 8-byte character field containing the report id of the report whose information was recorded in this slot by DRS. <b>Assembler name:</b> DRQSRID <b>COBOL name:</b> DRS-DRQBG-RPT-ID
<b>CURRENT LINE COUNT</b>	A four-byte binary field containing the number of lines in this report at the time DRS recorded the information in this slot. <b>Assembler name:</b> DRQSLIN <b>COBOL name:</b> DRS-DRQBG-LINE-COUNT
<b>LAST REQUEST TIME</b>	A four-byte binary field containing the time stamp of the last DRS call for this report. <b>Assembler name:</b> DRQSTIM <b>COBOL name:</b> DRS-DRQBG-TIME-STAMP
<b>NUMBER OF ATTRS</b>	A two-byte binary field containing the number of attribute groups in this report. <b>Assembler name:</b> DRQSNAT <b>COBOL name:</b> DRS-DRQBG-NBR-ATTRS

---

For a detail query, the extension consists of either the “basic” attribute group information or both the “basic” attribute group information and the “extended” attribute group information. The description of these fields is identical to that used in the DRIB. Assembler field names start with “DRQ”, and COBOL field names start with “DRS-DRQBS-”.

For a system query, the fields in the extension are:

<b>RELEASE NUMBER</b>	<p>A four-byte character field containing the DRS version and release number (currently “13.3”).</p> <p><b>Assembler name:</b> DRQSRLSE</p> <p><b>COBOL name:</b> DRS-DRQBS-RELEASE</p> <p>The first byte of the 4-byte RELEASE NUMBER field is the DRS version number (currently “1”).</p> <p><b>Assembler name:</b> DRQSVER#</p> <p><b>COBOL name:</b> DRS-DRQBS-VERSION-NUMBER</p> <p>Bytes 2, 3 and 4 of the 4-byte RELEASE NUMBER field is the DRS release number (currently “3.3”).</p> <p><b>Assembler name:</b> DRQSREL#</p> <p><b>COBOL name:</b> DRS-DRQBS-RELEASE-NUMBER</p>
<b>FIX LEVEL</b>	<p>A one-byte binary field containing the DRS fix level.</p> <p><b>Assembler name:</b> DRQSFIX</p> <p><b>COBOL name:</b> DRS-DRQBS-FIX-LEVEL</p>
<b>MAINTENANCE LEVEL</b>	<p>A one-byte character field containing the maintenance level of DRS.</p> <p><b>Assembler name:</b> DRQSMNT</p> <p><b>COBOL name:</b> DRS-DRQBS-MAINT-LEVEL</p>
<b>PRODUCT STATUS</b>	<p>A one-byte character field containing the DRS product status.</p> <p>“L” = product is licensed; “T” = product is on an unexpired trial; “X” = product is on an expired trial; “I” = product key is invalid.</p> <p><b>Assembler name:</b> DRQSSTAT</p> <p><b>COBOL name:</b> DRS-DRQBS-PRODUCT-STATUS</p>
<b>CUSTOMER ID</b>	<p>A six-byte character field containing the customer ID.</p> <p><b>Assembler name:</b> DRQSCUST</p> <p><b>COBOL name:</b> DRS-DRQBS-CUSTID</p>
<b>JOB NAME LENGTH</b>	<p>A one-byte binary field containing the length of the DRS job name.</p> <p><b>Assembler name:</b> DRQSJOBL</p> <p><b>COBOL name:</b> DRS-DRQBS-JOBNAME-LENGTH</p>
<b>JOB NAME</b>	<p>An eight-byte character field containing the DRS job name.</p> <p><b>Assembler name:</b> DRQSJOBN</p> <p><b>COBOL name:</b> DRS-DRQBS-JOBNAME</p>
<b>DRS INIT DATE</b>	<p>A four-byte packed field containing the DRS initialization date in 00YYDDSS format.</p> <p><b>Assembler name:</b> DRQSIDAT</p> <p><b>COBOL name:</b> DRS-DRQBS-INIT-DATE</p>

---

<b>DRS INIT TIME</b>	<p>A four-byte field containing the DRS initialization time in HHMMSSSTH format.</p> <p><b>Assembler name:</b> DRQSITIM</p> <p><b>COBOL name:</b> DRS-DRQBS-INIT-TIME</p>
<b>EXIT WORDS</b>	<p>An eight-byte field containing information passed to user exits.</p> <p><b>Assembler name:</b> DRQSUSER</p> <p><b>COBOL name:</b> DRS-DRQBS-USER-WORDS</p>
<b>INITIAL SUBTASKS</b>	<p>A four-byte binary field containing the number of subtasks DRS was initially requested to use to process DRS calls.</p> <p><b>Assembler name:</b> DRQSTCBI</p> <p><b>COBOL name:</b> DRS-DRQBS-INITIAL-SUBTASKS</p>
<b>ACTIVE SUBTASKS</b>	<p>A four-byte binary field containing the number of subtasks DRS is currently using to process DRS calls.</p> <p><b>Assembler name:</b> DRQSTCBA</p> <p><b>COBOL name:</b> DRS-DRQBS-ACTIVE-SUBTASKS</p>
<b>NUMBER OF ACTIVE REPORTS</b>	<p>A four-byte binary field containing the number of reports currently active to DRS.</p> <p><b>Assembler name:</b> DRQSARCA</p> <p><b>COBOL name:</b> DRS-DRQBS-NBR-ACTIVE-REPORTS</p>
<b>REPORT ID COUNTER</b>	<p>A four-byte binary field containing the number of reports processed by DRS since initialization.</p> <p><b>Assembler name:</b> DRQSRID#</p> <p><b>COBOL name:</b> DRS-DRQBS-REPORT-ID-CNT</p>
<b>USER EXIT STATUS</b>	<p>A four-byte binary field where each bit represents a DRS exit. An enabled exit is represented by a bit being on.</p> <p><b>Assembler name:</b> DRQSXSTA</p> <p><b>COBOL name:</b> DRS-DRQBS-USER-EXIT-STATUS</p>
<b>USER EXIT LENGTHS</b>	<p>Eight four-byte binary fields containing the size of DRS exit modules.</p> <p><b>Assembler name:</b> DRQSXLEN</p> <p><b>COBOL name:</b> DRS-DRQBS-USER-EXIT-LENGTHS</p>
<b>USER EXIT ADDRESSES</b>	<p>Eight four-byte binary fields containing the entry point addresses of DRS exit modules.</p> <p><b>Assembler name:</b> DRQSXEPA</p> <p><b>COBOL name:</b> DRS-DRQBS-USER-EXIT-ADDRESSES</p>
<b>DRS STATISTICS</b>	<p>Seven 56-byte areas containing statistics concerning INIT, PUT, TERM, QUERY, SHUT, CMND, and OUTP calls.</p> <p><b>Assembler name:</b> DRQSINIT, DRQSPUT, DRQSTERM, DRQSQURY, DRQSSHUT, DRQSCMND, and DRQSOUTP</p> <p><b>COBOL name:</b> DRS-DRQBS-INIT-CALL, DRS-DRQBS-PUT-CALL, DRS-DRQBS-TERM-CALL, DRS-DRQBS-QUERY-CALL, DRS-DRQBS-SHUT-CALL, DRS-DRQBS-CMND-CALL, DRS-DRQBS-OUTP-CALL</p>

---

---

<b>TOTAL DATASETS STARTED</b>	A four-byte binary field containing a count of datasets initialized by DRS.
	<b>Assembler name:</b> DRQSDSST
	<b>COBOL name:</b> DRS-DRQBS-DS-STARTS
<b>TOTAL DATASETS COMPLETED (NORMAL)</b>	A four-byte binary field containing a count of datasets completed normally.
	<b>Assembler name:</b> DRQSDSKP
	<b>COBOL name:</b> DRS-DRQBS-DS-COMPLT-NORMAL
<b>TOTAL DATASETS COMPLETED (DELETE)</b>	A four-byte binary field containing a count of datasets completed with deletion requested at termination.
	<b>Assembler name:</b> DRQSDSDL
	<b>COBOL name:</b> DRS-DRQBS-DS-COMPLT-DELETE
<b>TOTAL LINES WRITTEN</b>	A four-byte binary field containing a count of lines written to all DRS reports.
	<b>Assembler name:</b> DRQSTLIN
	<b>COBOL name:</b> DRS-DRQBS-DS-TOTAL-LINES
<b>TOTAL OUTPUT STATEMENTS ADDED</b>	A four-byte binary field containing a count of OUTPUT JCL statements added via the OUTP call.
	<b>Assembler name:</b> DRQSOUTA
	<b>COBOL name:</b> DRS-DRQBS-OUTJCL-ADD
<b>TOTAL OUTPUT STATEMENTS DELETED</b>	A four-byte binary field containing a count of OUTPUT JCL statements deleted via the OUTP call.
	<b>Assembler name:</b> DRQSOUTD
	<b>COBOL name:</b> DRS-DRQBS-OUTJCL-DEL
<b>TOTAL GETMAINs BELOW</b>	A four-byte binary field containing a count of virtual storage GETMAINs below the 16M line.
	<b>Assembler name:</b> DRQSVBGT
	<b>COBOL name:</b> DRS-DRQBS-VS-GM-BELOW-CNT
<b>TOTAL GETMAIN ERRORS BELOW</b>	A four-byte binary field containing a count of failures of virtual storage GETMAINs below the 16M line.
	<b>Assembler name:</b> DRQSVBGE
	<b>COBOL name:</b> DRS-DRQBS-VS-GM-BELOW-ERROR
<b>TOTAL FREEMAINs BELOW</b>	A four-byte binary field containing a count of virtual storage FREEMAINs below the 16M line.
	<b>Assembler name:</b> DRQSVBFR
	<b>COBOL name:</b> DRS-DRQBS-VS-FM-BELOW-CNT



---

<b>NET STORAGE BELOW</b>	<p>A four-byte binary field containing the net amount of storage acquired below the 16M line.</p> <p><b>Assembler name:</b> DRQSVBNT</p> <p><b>COBOL name:</b> DRS-DRQBS-NS-BELOW</p>
<b>MAX STORAGE BELOW</b>	<p>A four-byte binary field containing the highest amount of storage acquired below the 16M line.</p> <p><b>Assembler name:</b> DRQSVBMX</p> <p><b>COBOL name:</b> DRS-DRQBS-MAX-BELOW</p>
<b>TOTAL GETMAINS ABOVE</b>	<p>A four-byte binary field containing a count of virtual storage GETMAINS above the 16M line.</p> <p><b>Assembler name:</b> DRQSVAGT</p> <p><b>COBOL name:</b> DRS-DRQBS-VS-GM-ABOVE-CNT</p>
<b>TOTAL GETMAIN ERRORS ABOVE</b>	<p>A four-byte binary field containing a count of failures of virtual storage GETMAINS above the 16M line.</p> <p><b>Assembler name:</b> DRQSVAGE</p> <p><b>COBOL name:</b> DRS-DRQBS-VS-GM-ABOVE-ERROR</p>
<b>TOTAL FREEMAINS ABOVE</b>	<p>A four-byte binary field containing a count of virtual storage FREEMAINS above the 16M line.</p> <p><b>Assembler name:</b> DRQSVAFR</p> <p><b>COBOL name:</b> DRS-DRQBS-VS-FM-ABOVE-CNT</p>
<b>NET STORAGE ABOVE</b>	<p>A four-byte binary field containing the net amount of storage acquired above the 16M line.</p> <p><b>Assembler name:</b> DRQSVANT</p> <p><b>COBOL name:</b> DRS-DRQBS-VS-NS-ABOVE</p>
<b>MAX STORAGE ABOVE</b>	<p>A four-byte binary field containing the highest amount of storage acquired above the 16M line.</p> <p><b>Assembler name:</b> DRQSVAMX</p> <p><b>COBOL name:</b> DRS-DRQBS-VS-MAX-ABOVE</p>
<b>LOG OPTIONS</b>	<p>A one-byte binary field specifying the log options in effect in DRS. Bit 0 on indicates that logging is in effect. Bit 1 on indicates that the log dataset should be “held”.</p> <p><b>Assembler name:</b> DRQSLOPT</p> <p><b>COBOL name:</b> DRS-DRQBS-LOG-OPTS</p>
<b>LOG DDNAME</b>	<p>An eight-byte character field containing the DDNAME of the DRS log dataset.</p> <p><b>Assembler name:</b> DRQSLDD</p> <p><b>COBOL name:</b> DRS-DRQBS-LOG-DDNAME</p>
<b>LOG CLASS</b>	<p>A one-byte character field containing the SYSOUT class of the DRS log dataset.</p> <p><b>Assembler name:</b> DRQSLCLS</p> <p><b>COBOL name:</b> DRS-DRQBS-LOG-CLASS</p>

---

---

<b>LOG DESTINATION</b>	An eight-byte character field containing the destination of the DRS log dataset. <b>Assembler name:</b> DRQSLDST <b>COBOL name:</b> DRS-DRQBS-LOG-DEST
<b>LOG FORM</b>	An eight-byte character field containing the form name of the DRS log dataset. <b>Assembler name:</b> DRQSLFRM <b>COBOL name:</b> DRS-DRQBS-LOG-FORM
<b>LOG WRITER</b>	An eight-byte character field containing the writer name of the DRS log dataset. <b>Assembler name:</b> DRQSLWTR <b>COBOL name:</b> DRS-DRQBS-LOG-WRITER
<b>LOG FCB</b>	A four-byte character field containing the FCB name of the DRS log dataset. <b>Assembler name:</b> DRQSLFCB <b>COBOL name:</b> DRS-DRQBS-LOG-FCB
<b>LOG UCS</b>	A four-byte character field containing the UCS name of the DRS log dataset. <b>Assembler name:</b> DRQSLUCS <b>COBOL name:</b> DRS-DRQBS-LOG-UCS
<b>SNAP OPTIONS</b>	A one-byte binary field specifying the snap options in effect in DRS. Bit 0 on indicates that the snap dataset should be “held”. <b>Assembler name:</b> DRQSSOPT <b>COBOL name:</b> DRS-DRQBS-SNAP-OPTS
<b>SNAP CLASS</b>	A one-byte character field containing the SYSOUT class of the DRS snap dataset. <b>Assembler name:</b> DRQSSCLS <b>COBOL name:</b> DRS-DRQBS-SNAP-CLASS
<b>SNAP DESTINATION</b>	An eight-byte character field containing the destination of the DRS snap dataset. <b>Assembler name:</b> DRQSSDST <b>COBOL name:</b> DRS-DRQBS-SNAP-DEST
<b>SNAP FORM</b>	An eight-byte character field containing the form name of the DRS snap dataset. <b>Assembler name:</b> DRQSSFRM <b>COBOL name:</b> DRS-DRQBS-SNAP-FORM
<b>SNAP WRITER</b>	An eight-byte character field containing the writer name of the DRS snap dataset. <b>Assembler name:</b> DRQSSWTR <b>COBOL name:</b> DRS-DRQBS-SNAP-WRITER
<b>SNAP FCB</b>	A four-byte character field containing the FCB name of the DRS snap dataset. <b>Assembler name:</b> DRQSSFCB <b>COBOL name:</b> DRS-DRQBS-SNAP-FCB

---

<b>SNAP UCS</b>	A four-byte character field containing the UCS name of the DRS snap dataset. <b>Assembler name:</b> DRQSSUCS <b>COBOL name:</b> DRS-DRQBS-SNAP-UCS
<b>TRACE TYPES</b>	A one-byte binary field containing the bit settings indicating what type of DRS events should be traced. <b>Assembler name:</b> DRQSTTYP <b>COBOL name:</b> DRS-DRQBS-TRACE-OPTS
<b>TRACE TABLE SIZE</b>	A two-byte binary field containing the number of 4K pages which should be allocated to the DRS trace table. <b>Assembler name:</b> DRQSTSIZ <b>COBOL name:</b> DRS-DRQBS-TRACE-TBL-SIZE
<b>GTF TRACE FID</b>	A two-byte binary field containing the format appendage id to be used in GTF trace entries created by DRS, when TRACE TYPES indicates that DRS trace entries should be recorded in the GTF trace file. <b>Assembler name:</b> DRQSTGTF <b>COBOL name:</b> DRS-DRQBS-TRACE-GTF-ID
<b>TRACE TABLE BEGINNING ADDRESS</b>	A four-byte binary field containing the address of the start of the DRS trace table. <b>Assembler name:</b> DRQSTBEG <b>COBOL name:</b> DRS-DRQBS-TRACE-TBL-START
<b>TRACE TABLE ENDING ADDRESS</b>	A four-byte binary field containing the address of the end of the DRS trace table. <b>Assembler name:</b> DRQSTEND <b>COBOL name:</b> DRS-DRQBS-TRACE-TBL-END
<b>TRACE TABLE CURRENT ADDRESS</b>	A four-byte binary field containing the address of the current entry in the DRS trace table. <b>Assembler name:</b> DRQSTCUR <b>COBOL name:</b> DRS-DRQBS-TRACE-TBL-CURR

---

For a general output statement query, the fields in each occurrence of the extension are:

<b>OUTPUT STATEMENT NAME</b>	An 8-byte character field containing the name assigned to the output statements. <b>Assembler name:</b> DRQOONAM <b>COBOL name:</b> DRS-DRQBO-OUTPUT-STMT-NAME
<b>CREATE DATE</b>	A 4-byte packed field in “0CYYDDDS” format representing the date that the output statement was created. <b>Assembler name:</b> DRQOOC DT <b>COBOL name:</b> DRS-DRQBO-CREATE-DATE
<b>CREATE TIME</b>	A 4-byte binary field in “HHMMDDTH” format representing the time that the output statement was created. <b>Assembler name:</b> DRQOOCTM <b>COBOL name:</b> DRS-DRQBO-CREATE-TIME

For a detail output statement query, the fields in the extension are:

<b>OUTPUT STATEMENT NAME</b>	An 8-byte character field containing the name assigned to the output statements. <b>Assembler name:</b> DRQONAME <b>COBOL name:</b> DRS-DRQBL-OUTPUT-STMT-NAME
<b>CREATE DATE</b>	A 4-byte packed field in “0CYYDDDS” format representing the date that the output statement was created. <b>Assembler name:</b> DRQOCDAT <b>COBOL name:</b> DRS-DRQBL-CREATE-DATE
<b>CREATE TIME</b>	A 4-byte binary field in “HHMMDDTH” format representing the time that the output statement was created. <b>Assembler name:</b> DRQOCTIM <b>COBOL name:</b> DRS-DRQBL-CREATE-TIME
<b>ADDRESS</b>	Four 60-byte fields specifying the address to be printed on output separator pages. <b>Assembler name:</b> DRQOADR1, DRQOADR2, DRQOADR3, and DRQOADR4 <b>COBOL name:</b> DRS-DRQBL-ADDR1, DRS-DRQBL-ADDR2, DRS-DRQBL-ADDR3, and DRS-DRQBL-ADDR4
<b>BUILDING</b>	A 60-byte field specifying the building to be printed on output separator pages. <b>Assembler name:</b> DRQOBLDG <b>COBOL name:</b> DRS-DRQBL-BUILDING
<b>BURST</b>	A 1-byte field which directs output to a stacker on a 3800 Printing Subsystem. “Y” = burster_trimmer-stacker; “N” = continuous forms stacker. <b>Assembler name:</b> DRQOBURS <b>COBOL name:</b> DRS-DRQBL-BURST

---

<b>CHARS</b>	<p>Four 4-byte fields specifying the names of character arrangement tables to be used when printing on a 3800.</p> <p><b>Assembler name:</b> DRQOCHR1, DRQOCHR2, DRQOCHR3, and DRQOCHR4</p> <p><b>COBOL name:</b> DRS-DRQBL-CHAR1, DRS-DRQBL-CHAR2, DRS-DRQBL-CHAR3, and DRS-DRQBL-CHAR4</p>
<b>CKPTLINE</b>	<p>A 2-byte binary field specifying the checkpoint value in lines.</p> <p><b>Assembler name:</b> DRQOCKLN</p> <p><b>COBOL name:</b> DRS-DRQBL-CHECKPOINT-LINES</p>
<b>CKPTPAGE</b>	<p>A 2-byte binary field specifying the checkpoint value in pages.</p> <p><b>Assembler name:</b> DRQOCKPG</p> <p><b>COBOL name:</b> DRS-DRQBL-CHECKPOINT-PAGES</p>
<b>CKPTSEC</b>	<p>A 2-byte binary field specifying the checkpoint values in seconds.</p> <p><b>Assembler name:</b> DRQOCKSC</p> <p><b>COBOL name:</b> DRS-DRQBL-CHECKPOINT-SECONDS</p>
<b>CLASS</b>	<p>A 1-byte field specifying the SYSOUT class.</p> <p><b>Assembler name:</b> DRQOCLAS</p> <p><b>COBOL name:</b> DRS-DRQBL-CLASS</p>
<b>COLORMAP</b>	<p>An 8-byte field specifying the AFP resource for the print file which contains color translation information.</p> <p><b>Assembler name:</b> DRQOXCMP</p> <p><b>COBOL name:</b> DRS-DRQBL-COLOR-MAP</p>
<b>COMPACT</b>	<p>An 8-byte field specifying the name of a compaction table for JES to use when sending the SYSOUT dataset to an SNA remote terminal.</p> <p><b>Assembler name:</b> DRQOCMPT</p> <p><b>COBOL name:</b> DRS-DRQBL-COMPACTION-TABLE</p>
<b>COMSETUP</b>	<p>An 8-byte field specifying the AFP resource for the print file which contains setup information.</p> <p><b>Assembler name:</b> DRQOXCST</p> <p><b>COBOL name:</b> DRS-DRQBL-COM-SETUP</p>
<b>CONTROL</b>	<p>An 8-byte field specifying the type of spacing to be applied to the SYSOUT dataset. Valid values consist of "SINGLE", "DOUBLE", "TRIPLE", and "PROGRAM".</p> <p><b>Assembler name:</b> DRQOCNTL</p> <p><b>COBOL name:</b> DRS-DRQBL-CONTROL</p>
<b>COPIES</b>	<p>An 2-byte binary field specifying the number of copies to be printed. Valid values range from 1 to 255.</p> <p><b>Assembler name:</b> DRQOCOPY</p> <p><b>COBOL name:</b> DRS-DRQBL-COPIES</p>
<b>COPY GROUP(S)</b>	<p>Eight 1-byte binary fields specifying copy groups.</p> <p><b>Assembler name:</b> DRQOCPY1 through DRQOCPY8</p> <p><b>COBOL name:</b> DRS-DRQBL-CPYG1 through DRS-DRQBL-CPYG8</p>

---

---

<b>DATAACK</b>	<p>An 8-byte field specifying whether or not print-positioning and invalid-character data-check errors are to be blocked or unblocked for printers accessed through the functional subsystem Print Services Facility (PSF). Valid values are "BLOCK", "UNBLOCK", "BLKCHAR", and "BLKPOS".</p> <p><b>Assembler name:</b> DRQODATC  <b>COBOL name:</b> DRS-DRQBL-DATAACK</p>
<b>DEFAULT</b>	<p>A 1-byte field specifying whether or not the OUTPUT statement can be implicitly referenced by SYSOUT DD statements in the same job.</p> <p><b>Assembler name:</b> DRQODFLT  <b>COBOL name:</b> DRS-DRQBL-DEFLT</p>
<b>DEPT</b>	<p>A 60-byte field specifying the department to be printed on output separator pages.</p> <p><b>Assembler name:</b> DRQODEPT  <b>COBOL name:</b> DRS-DRQBL-DEPARTMENT</p>
<b>DEST</b>	<p>A 17-byte field specifying the destination for the SYSOUT dataset.</p> <p><b>Assembler name:</b> DRQODEST  <b>COBOL name:</b> DRS-DRQBL-DESTINATION</p>
<b>DEST (LONG)</b>	<p>A 127-byte field specifying the destination for the SYSOUT dataset. The longer destination field is intended for a TCP/IP address in the format "IP:xxxxxxx" or "node.IP:xxxxxxx". This field and the 17-byte DEST field are mutually exclusive.</p> <p><b>Assembler name:</b> DRQOXDSL  <b>COBOL name:</b> DRS-DRQBL-DESTINATION-LONG</p>
<b>DPAGELBL</b>	<p>A 1-byte field specifying whether or not the system should print the security label on each page of printed output. Valid values are "Y" or "N".</p> <p><b>Assembler name:</b> DRQODLBL  <b>COBOL name:</b> DRS-DRQBL-DPAGELBL</p>
<b>DUPLEX</b>	<p>An 8-byte field specifying whether printing is to be done on both sides of the sheet. Valid values consist of "NO", "NORMAL" or "TUMBLE".</p> <p><b>Assembler name:</b> DRQOXDUP  <b>COBOL name:</b> DRS-DRQBL-DUPLEX</p>
<b>FCB</b>	<p>A 4-byte field specifying the FCB name.</p> <p><b>Assembler name:</b> DRQOFCB  <b>COBOL name:</b> DRS-DRQBL-FCB</p>
<b>FLASH</b>	<p>A 4-byte field specifying the forms overlay to be used in printing the SYSOUT dataset on a 3800 Printing Subsystem.</p> <p><b>Assembler name:</b> DRQOFLSH  <b>COBOL name:</b> DRS-DRQBL-FLASH</p>
<b>FLASH COUNT</b>	<p>A 2-byte binary field specifying the number of copies on which the forms overlay is to be printed. Valid values consist of 0 through 255.</p> <p><b>Assembler name:</b> DRQOFLCT  <b>COBOL name:</b> DRS-DRQBL-FLASH-COUNT</p>

---

---

<b>FORMDEF</b>	<p>A 6-byte field specifying the name of a FORMDEF member in a PSF library.</p> <p><b>Assembler name:</b> DRQOFMDF</p> <p><b>COBOL name:</b> DRS-DRQBL-FORMDEF</p>
<b>FORMLEN</b>	<p>A 10-byte field specifying the numeric length and unit type that will be used to change the physical paper length without reconfiguring the printer.</p> <p>Valid values are nn.nnnUU, where n is a digit 0-9, and UU represents one of the following units: IN (inches) or CM (centimeters).</p> <p><b>Assembler name:</b> DRQOXFLN</p> <p><b>COBOL name:</b> DRS-DRQBL-FORM-LENGTH</p>
<b>FORMS</b>	<p>An 8-byte field specifying the form name to be used for the dataset.</p> <p><b>Assembler name:</b> DRQOFORM</p> <p><b>COBOL name:</b> DRS-DRQBL-FORM</p>
<b>GROUPID</b>	<p>An 8-byte field specifying the name of an output group to which this dataset belongs.</p> <p><b>Assembler name:</b> DRQOGRID</p> <p><b>COBOL name:</b> DRS-DRQBL-GROUP-ID</p>
<b>INDEX</b>	<p>A 2-byte binary field specifying the left margin on a 3211 printer with the indexing feature. Valid values are 1 through 31.</p> <p><b>Assembler name:</b> DRQOINDEX</p> <p><b>COBOL name:</b> DRS-DRQBL-INDEX</p>
<b>INTRAY</b>	<p>A 4-byte binary field specifying the paper source when printing AFP files. Valid values are 1- 255.</p> <p><b>Assembler name:</b> DRQOXNTR</p> <p><b>COBOL name:</b> DRS-DRQBL-INTRAY</p>
<b>LINDEX</b>	<p>A 2-byte binary field specifying the right margin on a 3211 printer with the indexing feature. Valid values are 1 through 31.</p> <p><b>Assembler name:</b> DRQOLNDX</p> <p><b>COBOL name:</b> DRS-DRQBL-LINDEX</p>
<b>LINECT</b>	<p>A 2-byte binary field specifying the maximum number of lines JES2 is to printer on each output page. Valid values are 0 through 255.</p> <p><b>Assembler name:</b> DRQOLNCT</p> <p><b>COBOL name:</b> DRS-DRQBL-LINECT</p>
<b>MODIFY</b>	<p>A 4-byte field specifying a copy modification module name that tells JES how to print the SYSOUT dataset on a 3800 Printing Subsystem.</p> <p><b>Assembler name:</b> DRQOCMOD</p> <p><b>COBOL name:</b> DRS-DRQBL-COPY-MOD</p>
<b>MODIFY TRC</b>	<p>A 1-byte field specifying the table name in the CHARS parameter (0 for first, 1 for second, 2 for third, etc.).</p> <p><b>Assembler name:</b> DRQOCTRC</p> <p><b>COBOL name:</b> DRS-DRQBL-COPY-MOD-TRC</p>

---

---

<b>NAME</b>	<p>A 60-byte field specifying the preferred name to be printed on output separator pages.</p> <p><b>Assembler name:</b> DRQONAM</p> <p><b>COBOL name:</b> DRS-DRQBL-NAME</p>
<b>NOTIFY</b>	<p>Four 17-byte fields specifying the 1 to 4 users to be notified when a job completes.</p> <p><b>Assembler name:</b> DRQONTFY, DRQONTF2, DRQONTF3, DRQONTF4</p> <p><b>COBOL name:</b> DRS-DRQBL-NOTIFY, DRS-DRQBL-NOTIFY2, DRS-DRQBL-NOTIFY3, DRS-DRQBL-NOTIFY4</p>
<b>OFFSETXB</b>	<p>A 13-byte field specifying the offset in the x direction from the page origin for the back side of each page of output.</p> <p>Valid values are mmm.nnnUU, where m is a digit from 0 - 9, n is a digit from 0 - 9 and UU represents one of the following units: IN (inches), CM (centimeters), MM (millimeters), PELS or POINTS</p> <p><b>Assembler name:</b> DRQOXOXB</p> <p><b>COBOL name:</b> DRS-DRQBL-BACK-X-OFFSET</p>
<b>OFFSETXF</b>	<p>A 13-byte field specifying the offset in the x direction from the page origin for the front side of each page of output.</p> <p>Valid values are mmm.nnnUU, where m is a digit from 0 - 9, n is a digit from 0 - 9 and UU represents one of the following units: IN (inches), CM (centimeters), MM (millimeters), PELS or POINTS</p> <p><b>Assembler name:</b> DRQOXOXF</p> <p><b>COBOL name:</b> DRS-DRQBL-FRONT-X-OFFSET</p>
<b>OFFSETYB</b>	<p>A 13-byte field specifying the offset in the y direction from the page origin for the back side of each page of output.</p> <p>Valid values are mmm.nnnUU, where m is a digit from 0 - 9, n is a digit from 0 - 9 and UU represents one of the following units: IN (inches), CM (centimeters), MM (millimeters), PELS or POINTS</p> <p><b>Assembler name:</b> DRQOXOYB</p> <p><b>COBOL name:</b> DRS-DRQBL-BACK-Y-OFFSET</p>
<b>OFFSETYF</b>	<p>A 13-byte field specifying the offset in the y direction from the page origin for the front side of each page of output.</p> <p>Valid values are mmm.nnnUU, where m is a digit from 0 - 9, n is a digit from 0 - 9 and UU represents one of the following units: IN (inches), CM (centimeters), MM (millimeters), PELS or POINTS</p> <p><b>Assembler name:</b> DRQOXOYF</p> <p><b>COBOL name:</b> DRS-DRQBL-FRONT-Y-OFFSET</p>
<b>OUTBIN</b>	<p>A 4-byte binary field specifying the printer output bin identifier to be used for the SYSOUT data set.</p> <p><b>Assembler name:</b> DRQOBIN</p> <p><b>COBOL name:</b> DRS-DRQBL-OUTBIN</p>
<b>OUTDISP (NORMAL)</b>	<p>An 8-byte field specifying the output disposition when the job ends normally.</p> <p><b>Assembler name:</b> DRQOODSN</p> <p><b>COBOL name:</b> DRS-DRQBL-NORMAL-DISP</p>



---

<b>OUTDISP (ABNORMAL)</b>	An 8-byte field specifying the output disposition when the job ends abnormally. <b>Assembler name:</b> DRQOODSA <b>COBOL name:</b> DRS-DRQBL-ABNORMAL-DISP
<b>OVERLAYB</b>	An 8-byte field specifying that the named medium overlay is to be placed on the back side of each sheet to be printed. <b>Assembler name:</b> DRQOXOVB <b>COBOL name:</b> DRS-DRQBL-BACK-OVERLAY
<b>OVERLAYF</b>	An 8-byte field specifying that the named medium overlay is to be placed on the front side of each sheet to be printed. <b>Assembler name:</b> DRQOXOVF <b>COBOL name:</b> DRS-DRQBL-FRONT-OVERLAY
<b>OVFL</b>	An 8-byte field specifying whether or not JES3 should test for page overflow on an output printer. (JES3 only) Valid values are "ON" or "OFF". <b>Assembler name:</b> DRQOXOFL <b>COBOL name:</b> DRS-DRQBL-OVERFLOW
<b>PAGEDEF</b>	A 6-byte field specifying the name of a PAGEDEF member in a PSF library. <b>Assembler name:</b> DRQOPGDF <b>COBOL name:</b> DRS-DRQBL-PAGEDEF
<b>PIMSG</b>	A 1-byte field specifying whether or not a functional subsystem should print its messages in the output listing. Valid values are "Y" or "N". <b>Assembler name:</b> DRQOPMSG <b>COBOL name:</b> DRS-DRQBL-PIMSG
<b>PIMSG COUNT</b>	A 2-byte binary field specifying the number of errors to cause printing of PIMSG to be terminated. Valid values are 0 through 999. <b>Assembler name:</b> DRQOPMCT <b>COBOL name:</b> DRS-DRQBL-PIMSG-COUNT
<b>PORTNO</b>	A 4-byte binary field specifying the TCP/IP port number at which the printing application connects to the printer. Valid values are 1 - 65535. <b>Assembler name:</b> DRQOXPRT <b>COBOL name:</b> DRS-DRQBL-PORT
<b>PRMODE</b>	An 8-byte field specifying the process mode required to print the dataset. Valid values are "LINE", "PAGE", or any valid installation-defined process mode. <b>Assembler name:</b> DRQOPRMD <b>COBOL name:</b> DRS-DRQBL-PRMODE
<b>PRTERORR</b>	An 8-byte field specifying how a SYSOUT data set that has had printing terminated by a functional subsystem is to be released by JES. Valid values are "DEFAULT", "HOLD" or "QUIT". <b>Assembler name:</b> DRQOXPTE <b>COBOL name:</b> DRS-DRQBL-ERROR-DISPOSITION

---

---

<b>PRTOPTNS</b>	<p>An 16- byte field specifying the named entity that contains additional print options for an IP-destined data set that is being sent by a functional subsystem.</p> <p><b>Assembler name:</b> DRQOXPRO</p> <p><b>COBOL name:</b> DRS-DRQBL-PRINT-OPTIONS</p>
<b>PRTQUEUE</b>	<p>A 127-byte field specifying the print queue name used when printing the IP-destined data set.</p> <p><b>Assembler name:</b> DRQOXPRQ</p> <p><b>COBOL name:</b> DRS-DRQBL-PRINT-QUEUE</p>
<b>PRTY</b>	<p>A 2-byte binary field specifying the initial priority at which the SYSOUT dataset enters the output queue. Value values are 0 through 255.</p> <p><b>Assembler name:</b> DRQOPRTY</p> <p><b>COBOL name:</b> DRS-DRQBL-INITIAL-PRIORITY</p>
<b>RESFMT</b>	<p>An 8-byte field specifying the resolution used to format the print data set. Valid values are "P240" or "P300".</p> <p><b>Assembler name:</b> DRQOXRES</p> <p><b>COBOL name:</b> DRS-DRQBL-RESOLUTION</p>
<b>RETAINF</b>	<p>A 10-byte field specifying how long a functional subsystem will retain an IP-destined data set after a failed transmission.</p> <p><b>Assembler name:</b> DRQOXRTF</p> <p><b>COBOL name:</b> DRS-DRQBL-FAILED-RETAIN-TIME</p>
<b>RETAINS</b>	<p>A 10 byte field specifying how long a functional subsystem will retain an IP-destined data set after a successful transmission.</p> <p><b>Assembler name:</b> DRQOXRTS</p> <p><b>COBOL name:</b> DRS-DRQBL-SUCCESS-RETAIN-TIME</p>
<b>RETRYL</b>	<p>A 4-byte binary field specifying the number of attempts an FSS will try for transmission of an IP-destined data set.</p> <p>Valid values are 0 -32767.</p> <p><b>Assembler name:</b> DRQOXRTL</p> <p><b>COBOL name:</b> DRS-DRQBL-MAX-RETRIES</p>
<b>RETRYT</b>	<p>A 10-byte field specifying how much time a functional subsystem will wait between retries of transmission attempts of a data set.</p> <p><b>Assembler name:</b> DRQOXRTT</p> <p><b>COBOL name:</b> DRS-DRQBL-TIME-BETWEEN-RETRIES</p>
<b>ROOM</b>	<p>A 60-byte field specifying the room identification to be printed on output separator pages.</p> <p><b>Assembler name:</b> DRQOROOM</p> <p><b>COBOL name:</b> DRS-DRQBL-ROOM</p>
<b>SYSAREA</b>	<p>A 1-byte field specifying whether or not the system should reserve an area for the security label on each page of printed output. Valid values are "Y" or "N".</p> <p><b>Assembler name:</b> DRQOSARE</p> <p><b>COBOL name:</b> DRS-DRQBL-SYSAREA</p>

---

<b>TITLE</b>	A 60-byte field specifying the title to be printed on JES banner pages. <b>Assembler name:</b> DRQOTITL <b>COBOL name:</b> DRS-DRQBL-TITLE
<b>TRC</b>	A 1-byte field specifying whether or not the SYSOUT dataset should have a trc character in the second character of each record. <b>Assembler name:</b> DRQOTRC <b>COBOL name:</b> DRS-DRQBL-TRC
<b>UCS</b>	A 4-byte field specifying a universal character set, print train, or character-arrangement table for a 3800 Printing Subsystem. <b>Assembler name:</b> DRQOUCS <b>COBOL name:</b> DRS-DRQBL-UCS
<b>WRITER</b>	An 8-byte field specifying the writer name. <b>Assembler name:</b> DRQOWRTR <b>COBOL name:</b> DRS-DRQBL-WRITER
<b>OUTPUT DESCRIPTION</b>	A 60-byte character field that is NOT used when allocating the OUTPUT JCL statement. There is no OUTPUT JCL statement keyword that corresponds to this field. This field simply allows an installation to associate a description with the OUTPUT JCL statement. The DRS/VPI @OUTDESC keyword can be used for this purpose when defining dynamic OUTPUT JCL statements to DRS/VPI. <b>Assembler name:</b> DRQODESC <b>COBOL name:</b> DRS-DRQBL-OUTDESC

For an extended output statement, the following fields will also appear in the extension:

<b>USERLIB(S)</b>	Eight 44-byte fields containing the name(s) assigned to the output statements. <b>Assembler name:</b> DRQOXUL1 through DRQOXUL8 <b>COBOL name:</b> DRS-DRQBL-USERLIB-1 through DRS-DRQBL-USERLIB-8
<b>USER DATA FIELD(S)</b>	Sixteen 60-byte fields containing USERDATA values. <b>Assembler name:</b> DRQOXD01 through DRQOXD16 <b>COBOL name:</b> DRS-DRQBL-USERDATA-1 through DRS-DRQBL-USERDATA-16

---

The macro “DRSDRQB” in file LRS.DRS.V1R33.MACLIB is supplied to generate a DSECT of the QUERY BLOCK.

The copy member “DRSDRQBG” in file LRS.DRS.V1R33.MACLIB is supplied to generate a COBOL description of the QUERY BLOCK to be used for general queries.

Member “DRSDRQBD” generates a COBOL description of the QUERY BLOCK to be used for detail queries.

Member “DRSDRQBS” generates a COBOL description of the QUERY BLOCK to be used for system queries.

Member “DRSDRQBO” generates a COBOL description of the QUERY BLOCK to be used for general output statement queries.

Member “DRSDRQBL” generates a COBOL description of the QUERY BLOCK to be used for detail output statement queries.

---

## Command Block (DRCB)

The COMMAND block can be used to issue commands to DRS via the CMND call.

The individual fields which make up the COMMAND BLOCK are:

<b>'DRCB'</b>	The control block identifier. <b>Assembler name:</b> DRCID <b>COBOL name:</b> DRS-DRCB-ID
<b>COMMAND VERB</b>	An eight-byte character field which must contain the name of the command to be issued to DRS. Valid values are "CLOSELOG", "QUIESCE", "RESTART", "SNAP", and "SSET".  CLOSELOG indicates that the current DRS log file should be closed so it is available for printing. DRS will immediately allocate a new log file and will begin recording entries in the new log.  QUIESCE indicates that DRS should stop accepting INIT calls. DRS will still accept PUT and TERM calls for active reports.  RESTART indicates that DRS should start accepting INIT calls. This means that new reports can become active to DRS.  SNAP indicates that a formatted dump of DRS control blocks should be created, based on the SNAP parameters (class, destination, form, etc.) specified.  SSET indicates that system log or trace options are to be changed. This option requires that one or both of the following fields be specified. <b>Assembler name:</b> DRCCMND <b>COBOL name:</b> DRS-DRCB-VERB
<b>NEW LOG OPTION</b>	For an SSET command, this one-byte character field contains a "Y" to enable DRS logging and an "N" to disable DRS logging. <b>Assembler name:</b> DRCLOG <b>COBOL name:</b> DRS-DRCB-LOG-OPTION
<b>NEW TRACE TYPES</b>	For an SSET command, this two-byte character field contains the trace options to be specified to DRS. This should be an EBCDIC representation of the bit setting. For example, if the trace options are to be set to a value of X'30', this two-byte field should contain the characters 30 (X'F3F0'). <b>Assembler name:</b> DRCTRYP <b>COBOL name:</b> DRS-DRCB-TRACE-TYPES

The macro "DRSDRCB" in file LRS.DRS.V1R33.MACLIB is supplied to generate a DSECT of the COMMAND BLOCK. The copy member "DRSDRCBC" in file LRS.DRS.V1R33.MACLIB is supplied to generate a COBOL description of the COMMAND BLOCK.

---

## Output Block (DROB)

The OUTPUT block can be used to dynamically add or delete OUTPUT JCL statements in the DRS address space, under MVS/ESA JES2.

The individual fields which make up the OUTPUT BLOCK are:

<b>'DROB'</b>	The control block identifier. <b>Assembler name:</b> DROID <b>COBOL name:</b> DRS-DROB-ID
<b>OUTPUT ID</b>	An eight-byte character field which can contain the name to be assigned to the OUTPUT JCL statement to be defined, when adding an OUTPUT JCL statement. If this field is blank, a name will be assigned. When deleting an existing dynamically defined OUTPUT JCL statement, this field must be supplied. <b>Assembler name:</b> DROOUTID <b>COBOL name:</b> DRS-DROB-OUTPUT-ID
<b>FUNCTION</b>	A three-byte character field indicating whether the OUTPUT JCL statement is to be defined ("ADD") or deleted ("DEL"). <b>Assembler name:</b> DROFUNC <b>COBOL name:</b> DRS-DROB-FUNCTION
<b>EXTENDED ATTR INDICATOR</b>	A one-byte character field indicating whether the "extended" DROB fields will be present in the DROB. Valid values are "Y" or "N". <b>Assembler name:</b> DROXATTR <b>COBOL name:</b> DRS-DROB-XATTR
<b>RETURN CODE</b>	A four-byte binary field, filled in by DRS, specifying the results of the call. A non-zero value indicates that a failure occurred. <b>Assembler name:</b> DRORC <b>COBOL name:</b> DRS-DROB-RETURN-CODE
<b>REASON CODE</b>	A four-byte binary field, filled in by DRS, specifying the particular reason that a call failed. A non-zero value indicates that a failure occurred. <b>Assembler name:</b> DROREAS <b>COBOL name:</b> DRS-DROB-REASON-CODE
<b>TEXT UNIT KEY</b>	A four-byte binary field, filled in by DRS, specifying the text unit key of the text unit that caused the call to fail. <b>Assembler name:</b> DROTUKEY <b>COBOL name:</b> DRS-DROB-TEXT-UNIT-KEY

---

**ZERO-VALUE FLAGS** A four-byte hex field containing flags that indicate which DROB fields contain a value of zero. Note that you must set this flag if a zero-value is specified.

For example, if LINECT=0 is required, then set DROZFLG1=X'20' and set DROLINCT=0.

**Assembler name:** DROZFLGS  
DROZFLG1  
DROZ1CKL X'80' CHECKPOINT  
LINES = 0  
DROZ1FLC X'20' FLASH COUNT = 0  
DROZ1LCT X'10' LINECT = 0  
DROZ1PIM X'08' PIMSG COUNT = 0  
DROZ1PRI X'04' PRIORITY = 0  
DROZFLG2  
DROZFLG3  
DROZFLG4

**COBOL name:** DRS-DROB-ZFLAGS  
DRS-DROB-ZFLAG-1  
DRS-DROB-ZFLAG-2  
DRS-DROB-ZFLAG-3  
DRS-DROB-ZFLAG-4

**ADDRESS** Four 60-byte fields specifying the address to be printed on output separator pages.

**Assembler name:** DROADDR1, DROADDR2,  
DROADDR3, and DROADDR4

**COBOL name:** DRS-DROB-ADDR1,  
DRS-DROB-ADDR2,  
DRS-DROB-ADDR3, and  
DRS-DROB-ADDR4

**BUILDING** A 60-byte field specifying the building to be printed on output separator pages.

**Assembler name:** DROBLDG

**COBOL name:** DRS-DROB-BUILDING

**BURST** A 1-byte field which directs output to a stacker on a 3800 Printing Subsystem. "Y" = burster-trimmer-stacker; "N" = continuous forms stacker.

**Assembler name:** DROBURST

**COBOL name:** DRS-DROB-BURST

**CHARS** Four 4-byte fields specifying the names of character arrangement tables to be used when printing on a 3800.

**Assembler name:** DROCHAR1, DROCHAR2,  
DROCHAR3, and DROCHAR4

**COBOL name:** DRS-DROB-CHAR1,  
DRS-DROB-CHAR2,  
DRS-DROB-CHAR3, and  
DRS-DROB-CHAR4

---

<b>CKPTLINE</b>	A 2-byte binary field specifying the checkpoint value in lines. <b>Assembler name:</b> DROCKLIN <b>COBOL name:</b> DRS-DROB-CHECKPOINT-LINES
<b>CKPTPAGE</b>	A 2-byte binary field specifying the checkpoint value in pages. <b>Assembler name:</b> DROCKPAG <b>COBOL name:</b> DRS-DROB-CHECKPOINT-PAGES
<b>CKPTSEC</b>	A 2-byte binary field specifying the checkpoint value in seconds. <b>Assembler name:</b> DROCKSEC <b>COBOL name:</b> DRS-DROB-CHECKPOINT-SECOND S
<b>CLASS</b>	A 1-byte field specifying the SYSOUT class. <b>Assembler name:</b> DROCLASS <b>COBOL name:</b> DRS-DROB-CLASS
<b>COLORMAP</b>	An 8-byte field specifying the AFP resource for the print file which contains color translation information. <b>Assembler name:</b> DROXCMAP <b>COBOL name:</b> DRS-DROX-COLOR-MAP
<b>COMPACT</b>	An 8-byte field specifying the name of a compaction table for JES to use when sending the SYSOUT dataset to an SNA remote terminal. <b>Assembler name:</b> DROCOMPT <b>COBOL name:</b> DRS-DROB-COMPACTION-TABLE
<b>COMSETUP</b>	An 8-byte field specifying the AFP resource for the print file which contains setup information. <b>Assembler name:</b> DROXCSET <b>COBOL name:</b> DRS-DROX-COM-SETUP
<b>CONTROL</b>	An 8-byte field specifying the type of spacing to be applied to the SYSOUT dataset. Valid values consist of "SINGLE", "DOUBLE", "TRIPLE", and "PROGRAM". <b>Assembler name:</b> DROCNTL <b>COBOL name:</b> DRS-DROB-CONTROL
<b>COPIES</b>	An 2-byte binary field specifying the number of copies to be printed. Valid values range from 1 to 255. <b>Assembler name:</b> DROCOPY <b>COBOL name:</b> DRS-DROB-COPIES
<b>COPY GROUP(S)</b>	Eight 1-byte binary fields specifying copy groups. <b>Assembler name:</b> DROCPYG1 through DROCPYG8 <b>COBOL name:</b> DRS-DROB-CPYG1 through DRS-DROB-CPYG8
<b>DATAACK</b>	An 8-byte field specifying whether or not print-positioning and invalid-character data-check errors are to be blocked or unblocked for printers accessed through the functional subsystem Print Services Facility (PSF). Valid values are "BLOCK", "UNBLOCK", "BLKCHAR", and "BLKPOS". <b>Assembler name:</b> DRODATCK <b>COBOL name:</b> DRS-DROB-DATAACK

---



---

<b>DEFAULT</b>	<p>A 1-byte field specifying whether or not the OUTPUT statement can be implicitly referenced by SYSOUT DD statements in the same job. Valid values are “Y” or “N”.</p> <p><b>Assembler name:</b> DRODEFLT</p> <p><b>COBOL name:</b> DRS-DROB-DEFLT</p>
<b>DEPT</b>	<p>A 60-byte field specifying the department to be printed on output separator pages.</p> <p><b>Assembler name:</b> DRODEPT</p> <p><b>COBOL name:</b> DRS-DROB-DEPARTMENT</p>
<b>DEST</b>	<p>A 17-byte field specifying the destination for the SYSOUT dataset.</p> <p><b>Assembler name:</b> DRODEST</p> <p><b>COBOL name:</b> DRS-DROB-DESTINATION</p>
<b>DEST (LONG)</b>	<p>A 127-byte field specifying the destination for the SYSOUT dataset. The longer destination field is intended for a TCP/IP address in the format “IP:xxxxxxx” or “node.IP:xxxxxxx”. This field and the 17-byte DEST field are mutually exclusive.</p> <p><b>Assembler name:</b> DROXDSTL</p> <p><b>COBOL name:</b> DRS-DROX-DESTINATION-LONG</p>
<b>DPAGELBL</b>	<p>A 1-byte field specifying whether or not the system should print the security label on each page of printed output. Valid values are “Y” or “N”.</p> <p><b>Assembler name:</b> DRODPLBL</p> <p><b>COBOL name:</b> DRS-DROB-DPAGELBL</p>
<b>DUPLEX</b>	<p>An 8-byte field specifying whether printing is to be done on both sides of the sheet. Valid values consist of “NO”, “NORMAL” or “TUMBLE”.</p> <p><b>Assembler name:</b> DROXDUPX</p> <p><b>COBOL name:</b> DRS-DROX-DUPLEX</p>
<b>FCB</b>	<p>A 4-byte field specifying the FCB name.</p> <p><b>Assembler name:</b> DROFCB</p> <p><b>COBOL name:</b> DRS-DROB-FCB</p>
<b>FLASH</b>	<p>A 4-byte field specifying the forms overlay to be used in printing the SYSOUT dataset on a 3800 Printing Subsystem.</p> <p><b>Assembler name:</b> DROFLASH</p> <p><b>COBOL name:</b> DRS-DROB-FLASH</p>
<b>FLASH COUNT</b>	<p>A 2-byte binary field specifying the number of copies on which the forms overlay is to be printed. Valid values consist of 0 through 255.</p> <p><b>Assembler name:</b> DROFLSCT</p> <p><b>COBOL name:</b> DRS-DROB-FLASH-COUNT</p>
<b>FORMDEF</b>	<p>A 6-byte field specifying the name of a FORMDEF member in a PSF library.</p> <p><b>Assembler name:</b> DROFMDEF</p> <p><b>COBOL name:</b> DRS-DROB-FORMDEF</p>

---

---

<b>FORMLEN</b>	<p>A 10-byte field specifying the numeric length and unit type that will be used to change the physical paper length without reconfiguring the printer.</p> <p>Valid values are nn.nnnUU, where n is a digit 0-9, and UU represents one of the following units:</p> <p>IN (inches) or CM (centimeters).</p> <p><b>Assembler name:</b> DROXFMLN <b>COBOL name:</b> DRS-DROX-FORM-LENGTH</p>
<b>FORMS</b>	<p>An 8-byte field specifying the form name to be used for the dataset.</p> <p><b>Assembler name:</b> DROFORM <b>COBOL name:</b> DRS-DROB-FORM</p>
<b>GROUPID</b>	<p>An 8-byte field specifying the name of an output group to which this dataset belongs.</p> <p><b>Assembler name:</b> DROGRPID <b>COBOL name:</b> DRS-DROB-GROUP-ID</p>
<b>INDEX</b>	<p>A 2-byte binary field specifying the left margin on a 3211 printer with the indexing feature. Valid values are 1 through 31.</p> <p><b>Assembler name:</b> DROINDEX <b>COBOL name:</b> DRS-DROB-INDEX</p>
<b>INTRAY</b>	<p>A 4-byte binary field specifying the paper source when printing AFP files. Valid values are 1- 255.</p> <p><b>Assembler name:</b> DROXNTRA <b>COBOL name:</b> DRS-DROX-INTRAY</p>
<b>LINDEX</b>	<p>A 2-byte binary field specifying the right margin on a 3211 printer with the indexing feature. Valid values are 1 through 31.</p> <p><b>Assembler name:</b> DROLINDEX <b>COBOL name:</b> DRS-DROB-LINDEX</p>
<b>LINECT</b>	<p>A 2-byte binary field specifying the maximum number of lines JES2 is to printer on each output page. Valid values are 0 through 255.</p> <p><b>Assembler name:</b> DROLINCT <b>COBOL name:</b> DRS-DROB-LINECT</p>
<b>MODIFY</b>	<p>A 4-byte field specifying a copy modification module name that tells JES how to print the SYSOUT dataset on a 3800 Printing Subsystem.</p> <p><b>Assembler name:</b> DROCMOD <b>COBOL name:</b> DRS-DROB-COPY-MOD</p>
<b>MODIFY TRC</b>	<p>A 1-byte field specifying the table name in the CHARS parameter (0 for first, 1 for second, 2 for third, etc.).</p> <p><b>Assembler name:</b> DROCMTRC <b>COBOL name:</b> DRS-DROB-COPY-MOD-TRC</p>
<b>NAME</b>	<p>A 60-byte field specifying the preferred name to be printed on output separator pages.</p> <p><b>Assembler name:</b> DRONAME <b>COBOL name:</b> DRS-DROB-NAME</p>

---

<b>NOTIFY</b>	<p>Four 17-byte fields specifying the 1 to 4 users to be notified when a job completes.</p> <p><b>Assembler name:</b> DRONOTFY, DRONOTF2, DRONOTF3, DRONOTF4</p> <p><b>COBOL name:</b> DRS-DROX-NOTIFY, DRS-DROX-NOTIFY2, DRS-DROX-NOTIFY3, DRS-DROX-NOTIFY4</p>
<b>OFFSETXB</b>	<p>A 13-byte field specifying the offset in the x direction from the page origin for the back side of each page of output.</p> <p>Valid values are mmm.nnnUU, where m is a digit from 0 - 9, n is a digit from 0 - 9 and UU represents one of the following units:</p> <p>IN (inches)</p> <p>CM (centimeters)</p> <p>MM (millimeters)</p> <p>PELS or POINTS</p> <p><b>Assembler name:</b> DROXOFXB</p> <p><b>COBOL name:</b> DRS-DROX-BACK-X-OFFSET</p>
<b>OFFSETXF</b>	<p>A 13-byte field specifying the offset in the x direction from the page origin for the front side of each page of output.</p> <p>Valid values are mmm.nnnUU, where m is a digit from 0 - 9, n is a digit from 0 - 9 and UU represents one of the following units:</p> <p>IN (inches)</p> <p>CM (centimeters)</p> <p>MM (millimeters)</p> <p>PELS or POINTS</p> <p><b>Assembler name:</b> DROXOFXF</p> <p><b>COBOL name:</b> DRS-DROX-FRONT-X-OFFSET</p>
<b>OFFSETYB</b>	<p>A 13-byte field specifying the offset in the y direction from the page origin for the back side of each page of output. Valid values are mmm.nnnUU, where m is a digit from 0 - 9, n is a digit from 0 - 9 and UU represents one of the following units: IN (inches), CM (centimeters), MM (millimeters), PELS or POINTS</p> <p><b>Assembler name:</b> DROXOFYB</p> <p><b>COBOL name:</b> DRS-DROX-BACK-Y-OFFSET</p>
<b>OFFSETYF</b>	<p>A 13-byte field specifying the offset in the y direction from the page origin for the front side of each page of output.</p> <p>Valid values are mmm.nnnUU, where m is a digit from 0 - 9, n is a digit from 0 - 9 and UU represents one of the following units:</p> <p>IN (inches)</p> <p>CM (centimeters)</p> <p>MM (millimeters)</p> <p>PELS or POINTS</p> <p><b>Assembler name:</b> DROXOFYF</p> <p><b>COBOL name:</b> DRS-DROX-FRONT-Y-OFFSET</p>

---

---

<b>OUTBIN</b>	<p>A 4-byte binary field specifying the printer output bin identifier to be used for the SYSOUT dataset.</p> <p><b>Assembler name:</b> DROOBIN</p> <p><b>COBOL name:</b> DRS-DROB-OUTBIN</p>
<b>OUTDISP (NORMAL)</b>	<p>An 8-byte field specifying the output disposition when the job ends normally.</p> <p><b>Assembler name:</b> DROODSPN</p> <p><b>COBOL name:</b> DRS-DROB-NORMAL-DISP</p>
<b>OUTDISP (ABNORMAL)</b>	<p>An 8-byte field specifying the output disposition when the job ends abnormally.</p> <p><b>Assembler name:</b> DROODSPA</p> <p><b>COBOL name:</b> DRS-DROB-ABNORMAL-DISP</p>
<b>OVERLAYB</b>	<p>An 8-byte field specifying that the named medium overlay is to be placed on the back side of each sheet to be printed.</p> <p><b>Assembler name:</b> DROXOVLB</p> <p><b>COBOL name:</b> DRS-DROX-BACK-OVERLAY</p>
<b>OVERLAYF</b>	<p>An 8-byte field specifying that the named medium overlay is to be placed on the front side of each sheet to be printed.</p> <p><b>Assembler name:</b> DROXOVLF</p> <p><b>COBOL name:</b> DRS-DROX-FRONT-OVERLAY</p>
<b>OVFL</b>	<p>An 8-byte field specifying whether or not JES3 should test for page overflow on an output printer. (JES3 only)</p> <p>Valid values are "ON" or "OFF".</p> <p><b>Assembler name:</b> DROXOVFL</p> <p><b>COBOL name:</b> DRS-DROX-OVERFLOW</p>
<b>PAGEDEF</b>	<p>A 6-byte field specifying the name of a PAGEDEF member in a PSF library.</p> <p><b>Assembler name:</b> DROPGDEF</p> <p><b>COBOL name:</b> DRS-DROB-PAGEDEF</p>
<b>PIMSG</b>	<p>A 1-byte field specifying whether or not a functional subsystem should print its messages in the output listing. Valid values are "Y" or "N".</p> <p><b>Assembler name:</b> DROPIMSG</p> <p><b>COBOL name:</b> DRS-DROB-PIMSG</p>
<b>PIMSG COUNT</b>	<p>A 2-byte binary field specifying the number of errors to cause printing of PIMSG to be terminated. Valid values are 0 through 999.</p> <p><b>Assembler name:</b> DROPIMCT</p> <p><b>COBOL name:</b> DRS-DROB-PIMSG-COUNT</p>
<b>PORTNO</b>	<p>A 4-byte binary field specifying the TCP/IP port number at which the printing application connects to the printer.</p> <p>Valid values are 1 - 65535.</p> <p><b>Assembler name:</b> DROXPORT</p> <p><b>COBOL name:</b> DRS-DROX-PORT</p>

---

<b>PRMODE</b>	An 8-byte field specifying the process mode required to print the dataset. Valid values are "LINE", "PAGE", or any valid installation-defined process mode. <b>Assembler name:</b> DROPRMOD <b>COBOL name:</b> DRS-DROB-PRMODE
<b>PRERROR</b>	An 8-byte field specifying how a SYSOUT data set that has had printing terminated by a functional subsystem is to be released by JES. Valid values are "DEFAULT", "HOLD" or "QUIT". <b>Assembler name:</b> DROXPTER <b>COBOL name:</b> DRS-DROX-ERROR-DISPOSITION
<b>PRTOPTNS</b>	An 16- byte field specifying the named entity that contains additional print options for an IP-destined data set that is being sent by a functional subsystem. <b>Assembler name:</b> DROXPRT0 <b>COBOL name:</b> DRS-DROX-PRINT-OPTIONS
<b>PRTQUEUE</b>	A 127-byte field specifying the print queue name used when printing the IP-destined data set. <b>Assembler name:</b> DROXPRTQ <b>COBOL name:</b> DRS-DROX-PRINT-QUEUE
<b>PRTY</b>	A 2-byte binary field specifying the initial priority at which the SYSOUT dataset enters the output queue. Value values are 0 through 255. <b>Assembler name:</b> DROPRTY <b>COBOL name:</b> DRS-DROB-INITIAL-PRIORITY
<b>RESFMT</b>	An 8-byte field specifying the resolution used to format the print data set. Valid values are "P240" or "P300". <b>Assembler name:</b> DROXRFMT <b>COBOL name:</b> DRS-DROX-RESOLUTION
<b>RETAINF</b>	A 10-byte field specifying how long a functional subsystem will retain an IP-destined data set after a failed transmission. <b>Assembler name:</b> DROXRETF <b>COBOL name:</b> DRS-DROX-FAILED-RETAIN-TIME
<b>RETAINS</b>	A 10 byte field specifying how long a functional subsystem will retain an IP-destined data set after a successful transmission. <b>Assembler name:</b> DROXRETS <b>COBOL name:</b> DRS-DROX-SUCCESS-RETAIN-TIME
<b>RETRYL</b>	A 4-byte binary field specifying the number of attempts an FSS will try for transmission of an IP-destined data set. Valid values are 0 -32767. <b>Assembler name:</b> DROXRETL <b>COBOL name:</b> DRS-DROX-MAX-RETRIES

---

---

<b>RETRYT</b>	<p>A 10-byte field specifying how much time a functional subsystem will wait between retries of transmission attempts of a data set.</p> <p><b>Assembler name:</b> DROXRETT</p> <p><b>COBOL name:</b> DRS-DROX-TIME-BETWEEN-RETRIES</p>
<b>ROOM</b>	<p>A 60-byte field specifying the room identification to be printed on output separator pages.</p> <p><b>Assembler name:</b> DROROOM</p> <p><b>COBOL name:</b> DRS-DROB-ROOM</p>
<b>SYSAREA</b>	<p>A 1-byte field specifying whether or not the system should reserve an area for the security label on each page of printed output. Valid values are "Y" or "N".</p> <p><b>Assembler name:</b> DROSAREA</p> <p><b>COBOL name:</b> DRS-DROB-SYSAREA</p>
<b>TITLE</b>	<p>A 60-byte field specifying the title to be printed on JES banner pages.</p> <p><b>Assembler name:</b> DROTITLE</p> <p><b>COBOL name:</b> DRS-DROB-TITLE</p>
<b>TRC</b>	<p>A 1-byte field specifying whether or not the SYSOUT dataset should have a trc character in the second character of each record.</p> <p><b>Assembler name:</b> DROTRC</p> <p><b>COBOL name:</b> DRS-DROB-TRC</p>
<b>UCS</b>	<p>A 4-byte field specifying a universal character set, print train, or character-arrangement table for a 3800 Printing Subsystem.</p> <p><b>Assembler name:</b> DROUCS</p> <p><b>COBOL name:</b> DRS-DROB-UCS</p>
<b>WRITER</b>	<p>An 8-byte field specifying the writer name.</p> <p><b>Assembler name:</b> DROWRTR</p> <p><b>COBOL name:</b> DRS-DROB-WRITER</p>
<b>OUTPUT DESCRIPTION</b>	<p>A 60-byte character field that is NOT used when allocating the OUTPUT JCL statement. There is no OUTPUT JCL statement keyword that corresponds to this field. This field simply allows an installation to associate a description with the OUTPUT JCL statement. The DRS/VPI @OUTDESC keyword can be used for this purpose when defining dynamic OUTPUT JCL statements to DRS/VPI.</p> <p><b>Assembler name:</b> DROODESC</p> <p><b>COBOL name:</b> DRS-DROB-OUTPUT-DESCRIPTION</p>

---

The following fields are present in the “extended” version of the DROB.

<b>USERLIB(S)</b>	Eight 44-byte character fields containing the dataset name of a USERLIB. <b>Assembler name:</b> DROXULB1 through DROXULB8 <b>COBOL name:</b> DRS-DROBX-USERLIB-1 through DRS-DROBX-USERLIB-8
<b>USER DATA FIELD(S)</b>	Sixteen 60-byte character fields containing USERDATA values. <b>Assembler name:</b> DROXUD01 through DROXUD16 <b>COBOL name:</b> DRS-DROBX-USERDATA-1 through DRS-DROBX-USERDATA-16

The macro “DRSDROB” in file LRS.DRS.V1R33.MACLIB is supplied to generate a DSEC of the OUTPUT BLOCK. The copy member “DRSDROBC” in file LRS.DRS.V1R33.MACLIB is supplied to generate a COBOL description of the OUTPUT BLOCK.





---

# Section 22

## DRS/API User Exits

### Introduction

Release 3.2 of DRS has seven optional user exits that can be used to customize DRS operation. None of these exits is required. Sample source code for all user exits is supplied to you in the DRS source library (file LRS.DRS.V1R33.ASM) on the distribution cartridge.

DRS loads all available user exits during DRS initialization. Therefore, user exits must be linked into the appropriate JOBLIB, STEPLIB, or LINKLIST library prior to starting the DRS application.

Sample JCL to assemble and link an exit is included on the DRS distribution cartridge as member EXITASML in file LRS.DRS.V1R33.CNTL. That JCL is reproduced here:

```
//JOBNAME JOB (YOUR JOB CARD INFORMATION)
//ASMH EXEC PGM=IEV90,PARM='OBJECT,NODECK,RENT,XREF(SHORT)'
//SYSLIB DD DSN=SYS1.MACLIB,DISP=SHR
// DD DSN=SYS1.AMODGEN,DISP=SHR
// DD DSN=LRS.DRS.V1R33.MACLIB,DISP=SHR
// DD DSN=LRS.DRS.V1R33.ASM,DISP=SHR
//SYSUT1 DD UNIT=VIO,SPACE=(CYL,(5,2))
//SYSUT2 DD UNIT=VIO,SPACE=(CYL,(5,2))
//SYSUT3 DD UNIT=VIO,SPACE=(CYL,(5,2))
//SYSPUNCH DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSIN DD DSN=LRS.DRS.V1R33.ASM(DRSSUEXX),DISP=SHR
//SYSLIN DD DSN=&&OBJ,DISP=(,PASS),SPACE=(TRK,(6,3)),
// UNIT=VIO,DCB=BLKSIZE=1600
// *
//LKED EXEC PGM=IEWL,PARM='XREF,LET,LIST,RENT,REUS,REFR,MAP,AC=0',
// COND=(5,LT,ASMH)
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD UNIT=VIO,SPACE=(CYL,(1,1))
//SYSLMOD DD DSN=LRS.DRS.V1R33.LOAD(DRSSUEXX),DISP=SHR
//SYSLIN DD DSN=&&OBJ,DISP=(OLD,DELETE,DELETE)
/*
```

### Language

All user exits must be written in Assembler Language.

### Reentrancy Considerations

All user exits must be reentrant.

---

## Exit Types & Recovery Considerations

There are two types of DRS user exits: interface and subtask. **Interface** user exits execute under the same TCB as the DRS caller. **Subtask** user exits execute under the DRS service TCB (DRSSMAIN). The only difference between the two types of user exits is whether or not DRS will establish a recovery environment prior to invoking the user exit. DRS will not establish a recovery environment for interface related user exits. If the user exit abends, control will be passed to the DRS caller's recovery routine, if any. Subtask related user exit abends are intercepted by DRS. DRS disables the user exit, takes a dump (SYSUDUMP), and continues processing as if the user exit did not exist. Note that DRS will no longer invoke the user exit after an abend has been detected. The user should examine the dump to determine the cause of the abend. The type of each user exit is documented in the detailed description for each user exit.

## Execution Environment

All user exits are invoked in problem state and storage protect key 8.

## Linkage Conventions

When a user exit receives control, register 15 contains the entry address, register 14 contains the address to which the user exit must return control, and register 13 contains the address of OS-style 18 word savearea. All user exits must save and restore registers according to standard MVS linkage conventions.

## Parameters Passed

When control is passed to a DRS user exit routine, register 0 contains a request code and register 1 points to a parameter list. The request code and parameter list are user exit dependent and are documented in the detailed description for each user exit.

## Return Codes

When a user exit returns control to DRS, it must set an appropriate return code in register 15. The return codes are user exit dependent and are documented in the detailed description for each user exit.

## Exit-to-Exit Communication

The DRS System Attribute DSECT (mapping macro DRSSYAT), which is passed as the first parameter to all user exits, contains a two word field (SYAUSRWD) which can be used to pass information between user exits. These two words are never modified by DRS, so one user exit can place a value in it, or perhaps a pointer to some GETMAINED area, and other user exits will have that information available to them. Additionally, the DRS report control area (mapping macro DRSRCA), contains a two word field (RCAUSRWD) which can be used to pass information between the user exits that have access to it.

---

## Exit Tracing

DRS will trace entry to and exit from each user exit whenever the user exit trace type mask (X'10') is set on in the DRSSOPTS module. The DRS trace types may also be changed dynamically by issuing the DRS CMND request call specifying the SSET command.

The general format of each trace type is as follows.

<b>BYTES</b>	<b>CONTENTS</b>
<b>00-03</b>	Trace Type (UnnE = entry or UnnX = exit, nn = exit number)
<b>04-07</b>	Time Stamp (Format = HHMMSSSTH)
<b>08-09</b>	Request Code
<b>10-11</b>	Return Code
<b>12-15</b>	TCB address
<b>16-31</b>	Exit dependent data (documented with user exit description)

A copy of the DRS trace table can be obtained by issuing the DRS CMND request call specifying the SNAP command.

The parameters passed to each exit, along with all internal trace entries, can be traced to the MVS GTF trace file. See [“Customizing the DRS System Options” on page 20.9](#) for information on setting trace options.

---

## Exit Selection Table

Exit	Name	Title	Type	Recovery
00	DRSSUE00	DRS Initialization	Interface	No
01	DRSSUE01	DRS Termination	Interface	No
02	DRSSUE02	Request	Subtask	Yes
03	DRSSUE03	Add Line(s)	Subtask	Yes
04	DRSSUE04	Snap Dump	Subtask	Yes
05	DRSSUE05	WTO	Both	Yes & No
06	DRSSUE06	OS Services	Interface	No

**Table 22.1: Exit Selection Table**

---

## Initialization Exit (Exit 00)

<b>Module Name:</b>	DRSSUE00
<b>Invoked by:</b>	DRSSINIT
<b>Execution TCB:</b>	Interface TCB
<b>Recovery:</b>	None
<b>Function:</b>	This exit is called after DRS initialization has been completed.
<b>Request Code (R0):</b>	Always 00
<b>Parameter List (R1):</b>	Word 0 - A (System attributes)
<b>Trace Data (Entry):</b>	Request code System user words
<b>Trace Data (Exit):</b>	Return code System user words
<b>Return Codes (R15):</b>	00 - Successful completion
<b>Sample Program:</b>	Member DRSSUE00 in file LRS.DRS.V1R33.ASM is an example of the DRS Initialization Exit. This example merely issues a message via WTO, which includes the job name and the DRS version, release number and fix level.

---

## Termination Exit (Exit 01)

<b>Module Name:</b>	DRSSUE01
<b>Invoked by:</b>	DRSSTERM
<b>Execution TCB:</b>	Interface TCB
<b>Recovery:</b>	None
<b>Function:</b>	This exit is called after DRS termination has been completed.
<b>Request Code (R0):</b>	Always 00
<b>Parameter List (R1):</b>	Word 0 - A (System attributes)
<b>Trace Data (Entry):</b>	Request code System user words
<b>Trace Data (Exit):</b>	Return code System user words
<b>Return Codes (R15):</b>	00 - Successful completion
<b>Sample Program:</b>	Member DRSSUE01 in file LRS.DRS.V1R33.ASM is an example of the DRS Termination Exit. This example merely issues a message via WTO, which includes the job name and processing statistics.

---

## Request Exit (Exit 02)

<b>Module Name:</b>	DRSSUE02
<b>Invoked by:</b>	DRSSMAIN
<b>Execution TCB:</b>	Subtask TCB
<b>Recovery:</b>	ESTAE Exit
<b>Function:</b>	This exit is called prior to processing each DRS request. The exit may reject the request by returning with RC=04.
<b>Request Code (R0):</b>	1 - INIT Request 2 - PUT Request 3 - TERM Request 4 - QURY Request 5 - CMND Request 6 - SHUT Request 7 - OUTP Request
<b>Parameter List (R1):</b>	Word 0 - A (System attributes) Word 1 - A (DRRB) Word 2 - A (2 <sup>nd</sup> argument) or zeros Word 3 - A (Report control area) or zeros Word 4 - A (Alternate data buffer) (Initialized by the exit)
<b>Trace Data (Entry):</b>	Request code Address of DRRB Address of 2 <sup>nd</sup> argument
<b>Trace Data (Exit):</b>	Return code Address of DRRB Address of 2 <sup>nd</sup> argument
<b>Return Codes (R15):</b>	00 - Process request 04 - Reject request
<b>Sample Program:</b>	Member DRSSUE02 in file LRS.DRS.V1R33.ASM is an example of the DRS Request Exit. This example reformats 3270 data into print lines.

---

## Add Lines Exit (Exit 03)

<b>Module Name:</b>	DRSSUE03
<b>Invoked by:</b>	DRSSMAIN
<b>Execution TCB:</b>	Subtask TCB
<b>Recovery:</b>	ESTAE Exit
<b>Function:</b>	This exit is called after a dataset has been successfully allocated and opened and prior to unallocation. The purpose of this exit is to allow insertion of data before and after each SYSOUT dataset created by DRS.
<b>Request Code (R0):</b>	00 - Add line(s) at start of dataset 04 - Add line(s) at end of dataset
<b>Parameter List (R1):</b>	WORD 0 - A (System attributes) WORD 1 - A (Report control area) WORD 2 - A (Report attributes) WORD 3 - A (Line workarea) WORD 4 - A (User words)
<b>Trace Data (Entry):</b>	Request code Address of RAT Address of line work area Contents of user words
<b>Trace Data (Exit):</b>	Return code Address of RAT Address of line work area Contents of user words
<b>Return Codes (R15):</b>	00 - Write line (invoke exit again) 04 - Do not write line (exit completed)
<b>Sample Program:</b>	Member DRSSUE03 in file LRS.DRS.V1R33.ASM is an example of the DRS Add Lines Exit. This example adds separator or banner pages at the start and end of each data file.
<b>Programming Considerations:</b>	For DRS/VPI users, the SEPAR keyword now allows banner pages to be added by routines which can be unique for each virtual printer definition.



---

## Snap Dump Exit (Exit 04)

<b>Module Name:</b>	DRSSUE04
<b>Invoked by:</b>	DRSSSNAP
<b>Execution TCB:</b>	Subtask TCB
<b>Recovery:</b>	ESTAE exit
<b>Function:</b>	This exit is called whenever an error is detected while processing a user DRS request (e.g., INIT, PUT, TERM, etc). The purpose of this exit is to allow the user to determine whether or not a SNAP dump should be taken.
<b>Request Code (R0):</b>	00 - Invoked by error processing 04 - Invoked by command processing
<b>Parameter List (R1):</b>	Word 0 - A (System attributes) Word 1 - A (Report control area) Word 2 - A (Error attributes)
<b>Trace Data (Entry):</b>	Request code Report ID Default DRS action
<b>Trace Data (Exit):</b>	Return code Report ID Updated DRS action
<b>Return Codes (R15):</b>	00 - Successful completion
<b>NOTES:</b>	To indicate that a snap dump should not be created, the snap dump flag (\$DUMP) on the field ERAFLAG1 should be turned off. This is demonstrated in the sample User Exit 04 distributed as member DRSSUE04 in file LRS.DRS.V1R33.ASM.
<b>Sample Program:</b>	Member DRSSUE04 in file LRS.DRS.V1R33.ASM is an example of the DRS Snap Dump Exit. This example bypasses SNAP dumps for dynamic allocation errors due to invalid destinations.

---

## WTO Exit (Exit 05)

<b>Module Name:</b>	DRSSUE05
<b>Invoked by:</b>	DRSSMSGH
<b>Execution TCB:</b>	All DRS subtasks
<b>Recovery:</b>	None
<b>Function:</b>	This exit is called prior to issuing a WTO. The exit may modify the WTO, request that the WTO not be issued, or request that the WTO not be issued and not be logged to SYSLOG.
<b>Request Code (R0):</b>	Always 00
<b>Parameter List (R1):</b>	Word 0 - A (System attributes) Word 1 - A (WTO attributes)
<b>Trace Data (Entry):</b>	Request code Message ID Message attributes Message text length Message text (first 10 bytes)
<b>Trace Data (Exit):</b>	Return code Message ID Message attributes Message text length Message text (first 10 bytes)
<b>Return Codes (R15):</b>	00 - Issue WTO 04 - Do not issue WTO 04 - Do not issue WTO and do not log message to SYSLOG
<b>Sample Program:</b>	Member DRSSUE05 in file LRS.DRS.V1R33.ASM is an example of the DRS WTO Exit. This example cancels message DRS009, changes the WTO descriptor code for messages DRS021 and DRS022, and bypasses writing messages DRS100 and DRS101 (ALLOCATION/UNALLOCATION) messages to SYSLOG.

---

## OS Services Exit (Exit 06)

<b>Module Name:</b>	DRSSUE06
<b>Invoked by:</b>	DRSSINTB
<b>Execution TCB:</b>	Interface TCB
<b>Recovery:</b>	None
<b>Function:</b>	This exit is called prior to issuing an OS services request (e.g., WAIT, ABEND, etc.) when using the non-CICS interface of DRS. The purpose of this exit is to allow non-CICS TP monitors to substitute a TP monitor request in place of the OS request. This will prevent DRS from placing the entire TP monitor in a wait state (WAIT/ENQ request) or abending the TP monitor (ABEND request).
<b>Request Code (R0):</b>	Always 00
<b>Parameter List (R1):</b>	Word 0 - A (System attributes) Word 1 - A (OS services attributes)
<b>Trace Data (Entry):</b>	Request code OS services request type ABEND code (ABEND request) ENQ type (SHR EXCL) (ENQ/DEQ request) ENQ/DEQ RNAME length (ENQ/DEQ request) ENQ/DEQ RNAME (ENQ/DEQ request) ECB address (WAIT request) ECB value (WAIT request)
<b>Trace Data (Exit):</b>	Return code OS services request type ABEND code (ABEND request) ENQ type (SHR EXCL) (ENQ/DEQ request) ENQ/DEQ RNAME length (ENQ/DEQ request) ENQ/DEQ RNAME (ENQ/DEQ request) ECB address (WAIT request) ECB value (WAIT request)
<b>Return Codes (R15):</b>	00 - Request processed 04 - Function not implemented
<b>Sample Program:</b>	Member DRSSUE06 in file LRS.DRS.V1R33.ASM is an example of the DRS OS Services Exit. This example does not change the processing for any OS services request.



---

## Section 23

# DRS/API JES Considerations

In this section, we list a number of items related to JES operation which should be considered before you begin to use DRS. These items pertain to certain internal limits in JES2 and JES3.

Most users of DRS implement DRS in a CICS or other TP monitor environment to create a large number of small reports. A common example would be using DRS to create pick tickets for a warehouse, where each “set” of DRS calls (INIT, PUT, TERM) creates one ticket. This type of application can result in a significant increase in the number of individual SYSOUT datasets in existence on the JES spool at any point in time. Most of the considerations listed in this section pertain to JES limitations on the number of datasets.

## JES2 Considerations

Users of non-MVS/ESA versions of JES2 (versions prior to JES2 3.1.1) should apply APAR OZ93770. This APAR increases the number of SYSOUT datasets that can be created per job from 9,899 to 32,667. Each attribute group in a DRS “INIT” call creates a new SYSOUT dataset. An attempt to create more than the maximum will result in an allocation failure at the time of the “INIT” call. The DRS return code field in the DRRB will contain decimal 182. The OS return code field in the DRRB will contain X'04780000', which indicates JES is unable to process the allocation request.

MVS/ESA versions of JES2 allow creation of a maximum of 9,999,999 SYSOUT datasets per job.

The additional output generated by DRS may require increasing the size of the JES2 output queue (number of JOEs). If the SYSOUT datasets created by DRS are retrieved from the JES spool and printed in a timely fashion, the number of JOEs now specified may be adequate. However, exceptional situations, such as “broken” printers or an extremely high volume of processing, resulting in many more SYSOUT datasets than normal, should be considered.

If the JES2 output queues are full (no JOEs available), SYSOUT datasets being spun or becoming available as jobs end will not be queued for processing. These SYSOUT datasets will remain unavailable until the JOE shortage is relieved. The JES2 spool offload facility can be used to relieve a shortage of JOEs.

### Spool Space

While the spool space for the dataset itself is made available for reuse after a spin dataset has been processed, the space for the associated spool control block remains until the whole job is purged. One spool block is required, to contain a spin IOT control block, for each SYSOUT dataset created by DRS.

To reclaim this space, you need to stop and start the job which is issuing the DRS calls (such as CICS). After all of the datasets from the job have been processed, the job will be purged and the spool space will be reclaimed. Installations should consider increasing the amount of spool space allocated to mitigate the impact of unpurged control blocks. The amount of additional space needed depends upon the number of datasets being created and the frequency with which the application which uses DRS (such as CICS) is restarted.

In addition to taking spool space, these control blocks are processed during certain types of JES2 warm starts, which may extend the length of the warm start.

---

## JES3 Considerations

JES3 uses JSAM buffers to hold control information about the spin datasets when it is adding or processing a dataset. If a backlog occurs, more buffers will be needed. If there are not enough buffers, JES3 and the task creating datasets (such as CICS) will hang. Installations may have to increase the buffer space definition in their JES3 initialization decks to provide sufficient buffers so that a backlog can be detected and cleared before JES3 runs out of buffers.

### Spool Space

While the spool space for the dataset itself is made available for reuse after a spin dataset has been processed, the space occupied by the JDS and OSE control blocks remains allocated until the whole job is purged. Each spin dataset uses one entry in a JDS record as well as in an OSE record.

To reclaim this space, you need to stop and start the job which is issuing the DRS calls (such as CICS). After all of the datasets from the job have been processed, the job will be purged and the spool space will be reclaimed. Installations should consider increasing the amount of spool space allocated to mitigate the impact of unpurged control blocks. The amount of additional space needed depends upon the number of datasets being created and the frequency with which the application which uses DRS (such as CICS) is restarted.

If the SYSOUT datasets created by DRS are retrieved from the JES spool and printed in a timely fashion, the amount of spool space now specified may be adequate. However, exceptional situations, such as “broken” printers or an extremely high volume of processing, resulting in many more SYSOUT datasets than normal, should be considered.

If a shortage of spool space occurs, the JES3 dump job facility can be used to free spool space. After the shortage has been relieved, the facility can be used to restore the jobs to the system.

## Section 24

### DRS/API Return Codes

Return code (decimal)	Return code (hex)	Meaning
0000	X'0000'	GOOD RETURN CODE
0002	X'0002'	DRRB IS IN BAD STORAGE
0003	X'0003'	INVALID DRRB
0004	X'0004'	REQUEST REJECTED BY USER EXIT
0020	X'0014'	DRRB INVALID REPORT ID
0021	X'0015'	DRRB FUNCTION IS INVALID
0023	X'0017'	DRRB OPTION 1 IS INVALID
0024	X'0018'	DRRB OPTION 2 IS INVALID
0025	X'0019'	DRRB OPTION 3 IS INVALID
0026	X'001A'	DRRB OPTION 4 IS INVALID
0027	X'001B'	DRRB LINE COUNT IS INVALID
0050	X'0032'	DRS SHUTDOWN IN PROGRESS
0051	X'0033'	MULTIPLE SUBTASKS ATTEMPTING TO UPDATE SAME REPORT
0060	X'003C'	NO STORAGE FOR SWA
0061	X'003D'	NO STORAGE FOR SCA
0080	X'0050'	ATTACH ERROR
0081	X'0051'	SUBTASK ABENDED
0082	X'0052'	DRS INITIALIZATION ERROR
0101	X'0065'	BAD PARMS PASSED (ONLY ONE)
0102	X'0066'	DRIB BASE IS IN BAD STORAGE
0103	X'0067'	INVALID DRIB
0104	X'0068'	DRIB ATTR IS IN BAD STORAGE
0105	X'0069'	DRIB EXTENDED ATTR IS IN BAD STORAGE
0110	X'006E'	INVALID DATASET NAME

<b>Return code (decimal)</b>	<b>Return code (hex)</b>	<b>Meaning</b>
0111	X'006F'	INVALID MEMBER NAME
0112	X'0070'	INVALID UNIT
0113	X'0071'	INVALID ALLOCATION TYPE
0115	X'0073'	INVALID PRIMARY ALLOCATION
0116	X'0074'	INVALID SECONDARY ALLOCATION
0117	X'0075'	INVALID INITIAL ALLOCATION DISPOSITION
0118	X'0076'	INVALID VOLUME PARAMETER
0119	X'0077'	INVALID NORMAL DISPOSITION
0120	X'0078'	INVALID NUMBER OF ATTR SECTIONS
0121	X'0079'	DRIB RECORD FORMAT IS INVALID
0122	X'007A'	DRIB LRECL IS INVALID
0123	X'007B'	DRIB BLKSIZE IS INVALID
0124	X'007C'	DRIB JOBNAME IS INVALID
0127	X'007F'	DRIB CNTRL IS INVALID
0128	X'0080'	DRIB DEST IS INVALID
0129	X'0081'	DRIB WRITER IS INVALID
0130	X'0082'	DRIB FORM IS INVALID
0131	X'0083'	DRIB FCB IS INVALID
0132	X'0084'	DRIB UCS IS INVALID
0133	X'0085'	DRIB CLASS IS INVALID
0134	X'0086'	DRIB HOLD IS INVALID
0135	X'0087'	DRIB COPIES IS INVALID
0136	X'0088'	DRIB EXTEND ATTR FLAG IS INVALID
0137	X'0089'	DRIB DELETE OPTION IS INVALID
0138	X'008A'	DRIB DDNAME IS INVALID
0139	X'008B'	DRIB EXTEND FORM IS INVALID
0140	X'008C'	DRIB SYSOUT USERID IS INVALID
0141	X'008D'	DRIB OUTPUT REF IS INVALID
0142	X'008E'	DRIB COPY GROUP(S) ARE INVALID
0143	X'008F'	DRIB CHAR TABLE(S) ARE INVALID
0144	X'0090'	DRIB BURST IS INVALID
0145	X'0091'	DRIB FLASH IS INVALID



<b>Return code (decimal)</b>	<b>Return code (hex)</b>	<b>Meaning</b>
0146	X'0092'	DRIB FLASH COUNT IS INVALID
0147	X'0093'	DRIB COPY MODULE IS INVALID
0148	X'0094'	DRIB COPY MOD REF IS INVALID
0150	X'0096'	INIT ON EXISTING REPORT
0151	X'0097'	DRS SYSTEM IS QUIESCING
0152	X'0098'	DRIB JOBNAME SPECIFIED; DRS/API LOADED FROM NON-AUTHORIZED LIBRARY
0153	X'0099'	DRIB JOBNAME SPECIFIED; MVS RELEASE DOES NOT SUPPORT JOBNAME CHANGE
0160	X'00A0'	NO STORAGE FOR RCA
0162	X'00A2'	NO STORAGE FOR BFCB
0163	X'00A3'	NO STORAGE FOR DATA LINE
0170	X'00AA'	DRIB UCS VERIFICATION IS INVALID
0171	X'00AB'	DRIB FOLD IS INVALID
0172	X'00AC'	DRIB SUBSYSTEM NAME IS INVALID
0173	X'00AD'	DRIB SUBSYSTEM PARM IS INVALID
0174	X'00AE'	INVALID DATSET NUMBER
0175	X'00AF'	INVALID DIRECTORY BLOCKS
0176	X'00B0'	INVALID RETENTION PERIOD
0177	X'00B1'	INVALID RETENTION PERIOD INDICATOR
0178	X'00B2'	INVALID AVERAGE RECORD UNIT
0179	X'00B3'	INVALID AVERAGE RECORD SIZE
0182	X'00B6'	DYNAMIC ALLOCATION FAILURE
0183	X'00B7'	OPEN FAILURE
0185	X'00B9'	DCB ABEND EXIT DRIVEN
0187	X'00BB'	SUBTASK ABENDED
0188	X'00BC'	SYNAD EXIT DRIVEN - ADD LINE(S) EXIT
0189	X'00BD'	ABEND EXIT DRIVEN - ADD LINE(S) EXIT
0191	X'00BF'	INVALID EXPIRATION DATE
0192	X'00C0'	INVALID DATA SET TYPE
0193	X'00C1'	INVALID RELEASE UNUSED SPACE INDICATOR
0194	X'00C2'	INVALID VOLUME COUNT
0195	X'00C3'	INVALID UNIT COUNT

<b>Return code (decimal)</b>	<b>Return code (hex)</b>	<b>Meaning</b>
0196	X'00C4'	INVALID SMS STORAGE CLASS
0197	X'00C5'	INVALID SMS MANAGEMENT CLASS
0198	X'00C6'	INVALID SMS DATA CLASS 11
0201	X'00C9'	BAD PARMS PASSED
0203	X'00CB'	DATA LINE IS IN BAD STORAGE
0220	X'00DC'	VARIABLE LENGTH RECORD TOO SMALL
0221	X'00DD'	VARIABLE LENGTH RECORD TOO LARGE
0222	X'00DE'	UNDEFINED LENGTH RECORD TOO SMALL
0223	X'00DF'	UNDEFINED LENGTH RECORD TOO LARGE
0250	X'00FA'	NO REPORT EXISTS
0251	X'00FB'	PREVIOUS TERM ERROR DETECTED
0280	X'0118'	SYNAD EXIT DRIVEN
0281	X'0119'	DCB ABEND EXIT DRIVEN
0283	X'011B'	SUBTASK ABENDED
0301	X'012D'	BAD PARMS PASSED
0303	X'012F'	DRTB BASE IS IN BAD STORAGE
0304	X'0130'	INVALID DRTB
0305	X'0131'	DRTB ATTR IS IN BAD STORAGE
0320	X'0140'	INVALID NUMBER OF ATTR SECTIONS
0321	X'0141'	INVALID OVERRIDING ATTRIBUTE INDEX
0322	X'0142'	INVALID OVERRIDING DISPOSITION
0323	X'0143'	INVALID OVERRIDING SYSOUT CLASS
0324	X'0144'	INVALID OVERRIDING HOLD
0325	X'0145'	INVALID OVERRIDING NOHOLD
0326	X'0146'	INVALID OVERRIDING DESTINATION
0350	X'015E'	NO REPORT EXISTS

<b>Return code (decimal)</b>	<b>Return code (hex)</b>	<b>Meaning</b>
0381	X'017D'	SUBTASK ABENDED
0382	X'017E'	SYNAD EXIT DRIVEN - ADD LINE(S) EXIT
0383	X'017F'	ABEND EXIT DRIVEN - ADD LINE(S) EXIT
0384	X'0180'	CLOSE FAILURE
0385	X'0181'	DYNAMIC UNALLOCATION FAILURE
0401	X'0191'	BAD PARMS PASSED (ONLY ONE)
0402	X'0192'	DRQB BASE IS IN BAD STORAGE
0403	X'0193'	INVALID DRQB
0404	X'0194'	DRQB EXTENSION IS IN BAD STORAGE
0420	X'01A4'	INVALID QUERY TYPE
0421	X'01A5'	INVALID NUMBER OF SLOTS
0422	X'01A6'	INVALID EXTENSION SIZE
0423	X'01A7'	INVALID ATTR SECTION NUMBER
0424	X'01A8'	INVALID QUERY REPORT ID
0450	X'01C2'	MISSING QUERY REPORT ID
0451	X'01C3'	ATTR SECTION NUMBER IS TOO LARGE FOR REPORT
0481	X'01E1'	SUBTASK ABENDED
0501	X'01F5'	BAD PARMS PASSED
0582	X'0246'	SOME REPORTS FAILED TERM PROCESSING
0587	X'024B'	SUBTASK ABENDED
0601	X'0259'	BAD PARMS PASSED (ONLY ONE)
0602	X'025A'	DRCB BASE IS IN BAD STORAGE
0603	X'025B'	INVALID DRCB
0620	X'026C'	INVALID COMMAND VERB
0621	X'026D'	INVALID LOG OPTION

<b>Return code (decimal)</b>	<b>Return code (hex)</b>	<b>Meaning</b>
0622	X'026E'	INVALID TRACE TYPES
0623	X'026F'	INVALID TRACE ADDRESS
0623	X'0270'	INVALID MESSAGE ID
0650	X'028A'	CLOSELOG REJECTED - LOGGING DISABLED
0660	X'0294'	NO STORAGE FOR LQE
0680	X'02A8'	SUBTASK ABENDED
0701	X'02BD'	BAD PARMS PASSED (ONLY ONE)
0702	X'02BE'	DROB BASE IS IN BAD STORAGE
0703	X'02BF'	INVALID DROB
0704	X'02C0'	DROB EXTENDED ATTR IS IN BAD STORAGE
0720	X'02D0'	DROB OUTPUT NAME IS INVALID
0721	X'02D1'	DROB FUNCTION IS INVALID
0722	X'02D2'	DROB ADDRESS IS INVALID
0723	X'02D3'	DROB BUILDING IS INVALID
0724	X'02D4'	DROB BURST IS INVALID
0725	X'02D5'	DROB CHAR TABLE(S) ARE INVALID
0726	X'02D6'	DROB CKPTLINE IS INVALID
0727	X'02D7'	DROB CKPTPAGE IS INVALID
0728	X'02D8'	DROB CKPTSEC IS INVALID
0729	X'02D9'	DROB CLASS IS INVALID
0730	X'02DA'	DROB COMPACT IS INVALID
0731	X'02DB'	DROB CONTROL IS INVALID
0732	X'02DC'	DROB COPY IS INVALID
0733	X'02DD'	DROB COPY GROUP(S) ARE INVALID
0734	X'02DE'	DROB DATAACK IS INVALID
0735	X'02DF'	DROB DEFAULT IS INVALID
0736	X'02E0'	DROB DEPARTMENT IS INVALID
0737	X'02E1'	DROB DESTINATION IS INVALID
0738	X'02E2'	DROB DPAGELBL IS INVALID
0739	X'02E3'	DROB FCB IS INVALID

<b>Return code (decimal)</b>	<b>Return code (hex)</b>	<b>Meaning</b>
0740	X'02E4'	DROB FLASH IS INVALID
0741	X'02E5'	DROB FLASH COUNT IS INVALID
0742	X'02E6'	DROB FORMDEF IS INVALID
0743	X'02E7'	DROB FORM IS INVALID
0744	X'02E8'	DROB GROUPID IS INVALID
0745	X'02E9'	DROB INDEX IS INVALID
0746	X'02EA'	DROB LINDEX IS INVALID
0747	X'02EB'	DROB LINECT IS INVALID
0748	X'02EC'	DROB COPY MODULE IS INVALID
0749	X'02ED'	DROB COPY MOD REF IS INVALID
0750	X'02EE'	NOT EXECUTING UNDER MVS/ESA
0751	X'02EF'	DRS SYSTEM QUIESCING
0752	X'02F0'	OUTPUT STATEMENT DOES NOT EXIST
0753	X'02F1'	OUTPUT STATEMENT ALREADY EXISTS
0754	X'02F2'	NULL OUTPUT STATEMENT SPECIFIED
0755	X'02F3'	OUTPUT STATEMENT DESTINATION CONFLICT
0760	X'02F8'	NO STORAGE FOR OCA
0770	X'0302'	DROB NAME IS INVALID
0771	X'0303'	DROB NOTIFY IS INVALID
0772	X'0304'	DROB OUTDISP (NORMAL) IS INVALID
0773	X'0305'	DROB OUTDISP (ABNORMAL) IS INVALID
0774	X'0306'	DROB PAGEDEF IS INVALID
0775	X'0307'	DROB PIMSG IS INVALID
0776	X'0308'	DROB PIMSG COUNT IS INVALID
0777	X'0309'	DROB PRMODE IS INVALID
0778	X'030A'	DROB PRIORITY IS INVALID
0779	X'030B'	DROB ROOM IS INVALID
0780	X'030C'	SUBTASK ABENDED
0781	X'030D'	DYNAMIC OUTPUT FAILURE (OUTADD)
0782	X'030E'	DYNAMIC OUTPUT FAILURE (OUTDEL)

<b>Return code (decimal)</b>	<b>Return code (hex)</b>	<b>Meaning</b>
0820	X'0334'	DROB SYSAREA IS INVALID
0821	X'0335'	DROB TITLE IS INVALID
0822	X'0336'	DROB TRC IS INVALID
0823	X'0337'	DROB UCS IS INVALID
0824	X'0338'	DROB WRITER IS INVALID
0825	X'0339'	DROB EXTEND ATTR FLAG IS INVALID
0826	X'033A'	DROB USERLIB IS INVALID
0827	X'033B'	DROB USERDATA IS INVALID
0828	X'033C'	DROB OUTBIN IS INVALID
0829	X'033D'	DROB COLORMAP IS INVALID
0830	X'033E'	DROB COMSETUP IS INVALID
0831	X'033F'	DROB DEST (LONG) IS INVALID
0832	X'0340'	DROB DUPLEX IS INVALID
0833	X'0341'	DROB FORMLEN IS INVALID
0834	X'0342'	DROB INTRAY IS INVALID
0835	X'0343'	DROB OFFSETXB IS INVALID
0836	X'0344'	DROB OFFSETXF IS INVALID
0837	X'0345'	DROB OFFSETYB IS INVALID
0838	X'0346'	DROB OFFSETYF IS INVALID
0839	X'0347'	DROB OVERLAYB IS INVALID
0840	X'0348'	DROB OVERLAYF IS INVALID
0841	X'0349'	DROB OVFL IS INVALID
0842	X'034A'	DROB PORTNO IS INVALID
0843	X'034B'	DROB PRTERORR IS INVALID
0844	X'034C'	DROB PRTOPTNS IN INVALID
0845	X'034D'	DROB PRTQUEUE IS INVALID
0846	X'034E'	DROB RETAINF IS INVALID
0847	X'034F'	DROB RETAINS IS INVALID
0848	X'0350'	DROB RETRYL IS INVALID

<b>Return code (decimal)</b>	<b>Return code (hex)</b>	<b>Meaning</b>
0849	X'0351'	DROB RETRYT IS INVALID
0850	X'0352'	DROB RESFMT IS INVALID
0851	X'0353'	DROB ZERO-VALUE FLAGS ARE INVALID
0950	X'03B6'	INVALID DRRB FUNCTION CODE
0951	X'03B7'	INVALID DRSSCBVL REQUEST CODE
0952	X'03B8'	UNEXPECTED ABEND IN DRSSMAIN SUBTASK
0980	X'03D4'	ALL SUBTASKS ABENDED
0981	X'03D5'	INVALID SCA CONTROL BLOCK
0998	X'03E6'	INVALID DRS KEY SPECIFIED
0999	X'03E7'	DRS TRIAL HAS EXPIRED
2801	X'0AF1'	VSIO BAD PARMS PASSED (ONLY ONE)
2861	X'0B2D'	VSIO NO STORAGE FOR SFCB
2862	X'0B2E'	VSIO NO STORAGE FOR VSCB
2881	X'0B41'	VSIO DYNAMIC ALLOCATION FAILURE
2887	X'0B47'	VSIO SUBTASK ABENDED
3001	X'0BB9'	UNSUPPORTED ACCESS METHOD VERSION
3002	X'0BBA'	INVALID ACCESS METHOD REQUEST CODE. (The reason code contains the invalid request code)
3003	X'0BBB'	UNSUPPORTED OPERATING ENVIRONMENT
3004	X'0BBC'	UNSUPPORTED NATURAL FILE TYPE
3005	X'0BBD'	OPEN FOR INPUT REQUEST IS NOT SUPPORTED
3006	X'0BBE'	READ REQUEST IS NOT SUPPORTED
3007	X'0BBF'	UTILITY REQUEST IS NOT SUPPORTED
3010	X'0BC2'	NATURAL REGION OR SYSTEM SWITCH IS NOT SUPPORTED DURING PRINT REQUEST.
3011	X'0BC3'	RECORD LENGTH IS GREATER THAN FILE BLKSIZE AND TRUNCATION IS DISABLED.
3012	X'0BC4'	CICS DFHEISTG ADDRESS IS INVALID
3013	X'0BC5'	CICS DFHEIB ADDRESS IS INVALID

Return code (decimal)	Return code (hex)	Meaning
3051	X'0BEB'	LOGIC ERROR – INVALID OPTSUTIL REQUEST CODE
3052	X'0BEC'	LOGIC ERROR – INVALID DRSSTOR REQUEST CODE
3060	X'0BF4'	NATURAL THREAD WORKING STORAGE IS NOT AVAILABLE
3061	X'0BF5'	GETMAIN REQUEST FAILED (The reason code contains the OS GETMAIN return code or the CICS EIBRESP return code.)
3062	X'0BF6'	FREEMAIN REQUEST FAILED (The reason code contains the OS FREEMAIN return code or the CICS EIBRESP return code.)
3080	X'0C08'	LOAD REQUEST FOR OUTPUT CHARACTERISTICS TABLE FAILED (DRSNSDEF) (The reason code contains the OS ABEND code or the CICS EIBRESP return code.)
3081	X'0C09'	DELETE FAILED FOR OUTPUT CHARACTERISTICS TABLE (DRSNSDEF) (The reason code contains the CICS EIBRESP return code.)
3082	X'0C0A'	CICS LINK REQUEST TO MODULE DRSSINTC FAILED (The reason code contains the CICS EIBRESP return code.)
3083	X'0C0B'	LINK TO DRSSINTB FAILED (The reason code contains the OS ABEND code.)
3084	X'0C0C'	OUTPUT CHARACTERISTICS TABLE HEADER IS INVALID
3085	X'0C0D'	OUTPUT CHARACTERISTICS TABLE CONTAINS UNSUPPORTED VERSION NUMBER.
3086	X'0C0E'	INCORRECT DRS/API VERSION. MUST BE V1 R3.3 AT FIX LEVEL 35 OR HIGHER.
3098	X'0C1A'	DRS/NATURAL PRODUCT KEY IS INVALID
3099	X'0C1B'	DRS/NATURAL PRODUCT KEY HAS EXPIRED
3198	X'0C7E'	DRS/SAPR2 KEY IS INVALID
3199	X'0C7F'	DRS/SAPR2 KEY HAS EXPIRED

**Table 24.1: DRS/API Return Codes**



---

## Section 25

# Sample DRS/API Programs

The distribution cartridge contains the source code for several sample programs which invoke DRS/API. These programs demonstrate the formats of the calls and the control blocks passed to the calls.

“[Sample COBOL and Assembler Programs](#)” on page 25.2 are very simple examples intended to show the basic formats of calls. “[DRS Call Utility](#)” on page 25.5 is an application that can be used to test DRS calls and to monitor the activity of DRS/API in a CICS address space.

---

## Sample COBOL and Assembler Programs

Execution JCL for the non-CICS programs should include both a SYSOUT and a SYSUDUMP DD statement. The non-CICS COBOL programs use the “DISPLAY” verb to inform the user of any non-zero return codes from the DRS calls; the non-CICS assembler language programs write messages to DDNAME SYSOUT to inform the user of any non-zero return codes from the DRS calls. Both the COBOL and assembler language programs generate a dump with abend code U0100 when any non-zero DRS return code occurs.

You can associate any transaction ID with the CICS programs to test them. They do not use any CICS maps. An example of the PCT entry for one of these programs is:

```
DFHPCT TYPE=ENTRY, X
      TRANSID=XXXX, (use any transid you choose) X
      PROGRAM=DRSXC04, X
      PRIVATE=YES, X
      DTB=NO
```

An example of the PPT entry for one of these programs is:

```
DFHPPT TYPE=ENTRY, X
      PROGRAM=DRSXC04, X
      PGMLANG=COBOL
```

or

```
DFHPPT TYPE=ENTRY, X
      PROGRAM=DRSXA04, X
      PGMLANG=ASSEMBLER
```

The sample programs and their functions are:

### Example # 1:

COBOL source member name: DRSXC01

Assembler source member name: DRSXA01

This non-CICS program uses the INIT, PUT, and TERM calls to create a report. The report is routed to a destination of U3999 and SYSOUT class of A. Each PUT call adds one print line to the report. The resulting report contains a total of four print lines.

---

**Example # 2:**

COBOL source member name: DRSXC02

Assembler source member name: DRSXA02

This non-CICS program uses the INIT, PUT, and TERM calls to create a report. Three attribute groups are described on the INIT call, so three datasets are created on the JES spool, each with a unique combination of destination, class, etc. This program demonstrates adding more than one line to a report with a single PUT call.

This program also demonstrates the use of the detail QURY call to retrieve information about a dataset, as well as using the termination control block on the TERM call to modify the characteristics of a dataset.

**Example # 3:**

COBOL source member name: DRSXC03

Assembler source member name: DRSXA03

This non-CICS program demonstrates the use of the CMND call. After the INIT and PUT calls are used to create an active report, the SNAP command is issued. This is followed by a TERM call.

**Example # 4:**

COBOL source member name: DRSXC04

Assembler source member name: DRSXA04

This CICS program uses the INIT, PUT, and TERM calls to create a report. Three attribute groups are described on the INIT call, so three datasets are created on the JES spool, each with a unique combination of destination, class, etc. This program demonstrates adding more than one line to a report with a single PUT call.

**Example # 5:**

COBOL source member name: DRSXC05

Assembler source member name: DRSXA05

This CICS program uses the INIT, PUT, and TERM calls to submit a batch job via the internal reader. In this case, the INIT call indicates a writer name of INTRDR, and the PUT call is passing JCL statements to DRS rather than print lines.

---

**Example # 6:**

COBOL source member name: DRSXC06

Assembler source member name: DRSXA06

This non-CICS program uses the INIT, PUT, and TERM calls to create two datasets. One is a SYSOUT dataset, and the other is a DASD dataset.

**Example #7:**

COBOL source member name: DRSXC07

Assembler source member name: DRSXA07

This non-CICS program uses the OUTP ADD and OUTP DEL calls. It dynamically adds an OUTPUT statement and refers to that OUTPUT statement when making the INIT call. Several lines are added to the report with the PUT call. The TERM call is used to close the report and the OUTP DEL call is used to delete the OUTPUT statement.

---

## DRS Call Utility

The DRS Call Utility provides for the interactive construction and execution of DRS System calls from a CICS transaction. The following options are available:

<b>INIT</b>	Establishes the attributes of a report. These are similar to the parameters you would code in SYSOUT JCL statements.
<b>PUT</b>	Adds print lines to a report that has been previously initiated.
<b>TERM</b>	Ends a previously initiated report. The report is then available for printing from the JES spool.
<b>OUTP</b>	Dynamically defines an OUTPUT JCL statement, which can then be referenced in a DRS "INIT" call.
<b>QUERY GENL</b>	Returns information on which reports are active in DRS.
<b>QUERY DETL</b>	Returns information on the attributes of a specific initiated report.
<b>QUERY OUTG</b>	Returns information about dynamically defined OUTPUT JCL statements.
<b>QUERY OUTD</b>	Returns information about a specific dynamically defined OUTPUT JCL statement.
<b>QUERY SYS</b>	Returns information about DRS system status and collected statistics.
<b>CMND</b>	Permits issuing special commands to DRS.
<b>TRACE</b>	Browses the contents of the DRS system trace table (not a DRS call).
<b>ERROR CODES</b>	Browses the contents of a list of DRS error codes and dynamic allocation error codes.

Each screen displayed by the DRS Call Utility has a HELP screen, which can be viewed by using the PF-1 key. These HELP screens describe the function which the screen supports and the types of information which appear on the screen.

Installation procedures for the DRS Call Utility are given on the next page.

---

## Installing the DRS Call Utility

The steps required to install the DRS Call Utility are:

1. Update the CICS tables to define the DRS Call Utility (i.e., transaction, program, and map).

```
DFHPCT TYPE=ENTRY,          *
      TRANSID=XXXX,          (use any transid you choose) *
      PROGRAM=DRSXUTL,      *
      PRIVATE=YES,          *
      DTB=NO                *

DFHPPT TYPE=ENTRY,          DRS Call Utility Program      *
      PROGRAM=DRSXUTL,      *
      PGMLANG=ASSEMBLER     *

DFHPPT TYPE=ENTRY,          DRS Call Utility Mapset       *
      MAPSET=DRSXUTM,      *
      PGMLANG=ASSEMBLER,   *
      USAGE=MAP             *
```

2. Copy the following modules from LRS.DRS.V1R33.LOAD into one of the CICS DFHRPL libraries.

- DRSXUTL (DRS Call Utility program)
- DRSXUTM (DRS Call Utility mapset)

Sample JCL to assemble and linkedit the DRS Call Utility is supplied in members D2UTASM, D2UTLINK, and D2UTMAP on file #28 of the distribution cartridge. Installations that choose to modify the DRS Call Utility can use the sample JCL to re-assemble and re-link the utility.

Source to the DRS Call Utility is supplied on the distribution cartridge. The source consists of the following members:

LRS.DRS.V1R33.ASM (file #26)

- DRSXUTCA (COMMAREA definition)
- DRSXUTMS (Mapset)
- DRSXUT00 (Primary Menu processor)
- DRSXUT01 (INIT Call processor)
- DRSXUT02 (PUT Call processor)
- DRSXUT03 (TERM Call processor)
- DRSXUT04 (OUTP Call processor)
- DRSXUT05 (GENL QURY Call processor)
- DRSXUT06 (DETL QURY Call processor)
- DRSXUT07 (OUTG QURY Call processor)
- DRSXUT08 (OUTD QURY Call processor)
- DRSXUT09 (SYS QURY Call processor)
- DRSXUT10 (CMND Call processor)
- DRSXUT11 (Browse DRS trace table processor)
- DRSXUTAA (Browse DRS and/or DAIR error codes)

LRS.DRS.V1R33.MACLIB (file #27)

- DRSXUTM (Mapset definition)

---

## Section 26

# Introduction To DRS/SAPR2

The DRS/SAPR2 interface is a direct interface from SAP R2 applications to the JES spool. It enables any existing SAP R2 application running in CICS to create a report on the JES spool without any application changes and without the use of either SAP R2 exits or the SAP R2 external spool interface.

Using DRS/SAPR2 provides complete control over the JES spool attributes assigned to a report and enables existing reports to use the sophisticated printing facilities of Advanced Function Presentation (AFP) to enhance the quality of printed output from existing SAP R2 applications.

With the DRS/SAPR2 interface, JES spool attributes can be specified at the system level as global defaults for the entire SAP R2 system, at the individual printer level as defaults for a specific report, or can be specified for individual reports.

The JES spool attributes assigned to a report are controlled by the SAP R2 format names which are specified in SAP R2 table 022D. By selecting different format names you can control the spooling attributes of specific reports, or you can specify a default format to be used by a specific printer in the SAP R2 printer table 022B.

The DRS/SAPR2 interface provides complete control over all SYSOUT attributes including class, destination, form name, writer name, FORMDEF, PAGEDEF, etc.

DRS/SAPR2 is simple to install and requires no changes to SAP R2 software. It can be implemented at the individual printer level to enable simple migration from existing printing facilities and has no effect on existing printers which are directly controlled by SAP R2 print facilities.





---

# Section 27

## DRS/SAPR2 Installation

### Introduction

This section gives you the information you need to install DRS/SAPR2 on your system. The details of this procedure are listed in “Installation Steps” on the following page. Before you do the actual installation, you must install DRS/API. Please see [“DRS/API Installation” on page 20.1](#).

---

# Installation Steps

The steps to install the DRS/SAPR2 Print Interface are:

1. **Read the entire installation procedure.**
2. **Install the DRS/API product.** (See [“DRS/API Installation”](#) on page 20.1.)
3. **Build the DRS/SAPR2 defaults module.** (See [“Building the DRS/SAPR2 Defaults Module”](#) on page 27.4.)
4. **Build the DRS/SAPR2 printer alias table (optional).** (See [“Building the DRS/SAPR2 Printer Alias Table”](#) on page 27.13.)
5. **Update the CICS tables to define the DRS/SAPR2 interface modules, transactions and message queues.** (See [“Updating the CICS Tables”](#) on page 27.15.)
6. **Copy the DRS/SAPR2 programs to a load library in the CICS DFHRPL concatenation.** (See [“Copying the DRS/SAPR2 Modules for CICS”](#) on page 27.17.)
7. **Define a printer TD queue and restart CICS with a COLD start.** (See [“Defining the Printer TD Queue”](#) on page 27.18.)
8. **Define the new printer TD queue to SAP R2 via table 022B.** (See [“Defining the Printer to SAP R2”](#) on page 27.19.)
9. **The DRS/SAPR2 interface installation is now complete.**

The remainder of this section provides the information required to perform the above steps.

---

## Installing the DRS/API Product

The DRS/SAPR2 print interface uses the facilities of DRS/API to create output on the JES spool. Before installing the DRS/SAPR2 interface, the DRS/API product must be installed and defined to CICS.

For information on installing DRS/API, please refer to [“DRS/API Installation”](#) on page 20.1.

---

## Building the DRS/SAPR2 Defaults Module

The DRS/SAPR2 print interface uses an assembled module named DPSSSDEF to define the default SYSOUT attributes which should be assigned to output created from SAP R2 applications running in a CICS system. These global defaults can then be overridden for individual printers or reports.

The DRS/SAPR2 interface defaults module (DPSSSDEF) allows an installation to define default values for any of the SYSOUT attributes that would be available using the SYSOUT DD JCL statement or the OUTPUT JCL statement. The SYSOUT attributes defined in DPSSSDEF are used, along with the ERRACTN, ERRCLASS and ERRDEST keywords, to generate a SYSOUT dataset if an error is encountered, including reports which are incomplete.

In addition to SYSOUT parameters, certain processing options are defined in the defaults module.

The defaults module is generated by performing the following steps:

Modify member DPSSSDEF in library LRS.DRS.V1R33.ASM, specifying the values you want to use as a default for each parameter.

Create the defaults module named DPSSSDEF by assembling and linking your modified version of the DPSSSDEF member into a CICS DFHRPL library. Sample JCL to assemble and link the DRS/SAPR2 defaults module is contained in member SDEFASML or SDEFASMH in dataset LRS.DRS.V1R33.CNTL.

The keywords which are used to define the defaults modules are described on the following pages.

The following is a sample DPSSSDEF source member:

```
$DRSSDEF @OUTDESC='SAMPLE DEFAULTS MEMBER',      X
      CLASS=A,                                     X
      NODE=N1,                                     X
      DEST=R99,                                    X
      FORMS=STD,                                   X
      FORMDEF=A10110,                              X
      PAGEDEF=B11082
END
```

---

## DRS/SAPR2 Defaults Module

```
b      One or more blanks must precede $DRSSDEF
$DRSSDEF
  { @OUTDESC= }
  { ,ADDR1= }
  { ,ADDR2= }
  { ,ADDR3= }
  { ,ADDR4= }
  { ,BUILDING= }
  { ,BURST= }
  { ,CHARS= }
  { ,CKPTLINE= }
  { ,CKPTPAGE= }
  { ,CKPTSEC= }
  { ,CLASS= }
  { ,COLORMAP= }
  { ,COMPACT= }
  { ,COMSETUP= }
  { ,CONTROL= }
  { ,COPIES= }
  { ,DATAACK= }
  { ,DEPT= }
  ( ,DEST= )
  { ,DPAGELBL= }
  { ,DUPLEX= }
  { ,ERRACTN= }
  { ,ERRCLASS= }
  { ,ERRDEST= }
  { ,FCB= )
  { ,FFOPT= }
  { ,FLASH= }
  { ,FORMDEF= }
  { ,FORMLEN= }
  ( ,FORMS= )
  { ,GROUPID= )
  ( ,INDEX= )
  ( ,LINDEX= )
  { ,LINECT= }
  { ,MAXLRECL= }
  { ,MODIFY= }
  { ,NAME= }
  { ,NODE= }
  { ,NOTIFY= }
  { ,NOTIFY2= }
  { ,NOTIFY3= }
  { ,NOTIFY4= }
  { ,OFFSETXB= }
  { ,OFFSETXF= }
  { ,OFFSETYB= }
  { ,OFFSETYF= }
  { ,OUTBIN= }
  { ,OUTDISP= }
```

---

```
{ ,OVERLAYB=}
{ ,OVERLAYF=}
{ ,OVLF=}
{ ,PAGEDEF=}
{ ,PIMSG=}
{ ,PORTNO=}
{ ,PRMODE=}
{ ,PRTEROR=}
{ ,PRTOPTS=}
{ ,PRTQUEUE=}
{ ,PRTY=}
{ ,RESFMT=}
{ ,RETAINF=}
{ ,RETAINS=}
{ ,RETRY=}
{ ,RETRYL=}
{ ,RETRYT=}
{ ,ROOM=}
{ ,SYSAREA=}
{ ,TDQMSG=}
{ ,TDQTRC=}
{ ,TITLE=}
{ ,TRC=}
{ ,UCS=}
{ ,USEROPTS=}
{ ,USRDTA01=}
{ ,USRDTA02=}
{ ,USRDTA03=}
{ ,USRDTA04=}
{ ,USRDTA05=}
{ ,USRDTA06=}
{ ,USRDTA07=}
( ,USRDTA08=}
{ ,USRDTA09=}
{ ,USRDTA10=}
{ ,USRDTA11=}
{ ,USRDTA12=}
{ ,USRDTA13=}
{ ,USRDTA14=}
{ ,USRDTA15=}
{ ,USRDTA16=}
{ ,USRLIB01=}
{ ,USRLIB02=}
{ ,USRLIB03=}
{ ,USRLIB04=}
{ ,USRLIB05=}
{ ,USRLIB06=}
{ ,USRLIB07=}
{ ,USRLIB08=}
{ ,WAIT=}
{ ,WRITER=}
END
```

---

Brackets, {}, are used to enclose optional parameters, which may or may not be specified. If a parameter is not coded, the default for that parameter will be used.

The following are operational parameters which control the DRS/SAPR2 interface processing:

**ERRACTN=**  
**ERRCLASS=**  
**ERRDEST=**  
**FFOPT=**  
**MAXLRECL=**  
**RETRY=**  
**TDQMSG=**  
**TDQTRC=**  
**USEROPTS=**  
**WAIT=**

The @OUTDESC keyword allows 60 bytes of text to be added as a comment to the DRS control block which represents the OUTPUT statement (DROB). For more information about the DROB, see [“DRS/API Request Calls” on page 21.1](#).

All of the remaining parameters refer to SYSOUT characteristics which could be specified on a SYSOUT DD statement or an OUTPUT JCL statement. For a complete description of these parameters, refer to the MVS JCL Reference Manual.

**Note:** A value of '\*' can be specified in the DEST, FORMS, GROUPID, NODE or WRITER parameters to indicate that the current SAP R2 printer name should be inserted into that field. Optionally, an eight character SAP R2 printer alias can be substituted. Refer to [“Building the DRS/SAPR2 Printer Alias Table” on page 27.13](#) and to [“Using the Printer Alias Table” on page 28.8](#) for more details about the printer alias table.

---

The parameters of the \$DRSSDEF macro are explained below:

<b>ADDR1=</b>	A 60-byte field specifying the first address to be printed on separator pages.
<b>ADDR2=</b>	A 60-byte field specifying the second address to be printed on separator pages.
<b>ADDR3=</b>	A 60-byte field specifying the third address to be printed on separator pages.
<b>ADDR4=</b>	A 60-byte field specifying the fourth address to be printed on separator pages.
<b>BUILDING=</b>	A 60-byte field specifying the building to be printed on separator pages.
<b>BURST=</b>	A 1-byte field which directs output to a stacker on a 3800 Printing Subsystem. 'B' = burster-trimmer-stacker; 'C' = continuous forms stacker.
<b>CHARS=</b>	Four 4-byte fields specifying the names of character arrangement tables to be used.
<b>CKPTLINE=</b>	Checkpoint lines. Valid values are 0-32767.
<b>CKPTPAGE=</b>	Checkpoint pages. Valid values are 0-32767.
<b>CKPTSEC=</b>	Checkpoint seconds. Valid values are 0-32767.
<b>CLASS=</b>	A 1-byte field specifying the SYSOUT class. Valid values are 0-9 or A-Z.
<b>COLORMAP=</b>	An 8-byte field specifying the AFP resource for the print file which contains color translation information
<b>COMPACT=</b>	An 8-byte field specifying the name of a compaction table for JES to use.
<b>COMSETUP=</b>	An 8-byte field specifying the APF resource for the print file which contains setup information.
<b>CONTROL=</b>	An 8-byte field specifying the type of spacing to be applied to the SYSOUT dataset. Valid values consist of "SINGLE", "DOUBLE", "TRIPLE" or "PROGRAM".
<b>COPIES=</b>	The copies or copy groups to be used to print the field. One copies field or up to 8 copy group values may be specified. Valid values are 0-255.
<b>DATAACK=</b>	An 8-byte field specifying action to be taken for AFP print errors; valid values are "BLOCK", "UNBLOCK", "BLKCHAR" or "BLKPOS".
<b>DEPT=</b>	A 60-byte field specifying the department to be printed on separator pages.
<b>DEST=</b>	An 8-character destination or userid. The DEST value may contain an '*' at any point to indicate that the current printer id should be inserted.
<b>DEST (LONG)=</b>	A 127-byte field specifying the destination for the SYSOUT dataset. The longer destination field is intended for a TCP/IP address in the format "IP:xxxxxxx" or "node.IP:xxxxxxx". This field and the 17-byte DEST field are mutually exclusive.
<b>DPAGELBL=</b>	A 1-byte field specifying whether the security label should be printed on each page. Valid values are "Y" or "N".



---

<b>DUPLEX=</b>	An 8-byte field specifying whether printing is to be done on both sides of the sheet. Valid values consist of “NO”, “NORMAL” or “TUMBLE”.
<b>ERRACTN=</b>	The action to be taken for any report in error, including reports with SYSOUT parameters that are invalid and reports which are incomplete. Valid values are “KEEP”, “HOLD” or “DELETE”. The default is “KEEP”.
<b>ERRCLASS=</b>	The SYSOUT class to be specified on the TERM call if the report is in error or is incomplete.
<b>ERRDEST=</b>	The destination to be specified on the TERM call if the report is in error or is incomplete.
<b>FCB=</b>	A 4-byte field containing the FCB name.
<b>FFOPT=</b>	The action to be taken if a form feed is encountered at the beginning of a report. Valid values are “KEEP”, which indicates to keep the form feed character in the report; “DELETE”, which indicates to remove the form feed character; and “SAP”, which indicates to use the option specified in the SAP R2 tables to keep or remove the form feed character. The default is “SAP”.
<b>FLASH=</b>	Two fields to represent the forms overlay name and the number of copies on which the forms overlay is to be printed. Valid values for the flash name include any 4 character flash name. Valid values for the flash count are 0 to 255.
<b>FORMDEF=</b>	A 6-byte field representing the name of a FORMDEF for AFP processing.
<b>FORMLEN=</b>	A 10-byte field specifying the numeric length and unit type that will be used to change the physical paper length without reconfiguring the printer. Valid values are nn.nnnUU, where n is a digit 0-9, and UU represents one of the following units: IN (inches) or CM (centimeters).
<b>FORMS=</b>	An 8-byte field to specify the form name to be used for this dataset. The FORMS value may contain an ‘*’ at any point to indicate that the current printer id should be inserted.
<b>GROUPID=</b>	An 8-byte field to specify the name of an OUTPUT GROUP to which this dataset belongs. The GROUPID value may contain an ‘*’ at any point to indicate that the current printer id should be inserted.
<b>INDEX=</b>	A field specifying the left margin on a printer with an indexing feature. Valid values are 1 to 31.
<b>INTRAY=</b>	A 4-byte binary field specifying the paper source when printing AFP files. Valid values are 1- 255.
<b>LINDEX=</b>	A field specifying the right margin on a printer with the indexing feature. Valid values are 1 to 31.
<b>LINECT=</b>	A field specifying the maximum number of lines JES is to print on each output page. Valid values are 0 to 255.
<b>MAXLRECL=</b>	Specifies the maximum length of print data to be added to a report using the DRS/SAPR2 interface. The value does not include the carriage control byte, which is always present. Lines which are longer will be truncated, unless USEROPTS=20 is specified to stop processing for lines which exceed the MAXLRECL. Valid values are 511 to 32755.

---

---

**MODIFY=** Two values which specify the 4-byte name of a copy modification module and the 1-byte field which indicates which table name in the CHARS parameter should be used. Valid values for the first field are any 1-4 character name of a copy modification module. Valid values for the second field are 0-3.

**NAME=** A 60-byte field to specify the name to be printed on the separator pages.

**NODE=** An 8-byte field specifying the JES Node name to be used. The NODE value may contain an "\*" at any point to indicate that the current printer id should be inserted.

**NOTIFY=** A 17-byte field specifying the first user to be notified when a job completes

**NOTIFY2=** A 17-byte field specifying the second user to be notified when a job completes

**NOTIFY3=** A 17-byte field specifying the third user to be notified when a job completes

**NOTIFY4=** A 17-byte field specifying the fourth user to be notified when a job completes

**OFFSETXB=** A 13-byte field specifying the offset in the x direction from the page origin for the back side of each page of output. Valid values are mmm.nnnUU, where m is a digit from 0 - 9, n is a digit from 0 - 9 and UU represents one of the following units: IN (inches), CM (centimeters), MM (millimeters), PELS or POINTS

**OFFSETXF=** A 13-byte field specifying the offset in the x direction from the page origin for the front side of each page of output. Valid values are mmm.nnnUU, where m is a digit from 0 - 9, n is a digit from 0 - 9 and UU represents one of the following units: IN (inches), CM (centimeters), MM (millimeters), PELS or POINTS

**OFFSETYB=** A 13-byte field specifying the offset in the y direction from the page origin for the back side of each page of output. Valid values are mmm.nnnUU, where m is a digit from 0 - 9, n is a digit from 0 - 9 and UU represents one of the following units: IN (inches), CM (centimeters), MM (millimeters), PELS or POINTS

**OFFSETYF=** A 13-byte field specifying the offset in the y direction from the page origin for the front side of each page of output. Valid values are mmm.nnnUU, where m is a digit from 0 - 9, n is a digit from 0 - 9 and UU represents one of the following units: IN (inches), CM (centimeters), MM (millimeters), PELS or POINTS

**OUTBIN=** A 4-byte binary field specifying the printer output bin identifier to be used. Valid values are 1 to 65535.

**OUTDISP=** Two fields which represent the output disposition when the job ends NORMALLY or ABNORMALLY. Valid values are WRITE, HOLD, KEEP, LEAVE and PURGE.

**OVERLAYB=** An 8-byte field specifying that the named medium overlay is to be placed on the back side of each sheet to be printed.

**OVERLAYF=** An 8-byte field specifying that the named medium overlay is to be placed on the front side of each sheet to be printed.

**OVFL=** An 8-byte field specifying whether or not JES3 should test for page overflow on an output printer. (JES3 only) Valid values are "ON" or "OFF".

**PAGEDEF=** A 6-byte field specifying the name of a PAGEDEF member to be used for AFP processing.

---

<b>PIMSG=</b>	Two fields which specify whether messages should be printed on the output listing during AFP processing, and the number of errors which should be printed. Valid values for the first field are “Y” or “N”; valid values for the second field are 0 to 999.
<b>PORTNO=</b>	A 4-byte binary field specifying the TCP/IP port number at which the printing application connects to the printer. Valid values are 1 - 65535.
<b>PRMODE=</b>	An 8-byte field specifying the process mode required to print the dataset. Valid values are “LINE”, “PAGE” or any valid installation-defined process mode.
<b>PRTEROR=</b>	An 8-byte field specifying how a SYSOUT data set that has had printing terminated by a functional subsystem is to be released by JES. Valid values are “DEFAULT”, “HOLD” or “QUIT”.
<b>PRTOPTNS=</b>	An 16- byte field specifying the named entity that contains additional print options for an IP-destined data set that is being sent by a functional subsystem.
<b>PRTQUEUE=</b>	A 127-byte field specifying the print queue name used when printing the IP-destined data
<b>PRTY=</b>	Specifies the initial priority at which the SYSOUT dataset enters the output queue. Valid values are 0 to 255.
<b>RESFMT=</b>	An 8-byte field specifying the resolution used to format the print data set. Valid values are “P240” or “P300”.
<b>RETAINF=</b>	A 10-byte field specifying how long a functional subsystem will retain an IP-destined data set after a failed transmission.
<b>RETAINS=</b>	A 10 byte field specifying how long a functional subsystem will retain an IP-destined data set after a successful transmission.
<b>RETRY=</b>	Specifies the number of times a CICS request will be made to read additional data for an incomplete report when the previous read indicated that the queue was empty. Valid values are 1 to 10; the default is 5. Between retry attempts, DRS/SAPR2 will wait for the time specified on the WAIT keyword.
<b>RETRYL=</b>	A 4-byte binary field specifying the number of attempts an FSS will try for transmission of an IP-destined data set. Valid values are 0 -32767.
<b>RETRYT=</b>	A 10-byte field specifying how much time a functional subsystem will wait between retries of transmission attempts of a dataset.
<b>ROOM=</b>	A 60-byte field specifying the room identification to be printed on separator pages.
<b>SYSAREA=</b>	Specifies whether or not the system should reserve an area for the security label on each page of printed output. Valid values are “Y” or “N”.
<b>TDQMSG=</b>	The 4-character name of the TD queue to be used to write the DRS/SAPR2 messages. If this queue is not available, the CSMT queue will be used.
<b>TDQTRC=</b>	The 4-character name of the TD queue to be used to write the DRS/SAPR2 messages. If this queue is not available, the CSMT queue will be used.
<b>TITLE=</b>	A 60-byte field specifying the title to be printed on separator pages.
<b>TRC=</b>	A 1-byte field specifying whether or not the SYSOUT dataset has a TRC character in the second character of each record. Valid values are “Y” or “N”.
<b>UCS=</b>	A 4-byte field specifying a universal character set, print train, or character-arrangement name.

---

---

**USEROPTS=** An 8-byte field which represents a 4-byte hex value with flags for special processing options.

80000000 Do not issue informational messages.  
40000000 Continue processing after warning messages for  
SYSOUT keyword parsing  
20000000 Stop processing if record exceeds MAXLRECL  
10000000 Do not create a DRS SNAP for processing errors

All others are currently unassigned.

**USRDTA01=** Sixteen 60-byte fields containing user data values.

**USRDTA02=**

**USRDTA03=**

**USRDTA04=**

**USRDTA05=**

**USRDTA06=**

**USRDTA07=**

**USRDTA08=**

**USRDTA09=**

**USRDTA10=**

**USRDTA11=**

**USRDTA12=**

**USRDTA13=**

**USRDTA13=**

**USRDTA15=**

**USRDTA16=**

**USRLIB01=** Eight 44-byte fields containing the dataset name(s) of user libraries for AFP processing.

**USRLIB02=**

**USRLIB03=**

**USRLIB04=**

**USRLIB05=**

**USRLIB06=**

**USRLIB07=**

**USRLIB08=**

**WAIT=** Specifies the time in seconds to wait between retries to obtain additional print lines for an incomplete report if the previous read indicated that the CICS queue was empty. Valid values are 1 to 60; the default is 3 seconds. The number of times to retry is specified using the RETRY keyword.

**WRITER=** An 8-byte field specifying the writer name. The WRITER value may contain an '\*' at any point to indicate that the current printer id should be inserted.

---

## Building the DRS/SAPR2 Printer Alias Table

The DRS/SAPR2 print interface uses an assembled module named DPSSPTAB to define the alias names for SAP R2 printers. This optional table allows an 8-character name to be used to replace the 4-character name defined in CICS for the SAP R2 print queue. This 8-character name will be substituted into the SYSOUT fields containing “\*”. The fields which allow such substitution are: DEST, FORMS, GROUPID, NODE and WRITER. If no printer alias table is found, or if the 4-character SAP R2 printer name is not located in the printer alias table, the 4-character name will be used for any substitution requested.

To define a DRS/SAPR2 printer alias table, perform the following steps:

1. **Modify member DPSSPTAB in library LRS.DRS.V1R33.ASM, specifying the values you want to use for the printer name and its alias.**
2. **Create the printer alias module named DPSSPTAB by assembling and linking your modified version of the DPSSPTAB member into a CICS DFHRPL library. Sample JCL to assemble and link the DRS/SAPR2 printer alias module is contained in member PTABASML or PTABASMH in dataset LRS.DRS.V1R33.CNTL.**

The keywords which are used to define the printer alias table are described on the following page.

The following is a sample DRSSPTAB source member:

```
$DRSPTAB  SAPID=PRT1,ALIAS=RMT99
$DRSPTAB  SAPID=PRT2,ALIAS=U4087
$DRSPTAB  SAPID=PRT3,ALIAS=LOCAL
END
```

---

## DRS/SAPR2 Printer Alias Table

```
b           One or more blanks must precede $DRSPTAB

$DRSPTAB

  SAPID=
  ,ALIAS=
```

The parameters of the \$DRSPTAB macro are explained below:

**SAPID=** The 1 to 4 character name of the printer in the SAP R2 CICS queue.

**ALIAS=** The 1 to 8 character name of the alias to be used for substitution, when an "\*" is located in the DEST, FORMS, GROUPID, NODE or WRITER SYSOUT keywords.

The \$DRSPTAB macro is used to define each printer for which an alias is desired.

For more information on using the printer alias table, see ["Using the Printer Alias Table" on page 28.8](#).

---

## Updating the CICS Tables

The DRS/SAPR2 Print Interface program (DP33SAPP), the DRS/SAPR2 defaults module (DPSSSDEF) and the optional printer alias module (DPSSPTAB) must be defined to CICS via PPT and PCT entries. You can add the required definitions using RDO or standard CICS tables. If you choose to use RDO, sample JCL is supplied in member SAPRDO in dataset LRS.DRS.V1R33.CNTL.

The transaction defined which is related to program DP33SAPP must have the same name as the TRANSID value on the DCT entry defined for the print queue. See [“Defining the Printer TD Queue” on page 27.18](#).

The following RDO or macro definitions are required:

Add to the PPT & PCT using RDO:

```
DEFINE PROGRAM(DP33SAPP) GROUP(gggg) LANG(ASSEMBLER)
DEFINE PROGRAM(DPSSSDEF) GROUP(gggg) LANG(ASSEMBLER)
DEFINE PROGRAM(DPSSPTAB) GROUP(gggg) LANG(ASSEMBLER)

DEFINE TRANSACTION(DRSP) GROUP(gggg)
PROGRAM(DP33SAPP)

ADD GROUP(gggg) LIST(1111)
```

or, add to the PPT using macro definitions:

```
DFHPPT TYPE=ENTRY,      DRS/SAPR2 interface program  *
PROGRAM=DP33SAPP,      *
PGMLANG=ASSEMBLER

DFHPPT TYPE=ENTRY,      DRS/SAPR2 defaults           *
PROGRAM=DPSSSDEF,      *
PGMLANG=ASSEMBLER

DFHPPT TYPE=ENTRY,      DRS/SAPR2 Printer alias table *
PROGRAM=DPSSPTAB,      *
PGMLANG=ASSEMBLER
```

And to the PCT using macro definitions:

```
DFHPCT TYPE=ENTRY,      DRS/SAPR2 Print interface    *
TRANSID=DRSP,          *
PROGRAM=DP33SAPP
```

In addition to the programs and transactions, the message queues for DRS/SAPR2 must be defined to CICS. If the TDQMSG and TDQTRC keywords in the defaults module are set to names for queues that are already defined to CICS, then no additional definitions would be necessary. However, it is recommended that unique queues be defined to CICS for the DRS/SAPR2 messages and for the DRS/SAPR2 trace datasets.

Assuming that TDQMSG=DPMS and TDQTRC=DPTR were specified, the DCT entries should be defined as:

```

DPSSMSG  DFHDCT  TYPE=SDSCI,          OUTPUT DATASET FOR DRS/SAPR2 MSGS  X
          BLKSIZE=136,              LRECL + RDW                        X
          BUFNO=1,                  WRITE OUT EACH BUFFER              X
          DSCNAME=DPSSMSG,          SYSOUT DDNAME                       X
          RECFORM=VARUNB,           VARIABLE UNBLOCKED                 X
          RECSIZE=132,              LRECL                               X
          TYPEFLE=OUTPUT
DPSSTRC  DFHDCT  TYPE=SDSCI,          OUTPUT DATASET FOR DRS/SAPR2 TRACE  X
          BLKSIZE=136,              LRECL + RDW                        X
          BUFNO=1,                  WRITE OUT EACH BUFFER              X
          DSCNAME=DPSSTRC,          SYSOUT DDNAME                       X
          RECFORM=VARUNB,           VARIABLE UNBLOCKED                 X
          RECSIZE=132,              LRECL                               X
          TYPEFLE=OUTPUT
DPMS     DFHDCT  TYPE=EXTRA,          DRS/SAPR2 MESSAGES                 X
          DESTID=DPMS,              MUST MATCH TDQMSG= IN DPSSDEF      X
          DSCNAME=DPSSMSG
DPTR     DFHDCT  TYPE=EXTRA,          DRS/SAPR2 TRACE MESSAGES           X
          DESTID=DPTR,              MUST MATCH TDQTRC= IN DPSSDEF      X
          DSCNAME=DPSSTRC

```

The DSCNAME for the DPMS queue must match the DSCNAME (DDNAME) for the DPSSMSG DCT entry. The DSCNAME for the DPTR queue must match the DSCNAME (DDNAME) for the DPSSTRC DCT entry. Both DDNAMES must be added to the JCL for the CICS system. Examples of the necessary JCL statements follow:

```

//DPSSMSG DD  SYSOUT=A,DCB=(DSORG=PS,RECFM=V,BLKSIZE=136)
//DPSSTRC DD  SYSOUT=A,DCB=(DSORG=PS,RECFM=V,BLKSIZE=136)

```



---

## Copying the DRS/SAPR2 Modules for CICS

The DRS/SAPR2 load library must be added to the CICS DFHRPL concatenation or the following modules copied to a library in the DFHRPL concatenation:

**DP33SAPP**

**DPSSSDEF**

**DPSSPTAB** (optional)

In addition, the DRS/API modules must be available in the DFHRPL libraries. See [“DRS/API Installation” on page 20.1](#) for information on which DRS/API modules are necessary for using DRS in CICS.

---

## Defining the Printer TD Queue

Each printer defined to SAP R2 requires a Transient Data (TD) queue to be defined to CICS. For SAP R2 printers that will use the DRS/SAPR2 print interface, a DCT entry as described below is required.

```
DFHDCT TYPE=INTRA,          DRS/SAPR2 Printer queue      *
DESTID=nnnn,              See note 1                    *
DESTFAC=FILE,             *
DESTRCV=LG,               *
TRANSID=DRSP,            See note 2                    *
TRIGLEV=1
```

**Note 1:** nnnn should be replaced with the printer name defined in SAP R2 table 022B.

**Note 2:** The TRANSID value must match the TRANSACTION defined for PROGRAM DP33SAPP.

**Note 3:** After adding the above entry, it is necessary to cold start CICS for the change to come into effect.

---

## Defining the Printer to SAP R2

The new printer must now be defined to SAP R2 via table 022B.

SAP R2 table 022B contains an entry for each printer defined to the SAP R2 system and the associated DCT entry. It can also contain a printer type which refers to SAP R2 table 022D. The following sample shows defining a printer named “PRT1” with a printer type “DRS”.

**Table 022B**

<u>Dest</u>	<u>R</u>	<u>DCT</u>	<u>022D-Type</u>	<u>Prolog</u>	<u>Epilog</u>	<u>LF</u>	<u>UPC</u>	<u>FFB</u>	<u>FFA</u>	<u>Comment</u>
PRT1	R	PRT1	DRS					X	X	Printer PRT1

The name specified for “DCT” should match the DESTID in the DCT definition in CICS as described in the previous section.

The DRS/SAPR2 Print Interface is now ready to use. Any output sent to this SAP R2 printer (PRT1) will now be sent directly to the DRS/SAPR2 interface, which will place the output on the JES spool using the default SYSOUT attributes defined in the output defaults module DPSSSDEF.

Individual SYSOUT attributes for specific printers or for specific reports can be specified via SAP R2 table 022D, which is described in the [“Setting SYSOUT Attributes With DRS/SAPR2” on page 28.1.](#)



---

## Section 28

# Setting SYSOUT Attributes With DRS/SAPR2

The DRS/SAPR2 Print Interface provides facilities to specify JES SYSOUT attributes at three levels:

- System defaults
- Printer defaults
- Specific report attributes

When assigning attributes to a report, the *SYSOUT* options are processed in the above order; options specified at the individual report level override options specified at the printer or system level.

---

## DRS/SAPR2 System Defaults

As described in the [“DRS/SAPR2 Installation” on page 27.1](#), the global defaults for all printers in a CICS region with SAP R2 are defined via the DPSSSDEF defaults module. It is possible to define different system defaults for individual SAP R2 systems by creating separate DPSSSDEF modules to be used by each SAP R2/CICS system. In this way, it is possible to have different default SYSOUT attributes for test and production SAP R2/CICS systems.

---

## Specifying Printer and Report SYSOUT Attributes

The SYSOUT attributes specified at the printer and report level are controlled via parameter strings defined in SAP R2 table 022D. This standard SAP R2 table is normally used to define printer command sequences to enable SAP R2 applications to control printer options. With the DRS/SAPR2 interface, this facility is used to control the JES SYSOUT options.

**Table 022D**

Type	Format	Seq	L	S	X	D	Format string	Comment
DRS	LAND			1			`*DRS* C=A FD=LAND PD=LAND <'	
DRS	PORT			1			`*DRS* C=A FD=PORT PD=PORT <'	
HPLASER	PORT			1			`1B&100	
HPLASER	LAND			1			`1B&110	

The SAP R2 022D table contains a printer type, a format name, and the command string to be sent.

The type allows defining groups of commands for a particular type of printer or interface. For example, "HPLASER" may be the type for HP/PCL commands that would be sent by SAP R2 to the printer. "PS" might be the type for postscript commands. "DRS" might be used as the type for DRS/SAPR2 SYSOUT keywords.

A format name is used when selecting a specific command string.

A command string can contain up to 52 bytes of data. By using the sequence field (Seq), it is possible to assign several strings to a single format name. For more information on the format of the command strings, see ["Defining DRS/SAPR2 Command Strings" on page 28.9](#).

---

# Printer Defaults

Default SYSOUT attributes can be defined at the individual printer level by specifying the name of the command string in the Prolog field of the SAP R2 printer table 022B.

**Table 022B**

<u>Dest</u>	<u>R</u>	<u>DCT</u>	<u>022D-Type</u>	<u>Prolog</u>	<u>Epilog</u>	<u>LF</u>	<u>UPC</u>	<u>FFB</u>	<u>FFA</u>	<u>Comment</u>
PRT1	R	PRT1	DRS	LAND				X	X	Printer PRT1

**Table 022D**

<u>Type</u>	<u>Format</u>	<u>Seq</u>	<u>L</u>	<u>S</u>	<u>X</u>	<u>D</u>	<u>Format string</u>	<u>Comment</u>
DRS	LAND			1			`*DRS* CLASS=A FORMDEF=LAND PAGEDEF=LAND <'	

The 022D-Type field in the SAP R2 printer table 022B indicates which group of commands should be selected from the format table 022D, and the Prolog field selects a specific command string to be used as the default for this printer.



---

## Report SYSOUT Attributes

The SYSOUT attributes for individual reports can be controlled in two ways, either by explicitly selecting the output options from within the SAP R2 application program (ABAP) that creates the report or by allowing the user to select a different printer name to select a different set of output options.

An individual ABAP can select specific output options for a report using the PRINT-CONTROL statement. This statement allows an ABAP to select one or more command strings from table 022D. The command strings selected via the PRINT-CONTROL statement override defaults specified at the system or printer level.

The format of the PRINT-CONTROL statement is:

**PRINT-CONTROL: FUNCTION: 'LAND'**

The above statement selects the output options defined in the 'LAND' command string in table 022D.

**Table 022B**

Dest	R	DCT	022D-Type	Prolog	Epilog	LF	UPC	FFB	FFA	Comment
PRT1	R	PRT1	DRS	PORT				X	X	Printer PRT1

**Table 022D**

Type	Format	Seq	L	S	X	D	Format string	Comment
DRS	PORT			1			'*DRS* CLASS=A FORMDEF=PORT PAGEDEF=PORT <'	
DRS	LAND			1			'*DRS* CLASS=A FORMDEF=LAND PAGEDEF=LAND <'	

### ABAP Program

```
PRINT-CONTROL:FUNCTION: 'LAND'  
  
WRITE /text
```

In the above example, printer 'PRT1' has a default command string of 'PORT' which selects PAGEDEF and FORMDEF of 'PORT', but the ABAP overrides the printer defaults by selecting command string 'LAND'.

The output options selected via the PRINT-CONTROL statement remain in effect until the end of the report or until another PRINT-CONTROL statement is used to select different options. Every time the output options are changed using the PRINT-CONTROL statement, the current SYSOUT dataset is closed and a new dataset with the modified output options is allocated.

The PRINT-CONTROL statement can be used to select more than one command string from table 022D. In that case, the parameter options from both strings are combined.

**Table 022D**

Type	Format	Seq	L	S	X	D	Format string	Comment
DRS	LAND			1			`*DRS* CLASS=A FORMDEF=LAND PAGEDEF=LAND <'	
DRS	ADDR			1			`*DRS* ADDR1="226 Berwick Avenue" <'	

### ABAP Program

```

PRINT-CONTROL: FUNCTION: 'LAND' , 'ADDR'

WRITE /text

```

When selecting more than one control string using the PRINT-CONTROL statement, SAP R2 applies special significance to the first two characters of the name. If the first two characters of the command name are the same, then only the information from the last string will be used. SAP R2 assumes both strings are of the same type so the second string overrides the first.

**PRINT-CONTROL: FUNCTION: 'AA010' , 'AA020' ← Only last will be used.**

**PRINT-CONTROL: FUNCTION: '010AA' , '020AA' ← Both command strings will be used.**

The first statement above would only select command string AA020; the second statement would correctly select both command strings.

An alternative way to allow the user to select different output options is to define the same printer multiple times with different names and assign different default output options to each. The user can then select output options by specifying the appropriate printer name.

### Multiple Printer Names

Dest	R	DCT	022D-Type	Prolog	Epilog	LF	UPC	FFB	FFA	Comment
PR1	R	PR1	DRS					X	X	Printer PR1
PR1L	R	PR1	DRS	LAND				X	X	Landscape
PR1P	R	PR1	DRS	PORT				X	X	Portrait

When using this technique, the different SAP R2 printer names should all be associated with a single Transient Data queue (DCT entry).

**Note:** When processing the '\*' character in the DEST, FORMS, GROUPID, NODE and WRITER parameters, the TD Queue name is inserted and not the SAP R2 printer name.

---

## Establishing User Defaults

SAP R2 provides a standard facility to assign a default printer for a user in the profile record. This information is normally administered via SAP R2 transaction TMU5 illustrated below:

<u>User Defaults</u>	<u>USERID</u>
Start transaction	-
Transaction menu	-
Automatic start	-
Printer	PRT1
Printer parameters	N G D
Control totals	-
Date format	- DD.MM.YYYY - MM/DD/YYYY - MM-DD-YYYY X YYYY.MM.DD
Decimal indicator	X Comma - Point
ABAP/4 job prefix	-
OK _	PF: 3=Back 13=Save

### FIELD

### DESCRIPTION

#### Printer

SAP R2 printer name from table 022B

#### Printer Parameters

1. L - Lists are stored in the SAP R2 spool file  
N - Lists are not stored
2. G - Print immediately  
H - Hold in SAP R2 spool file until released
3. D - Delete from SAP R2 spool file after printing  
K - Keep in SAP R2 spool file after printer

#### Example:

**N G D:** No intermediate storage, print immediately, delete after printing

---

## Using the Printer Alias Table

The SAP R2 printer alias table is an optional feature that provides a facility to translate a four character SAP R2 printer name into a 1-8 character printer alias. This printer alias can then be used in the DEST, FORMS, GROUPID, NODE or WRITER fields when DRS/SAPR2 is allocating a SYSOUT dataset. This is done by specifying an '\*' in the appropriate field to indicate that the TD Queue name (normally the same as the SAP R2 printer name) should be inserted. For example:

**DEST=\***                    **or**                    **DEST=PRT\***

The four character SAP R2 printer name is simply inserted starting at the position indicated by the '\*'. The DEST attribute for the above examples, assuming a SAP R2 printer name of 'SAP1', would be:

**DEST=SAP1**            **or**                    **DEST=PRTSAP1**

If a printer alias table (DPSSPTAB) is available, and it contains an entry that matches the SAP R2 printer ID, then the printer alias will be substituted when processing these special fields. For example, assume the following entry was in DPSSPTAB:

**\$DRSPTAB SAPID=SAP1,ALIAS=RMT99**

Using these values, the DEST attribute for the above examples would be:

**DEST=RMT99**            **or**                    **DEST=PRTRMT99**

For more information on defining the printer alias table, see [“Building the DRS/SAPR2 Printer Alias Table” on page 27.13](#).

---

## Defining DRS/SAPR2 Command Strings

The command strings used by the DRS/SAPR2 print interface consist of a command identifier ‘\*DRS\*’ followed by one or more SYSOUT option parameters and terminated with a single less than character (<).

### Example

Type	Format	Seq	L	S	X	D	Format string	Comment
DRS	LAND			1			`*DRS* CLASS=A FORMDEF=LAND PAGEDEF=LAND <`	

**Note:** When entering the command string in table 022D, the entire string must be enclosed by apostrophes (single quotes).

The DRS/SAPR2 SYSOUT parameters consist of an output keyword followed by an equal sign (=) and then a value to assign to the output option. Each parameter has a shorter alias name which can be used to reduce the length of the parameter string. For example:

**PAGEDEF=LAND**                      **or**  
**PD=LAND**

Parameter values can be character, numeric or Y/N. Character values containing blanks must be enclosed in quotation marks. For example:

**BUILDING=“Technical support office”**

Any output option which has been defined at a higher level can be reset by assigning a null value. For example, the following example will reset the FCB name to blanks:

**FCB=**

Multiple strings of \*DRS\* parameters can be specified by using the sequence (Seq) field.

### Example

Type	Format	Seq	L	S	X	D	Format string	Comment
DRS	LAND	1					`*DRS* CLASS=A FORMDEF=LAND PAGEDEF=LAND <`	
DRS	LAND	2					`*DRS* NODE=N1 DEST=* <`	

The SYSOUT parameters supported by the DRS/SAPR2 interface include all the attributes which can be specified on a SYSOUT DD statement or on an OUTPUT JCL statement. For a complete list of the SYSOUT keywords, please refer to “DRS/SAPR2 SYSOUT Parameters” beginning on the following page.

## DRS/SAPR2 SYSOUT Parameters

Keyword	Alias	Description
ADDRn n = 1 to 4	ADn	Delivery address to be printed on separator pages. Up to four address lines can be specified each containing 1 - 60 characters. If the address contains blanks, then the string must be enclosed in quotation marks.
BUILDING	BLD	Building name to be printed on separator pages. Building name can be 1 - 60 characters and must be enclosed in quotation marks if the value contains blanks.
BURST	BST	Indicates processing options for a 3800 printer. "B" indicates the output should be processed through the burster-trimmer-stacker feature or "C" indicates continuous forms.
CHARSn n = 1 to 4	CHn	One to four character arrangement table names. Each character arrangement table name can be 1 - 4 characters.
CKPTLINE	CKL	The checkpoint line parameter in combination with the checkpoint page parameter controls the frequency of checkpoints for printing the document. Valid values are 0 - 32767.
CKPTPAGE	CKP	The checkpoint page parameter in combination with the checkpoint line parameter controls the frequency of checkpoints for printing the document. Valid values are 0 - 32767.
CKPTSEC	CKS	The checkpoint seconds parameter controls the frequency in seconds of checkpoints for printing the dataset. Valid values are 0 - 32767.
CLASS	C	Indicates the JES SYSOUT class. Valid values are A-Z or 0-9.
COLORMAP	CMAP	An 8-byte field specifying the AFP resource for the print file which contains color translation information.
COMPACT	CMP	Specifies the name of a compaction table. Must be 1 - 8 alphanumeric characters.
COMSETUP	CSET	An 8-byte field specifying the APF resource for the print file which contains setup information.
CONTROL	CTL	Specifies that each logical record starts with a carriage control character <b>or</b> that the output is to be printed with single, double, or triple spacing. Valid values are PROGRAM, SINGLE, DOUBLE, or TRIPLE.
COPIES	CPY	Specifies how many copies of the dataset are to be printed. Valid values are 0 - 255.
COPYGRPn n = 1 to 8	CGn	Specifies up to 8 copy group values for this dataset. Valid values are 0 - 255.
DATAACK	DCK	Specifies the action to take for errors during AFP processing. Valid values are BLOCK, UNBLOCK, BLKCHAR, or BLKPOS.

<b>Keyword</b>	<b>Alias</b>	<b>Description</b>
DEPT	DPT	Department name to be printed on separator pages. Value can be 1-60 characters and must be enclosed in quotation marks if it contains blanks.
DEST	D	Destination name. Value can be 1 - 8 characters.
DPAGELBL	DPL	Indicates whether the system should print the security label on each page of printed output. Valid values are Y or N.
DUPLEX	DUPX	An 8-byte field specifying whether printing is to be done on both sides of the sheet. Valid values consist of "NO", "NORMAL" or "TUMBLE".
FCB	FCB	Forms Control Buffer name. Value can be 1 to 4 characters.
FLASH	FLS	Specifies the forms overlay to be used in printing the dataset on a 3800 Printing Subsystem. Value can be 1 to 4 alphanumeric characters.
FLASHCNT	FCT	Specifies the number of copies on which the forms overlay is to be printed. Valid values are 0 to 255.
FORMDEF	FD	Indicates the name of the FORMDEF to be used for AFP processing. Valid values are 1 - 6 characters.
FORMLEN	FMLN	A 10-byte field specifying the numeric length and unit type that will be used to change the physical paper length without reconfiguring the printer. Valid values are nn.nnnUU, where n is a digit 0-9, and UU represents one of the following units: IN (inches) or CM (centimeters).
FORMS	F	Indicates the form name for the dataset. Value can be 1 to 8 alphanumeric characters.
GROUPID	GRP	Specifies the output group name for the dataset. Value can be 1-8 alphanumeric characters.
INDEX	IDX	The INDEX parameter sets the left margin when printing on a 3211 with the indexing feature. Valid values are 1 - 31.
LINDEX	LDX	The LINDEX parameter sets the right margin when printing on a 3211 with the indexing feature. Valid values are 1 - 31.
LINECT	LCT	The line count parameter specifies the number of lines to print per page when no carriage control characters are present. Valid values are 0 - 255.
MODIFY	MDF	Copy modification module name. Value can be 1 to 4 characters.
MODTRC	MTR	Specifies the table name in the CHARS parameter (0 for the first, 1 for the second, 2 for the third or 3 for the fourth).
NAME	NAM	Specifies the name to be printed on separator pages. Value can be 1 - 60 characters and must be enclosed in quotation marks if it contains blanks.

<b>Keyword</b>	<b>Alias</b>	<b>Description</b>
NODE	N	Specifies the JES Node name to be used to print the dataset. Value can be 1 to 8 characters.
NOTIFY	NTF	Indicates the name of a TSO user who should be notified when the output prints. Value can be 1 to 17 characters.
NOTIFY2	NTF2	Specifies the second userid to be notified when a job completes; the user's ID can be prefixed with a valid JES node name.
NOTIFY3	NTF3	Specifies the third userid to be notified when a job completes; the user's ID can be prefixed with a valid JES node name.
NOTIFY4	NTF4	Specifies the fourth userid to be notified when a job completes; the user's ID can be prefixed with a valid JES node name.
OFFSETXB	OFXB	A 13-byte field specifying the offset in the x direction from the page origin for the back side of each page of output. Valid values are mmm.nnnUU, where m is a digit from 0 - 9, n is a digit from 0 - 9 and UU represents one of the following units: IN (inches), CM (centimeters), MM (millimeters), PELS or POINTS.
OFFSETXF	OFXF	A 13-byte field specifying the offset in the x direction from the page origin for the front side of each page of output. Valid values are mmm.nnnUU, where m is a digit from 0 - 9, n is a digit from 0 - 9 and UU represents one of the following units: IN (inches), CM (centimeters), MM (millimeters), PELS or POINTS.
OFFSETYB	OFYB	A 13-byte field specifying the offset in the y direction from the page origin for the back side of each page of output. Valid values are mmm.nnnUU, where m is a digit from 0 - 9, n is a digit from 0 - 9 and UU represents one of the following units: IN (inches), CM (centimeters), MM (millimeters), PELS or POINTS.
OFFSETYF	OFYF	A 13-byte field specifying the offset in the y direction from the page origin for the front side of each page of output. Valid values are mmm.nnnUU, where m is a digit from 0 - 9, n is a digit from 0 - 9 and UU represents one of the following units: IN (inches), CM (centimeters), MM (millimeters), PELS or POINTS.
OUTBIN	BIN	Specifies the output bin identifier to be used for this dataset. Valid values are 1 to 65535.
OUTDISPA	ODA	Indicates the SYSOUT output disposition for abnormal completion. Valid values are WRITE, HOLD, KEEP, LEAVE or PURGE.
OUTDISPN	ODN	Indicates the SYSOUT output disposition for normal completion. Valid values are WRITE, HOLD, KEEP, LEAVE or PURGE.
OVERLAYB	OVLB	An 8-byte field specifying that the named medium overlay is to be placed on the back side of each sheet to be printed.
OVERLAYF	OVLB	An 8-byte field specifying that the named medium overlay is to be placed on the front side of each sheet to be printed.



<b>Keyword</b>	<b>Alias</b>	<b>Description</b>
OVFL	OVFL	An 8-byte field specifying whether or not JES3 should test for page overflow on an output printer. (JES3 only) Valid values are "ON" or "OFF".
PAGEDEF	PD	Specifies the PAGEDEF which should be used for AFP processing. Value can be 1 to 6 characters.
PIMSG	PIM	Specifies whether or not messages should be printed on the output listing when processing AFP data. Valid values are Y or N.
PIMSGCNT	PIC	Specifies the number of errors to cause printing of PIMSG to be terminated. Valid values are 0 to 999.
PORTNO	PORT	A 4-byte binary field specifying the TCP/IP port number at which the printing application connects to the printer. Valid values are 1 - 65535.
PRMODE	PRM	Indicates the processing mode required to print the dataset. Value can be 1 to 8 characters, such as PAGE, LINE or an installation-defined value.
PRTERORR	PTER	An 8-byte field specifying how a SYSOUT data set that has had printing terminated by a functional subsystem is to be released by JES. Valid values are "DEFAULT", "HOLD" or "QUIT".
PRTOPTNS	PRTO	An 16- byte field specifying the named entity that contains additional print options for an IP-destined data set that is being sent by a functional subsystem.
PRTQUEUE	PRTQ	A 127-byte field specifying the print queue name used when printing the IP-destined data set.
PRTY	PTY	Indicates the initial output priority of the SYSOUT dataset. Valid values are 0 - 255.
RESFMT	RFMT	An 8-byte field specifying the resolution used to format the print data set. Valid values are "P240" or "P300".
RETAINF	RETF	A 10-byte field specifying how long a functional subsystem will retain an IP-destined data set after a failed transmission.
RETAINS	RETS	A 10 byte field specifying how long a functional subsystem will retain an IP-destined data set after a successful transmission.
RETRYL	RETL	A 4-byte binary field specifying the number of attempts an FSS will try for transmission of an IP-destined data set. Valid values are 0 -32767.
RETRYT	RETT	A 10-byte field specifying how much time a functional subsystem will wait between retries of transmission attempts of a data set.
ROOM	ROM	Specifies the room name to be printed on separator pages. Value can be 1 to 60 characters and must be enclosed in quotation marks if it contains blanks.
SYSAREA	SYA	Indicates whether the system should reserve an area on each page for the security label. Valid values are Y or N.

---

<b>Keyword</b>	<b>Alias</b>	<b>Description</b>
TITLE	TTL	Specifies the title to be printed on separator pages. Value can be 1 to 60 characters and must be enclosed in quotation marks if it contains blanks.
TRC	TRC	Indicates whether the SYSOUT dataset contains Table Reference Characters. Valid values are Y or N.
UCS	UCS	Specifies the name of the Universal Character Set to be used when printing this dataset. Value can be 1 to 4 characters.
USRDTAnn nn = 01 to 16	UDnn	Specifies 1-60 bytes of user data. Up to 16 fields can be specified.
USRLIBnn nn = 01 to 08	ULnn	Indicates the names of 1 to 8 AFP resource libraries which should be concatenated ahead of the standard libraries. Value can be 1 to 44 characters.
WRITER	W	Specifies the writer name to be associated with this SYSOUT dataset. Value can be 1 to 8 alphanumeric characters.

---

# Section 29

## DRS/SAPR2 Problem Determination

This section contains hints about diagnosing problems with DRS/SAPR2. Any of the following may help in problem resolution:

- DRS/SAPR2 messages, including trace messages
- DRS/API return codes
- DRS/API SNAP dumps
- Copies of SAP R2 spool record data
- SYSUDUMP for any ABEND that may occur

If you find you need assistance to resolve any problem, the LRS technical support is always ready to help you.

### DRS/SAPR2 Messages

The DRS/SAPR2 messages are written to the CICS TD queue whose name was specified by the TDQMSG keyword in the DRS/SAPR2 defaults module (DPSSSDEF). If trace records were requested to be written to a message file, the DRSP070I messages will be sent to a CICS TD queue whose name was specified by the TDQTRC keyword. It is recommended that these queues be defined as JES SYSOUT files using DDNAMES of DPSSMSG and DPSSTRC. For more information on defining these TD queues to CICS, see [“Updating the CICS Tables” on page 20.14](#).

Browsing the DRS/SAPR2 message queue may help resolve a problem with invalid SYSOUT keywords or invalid keyword values. If LRS technical support requests additional tracing, the DRSP070I messages in the trace message file will help to determine what had happened up to the point of an error that was encountered.

All messages begin with “DRSP” and contain the date and time that the message was issued, along with the TD queue name associated with the printer. The messages are documented on page [35.78](#).

### DRS/API Return Codes

DRS/API return codes will be displayed in the DRS/SAPR2 messages. These return codes are documented on page [24.1](#) and will be of assistance in determining the cause of SYSOUT keyword errors.

### DRS/API SNAP Dumps

If a DRS/API error is encountered, a SNAP dump will be created using the SNAP parameters from the DRSSOPTS module. For more information about defining these SNAP options, see [“Customizing the DRS System Options” on page 20.9](#).

---

## Copies of SAPR2 Spool Record Data

If you are having problems with the DRS/SAPR2 interface, the DRS support staff may request a copy of the SAP R2 print data so that the problem can be re-created in a test environment and a resolution found.

To capture the SAP R2 spool data an additional External DCT queue definition is required which moves the print data directly to an external sequential dataset.

### DCT Entry

DRSDOUT	DFHDCT	TYPE=SDSCI,	Output dataset for SAP R2 data	X
		DSCNAME=DRSDOUT,		X
		RECFORM=VARBLK,		X
		TYPEFLE=OUTPUT		
DRSD	DFHDCT	TYPE=EXTRA,	DRS/SAPR2 Interface Print data	X
		DESTID=DRSD,		X
		DSCNAME=DRSDOUT		

### CICS JCL

//DRSDOUT	DD	DSN=dataset.name,DISP=SHR
-----------	----	---------------------------

The output dataset should be pre-allocated with the following attributes:

```
RECFM=VB
DSORG=PS
LRECL=32756
BLKSIZE=32760
```

To capture the SAP R2 print data, you can define a new printer to SAP R2 which points to this new DCT entry and print to it as normal. Or, if you have a DRS/SAPR2 interface printer which has failed for some reason, the print data will still be in the printer queue and can be retrieved and sent to this external destination using the CEBR transaction.

To retrieve print data from a failed DRS/SAPR2 printer queue, use the CICS supplied transaction CEBR as follows:

1. Enter transaction 'CEBR'
2. On the command line enter 'GET prtr' (where prtr is the failed printer queue name). This will retrieve the print data from this queue and place it into temporary storage and will also display the data on the screen.
3. Enter 'PUT DRSD' (where DRSD is the name of the external transient data queue defined above). This will write the information to the external sequential dataset.
4. Use CEMT to close the external TD queue using: CEMT S TDQ(DRSD) CLOSE
5. The sequential dataset will now contain the SAP R2 printer data.

The SAP R2 print data can now be sent to LRS technical support on tape or via e-mail.

**Note:** To download the SAP R2 print data to a PC file for delivery, the file should be transferred from the host as a BINARY transfer but with the CRLF option specified. This will enable LRS to reconstruct the original data when uploaded to our MVS system.

---

## **SYSUDUMP from ABEND**

If DRS/SAPR2 abnormally terminates, a system dump will be created. These dumps will be extremely valuable for resolving problems where a DRS/SAPR2 program has abended. DRS/SAPR2 has an internal trace table which will be helpful in determining which events have most recently occurred. In addition, the DRS/SAPR2 control blocks will be dumped.

When you have a system dump from DRS/SAPR2, you should call LRS technical support for assistance. Some problems of this type can be resolved in a phone conversation. In most cases, you will be asked to send the dump by mail or e-mail. Please call LRS technical support before sending any dump.



---

## Section 30

# Introduction to DRS/NATURAL

DRS/NATURAL has been developed in partnership with Software/AG to provide a direct interface from the Natural environment to the JES spooling subsystem. The DRS/NATURAL product has been developed using a standardized interface to Natural that has been designed by Software/AG to provide a stable and supported interface to the LRS range of print management products.

DRS/NATURAL provides a direct interface from Natural applications to the JES spool. It enables any existing Natural application running in CICS or BATCH to create a report on the JES spool without any application changes.

The DRS/NATURAL interface provides complete control over the JES spool attributes assigned to a report and enables existing reports to use the printing facilities of Advanced Function Presentation (AFP) to enhance the quality of printed output from Natural applications.

With the DRS/NATURAL interface, JES spool attributes can be specified at the system level as global defaults for the entire Natural environment, at the individual printer level, or can be specified for individual reports. The spool file attributes supported include the standard JES SYSOUT attributes and the extended attributes normally specified via a JCL OUTPUT statement. This is all done dynamically and requires no JCL changes.

The JES spool attributes assigned to a report are controlled using the standard print file attributes available in Natural, which are mapped dynamically to the required JES attributes. The DRS/NATURAL product has been designed to use only standard Natural interfaces and programming statements. This has been done to ensure compatibility with all future versions of Natural and to isolate Natural applications from the specifics of the JES spooling environment. This separation also allows for the central control and definition of spooling attributes and will enable Natural to exploit future enhancements to JES without the requirement to change the Natural environment or the underlying applications.

The DRS/NATURAL interface enables Natural applications to exploit the full power of the JES spooling subsystem and the LRS range of Enterprise Print Management products. The interface has been designed to be simple to install and is loaded dynamically by Natural at session initialization. The enhanced spooling facilities of DRS/NATURAL can be enabled at the individual session level which enables it to run in parallel with existing printing solutions and enable a phased migration and implementation.





---

# Section 31

## DRS/NATURAL Installation

### Introduction

This section gives you the information you need to install DRS/NATURAL on your system. The details of this procedure are listed in “Installation Steps” on the following page. Before you do the actual installation, you must install the DRS/API product. Please see [“Installation Steps” on page 20.4.](#)

---

## Installation Steps

The steps to install the DRS/NATURAL print interface are:

1. Read the entire installation procedure.
2. Install the DRS/API product. (See [“Installation Steps” on page 20.4.](#))
3. Check Natural product level. (See page [31.4.](#))
4. Linkedit DRS/NATURAL interface. (See page [31.5.](#))
5. Define the DRS/NATURAL modules to CICS. (See [“Updating the CICS Tables” on page 31.6.](#))
6. Define the DRS/NATURAL interface routine to Natural. (See [“Define DRS/NATURAL Interface to Natural” on page 31.8](#))
7. Define a Natural print and hardcopy file using the DRS/NATURAL interface. (See [“Define a Print and Hardcopy File Using the DRS/Natural Interface” on page 31.9.](#))
8. The DRS/NATURAL interface installation is now complete.

The remainder of this section provides the information required to perform the above steps.

---

## Installing the DRS/API Product

The DRS/NATURAL print interface uses the facilities of DRS/API to create output on the JES spool. Before installing the DRS/NATURAL interface, the DRS/API product must be installed and defined to CICS.

For information on installing DRS/API, please refer to [“Installation Steps” on page 20.4](#).

---

## Check Natural Product Level

The DRS/NATURAL interface requires Natural Version 2.3.2 or higher. In addition to the base product level the following fixes must be applied to ensure correct operation of the DRS/NATURAL interface.

Fix Number	Pre-Requisite	Symptoms
NA33180	NA32178	Backend program not called in case of a non-zero session return code.
NA33208		Provide logical printer number to exit.
NA33209		CPRTADR incorrectly set on CLOSE for print files.
NA33245		CPRTADR incorrectly set on CLOSE for print files.
NA33260		OPEN errors never get reset, file unusable after error.
NA33264	NA32189 NA32270	Global area not fully initialized in some environments.
NA33277		Default to OPEN=ACC and CLOSE=CMD if no OPEN and/or CLOSE options have been specified for an AM=USER file.
NA34013		Settings of PRINT profile subparameters CLASS, COPIES, DISP, FORMS, NAME, PROFILE and PRTY are cleared after the print file is closed.
NA34015		Refresh file name at close.
NA34018		Enable user access method to disable DEFINE PRINTER OUTPUT routing.
NA34021	NA34015	Print file 0 file name change by User Access method not reflected in hardcopy destination name in IOCB.
NA34031	NA33277	Enable user access method to disable DEFINE PRINTER OUTPUT routing.

The status of the above fixes can be checked using the Natural 'DUMP ZAPS' command which will list all zaps that have been applied. Any zaps that are not applied can be obtained from Software AG technical support or are available online via the SAGNET support site.

If these fixes are applied to the Natural distribution libraries then it will be necessary to re-link the Natural run time nucleus. Please refer to the Software AG documentation for details.

**WARNING: Please ensure that all zaps have been applied before continuing with the next step of the installation.**

---

## Linkedit DRS/NATURAL Interface

The DRS/NATURAL interface module must now be re-linked to incorporate the Natural Use Access Method routine (NATAMUSR). Sample JCL to re-link the DN33MAIN module is supplied in member NATLKED of dataset LRS.DRS.V1R33.CNTL.

The NATAMUSR module is supplied in the Natural load library from version 2.3.2 onwards.

Note: If any zaps are applied which update the NATAMUSR routine then the DRS/NATURAL module (DN33MAIN) must be re-linked to incorporate these changes.

```
//JOBNAME JOB (YOUR JOB CARD INFORMATION)
/**
/**-----*
/** NATLKED - SAMPLE JCL TO LINKEDIT THE DRS/NATURAL MODULE WITH *
/**          THE NATURAL USER ACCESS METHOD STUB. *
/** *
/** NOTE #1 - CHANGE DSN TO NATURAL LOAD LIBRARY *
/** NOTE #2 - CHANGE DSN AS REQUIRED *
/**-----*
/**
/**
/**LKED EXEC PGM=IEWL,PARM='XREF,LET,LIST,RENT,REUS,MAP'
/**SYSPRINT DD SYSOUT=*
/**SYSUT1 DD UNIT=VIO,SPACE=(CYL,(1,1))
/**NATLIB DD DSN=SAGLIB.NAT233.LOAD,DISP=SHR <== SEE NOTE #1
/**DRSLIB DD DSN=LRS.DRS.V1R33.LOAD,DISP=SHR <== SEE NOTE #2
/**SYSLMOD DD DSN=LRS.DRS.V1R33.LOAD,DISP=SHR <== SEE NOTE #2
/**SYSLIN DD *
ORDER DN33MAIN,NATAMUSR
CHANGE NATAM9EX(DN33MAIN)
INCLUDE NATLIB(NATAMUSR)
INCLUDE DRSLIB(DN33MAIN)
ENTRY CMAM09
NAME DN33MAIN(R)
/**
```

---

## Updating the CICS Tables

The DRS/NATURAL interface program (DN33MAIN) and the DRS/NATURAL output characteristics table (DRSNSDEF) must be defined to CICS via PPT entries. You can add the required definitions using RDO or standard CICS tables. If you choose to use RDO, sample JCL is supplied in member NATRDO in dataset LRS.DRS.V1R33.CNTL.

The following RDO or macro definitions are required:

Add to the PPT using RDO:

```
DEFINE PROGRAM(DN33MAIN) GROUP(gggg) LANG(ASSEMBLER)
DEFINE PROGRAM(DRSNSDEF) GROUP(gggg) LANG(ASSEMBLER)

ADD GROUP(gggg) LIST(1111)
```

or, add to the PPT using macro definitions:

```
DFHPPT TYPE=ENTRY,DRS/NATURAL interface program *
PROGRAM=DN33MAIN, *
PGMLANG=ASSEMBLER

DFHPPT TYPE=ENTRY,DRS/NATURAL Output options table *
PROGRAM=DRSNSDEF, *
PGMLANG=ASSEMBLER
```

---

## Copying the DRS/NATURAL Modules for CICS

The DRS/NATURAL load library must be added to the CICS DFHRPL concatenation or the following modules copied to a library in the DFHRPL concatenation:

**DN33MAIN**

**DRSNSDEF**

In addition, the DRS/API modules must be available in the DFHRPL libraries. See [“Installation Steps” on page 20.4](#) for information on which DRS/API modules are necessary for using DRS in CICS.

---

## Define DRS/NATURAL Interface to Natural

The DRS/NATURAL interface is defined to the Natural environment using standard Natural profile parameters that instruct Natural to dynamically load the DRS/NATURAL interface at session initialization.

Natural profile parameters can be specified at two levels:

<b>Static Assignments</b>	These are made by profile parameters specified in the NTPRM macro which generate the NATPARM parameter module, which is then assembled and linked with the Natural nucleus.
<b>Dynamic Assignments</b>	<p>These are parameters specified for the Natural execution, which override the static assignments and are valid for the current Natural session.</p> <p>Dynamic parameters can be specified in BATCH via the EXEC PARM statement or the CMPRMIN DD statement.</p> <p>In a CICS environment, profile parameters can be created using the SYSPARM command and can either be selected explicitly when invoking Natural or selected based on the USERID or TERMINAL ID of the user.</p>

For a complete description of the Natural profile parameters and how they can be specified, please refer to the 'Natural Installation and Operations' manual in the section entitled 'Profile Parameters'.

The profile parameters required to instruct Natural to load the DRS/NATURAL interface are:

```
RCA=( NATAM09 ) ,  
RCALIAS=( NATAM09 ,DN33MAIN ) ,  
PROGRAM=DRSSHTB                (Batch Only)
```

The RCA parameter indicates that NaturalAccess Method 9 should be dynamically loaded at session initialization.

The RCALIAS parameter indicates that the module that should be loaded is called DN33MAIN, the DRS/NATURAL interface module.

The PROGRAM parameter is only required for Batch execution of the DRS/NATURAL interface. This parameter instructs Natural to call program DRSSHTB at session termination to shut down the DRS/API interface.

In a Batch environment, the DRS/NATURAL interface routine is dynamically loaded using a standard OS LOAD request. Therefore, the DN33MAIN module, the options module DRSNSDEF and the shutdown routine DRSSHTB must be available via the Linklist or a STEPLIB DD statement.

In the CICS environment, the DRS/NATURAL interface is loaded using a CICS load request. Therefore, the DN33MAIN module and the options module DRSNSDEF must be available in the CICS DFHRPL concatenation.



---

## Define a Print and Hardcopy File Using the DRS/Natural Interface

The previous step has instructed Natural to load the DRS/NATURAL interface. It is now necessary to define a print file and hardcopy file to use this new interface.

Natural print files are defined via Natural profile parameters and are allocated a file number in the range 1-31. Natural print file definitions can be specified as static definitions in the NATPARM parameter module using the NTPRINT macro or can be specified as dynamic definitions via the PRINT profile statement.

The following dynamic profile parameter is required to define a print file using DRS/NATURAL.

```
PRINT=( ( 1 ) ,AM=USER ,  
DEST=R123 ,  
CLASS=A ,  
BLKSIZE=200 )
```

<b>PRINT=(1)</b>	Defines Natural print file number 1.
<b>AM=USER</b>	Identifies the Access Method for this file as DRS/NATURAL
<b>DEST=R123</b>	This is the Natural print destination name which, by default, will be mapped directly to the JES SYSOUT destination name.
<b>CLASS=A</b>	This is the Natural output class which, by default, is mapped directly to the JES SYSOUT class.
<b>BLKSIZE=200</b>	The block size parameter controls the amount of Natural thread storage that is acquired as an output buffer for this file. This value should be large enough to contain the longest record written by a Natural application.

The above definition will generate a JES SYSOUT dataset with CLASS=A and DEST=R123 using the default output characteristics table. For details of how to control the SYSOUT characteristic, please refer to [“DRS/NATURAL - Controlling SYSOUT Attributes” on page 32.1.](#)

To enable Natural hardcopy requests to be processed by DRS/NATURAL, the following profile statements are required:

```
HCAM=USER ,  
HCDEST=R123
```

<b>HCAM=USER</b>	Indicates that DRS/NATURAL should process hardcopy requests.
<b>HCDEST=R123</b>	Defines the default hardcopy destination as R123. This setting can be overridden on the Natural hardcopy command ‘%H’ by including the destination name in the hardcopy request (i.e. ‘%HR567’).

The above definition will direct all Natural hardcopy requests (‘%H’ command) to DRS/NATURAL with a destination of R123 which, by default, will be mapped directly to the JES SYSOUT destination name.

For details of how to control SYSOUT characteristic for specific printers or for hardcopy requests, please refer to [“DRS/NATURAL - Controlling SYSOUT Attributes” on page 32.1.](#)

---

## Installation is Complete

The basic installation of DRS/NATURAL is now complete.

To verify the successful installation of DRS/NATURAL, you can now start a Natural session using the previously described profile parameters and issue the 'SYSFILE' command. In a BATCH environment this will display both Work and Print files available for this session. For CICS it is necessary to press 'PF10' to display print files.

The following print file information should be displayed:

```
12:40:41          ***** NATURAL SYSTP UTILITY *****          99-05-24
User LINEN06          - Print File Information -          TID A000

M No.   Type      Name   Recfm  Lrecl  Blksz  Status
-----
HC  DRS R3.3  R123    U           250  Available for Output
  1  DRS R3.3  R123   VBA           200  Available for Output

Top of List
Command ==>
Enter-PF1--PF2--PF3--PF4--PF5--PF6--PF7--PF8--PF9--PF10--PF11--PF12
Cont  Help Menu Exit Sel  Pos  --  -  +  ++  Print Work  Canc
```

If no print files are displayed, then either the profile parameters have not been specified correctly, or DRS/NATURAL initialization failed. Check the system log for messages indicating the reason for the failure. A list of failing return codes can be found on page [24.1](#).

If the SYSFILE command shows the correct information for the hardcopy file and print file, then a '%H' command should generate a JES SYSOUT dataset containing the current screen image with a destination of R123. If a SYSOUT dataset is not created, then check the JES LOG and DRSLOG dataset for error messages.

---

## Section 32

# DRS/NATURAL - Controlling SYSOUT Attributes

### Natural Print File Attributes

The DRS/NATURAL interface has been designed to use only standard Natural interfaces and programming statements. This has been done to ensure compatibility with future releases of Natural and also to avoid the requirement to include vendor specific non-standard statements within Natural applications.

Because the Natural environment provides direct support for a limited number of print files attributes, DRS/NATURAL incorporates a facility, which is external to Natural, to enable these attributes to be dynamically mapped to the more extensive JES spool file attributes. This enables Natural applications to fully exploit all current and future facilities of JES without requiring changes to the base Natural environment or the underlying Natural applications.

Print files are defined to Natural using the PRINT profile statement or NTPRINT macro. Each Natural user can have a maximum of thirty-one separate print files with Natural file numbers in the range 1-31. Because the print attributes of a single file can be overridden dynamically by the application, it is normally only necessary to define multiple print files for a user if an applications needs to output to more than one print file concurrently.

Below is a description of the Natural 'PRINT' profile statement which is used to define print output files and a description of how the Natural file attributes are interpreted by DRS/NATURAL.

```
PRINT=( ( n ) ,AM=USER ,  
        DEST=ddddddd ,  
        PROFILE=name ,  
        NAME=name ,  
        FORMS=name ,  
        DISP=disp ,  
        COPIES=nnn ,  
        CLASS=A ,  
        PRTY=nnn ,  
        OPEN=xxx ,  
        CLOSE=xxx ,  
        RECFM=rr ,  
        BLKSIZE=nnnn ,  
        LRECL=nnn ,  
        TRUNC=xxx ,  
        PAD=xx )
```

<b>PRINT((n))</b>	'n' indicates the required print file number that can be in the range 1-31.
<b>AM=USER</b>	The access method must specify 'USER' to indicate that DRS/NATURAL should process requests for this file.
<b>DEST=ddddddd</b>	This is the Natural print destination which by default is mapped directly to the JES SYSOUT destination.  Alternatively this value can be mapped to the JES SYSOUT WRITER attribute or can be ignored and only used as a look-up value when scanning the output options table DRSNSDEF that is described later.
<b>PROFILE=name</b>	This attribute is not mapped directly to any JES attribute but its value can be used as a look-up value when scanning the output options table DRSNSDEF that is described later.
<b>NAME=name</b>	The report name attribute is not mapped directly to any JES attribute but its value can be used as a look-up value when scanning the output options table DRSNSDEF that is described later.
<b>FORMS=name</b>	The Natural forms name parameter by default is mapped directly to the JES FORM name.
<b>DISP=disp</b>	Output file disposition:  DELETE = Print file and then delete from JES. HOLD = Hold print file. KEEP = Print file and then hold (JCL OUTDISP=KEEP). LEAVE = Hold print file and retain when released (JCL OUTDISP=LEAVE).
<b>COPIES=nnn</b>	The Natural copies attribute by default is used as the JES copies value.
<b>CLASS=A</b>	The Natural output class by default is used as the JES SYSOUT class.
<b>PRTY=nnn</b>	The Natural output priority by default is used as the JES SYSOUT priority.  <b>Note:</b> The setting of SYSOUT priority may have been disabled by your JES system programmer.
<b>OPEN=xxx</b>	This parameter is overridden by DRS/NATURAL to a value of 'ACC' which indicates that the JES SYSOUT dataset will be allocated and opened on first access.
<b>CLOSE=xxx</b>	This parameter is overridden by DRS/NATURAL to a value of 'CMD' which indicates that the file will be deallocated and closed when command mode, NEXT mode or MAINMENU is reached.
<b>RECFM=rr</b>	This value is used as the record format for the JES SYSOUT dataset.
<b>BLKSIZE=nnn</b>	This value is used by DRS/NATURAL to determine the size of the output buffer to allocate for this file. The value should be large enough to contain the longest record generated by a Natural application.  <b>Note:</b> This storage is allocated from Natural thread storage.
<b>LRECL=nnn</b>	This value is ignored by DRS/NATURAL.
<b>TRUNC=xxx</b>	If TRUNC=ON then records which are greater than the BLKSIZE defined for this file will be truncated. With TRUNC=OFF any write request for a record larger than the BLKSIZE value will cause an error to be returned.
<b>PAD=xx</b>	This value is ignored by DRS/NATURAL.

---

## Dynamic Control of Natural Print Attributes

The default Natural print file attributes specified in the profile definitions can be modified for individual reports using the Natural 'DEFINE PRINTER' programming statement.

The 'DEFINE PRINTER' statement has the following syntax and should be specified before beginning output operations to the specified print file.

```
DEFINE PRINTER (n) OUTPUT 'dest'  
    PROFILE 'name'  
    NAME 'name'  
    CLASS 'c'  
    DISP 'disp'  
    COPIES nnn  
    PRTY nnn  
  
WRITE (1) 'TEST LINE 1'  
CLOSE PRINTER (1)
```

The ability to dynamically change the print file attributes enables applications to dynamically select an alternate printer destination or to select special SYSOUT attributes for individual print dataset.

The following section will explain how Natural file attributes like PROFILE, NAME and FORMS can be used to instruct DRS/NATURAL to select specific SYSOUT attributes. The JES attributes may include operational parameters or complex AFP attributes to select specific document formatting options.

---

## Mapping Natural Attributes to JES SYSOUT Attributes

To enable Natural applications to exploit the full functionality of JES, the DRS/NATURAL interface provides a facility to map the limited Natural print attributes to the more extensive JES SYSOUT attributes.

This mapping is done via a simple configuration table which is scanned using the print file DEST, PROFILE, NAME and FORMS attributes to locate the required JES SYSOUT attributes.

The DRS/NATURAL output characteristic table consists of three main sections:

Output Characteristic Table DRSNSDEF

Table Header
Selection Table
Output Options

The Table Header contains configuration options and global defaults for the DRS/NATURAL interface.

The Selection Table section contains multiple entries which identify specific combinations of the print file DEST, PROFILE, NAME and FORMS values and a reference to the required JES output options. The DEST, PROFILE, NAME and FORMS values specified in the selection table can contain specific values, generic values or, if omitted, are assumed to match any incoming value.

The Selection Table is scanned from top to bottom until an entry is found which matches the incoming print file attributes. By arranging the order of this selection table, it is possible to define default output options for specific printers. The application can then override these options by specifying combinations of the PROFILE, NAME or FORMS attributes that match entries higher in the selection table.

The Output Options section of the configuration table defines specific, named, JES output characteristics, which can be selected by one or more selection table entries.

The DRS/NATURAL output characteristics table provides a very flexible method to select the required JES output options without the requirement to specify JES specific attributes directly within Natural applications. This enables JES attributes to be controlled centrally and changed without the requirement for application changes.

Below is an example DRS/NATURAL output characteristics table (DRSNSDEF) which shows the basic structure of the definitions.

```

*=====*
*
*      DRS/NATURAL SYSOUT CHARACTERISTICS TABLE
*
*=====*
*
*      $DRSNDEF TYPE=INITIAL,          GENERATE TABLE HEADER
*      ICMPACT=DELETE,                DELETE INCOMPLETE REPORTS
*      NATDEST=JESDEST,              USE NATURAL DEST AS JES DEST
*      TRACE=YES,                    ENABLE TRACING
*      USEROPTS=00000000,            USER OPTION FLAGS
*      DEFAULT=DEFOUT                DEFAULT OUTPUT OPTIONS
*-----*
*      SELECTION CRITERIA
*-----*
*
*      --- APPLICATION SPECIFIC SELECTIONS ---
*
*      $DRSNDEF NFORM=INVOICE,OUTPUT=INVOICE
*
*      --- HARDCOPY SELECTIONS ---
*
*      $DRSNDEF NPROF=HARDCOPY,OUTPUT=HCOPY
*
**     --- PRINTER SPECIFIC SELECTIONS ---
*
*      $DRSNDEF NDEST=PRT1,OUTPUT=PORT
*      $DRSNDEF NDEST=PRT2,OUTPUT=PORT
*      $DRSNDEF NDEST=P*,OUTPUT=LAND
*-----*
*      OUTPUT CHARACTERISTICS
*-----*
*
DEFOUT  $DRSNDEF @OUTDESC='DEFAULT OUTPUT OPTIONS',
        CLASS=H,
        FORM=STD
*
HCOPY   $DRSNDEF @OUTDESC='DEFAULT HARDCOPY OPTIONS',
        CLASS=A
*
INVOICE $DRSNDEF @OUTDESC='COMPANY INVOICE',
        CLASS=A,
        FORMDEF=INV,
        PAGEDEF=INV
*
LAND    $DRSNDEF @OUTDESC='LANDSCAPE PRINT 132 CHARS x 66 LINES',*
        CLASS=A,
        FORMDEF=SIMPLX,
        PAGEDEF=LAND
*
PORT    $DRSNDEF @OUTDESC='PORTRAIT PRINT 80 CHARS x 72 LINES',
        CLASS=A,
        FORMDEF=SIMPLX,
        PAGEDEF=PORT
*
        END

```

The above table contains some simple examples of how the output options can be specified using DRS/NATURAL.

---

The header section defines system options and identifies the default JES output attributes which should be used if no specific selection definition matches the current Natural print attributes.

The selection table then contains entries which identify specific Natural output options which will trigger the selection of the associated JES output characteristics defined at the bottom of the table. The selection table uses the NDEST, NPROF, NNAME and NFORM keywords to identify the Natural print attributes which should be checked during open processing. (If an attribute is not specified, then it is assumed to match any value).

For example, if a Natural report is generated with DEST=PRT1 and FORMS=INVOICE, the INVOICE selection will be triggered and the output would received a JES Destination of PRT1 and a FORMDEF and PAGEDEF of 'INV'.

Note: The Natural attributes are combined with the JES output options to generate the final SYSOUT attributes.

If the Natural report simply had DEST=PRT1 and no specific FORMS value, then the default selection for printer PRT1 will be selected, resulting in a JES destination of PRT1 and a PAGEDEF of 'PORT' and FORMDEF of 'SIMPLX'.

You will notice that the selection table can specify specific or generic values which are indicated by a '\*' appended to the end of the value.



## DRS/Natural Variable Substitution

To provide additional flexibility when defining output options DRS/NATURAL provides ten system variables that can be specified in any output field. These variable names are dynamically resolved during open processing with the appropriate values.

Variable name	Variable Description	Natural Variable
+NATDEST	NATURAL DESTINATION	
+NATFORM	NATURAL FORM NAME	
+NATPROF	NATURAL PROFILE NAME	
+NATRNAM	NATURAL REPORT NAME	
+NATUSER	NATURAL USERID	*USERID
+CICSUSR	CICS USERID	*INIT-USER
+CICSTRN	CICS TRANSACTION ID	*INIT-PROGRAM
+CICSTRM	CICS TERMINAL ID	*INIT-ID
+NATPROG	NATURAL PROGRAM NAME	*PROGRAM
+NATLIBR	NATURAL LIBRARY NAME	*LIBRARY-ID

The sample below illustrates the use of system variables in the output options table. In this example the SYSOUT TITLE is set to indicate the requesting user and terminal and the output WRITER name is assigned the value of the Natural Report name.

```

*=====*
*
*      DRS/NATURAL SYSOUT CHARACTERISTICS TABLE
*
*=====*
*
*      $DRSNDEF TYPE=INITIAL,      GENERATE TABLE HEADER
*      ICMPACT=DELETE,           DELETE INCOMPLETE REPORTS
*      NATDEST=JESDEST,          USE NATURAL DEST AS JES DEST
*      TRACE=YES,                 ENABLE TRACING
*      USEROPTS=00000000,        USER OPTION FLAGS
*      DEFAULT=DEFOUT            DEFAULT OUTPUT OPTIONS
*
*-----*
*      OUTPUT CHARACTERISTICS
*-----*
*
*      DEFOUT $DRSNDEF @OUTDESC='DEFAULT OUTPUT OPTIONS',
*              CLASS=A,
*              WRITER=+NATRNAM,
*              TITLE='REPORT CREATED BY +NATUSER ON TERMINAL +CICSTRM'
*

```

---

## Controlling Hardcopy Attributes

The Natural hardcopy facility enables a user to request a hardcopy of the current screen image. These hardcopy requests can be directed to DRS/NATURAL via the HCAM=USER profile parameter and output directly to the JES spool.

The only print file attribute that can be specified for hardcopy requests is the destination name. The default hardcopy destination is defined via the HCDEST profile parameter, although the user can override this value by specifying the destination name on the hardcopy command.

To enable the SYSOUT attributes of hardcopy requests to be specified independently of the printer defaults, the PROFILE file attribute is assumed to contain a value of 'HARDCOPY'. This enables specific selection statements to be coded in the output characteristics table to assign default attributes for hardcopy requests.

Below is an example selection table definition for hardcopy requests.

```
*  
*   --- HARDCOPY SELECTION ---  
*  
*           $DRSNDEF NPROF=HARDCOPY ,OUTPUT=HCOPY  
*
```

The above definition assigns the default output options for all hardcopy requests.

If required, you can also define specific hardcopy options for individual printer destinations by specifying both the NDEST=dest and NPROF=HARDCOPY keywords. These destination specific entries must be defined before the general hardcopy default as the selection table is scanned from top down.

If the selection table does not contain a specific hardcopy selection, then either the default attributes for the destination printer or the global default attributes will be used.

---

## Building the DRS/NATURAL Output Characteristics Table

The DRS/NATURAL output characteristics table is generated using the \$DRSNDEF macro which has three sets of keywords for the three sections in the table. The following pages will describe the keywords appropriate for each section of the table and the valid keyword values.

The output characteristics table is generated by performing the following steps:

1. Modify member DRSNSDEF in library LRS.DRS.V1R33.ASM, using the following pages to identify the required parameters.
2. Create the output characteristics module (DRSNSDEF) by assembling and link editing your modified version of the DRSNSDEF source. Sample JCL to assemble and link edit the output options module can be found in member NDEFASML or NDEFASMH of dataset LRS.DRS.V1R33.CNTL.

---

## Output Characteristics Table Header

The following definitions describe the \$DRSNDEF keywords which are valid for the header section of the DRS/NATURAL output characteristics table.

<b>b</b> <b>\$DRSNDEF</b> <b>TYPE=INITIAL,</b> <b>DEFAULT=,</b> <b>{,ICMPACT=}</b> <b>{,NATDEST=}</b> <b>{,TRACE=}</b> <b>{,USEROPTS}</b>	<b>\$DRSNDEF must be preceded by one or more blanks</b>
--	---

Brackets, { }, are used to enclose optional parameters, which may or may not be specified. If a parameter is not coded, the default for that parameter will be used.

The parameters for the table header are explained below:

- TYPE=** Must be 'INITIAL' for the table header
- DEFAULT=** This parameter identifies the DRS/NATURAL default output options and must specify the name of an 'output options' definition defined in the last section of the table generation.
- ICMPACT=** Specifies action for incomplete reports. 'DELETE' indicates that incomplete reports should be deleted at system termination; 'KEEP' indicates that these reports should be kept. Default: DELETE
- NATDEST=** This parameter indicates whether the Natural print file destination should be mapped to the JES SYSOUT destination or WRITER attributes. Valid values are 'JESDEST' or 'JESWRTR'. Default: 'JESDEST'
- TRACE=** Indicates whether tracing should be activated. Valid values are YES or NO. Default: NO
- USEROPTS=** An 8-byte field which represents a 4-byte hex value with flags for special processing options
- 80000000** - Do not use Natural print destination as JES destination or WRITER name
  - 40000000** - Do not use Natural print CLASS as JES SYSOUT class.
  - 20000000** - Do not use Natural FORMS name as JES SYSOUT FORMS name.
  - 10000000** - Do not use Natural COPIES as JES SYSOUT COPIES value.
  - 08000000** - Do not use Natural PRTY as JES SYSOUT priority value.
  - 04000000** - Do not use Natural DISP as JES output disposition.
- Other values are currently unassigned.

---

## Output Characteristics Selection Table

The following definitions describe the \$DRSNDEF keywords which are valid for the selection table section of the DRS/NATURAL output characteristics table. The selection table definitions must immediately follow the table header definition and any number of selection entries may be defined.

<b>b</b>	<b>\$DRSNDEF must be preceded by one or more blanks</b>
<b>\$DRSNDEF</b>	
{NDEST=}	
{,NPROF=}	
{,NNAME=}	
{,NFORM=}	
,OUTPUT=	

Brackets, { }, are used to enclose optional parameters, which may or may not be specified. If a parameter is not coded, the default for that parameter will be used.

The parameters for the selection table section are explained below:

- NDEST=** Identifies the Natural destination value that should be compared during open requests for output option selection. This keyword can specify a specific value or a generic value terminated with an '\*'.  
Default: '\*' (Match any value.)
- NPROF=** Identifies the Natural profile value that should be compared during open requests for output option selection. This keyword can specify a specific value or a generic value terminated with an '\*'.  
Default: '\*' (Match any value.)
- NNAME=** Identifies the Natural report name that which should be compared during open requests for output option selection. This keyword can specify a specific value or a generic value terminated with an '\*'.  
Default: '\*' (Match any value.)
- NFORM=** Identifies the Natural forms value that should be compared during open requests for output option selection. This keyword can specify a specific value or a generic value terminated with an '\*'.  
Default: '\*' (Match any value.)
- OUTPUT=** This parameter identifies the output options that should be selected if the Natural output options match the selection criteria defined via the NDEST, NPROF, NNAME and NFORM keywords. The output name must match the label associated with the output options defined in this third section of this table.

---

## Output Characteristics Table Output Options

The following definitions describe the \$DRSNDEF keywords which are valid for the output options section of the DRS/NATURAL output characteristics table. The output option definitions must follow all selection table definitions.

Each output option definition must specify a unique label name that can be referenced by one or more selection table entries.

&LABEL	\$DRSNDEF must be preceded by a unique label name
\$DRSNDEF	
{@OUTDESC=}	
{,ADDR1=}	
{,ADDR2=}	
{,ADDR3=}	
{,ADDR4=}	
{,BUILDING=}	
{,BURST=}	
{,CHARS=}	
{,CKPTLINE=}	
{,CKPTPAGE=}	
{,CKPTSEC=}	
{,CLASS=}	
{,COLORMAP=}	
{,COMPACT=}	
{,COMSETUP=}	
{,CONTROL=}	
{,COPIES=}	
{,DATAACK=}	
{,DEPT=}	
{,DEST=}	
{,DPAGELBL=}	
{,DUPLEX=}	
{,FCB=}	
{,FLASH=}	
{,FORMDEF=}	
{,FORMLEN=}	
{,FORMS=}	
{,GROUPID=}	
{,INDEX=}	
{,LINDEX=}	
{,LINECT=}	
{,MODIFY=}	
{,NAME=}	
{,NODE=}	
{,NOTIFY=}	

---

{,NOTIFY2=}  
{,NOTIFY3=}  
{,NOTIFY4=}  
{,OFFSETXB=}  
{,OFFSETXF=}  
{,OFFSETYB=}  
{,OFFSETYF=}  
{,OUTBIN=}  
{,OUTDISP=}  
{,OVERLAYB=}  
{,OVERLAYF=}  
{,OVL=}  
{,PAGEDEF=}  
{,PIMSG=}  
{,PORTNO=}  
{,PRMODE=}  
{,PRTEROR=}  
{,PRTOPTNS=}  
{,PRTQUEUE=}  
{,PRTY=}  
{,RESFMT=}  
{,RETAINF=}  
{,RETAINS=}  
{,RETRYL=}  
{,RETRYT=}  
{,ROOM=}  
{,SYSAREA=}  
{,TITLE=}  
{,TRC=}  
{,UCS=}  
{,USRDTA01=}  
{,USRDTA02=}  
{,USRDTA03=}  
{,USRDTA04=}  
{,USRDTA05=}  
{,USRDTA06=}  
{,USRDTA07=}  
{,USRDTA08=}  
{,USRDTA09=}  
{,USRDTA10=}  
{,USRDTA11=}  
{,USRDTA12=}  
{,USRDTA13=}

---

```
{,USRDTA14=}
{,USRDTA15=}
{,USRDTA16=}
{,USRLIB01=}
{,USRLIB02=}
{,USRLIB03=}
{,USRLIB04=}
{,USRLIB05=}
{,USRLIB06=}
{,USRLIB07=}
{,USRLIB08=}
{,WRITER=}
END
```

Brackets, { }, are used to enclose optional parameters, which may or may not be specified. If a parameter is not coded, the default for that parameter will be used.



---

The @OUTDESC keyword is provided for documentation purposes and allows 60 bytes of text to be associated with the output options to document the formatting options specified.

All of the remaining output option parameters refer to SYSOUT characteristics, which could be specified on a SYSOUT DD statement or an OUTPUT JCL statement. For a complete description of these parameters please refer to the MVS JCL Reference.

The parameters for the output options section are explained below:

<b>ADDR1=</b>	A 60-byte field specifying the first address to be printed on separator pages.
<b>ADDR2=</b>	A 60-byte field specifying the second address to be printed on separator pages.
<b>ADDR3=</b>	A 60-byte field specifying the third address to be printed on separator pages.
<b>ADDR4=</b>	A 60-byte field specifying the fourth address to be printed on separator pages.
<b>BUILDING=</b>	A 60-byte field specifying the building to be printed on separator pages.
<b>BURST=</b>	A 1-byte field which directs output to a stacker on a 3800 Printing Subsystem. 'B' = burster-trimmer-stacker; 'C' = continuous forms stacker.
<b>CHARS=</b>	Four 4-byte fields specifying the names of character arrangement tables to be used.
<b>CKPTLINE=</b>	Checkpoint lines. Valid values are 0-32767.
<b>CKPTPAGE=</b>	Checkpoint pages. Valid values are 0-32767.
<b>CKPTSEC=</b>	Checkpoint seconds. Valid values are 0-32767.
<b>CLASS=</b>	A 1-byte field specifying the SYSOUT class. Valid values are 0-9 or A-Z.
<b>COLORMAP=</b>	An 8-byte field specifying the AFP resource for the print file which contains color translation information
<b>COMPACT=</b>	An 8-byte field specifying the name of a compaction table for JES to use.
<b>COMSETUP=</b>	An 8-byte field specifying the AFP resource for the print file this contains setup information.
<b>CONTROL=</b>	An 8-byte field specifying the type of spacing to be applied to the SYSOUT dataset. Valid values consist of "SINGLE", "DOUBLE", "TRIPLE" or "PROGRAM".
<b>COPIES=</b>	The copies or copy groups to be used to print the field. One copies field or up to 8 copy group values may be specified. Valid values are 0-255.
<b>DATAACK=</b>	An 8-byte field specifying action to be taken for AFP print errors; valid values are "BLOCK", "UNBLOCK", "BLKCHAR" or "BLKPOS".
<b>DEPT=</b>	A 60-byte field specifying the department to be printed on separator pages.
<b>DEST=</b>	An 8-character destination or userid.

---

<b>DEST (LONG)=</b>	A 127-byte field specifying the destination for the SYSOUT dataset. The longer destination field is intended for a TCP/IP address in the format "IP:xxxxxxx" or "node.IP:xxxxxxx". This field and the 17-byte DEST field are mutually exclusive.
<b>DPAGELBL=</b>	A 1-byte field specifying whether the security label should be printed on each page. Valid values are "Y" or "N".
<b>DUPLEX=</b>	An 8-byte field specifying whether printing is to be done on both sides of the sheet. Valid values consist of "NO", "NORMAL" or "TUMBLE".
<b>FCB=</b>	A 4-byte field containing the FCB name.
<b>FLASH=</b>	Two fields to represent the forms overlay name and the number of copies on which the forms overlay is to be printed. Valid values for the flash name include any 4 character flash name. Valid values for the flash count are 0 to 255.
<b>FORMDEF=</b>	A 6-byte field representing the name of a FORMDEF for AFP processing.
<b>FORMLEN=</b>	A 10-byte field specifying the numeric length and unit type that will be used to change the physical paper length without reconfiguring the printer. Valid values are nn.nnnUU, where n is a digit 0-9, and UU represents one of the following units: IN (inches) or CM (centimeters).
<b>FORMS=</b>	An 8-byte field to specify the form name to be used for this dataset. The FORMS value may contain an '*' at any point to indicate that the current printer id should be inserted.
<b>GROUPID=</b>	An 8-byte field to specify the name of an OUTPUT GROUP to which this dataset belongs.
<b>INDEX=</b>	A field specifying the left margin on a printer with an indexing feature. Valid values are 1 to 31.
<b>INTRAY=</b>	A 4-byte binary field specifying the paper source when printing AFP files. Valid values are 1- 255.
<b>LINDEX=</b>	A field specifying the right margin on a printer with the indexing feature. Valid values are 1 to 31.
<b>LINECT=</b>	A field specifying the maximum number of lines JES is to print on each output page. Valid values are 0 to 255.
<b>MODIFY=</b>	Two values that specify the 4-byte name of a copy modification module and the 1-byte field, which indicates which table name in the CHARS parameter, should be used. Valid values for the first field are any 1-4 character name of a copy modification module. Valid values for the second field are 0-3.
<b>NAME=</b>	A 60-byte field to specify the name to be printed on the separator pages.
<b>NODE=</b>	An 8-byte field specifying the JES Node name to be used.
<b>NOTIFY=</b>	A 17-byte field specifying the first user to be notified when a job completes
<b>NOTIFY2=</b>	A 17-byte field specifying the second user to be notified when a job completes
<b>NOTIFY3=</b>	A 17-byte field specifying the third user to be notified when a job completes

---

---

<b>NOTIFY4=</b>	A 17-byte field specifying the fourth user to be notified when a job completes
<b>OFFSETXB=</b>	A 13-byte field specifying the offset in the x direction from the page origin for the back side of each page of output. Valid values are mmm.nnnUU, where m is a digit from 0 - 9, n is a digit from 0 - 9 and UU represents one of the following units: IN (inches), CM (centimeters), MM (millimeters), PELS or POINTS
<b>OFFSETXF=</b>	A 13-byte field specifying the offset in the x direction from the page origin for the front side of each page of output. Valid values are mmm.nnnUU, where m is a digit from 0 - 9, n is a digit from 0 - 9 and UU represents one of the following units: IN (inches), CM (centimeters), MM (millimeters), PELS or POINTS
<b>OFFSETYB=</b>	A 13-byte field specifying the offset in the y direction from the page origin for the back side of each page of output. Valid values are mmm.nnnUU, where m is a digit from 0 - 9, n is a digit from 0 - 9 and UU represents one of the following units: IN (inches), CM (centimeters), MM (millimeters), PELS or POINTS
<b>OFFSETYF=</b>	A 13-byte field specifying the offset in the y direction from the page origin for the front side of each page of output. Valid values are mmm.nnnUU, where m is a digit from 0 - 9, n is a digit from 0 - 9 and UU represents one of the following units: IN (inches), CM (centimeters), MM (millimeters), PELS or POINTS
<b>OUTBIN=</b>	A 4-byte binary field specifying the printer output bin identifier to be used. Valid values are 1 to 65535.
<b>OUTDISP=</b>	Two fields which represent the output disposition when the job ends NORMALLY or ABNORMALLY. Valid values are WRITE, HOLD, KEEP, LEAVE and PURGE.
<b>OVERLAYB=</b>	An 8-byte field specifying that the named medium overlay is to be placed on the back side of each sheet to be printed.
<b>OVERLAYF=</b>	An 8-byte field specifying that the named medium overlay is to be placed on the front side of each sheet to be printed.
<b>OVFL</b>	An 8-byte field specifying whether or not JES3 should test for page overflow on an output printer. (JES3 only) Valid values are "ON" or "OFF".
<b>PAGEDEF=</b>	A 6-byte field specifying the name of a PAGEDEF member to be used for AFP processing.
<b>PIMSG=</b>	Two fields which specify whether messages should be printed on the output listing during AFP processing, and the number of errors which should be printed. Valid values for the first field are "Y" or "N"; valid values for the second field are 0 to 999.
<b>PORTNO=</b>	A 4-byte binary field specifying the TCP/IP port number at which the printing application connects to the printer. Valid values are 1 - 65535.
<b>PRMODE=</b>	An 8-byte field specifying the process mode required to print the dataset. Valid values are "LINE", "PAGE" or any valid installation-defined process mode.
<b>PRERROR=</b>	An 8-byte field specifying how a SYSOUT data set that has had printing terminated by a functional subsystem is to be released by JES. Valid values are "DEFAULT", "HOLD" or "QUIT".

---

---

<b>PRTOPTNS=</b>	An 16- byte field specifying the named entity that contains additional print options for an IP-destined data set that is being sent by a functional subsystem.
<b>PRTQUEUE=</b>	A 127-byte field specifying the print queue name used when printing the IP-destined data
<b>PRTY=</b>	Specifies the initial priority at which the SYSOUT dataset enters the output queue. Valid values are 0 to 255.
<b>RESFMT=</b>	An 8-byte field specifying the resolution used to format the print data set. Valid values are "P240" or "P300".
<b>RETAINF=</b>	A 10-byte field specifying how long a functional subsystem will retain an IP-destined data set after a failed transmission.
<b>RETAINS=</b>	A 10 byte field specifying how long a functional subsystem will retain an IP-destined data set after a successful transmission.
<b>RETRYL=</b>	A 4-byte binary field specifying the number of attempts an FSS will try for transmission of an IP-destined data set. Valid values are 0 - 32767.
<b>RETRYT=</b>	A 10-byte field specifying how much time a functional subsystem will wait between retries of transmission attempts of a dataset.
<b>ROOM=</b>	A 60-byte field specifying the room identification to be printed on separator pages.
<b>SYSAREA=</b>	Specifies whether the system should reserve an area for the security label on each page of printed output. Valid values are "Y" or "N".
<b>TITLE=</b>	A 60-byte field specifying the title to be printed on separator pages.
<b>TRC=</b>	A 1-byte field specifying whether or not the SYSOUT dataset has a TRC character in the second character of each record. Valid values are "Y" or "N".
<b>UCS=</b>	A 4-byte field specifying a universal character set, print train, or character-arrangement name.
<b>USRDTA01=</b>	Sixteen 60-byte fields containing user data values.
<b>USRDTA02=</b>	
<b>USRDTA03=</b>	
<b>USRDTA04=</b>	
<b>USRDTA05=</b>	
<b>USRDTA06=</b>	
<b>USRDTA07=</b>	
<b>USRDTA08=</b>	
<b>USRDTA09=</b>	
<b>USRDTA10=</b>	
<b>USRDTA11=</b>	
<b>USRDTA12=</b>	
<b>USRDTA13=</b>	
<b>USRDTA13=</b>	
<b>USRDTA15=</b>	

---

**USRDTA16=**

**USRLIB01=** Eight 44-byte fields containing the dataset name(s) of user libraries for AFP processing.

**USRLIB02=**

**USRLIB03=**

**USRLIB04=**

**USRLIB05=**

**USRLIB06=**

**USRLIB07=**

**USRLIB08=**

**WRITER=** An 8-byte field specifying the writer name.

---

## Including DRS/Natural in the Natural Nucleus

The normal installation procedure explains how to dynamically load the DRS/Natural interface during session initialization. This is the recommended method of implementing the product as it isolates the DRS/Natural interface from the Natural environment and simplifies problem determination. This also has the advantage that new versions of the interface can be implemented without the requirement to re-build the Natural nucleus.

If dynamic Natural profile statements can not be used the DRS/Natural interface can be statically linked into the Natural nucleus by amending the Natural linkedit JCL as indicated below.

```
//LKD2230 EXEC PGM=IEWL,
//   PARM=('RENT,REUS,XREF,LET,LIST,NCAL',
//   'SIZE=(1024K,256K)'),
//   COND=(4,LT)
//SYSLMOD DD DSN=SAGLIB.SMA123.LOAD,
//   DISP=SHR
//SYSLIN DD DDNAME=SYSIN
//SYSUT1 DD UNIT=3390,SPACE=(1700,(500,100))
//SYSPRINT DD SYSOUT=*,
//   DCB=(RECFM=FB,LRECL=121,BLKSIZE=1210)
/* LIBRARIES NEEDED BY NATURAL/CICS
//TPSLIB DD DSN=CICSTS12.CICS.SDFHLOAD,DISP=SHR
//NCILIB DD DSN=SAGLIB.NCI233.LOAD,DISP=SHR
/* LIBRARIES NEEDED BY NATURAL
//SMALIB DD DSN=SAGLIB.SMA123.LOAD,DISP=SHR
//NATLIB DD DSN=SAGLIB.NAT233.LOAD,DISP=SHR
/*
/* LIBRARIES NEEDED FOR DRS/NATURAL
//DRSLIB DD DSN=DRS.V1R33.LOAD,DISP=SHR
/*
/* LIBRARIES NEEDED BY SAG-PRODUCTS
//ADALIB DD DSN=SAGLIB.ADA622.LOAD,DISP=SHR
//SYSIN DD *
MODE AMODE(31),RMODE(ANY)
INCLUDE TPSSLIB(DFHEAI) CICS COMMAND LEVEL STUB
INCLUDE SMALIB(NCISTART) NATURAL/CICS INTERFACE
INCLUDE TPSSLIB(DFHEAI0) CICS INITIAL DSA GETMAIN STUB
INCLUDE SMALIB(NCIPARM) NATCICS PARAMETER MODULE
INCLUDE NCILIB(NCINUC) STORAGE CONTROL PROGRAM
INCLUDE SMALIB(PRM001CI) GENERATED PARM.MODUL
CHANGE NATAM9EX(DN33MAIN) CHANGE NATAM9EX 'V' CONSTANT TO DN33MAIN
INCLUDE NATLIB(NATAMUSR) INCLUDE NATURAL USER ACCESS METHOD STUB
INCLUDE DRSLIB(DN33MAIN) INCLUDE DRS/NATURAL ACCESS METHOD
ENTRY CMSTART ENTRY (CICS)
NAME NC233RE(R)
```

The print file definitions can also be coded statically using the NTPRINT macro in the NATPARM module. For complete descriptions of the Natural profile macros please refer to the “Natural Installation and Operations” manual in the section entitled ‘Profile Parameters’.

```
*
      NTPRINT (1,2),AM=USER,DEST=PRT1
*
```

**Note:** This macro should be coded after the main NTPRM macro definitions.

---

# Section 33

## DRS/NATURAL - Problem Determination

This section contains hints about diagnosing problems with DRS/NATURAL. Any of the following may help in problem resolution:

- NATURAL file error messages
- DRS/NATURAL messages
- DRS/NATURAL and DRS/API return codes
- DRS/API SNAP dumps
- DRS/NATURAL trace data
- SYSUDUMP for any ABEND that may occur

If you find you need assistance to resolve any problem, LRS technical support is always ready to help you.

### NATURAL messages

If an error occurs during an output operation to a DRS/NATURAL controlled file, the following Natural error message will be issued.

```
NAT1524 NATAMUSR:xxx FILE nn EXIT yyy ERROR RTC ccc REASON rrr
```

Where:

```
xxx - NATURAL file type PRINT/WORK
nn  - NATURAL print file number
yyy - Failing function ( OPENO, OPENI, READ, WRITE,
      CLOSR, CLOSE or SETUP )
ccc - DRS/NATURAL or DRS/API return code
rrr - Reason code
```

The DRS/NATURAL and DRS/API return codes are documented on page [24.1](#).

### DRS/NATURAL Messages

DRS/NATURAL messages are written to the combined DRS/API and DRS/NATURAL log dataset. This dataset is dynamically allocated with a DDNAME of DRSLOG and contains a log of all activity. If the LOG dataset is not created, please check the logging options that have been specified in the DRS/API Options Module (DRSSOPTS). For a complete description of the DRS/API options keywords please refer to [“Installation Steps” on page 20.4](#).

### DRS/NATURAL and DRS/API Return Codes

A unique DRS/NATURAL or DRS/API return code will be displayed in all error messages. These return codes are documented on page [24.1](#) and will be of assistance in determining the cause of the error.

---

## DRS/API SNAP Dumps

If a DRS/API error is encountered, a SNAP dump will be created using the SNAP parameters from the DRSSOPTS module. For more information about defining these SNAP options, see [“Customizing the DRS System Options” on page 20.9](#).

## DRS/NATURAL Trace data

To help diagnose problems in the DRS/NATURAL interface you may be asked to enable tracing. DRS/NATURAL and the DRS/API share a common trace table that will be included in any SNAP dumps that are taken because of DRS/API request failures. The trace data can also be written externally to the MVS Generalized Trace Facility (GTF).

To enable tracing of DRS/NATURAL functions simply specify TRACE=YES in the Output Characteristics table header (DRSNSDEF). You must also ensure that the DRS/API options module defines an internal trace table size using the TRPAGES keyword and that DRS/API external tracing is enabled by specifying TRTYPES=01. For more information about defining these DRS/API options, see [“Customizing the DRS System Options” on page 20.9](#).

## SYSUDUMP from ABEND

If DRS/NATURAL abnormally terminates, a system dump will be created. These dumps will be extremely valuable for resolving problems where a DRS/NATURAL program has abended.

When you have a system dump from DRS/NATURAL, you should call LRS technical support for assistance. Some problems of this type can be resolved in a phone conversation. In most cases, you will be asked to send the dump by mail or e-mail. Please call LRS technical support before sending any dump.



---

## Section 34

# DRS/NATURAL - Messages and Codes

Each of the DRS/NATURAL messages is listed below, with a brief explanation of its meaning and the action that will be taken or that the user should take. The “Yes” or “No” after “WTO:” indicates if this message will be issued as a WTO, in addition to appearing in the DRS log dataset.

**DRSN000I**    stcname REPORT INIT SUCCESSFUL USER=uuuuuuuu PRT=pppp-  
pppp OPTIONS=oooooooo FN=nn  
uuuuuuuu        Userid taken from Natural \*INIT-USER system  
                  variable  
pppppppp        Natural print file destination  
oooooooo        DRS/NATURAL output options selected from  
                  table DRSNSDEF  
nn                Natural file number

**Message Meaning:** DRS/NATURAL has successfully allocated a JES spool file.

**System Action:** DRS/NATURAL is ready to process print records.

**Required Action:** None.

**WTO:** No.

**DRSN001I**    stcname REPORT COMPLETE USER=uuuuuuuu PRT=pppppppp  
FN=nn  
uuuuuuuu        Userid taken from Natural \*INIT-USER system  
                  variable  
pppppppp        Natural print file destination  
nn                Natural file number

**Message Meaning:** DRS/NATURAL has successfully deallocated and closed a JES spool file.

**System Action:** DRS/NATURAL has successfully processed a print request.

**Required Action:** None.

**WTO:** No.

---

**DRSN002E** stcname ffffffff REQUEST FAILED USER=uuuuuuuu RC=nnnnnnnn  
REASON=rrrrrrrr  
fffffff Natural function code  
uuuuuuuu Userid taken from Natural \*INIT-USER system  
variable  
nnnnnnnn DRS/NATURAL or DRS/API return code  
rrrrrrrr ERROR SPECIFIC REASON CODE

**Message Meaning:** DRS/NATURAL has encountered an error processing the requested function.

**System Action:** The requested function failed and an error will be returned to the requesting application.

**Required Action:** Check the return code and reason code for the cause of the error.  
DRS/API and DRS/NATURAL return codes can be found on page [24.1](#).

**WTO:** Yes.

**DRSN999E** stcname DRS/NATURAL INITIALIZATION FAILED RC=nnnnnnnn  
RSN=rrrrrrrr  
nnnnnnnn DRS/NATURAL or DRS/API return code  
rrrrrrrr ERROR SPECIFIC REASON CODE

**Message Meaning:** A fatal error occurred during the initialization of the DRS/NATURAL interface.

**System Action:** The Natural print file will be unavailable for use.

**Required Action:** Check the return code and reason code for the cause of the error and verify that the DRS/NATURAL and DRS/API installation tasks have been completed successfully.  
DRS/API and DRS/NATURAL return codes can be found on page [24.1](#).

**WTO:** Yes.

---

# Section 35

## Messages and Codes

### DRS/API Messages

Each of the DRS/API messages is listed below, with a brief explanation of its meaning and the action that DRS will take or that the user should take. The “Yes” or “No” after “WTO:” indicates if this message will be issued as a WTO, in addition to appearing in the DRS log dataset.

<b>DRS000</b>	stcname DRS INITIALIZATION SUCCESSFUL VERSION=V1R3.3.nm <b>Message Meaning:</b> DRS has been successfully initialized. <b>System Action:</b> DRS is ready to accept additional calls. <b>Required Action:</b> None. <b>WTO:</b> No.
<b>DRS001</b>	stcname GETMAIN FAILURE (IWA) <b>Message Meaning:</b> DRS is unable to acquire storage for the interface work area. <b>System Action:</b> DRS returns to the caller with RC=12. <b>Required Action:</b> Retry the request. If the request fails consistently, contact DRS technical support. <b>WTO:</b> Yes.
<b>DRS002</b>	stcname GETMAIN FAILURE (RIB) <b>Message Meaning:</b> DRS is unable to acquire storage in order to locate the main DRS system control block. <b>System Action:</b> DRS returns to the caller with RC=1 2. <b>Required Action:</b> Retry the request. If the request fails consistently, contact DRS technical support. <b>WTO:</b> Yes.
<b>DRS003</b>	stcname GETMAIN FAILURE (SYS) <b>Message Meaning:</b> DRS is unable to acquire storage for the main DRS system control block. <b>System Action:</b> DRS initialization is terminated. If DRS is executing under CICS, DRS abends with code DR05. Otherwise, DRS returns to the caller with RC=12. <b>Required Action:</b> Retry the request. If the request fails consistently, contact DRS technical support. <b>WTO:</b> Yes.

---

<b>DRS004</b>	<p>stcname ERROR READING DRS SYSTEM RECORD - RC=nn  nn: The return code from the read.</p> <p><b>Message Meaning:</b> DRS has encountered an error attempting to read the main DRS system control block.</p> <p><b>System Action:</b> If DRS is executing under CICS, DRS abends with code DR01. Otherwise, DRS returns to the caller with RC=12.</p> <p><b>Required Action:</b> Contact DRS technical support.</p> <p><b>WTO:</b> Yes.</p>
<b>DRS005</b>	<p>stcname ERROR WRITING DRS SYSTEM RECORD - RC=##  nn: The return code from the write.</p> <p><b>Message Meaning:</b> DRS has encountered an error attempting to write the main DRS system control block.</p> <p><b>System Action:</b> If DRS is executing under CICS, DRS abends with code DR02. Otherwise, DRS returns to the caller with RC=12.</p> <p><b>Required Action:</b> Contact DRS technical support.</p> <p><b>WTO:</b> Yes.</p>
<b>DRS007</b>	<p>stcname INVALID DRS SYSTEM RECORD</p> <p><b>Message Meaning:</b> DRS has encountered invalid data in the main DRS system control block.</p> <p><b>System Action:</b> If DRS is executing under CICS, DRS abends with code DR03. Otherwise, DRS returns to the caller with RC=12.</p> <p><b>Required Action:</b> Contact DRS technical support.</p> <p><b>WTO:</b> Yes.</p>
<b>DRS008</b>	<p>stcname ERROR LOADING DRS SUPPORT MODULE xxxxxxxx  xxxxxxx: The name of DRS support module.</p> <p><b>Message Meaning:</b> At the time of the first call to DRS, DRS is unable to load the DRS support module.</p> <p><b>System Action:</b> If DRS is executing under CICS, DRS abends with code DR07. Otherwise, DRS returns to the caller with RC=12.</p> <p><b>Required Action:</b> Make the DRS support module available for an OS LOAD in the steplib or linklist library.</p> <p><b>WTO:</b> Yes.</p>

---

<b>DRS009</b>	<p>stcname ERROR LOADING DRSSOPTS MODULE - RC=nn</p> <p>nn: The return code from the load.</p> <p><b>Message Meaning:</b> DRS is unable to load the DRS options module.</p> <p><b>System Action:</b> The default DRS system option values are used.</p> <p><b>Required Action:</b> Ignore the message or assemble and linkedit the DRS system options module.</p> <p><b>WTO:</b> Yes.</p>
<b>DRS010</b>	<p>stcname INVALID DRSSOPTS MODULE DETECTED</p> <p><b>Message Meaning:</b> The DRS options module is invalid.</p> <p><b>System Action:</b> The default DRS system option values are used.</p> <p><b>Required Action:</b> Ensure that the DRSSOPTS module is valid and retry the request.</p> <p><b>WTO:</b> Yes.</p>
<b>DRS011</b>	<p>stcname DRSSOPTS MODULE SUCCESSFULLY LOADED - VERSION= (ddddddd,tttttt)</p> <p>ddddddd: The date that the options module was assembled.</p> <p>tttttt: The time that the options module was assembled.</p> <p><b>Message Meaning:</b> DRS has successfully loaded the DRS options module.</p> <p><b>System Action:</b> None.</p> <p><b>Required Action:</b> None.</p> <p><b>WTO:</b> No.</p>
<b>DRS012</b>	<p>stcname DRS OPTIONS - LOG=(ENABLED DISABLED,HOLD NOHOLD, class,dest,form,writer,fc,ucs)</p> <p><b>Message Meaning:</b> DRS has successfully loaded the DRS options module, and is displaying the logging options specified in that module.</p> <p><b>System Action:</b> None.</p> <p><b>Required Action:</b> None.</p> <p><b>WTO:</b> No.</p>

---

<b>DRS013</b>	<p>stcname DRS OPTIONS - SNAP=(HOLD NOHOLD,class,dest,form,writer, fcb,ucs) SYSLOG=YES/NO TCB=aaa TRACE=(X'bb',cccK,dd)</p> <p>aaa:                   The number of TCBs to be used to process DRS calls.</p> <p>bb:                    The trace options.</p> <p>ccc:                   The size of the trace table.</p> <p>dd:                    The GTF format appendage id to be used for trace entries recorded in the system trace file.</p> <p><b>Message Meaning:</b> DRS has successfully loaded the DRS options module, and is displaying the SNAP, TCB, and trace options specified in that module.</p> <p><b>System Action:</b>    None.</p> <p><b>Required Action:</b>  None.</p> <p><b>WTO:</b>               No.</p>
<b>DRS014</b>	<p>stcname ERROR ALLOCATING DRS TRACE TABLE</p> <p><b>Message Meaning:</b>  DRS is unable to allocate the storage for the DRS trace table.</p> <p><b>System Action:</b>    The DRS tracing facility is disabled.</p> <p><b>Required Action:</b>  None.</p> <p><b>WTO:</b>               Yes.</p>
<b>DRS015</b>	<p>stcname ERROR ALLOCATING DRS SUBTASK CONTROL AREA TABLE</p> <p><b>Message Meaning:</b>  DRS is unable to allocate the subtask control area table.</p> <p><b>System Action:</b>    If DRS is executing under CICS, DRS abends with code DR06. Otherwise, DRS returns to the caller with RC=12.</p> <p><b>Required Action:</b>  Increase the region size and retry the request.</p> <p><b>WTO:</b>               Yes.</p>
<b>DRS016</b>	<p>stcname xxxxxxxx ATTACHED - TCB=yyyyyyyyy</p> <p>xxxxxxx:              The DRS module name (DRSSMAIN or DRSSLOG).</p> <p>yyyyyyyyy:            The TCB address.</p> <p><b>Message Meaning:</b>  DRS has attached a support subtask.</p> <p><b>System Action:</b>    None.</p> <p><b>Required Action:</b>  None.</p> <p><b>WTO:</b>               No.</p>

---

<b>DRS017</b>	<p>stcname xxxxxxxx ATTACH FAILURE - R15=nn  xxxxxxx:           The DRS module name (DRSSMAIN or DRSSLOG).</p> <p>nn:                 The return code from the ATTACH.</p> <p><b>Message Meaning:</b> DRS' attempt to attach a subtask has failed.</p> <p><b>System Action:</b> If DRS is executing under CICS, DRS abends with code DR08. Otherwise, DRS returns to the caller with RC=12.</p> <p><b>Required Action:</b> Contact DRS technical support.</p> <p><b>WTO:</b>            Yes.</p>
<b>DRS018</b>	<p>stcname SUCCESSFULLY ATTACHED nnn SUPPORT SUBTASK(S)  nnn:                The number of subtasks attached.</p> <p><b>Message Meaning:</b> DRS has completed attaching the requested number of subtasks.</p> <p><b>System Action:</b> None.</p> <p><b>Required Action:</b> None.</p> <p><b>WTO:</b>            No.</p>
<b>DRS019</b>	<p>stcname xxxxxxxx DETACHED - TCB=yyyyyyyy  xxxxxxx:           The DRS module name (DRSSMAIN or DRSSLOG).  yyyyyyyy:           The TCB address.</p> <p><b>Message Meaning:</b> DRS has detached a subtask.</p> <p><b>System Action:</b> None.</p> <p><b>Required Action:</b> None.</p> <p><b>WTO:</b>            No.</p>
<b>DRS020</b>	<p>stcname xxxxxxxx DETACH FAILURE - R15=nn  xxxxxxx:           The DRS module name (DRSSMAIN or DRSSLOG).  nn:                 The return code from DETACH.</p> <p><b>Message Meaning:</b> DRS' attempt to detach a subtask has failed.</p> <p><b>System Action:</b> None.</p> <p><b>Required Action:</b> Contact DRS technical support.</p> <p><b>WTO:</b>            Yes.</p>
<b>DRS021</b>	<p>stcname DRS INITIALIZATION FAILURE</p> <p><b>Message Meaning:</b> Due to one or more errors, DRS initialization has failed.</p> <p><b>System Action:</b> If DRS is executing under CICS, DRS abends with a code indicating the cause of the failure. Otherwise, DRS returns to the caller with RC=12.</p> <p><b>Required Action:</b> Refer to messages in the DRS log to determine the cause of the failure.</p> <p><b>WTO:</b>            Yes.</p>

---

---

<b>DRS022</b>	<p>stcname DRS PRODUCT WILL EXPIRE IN nn DAYS</p> <p><b>Message Meaning:</b> The trial copy of DRS that is being executed is due to expire in nn days.</p> <p><b>System Action:</b> None.</p> <p><b>Required Action:</b> If you have received a non-trial key for DRS, change the DRSSKEY module to use that key value, as explained in <a href="#">“Installing the DRSSKEY Module” on page 20.13</a>. If you will not be done with your trial version of DRS in the amount of time remaining, contact marketing at LRS.</p> <p><b>WTO:</b> Yes.</p>
<b>DRS023</b>	<p>stcname DRS PRODUCT HAS EXPIRED</p> <p><b>Message Meaning:</b> The trial copy of DRS that is being executed has reached or passed its expiration.</p> <p><b>System Action:</b> DRS initialization will fail.</p> <p><b>Required Action:</b> If you have received a non-trial key for DRS, change the DRSSKEY module to use that key value, as explained in <a href="#">“Installing the DRSSKEY Module” on page 20.13</a>.</p> <p><b>WTO:</b> Yes.</p>
<b>DRS025</b>	<p>stcname DRS PRODUCT KEY IS INVALID</p> <p><b>Message Meaning:</b> The DRS Product Key (KEYDRS=) was either not specified in the DRSSKEY module or was specified incorrectly.</p> <p><b>System Action:</b> DRS will reject all attempts to initialize new reports with error code 998.</p> <p><b>Required Action:</b> Ensure that the product key has been entered correctly in the DRSSKEY module. If you are unable to locate the product key, contact LRS marketing personnel.</p> <p><b>WTO:</b> Yes.</p>
<b>DRS099</b>	<p>stcname DRS TERMINATION COMPLETE</p> <p><b>Message Meaning:</b> DRS has terminated.</p> <p><b>System Action:</b> None.</p> <p><b>Required Action:</b> None.</p> <p><b>WTO:</b> No.</p>



---

**DRS100**      stcname report-id ALLOCATION SUCCESSFUL - I=nnnn  
JOBNAME=jobname DD=ddname SYSOUT=(class,dest,form,writer)  
nnnn:                    The attribute group number assigned to this dataset.  
jobname:                Name of job as specified using DRIB jobname  
                                 field.  
**Message Meaning:**    DRS has allocated a SYSOUT dataset, due to an  
                                 INIT call.  
**System Action:**        None.  
**Required Action:**    None.  
**WTO:**                    No.  
**Note:**                    JOBNAME=jobname will only be present if the  
                                 DRIB jobname field was used to change the job  
                                 name associated with the SYSOUT dataset.

**DRS101**      stcname report-id UNALLOCATION SUCCESSFUL - I=nnnn  
DD=ddname SYSOUT=(class,dest,HOLD|NOHOLD)  
DISP=KEEP|DELETE  
nnnn:                    The attribute group number assigned to this dataset.  
**Message Meaning:**    DRS has unallocated a SYSOUT dataset, due to a  
                                 TERM call.  
**System Action:**        None.  
**Required Action:**    None.  
**WTO:**                    No.

**DRS102**      stcname OUTADD SUCCESSFUL - NAME=xxxxxxx  
SYSOUT=(class,dest,form,writer,pagedef,formdef)  
**Message Meaning:**    DRS has added an OUTPUT JCL statement, due  
                                 to an “OUTP ADD” call.  
**System Action:**        None.  
**Required Action:**    None.  
**WTO:**                    No.

**DRS103**      stcname OUTDEL SUCCESSFUL - NAME=xxxxxxx  
SYSOUT=(class,dest,form,writer,pagedef,formdef)  
**Message Meaning:**    DRS has deleted an OUTPUT JCL statement, due  
                                 to an “OUTP DEL” call.  
**System Action:**        None.  
**Required Action:**    None.  
**WTO:**                    No.

---

**DRS104** stcname report-id ALLOC SUCCESSFUL - I=nnnn DD=ddname  
DS=dsname

nnnn: The attribute group number assigned to this dataset.

ddname: The ddname assigned to this dataset.

dsname: The dataset name assigned to this dataset.

**Message Meaning:** DRS has allocated a DASD dataset, due to an INIT call.

**System Action:** None.

**Required Action:** None.

**WTO:** No.

**DRS105** stcname report-id UNALLOC SUCCESSFUL - I=nnnn DD=ddname  
DS=dsname DISP=disp

nnnn: The attribute group number assigned to this dataset.

ddname: The ddname assigned to this dataset.

dsname: The dataset name assigned to this dataset.

disp: The dataset disposition.

**Message Meaning:** DRS has unallocated a DASD dataset, due to a TERM call.

**System Action:** None.

**Required Action:** None.

**WTO:** No.

**DRS106** stcname report-id NON-TERMINATING DYNAMIC ALLOCATION  
ERROR ENCOUNTERED, I=nnnn EC=eeee IC=iiii

nnnn: The attribute group number assigned to this dataset.

eeee: The error code from dynamic allocation.

iiii: The info code form dynamic allocation.

**Message Meaning:** DRS encountered a non-terminating error while allocating or unallocating a DASD dataset.

**System Action:** None.

**Required Action:** None.

**WTO:** No.

---

**DRS110**      stcname xxxxxxxx xxxxxxxxxxxx ERROR - I=nnnn R15=nn EC=eeee  
 IC=iiii text

nnnn:                    The attribute group number assigned to this dataset.

nn, eeee, iiiii:        Return code and error codes from dynamic allocation.

text:                    Description of error, if available.

**Message Meaning:**    An error occurred while attempting to allocate or unallocate a SYSOUT dataset.

**System Action:**      If the error occurred during allocation, DRS deletes all the datasets associated with the report id and returns an error code to the caller. If the error occurred during unallocation, DRS returns an error code to the caller and suspends further activity on all datasets associated with the report id.

**Required Action:**    Contact DRS technical support.

**WTO:**                    Yes.

**DRS111**      stcname xxxxxxxx K=#### N=#### xxx

**Message Meaning:**    Identifies the SVC99 text unit that caused the allocation/unallocation request to fail.

**System Action:**      Same as message DRS110.

**Required Action:**    Contact DRS technical support.

**WTO:**                    Yes.

**DRS112**      stcname OUTADD ERROR - R15=aa R0=bbbbbbbb R1=ccccccc

aa:                        OUTADD return code.

bbbbbbbb:                OUTADD reason code.

ccccccc:                 Text unit key causing the failure (or zero, if no text unit involved).

**Message Meaning:**    An error occurred while attempting to create an OUTPUT JCL statement.

**System Action:**      None.

**Required Action:**    Contact DRS technical support.

**WTO:**                    Yes.

**DRS113**      stcname xxxxxxxx K=#### N=#### xxx

**Message Meaning:**    Identifies the OUTPUT JCL text unit that caused the OUTADD request to fail.

**System Action:**      None.

**Required Action:**    Contact DRS technical support.

**WTO:**                    Yes.

---

**DRS114**      stcname xxxxxxxx OUTDEL ERROR - R15=aa R0=bbbbbbbb  
aa:                    OUTDEL return code.  
bbbbbbbb:            OUTDEL reason code.  
**Message Meaning:** An error occurred while attempting to delete an  
OUTPUT JCL statement.  
**System Action:**      None.  
**Required Action:**    Contact DRS technical support.  
**WTO:**                Yes.

**DRS115**      stcname report-id OBTAIN ERROR - I=nnnn R15=nn DSNAME=dsname  
VOLSER=volser  
nnnn:                The attribute group number assigned to the dataset.  
nn:                    The CAMLST OBTAIN return code.  
dsname:              The dataset name assigned to the dataset.  
volser:               The primary volume serial number assigned to the  
dataset.  
**Message Meaning:** DRS has allocated a DASD dataset and attempted to  
obtain the format-1 DSCB for the dataset in order to  
determine the dataset attributes. However, the  
attempt to obtain the format-1 DSCB failed.  
**System Action:**      DRS unallocates all the datasets associated with the  
report ID and returns an error code to the caller. Any  
datasets associated with the report ID that were  
created by DRS are deleted.  
**Required Action:**    Ensure that the volume assigned to the dataset is a  
valid DASD volume and is mounted. If unable to  
resolve the problem, contact DRS Technical Support.  
**WTO:**                Yes.

**DRS116**      stcname report-id DUPLICATE DS EXISTS - I=nnnn DSNAME=dsname  
VOLSER=volser  
nnnn:                The attribute group number assigned to the dataset.  
dsname:              The dataset name assigned to the dataset.  
volser:               The primary volume serial number assigned to the  
dataset.  
**Message Meaning:** DRS was invoked to allocate a DASD dataset. The  
attribute group for the dataset specified an initial  
disposition of either NEW or MOD and a normal  
disposition of CATLG. However, an existing dataset  
with the same name is already cataloged.  
**System Action:**      DRS unallocates all the datasets associated with the  
report ID and returns an error code to the caller. Any  
datasets associated with the report ID that were created  
by DRS are deleted.  
**Required Action:**    Specify a dataset name for a dataset that does not exist  
or change the disposition to overwrite or extend the  
existing dataset.  
**WTO:**                Yes.

---

<b>DRS117</b>	<p>stcname report-id INCOMPAT DS ATTRS - I=nnnn DSNAME=dsname VOLSER=volser</p> <p>nnnn: The attribute group number assigned to the dataset.</p> <p>dsname: The dataset name assigned to the dataset.</p> <p>volser: The primary volume serial number assigned to the dataset.</p> <p><b>Message Meaning:</b> DRS has allocated a DASD dataset and determined that the attributes of the dataset are incompatible with the report attributes. This can be caused by any of the following situations:</p> <ul style="list-style-type: none"> <li>• Specifying a partitioned dataset without providing a member name.</li> <li>• Specifying a sequential dataset and providing a member name.</li> <li>• Specifying a logical record length that is not identical to that of the dataset.</li> <li>• Specifying a block size that is not identical to that of the dataset.</li> <li>• Specifying a record format (fixed, variable or undefined) that is not identical to that of the dataset.</li> <li>• Specifying a carriage control (ANSI, machine, or none) that is not identical to that of the dataset.</li> </ul> <p><b>System Action:</b> DRS unallocates all the datasets associated with the report ID and returns an error code to the caller. Any datasets associated with the report ID that were created by DRS are deleted.</p> <p><b>Required Action:</b> Specify compatible report attributes and dataset allocation attributes.</p> <p><b>WTO:</b> Yes.</p>
<b>DRS600</b>	<p>stcname LOG INITIALIZATION SUCCESSFUL</p> <p><b>Message Meaning:</b> DRS has successfully initialized the logging function.</p> <p><b>System Action:</b> None.</p> <p><b>Required Action:</b> None.</p> <p><b>WTO:</b> No.</p>
<b>DRS601</b>	<p>stcname LOG ALLOCATION FAILURE - R15=nn EC=eeee IC=iiii nn, eeee, iii:</p> <p>The return code and error codes from dynamic allocation.</p> <p><b>Message Meaning:</b> Allocation of the DRS log dataset has failed.</p> <p><b>System Action:</b> DRS logging is disabled.</p> <p><b>Required Action:</b> Contact DRS technical support.</p> <p><b>WTO:</b> Yes.</p>

---

---

<b>DRS602</b>	stcname I/O ERROR WRITING TO LOG DATASET
	<b>Message Meaning:</b> DRS encountered an I/O error while attempting to write a record in the DRS log dataset.
	<b>System Action:</b> DRS logging is disabled.
	<b>Required Action:</b> Contact DRS technical support.
	<b>WTO:</b> Yes.
<b>DRS603</b>	stcname synad-error-message
	<b>Message Meaning:</b> DRS encountered an I/O error while attempting to write a record in the DRS log dataset. This message is preceded by message DRS602. This message contains the synad error message returned to DRS by the access method.
	<b>System Action:</b> DRS logging is disabled.
	<b>Required Action:</b> Contact DRS technical support.
	<b>WTO:</b> Yes.
<b>DRS604</b>	stcname LOG DCB ABEND - CODE=aaa-rc OPT=oo
	aaa: The abend code.
	rc: The return code.
	oo: The DCB recovery options.
	<b>Message Meaning:</b> An error occurred during OPEN, CLOSE, or EOV processing of the DRS log dataset.
	<b>System Action:</b> DRS logging is disabled.
	<b>Required Action:</b> Contact DRS technical support.
	<b>WTO:</b> Yes.
<b>DRS605</b>	stcname LOG OPEN FAILURE
	<b>Message Meaning:</b> DRS' attempt to open the log dataset has failed.
	<b>System Action:</b> DRS logging is disabled.
	<b>Required Action:</b> Contact DRS technical support.
	<b>WTO:</b> Yes.
<b>DRS606</b>	stcname LOG CLOSE FAILURE
	<b>Message Meaning:</b> DRS' attempt to close the log dataset has failed.
	<b>System Action:</b> DRS logging is disabled.
	<b>Required Action:</b> Contact DRS technical support.
	<b>WTO:</b> Yes.
<b>DRS607</b>	stcname LOG CLOSE COMPLETE
	<b>Message Meaning:</b> DRS has successfully closed the log dataset in response to a CLOSELOG command.
	<b>System Action:</b> A new log dataset is opened.
	<b>Required Action:</b> None.
	<b>WTO:</b> No.

---

<b>DRS608</b>	<p>stcname LOG DATASET IS FULL; LOGGING CONTINUES AT START OF DATASET</p> <p><b>Message Meaning:</b> The pre-allocated dataset being used for the DRS log is full.</p> <p><b>System Action:</b> DRS will continue logging at the beginning of the dataset.</p> <p><b>Required Action:</b> None.</p> <p><b>WTO:</b> No.</p>
<b>DRS609</b>	<p>stcname ERROR ALLOCATING DRSSLOG WORKAREA</p> <p><b>Message Meaning:</b> The GETMAIN failed attempting to allocate storage for the DRS log processor workarea.</p> <p><b>System Action:</b> DRS logging is disabled.</p> <p><b>Required Action:</b> Contact DRS technical support.</p> <p><b>WTO:</b> Yes.</p>
<b>DRS610</b>	<p>stcname LOGGING DISABLED</p> <p><b>Message Meaning:</b> DRS logging was disabled either because of an error condition or in response to a DRS SSET command.</p> <p><b>System Action:</b> DRS continues processing calls without recording messages in the log dataset.</p> <p><b>Required Action:</b> Contact DRS technical support.</p> <p><b>WTO:</b> No.</p>
<b>DRS615</b>	<p>stcname LOG TERMINATION COMPLETE</p> <p><b>Message Meaning:</b> The DRS log processor has terminated.</p> <p><b>System Action:</b> None.</p> <p><b>Required Action:</b> None.</p> <p><b>WTO:</b> No.</p>
<b>DRS650</b>	<p>stcname report-id SNAP ALLOCATION FAILURE - R15=rc EC=eeee IC=iiii</p> <p>rc, eeee, iii:</p> <p>The return code and error codes from dynamic allocation.</p> <p><b>Message Meaning:</b> Allocation of the SNAP dataset has failed.</p> <p><b>System Action:</b> The snap dump is not taken.</p> <p><b>Required Action:</b> Contact DRS technical support.</p> <p><b>WTO:</b> Yes.</p>

---

<b>DRS651</b>	<p>stcname report-id SNAP DCB ABEND - CODE=aaa-rc OPT=oo</p> <p>aaa:                   The abend code.</p> <p>rc:                    The return code.</p> <p>oo:                    The DCB recovery options.</p> <p><b>Message Meaning:</b>   An error occurred during OPEN, CLOSE, or EOV processing of the DRS snap dataset.</p> <p><b>System Action:</b>     The snap dump is not taken.</p> <p><b>Required Action:</b>   Contact DRS technical support.</p> <p><b>WTO:</b>                Yes.</p>
<b>DRS652</b>	<p>stcname report-id SNAP OPEN FAILURE</p> <p><b>Message Meaning:</b>   DRS' attempt to open a SNAP dataset has failed.</p> <p><b>System Action:</b>     The snap dump is not taken.</p> <p><b>Required Action:</b>   Contact DRS technical support.</p> <p><b>WTO:</b>                Yes.</p>
<b>DRS653</b>	<p>stcname report-id SNAP CLOSE FAILURE</p> <p><b>Message Meaning:</b>   DRS' attempt to close a SNAP dataset has failed.</p> <p><b>System Action:</b>     The snap dataset remains allocated.</p> <p><b>Required Action:</b>   Contact DRS technical support.</p> <p><b>WTO:</b>                Yes.</p>
<b>DRS654</b>	<p>stcname report-id I/O ERROR ISSUING SNAP</p> <p><b>Message Meaning:</b>   DRS encountered an I/O error while attempting to write to the DRS snap dataset. This message is followed by message DRS655. This message contains the synad error message returned to DRS by the access method.</p> <p><b>System Action:</b>     The snap dataset is closed.</p> <p><b>Required Action:</b>   Contact DRS technical support.</p> <p><b>WTO:</b>                Yes.</p>
<b>DRS655</b>	<p>stcname report-id synad-error-message</p> <p><b>Message Meaning:</b>   DRS encountered an I/O error when issuing a SNAP. This message is preceded by message DRS654. This message contains the synad error message returned to DRS by the access method.</p> <p><b>System Action:</b>     The snap dataset is closed.</p> <p><b>Required Action:</b>   Contact DRS technical support.</p> <p><b>WTO:</b>                Yes.</p>



---

<b>DRS659</b>	<p>stcname report-id SNAP DUMP COMPLETE</p> <p><b>Message Meaning:</b> DRS has successfully created a SNAP dump for the report identified by report-id.</p> <p><b>System Action:</b> None.</p> <p><b>Required Action:</b> None.</p> <p><b>WTO:</b> No.</p>
<b>DRS800</b>	<p>stcname DRSSMAIN SUBTASK ABEND - SYS=sss USR=uuu</p> <p>sss: System abend code.</p> <p>uuu: User abend code.</p> <p><b>Message Meaning:</b> The main DRS subtask has abended with the system or user abend code identified in the message.</p> <p><b>System Action:</b> Processing continues with the next DRS request.</p> <p><b>Required Action:</b> If the error was caused by passing invalid data to DRS then this message can be ignored. Otherwise, contact DRS technical support.</p> <p><b>WTO:</b> Yes.</p>
<b>DRS802</b>	<p>stcname DRSSLOG SUBTASK ABEND - SYS=sss USR=uuu</p> <p>sss: System abend code.</p> <p>uuu: User abend code.</p> <p><b>Message Meaning:</b> The DRS log subtask has abended with the system or user abend code identified in the message.</p> <p><b>System Action:</b> DRS logging is disabled.</p> <p><b>Required Action:</b> Contact DRS technical support.</p> <p><b>WTO:</b> Yes.</p>
<b>DRS804</b>	<p>stcname DRSSVSST SUBTASK ABEND - SYS=sss USR=uuu</p> <p>sss: System abend code.</p> <p>uuu: User abend code.</p> <p><b>Message Meaning:</b> The DRS VSAM I/O interface subtask has abended with the system or user code identified in the message.</p> <p><b>System Action:</b> Processing continues, if possible, with the next DRS VSAM I/O request. This message can be ignored if the DRS/STI interface is not being used.</p> <p><b>Required Action:</b> Contact DRS technical support.</p> <p><b>WTO:</b> Yes.</p>

---

---

**DRS808** stcname USER EXIT ABEND - EXITID=nn SYS=sss USR=uuu, EXIT DISABLED

nn: Exit number.

sss: System abend code.

uuu: User abend code.

**Message Meaning:** The DRS user exit identified by EXITID nn has abended with the system or user abend code identified in the message.

**System Action:** DRS continues processing without the exit. (The exit is disabled).

**Required Action:** Correct the user exit.

**WTO:** Yes.

**DRS809** stcname PSW=xxxxxxxx xxxxxxxx

**Message Meaning:** A DRS module has abended. This message contains the PSW at the time of the abend.

**System Action:** See messages DRS800, DRS802, and DRS808.

**Required Action:** See messages DRS800, DRS802, and DRS808.

**WTO:** Yes.

**DRS810** stcname R0-R7 rrrrrrrr ..... rrrrrrrr

**DRS810** stcname R8-R15 rrrrrrrr ..... rrrrrrrr

**Message Meaning:** A DRS module has abended. This message contains the register contents at the time of the abend.

**System Action:** See messages DRS800, DRS802, and DRS808.

**Required Action:** See messages DRS800, DRS802, and DRS808.

**WTO:** Yes.

**DRS820** stcname report-id SYSOUT DATASET ABEND - I=nnnn CODE=aaa-rc OPT=oo

nnnn: The attribute group assigned to this dataset.

aaa: The abend code.

rc: The return code.

oo: The DCB recovery options.

**Message Meaning:** An error occurred during OPEN, CLOSE, or EOV processing of the SYSOUT dataset.

**System Action:** DRS returns an error code to the caller and suspends further activity on all datasets associated with the report id.

**Required Action:** Contact DRS technical support.

**WTO:** Yes.

---

<b>DRS821</b>	<p>stcname report-id I/O ERROR PROCESSING DATA SET - I=nnnn RECNUM=rrr,rrr,rrr</p> <p>nnnn: The attribute group assigned to this dataset.</p> <p>rrr,rrr,rrr: The number of records processed successfully before the I/O error was encountered.</p> <p><b>Message Meaning:</b> An I/O error was encountered while processing a DRS dataset.</p> <p><b>System Action:</b> DRS returns an error code to the caller and suspends further activity on all datasets associated with the report id.</p> <p><b>Required Action:</b> Contact DRS technical support.</p> <p><b>WTO:</b> Yes.</p>
<b>DRS822</b>	<p>stcname report-id synad-error-message</p> <p><b>Message Meaning:</b> DRS encountered an I/O error while processing a DRS dataset. This message is preceded by message DRS821. This message contains the synad error message returned to DRS by the access method.</p> <p><b>System Action:</b> DRS returns an error code to the caller and suspends further activity on all datasets associated with the report id.</p> <p><b>Required Action:</b> Contact DRS technical support.</p> <p><b>WTO:</b> Yes.</p>
<b>DRS830</b>	<p>stcname INTERNAL LOGIC ERROR - MESSAGE ID nnn IS UNDEFINED</p> <p>nnn: The invalid message id number.</p> <p><b>Message Meaning:</b> DRS tried to issue a message with a message number that is not defined in DRS' message table.</p> <p><b>System Action:</b> The message is ignored.</p> <p><b>Required Action:</b> Contact DRS technical support.</p> <p><b>WTO:</b> Yes.</p>
<b>DRS831</b>	<p>stcname ERROR ALLOCATING LOG QUEUE ELEMENT - LOGGING DISABLED</p> <p><b>Message Meaning:</b> DRS' allocation of a log queue element failed.</p> <p><b>System Action:</b> DRS logging is disabled.</p> <p><b>Required Action:</b> Contact DRS technical support.</p> <p><b>WTO:</b> Yes.</p>

---

---

<b>DRS900</b>	<p>stcname QUIESCE COMMAND ACKNOWLEDGED</p> <p><b>Message Meaning:</b> DRS has acknowledged receiving a QUIESCE command.</p> <p><b>System Action:</b> DRS will no longer accept INIT calls.</p> <p><b>Required Action:</b> None.</p> <p><b>WTO:</b> No.</p>
<b>DRS901</b>	<p>stcname DRS SYSTEM QUIESCED</p> <p><b>Message Meaning:</b> DRS has processed a QUIESCE command.</p> <p><b>System Action:</b> DRS will no longer accept INIT calls.</p> <p><b>Required Action:</b> None.</p> <p><b>WTO:</b> Yes.</p>
<b>DRS902</b>	<p>stcname DRS SYSTEM RESTARTED</p> <p><b>Message Meaning:</b> DRS has processed a RESTART command.</p> <p><b>System Action:</b> DRS will now accept INIT calls.</p> <p><b>Required Action:</b> None.</p> <p><b>WTO:</b> Yes.</p>
<b>DRS903</b>	<p>stcname DRS SHUTDOWN REQUEST ACKNOWLEDGED</p> <p><b>Message Meaning:</b> DRS has acknowledged receiving a shutdown request.</p> <p><b>System Action:</b> DRS will begin termination processing.</p> <p><b>Required Action:</b> None.</p> <p><b>WTO:</b> Yes.</p>
<b>DRS904</b>	<p>stcname DRS SHUTDOWN REQUEST COMPLETE DRSRC=rc OSRC=osrc</p> <p>rc: DRS return code.</p> <p>osrc: OS return code.</p> <p><b>Message Meaning:</b> DRS has acknowledged receiving a shutdown request.</p> <p><b>System Action:</b> None.</p> <p><b>Required Action:</b> None.</p> <p><b>WTO:</b> Yes.</p>
<b>DRS910</b>	<p>stcname DRS OPTIONS - LOG=(ENABLED DISABLED,HOLD NOHOLD, class,dest,form,writer,fc,ucs)</p> <p><b>Message Meaning:</b> Displays the current status of the DRS log options.</p> <p><b>System Action:</b> None.</p> <p><b>Required Action:</b> None.</p> <p><b>WTO:</b> No.</p>

---

**DRS911**      stcname DRS OPTIONS - SNAP=(HOLD|NOHOLD,class,dest,form,writer,fc,  
ucs) SYSLOG=YES/NO TCB=aaa TRACE=(X'bb',cccK,dd)

aaa:                      The number of TCBs to be used to process DRS  
calls.

bb:                        The trace options.

ccc:                      The size of the trace table.

dd:                        The GTF format appendage id to be used for  
trace entries recorded in the system trace file.

**Message Meaning:**      Displays the current status of the DRS snap and  
trace options.

**System Action:**        None.

**Required Action:**      None.

**WTO:**                    No.

---

## DRS/VPI Messages

DRS/VPI messages are issued via OS/MVS WTO with the routing code specified in the fourth positional parameter value of the WTO parameter in the DRSSTART System Initialization Member. If the message is in response to a DRS/VPI command, the message will be issued to the console from which the command was issued.

Error or Action messages will be displayed with a WTO Descriptor Code of 2 (MVS Action), unless "N" is specified in the second positional parameter value of the WTO parameter in the DRSSTART System Initialization Member.

Informational messages will be bypassed if "N" is specified in the third positional parameter value of the WTO parameter in the DRSSTART System Initialization Member.

### DRS/VPI Message Format

All DRS/VPI messages will begin with an eight character message identifier formatted as follows:

Characters 1-4 - Always DRSV

Characters 5-7 - Message number uniquely identifying the message

Character 8 - One character identifying the message importance level

The possible message importance levels (the last character of the message identifier) are as follows:

**I** - Informational Message

**N** - Normal Message

**E** - Error or Action Message

**R** - Responding Message (responses to DRS/VPI Commands)

The DRS/VPI started task name will follow the message identifier, unless "N" is specified in the first positional parameter value of the WTO parameter in the DRSSTART System Initialization Member.

### DRS/VPI Message Text

The DRS/VPI messages use some or all of the following in the message text:

For printer related messages:

prtrid - the member name of the DRS/VPI virtual printer.

For VTAM related messages:

**r0** - the value in Register 0 at the time of the error.  
**r15** - the value in Register 15 at the time of the error.  
**ac** - the ACB error code field ACBERFLG  
**rc** - the RPL return code field "RPLRTNCD"  
**f2** - the RPL feedback 2 field "RPLFDB2"  
**sense** - the RPL sense code.

---

<b>DRSV000N</b>	<p>DRS/VPI INITIALIZATION SUCCESSFUL VERSION=V1R3.3.fff CUSTID=xxxxxx</p> <p>fff: DRS/VPI distribution fix level. xxxxxx: Six character customer ID</p> <p><b>Message Meaning:</b> The DRS/VPI System has been successfully initialized.</p> <p><b>System Action:</b> None.</p> <p><b>Required Action:</b> None.</p>
<b>DRSV001N</b>	<p>USER EXITnn LOADED NAME=aaaaaaaa LENGTH=lllll EP=eeeeeee ENABLED DISABLED RECOVERY=ON OFF</p> <p>nn: Exit number. aaaaaaaa: Name of the user exit module. lllll: Length of the user exit module. eeeeeee: Entry point of the user exit module.</p> <p><b>Message Meaning:</b> The DRS/VPI System has successfully loaded a user exit module.</p> <p><b>System Action:</b> None.</p> <p><b>Required Action:</b> None.</p>
<b>DRSV002E</b>	<p>DRS OPTIONS MODULE SPECIFIES UNSUPPORTED VALUE - TCBBTCH=0</p> <p><b>Message Meaning:</b> When the DRS/VPI system loaded the DRS system options module, the value for the TCBBTCH keyword was zero.</p> <p><b>System Action:</b> DRS/VPI will terminate.</p> <p><b>Required Action:</b> Update the DRS system options to specify a non-zero value for TCBBTCH (for example, TCBBTCH=1), re-assemble and linkedit the DRS system options module (DRSSOPTS), and restart DRS/VPI.</p>
<b>DRSV005E</b>	<p>DRSVLIB DCB ABEND CODE=xxx-xx</p> <p>xxx-xx: Abend/return codes.</p> <p><b>Message Meaning:</b> An error occurred during OPEN/CLOSE/EOV processing for the DRSVLIB dataset.</p> <p><b>System Action:</b> DRS/VPI will abend with a U005 abend.</p> <p><b>Required Action:</b> Determine the reason for the abend. If the reason cannot be determined, contact DRS technical support.</p>
<b>DRSV006E</b>	<p>DRSVLIB I/O ERROR xxxxxxxxxxxx</p> <p>xxxxxxxxxx: SYNAD message text.</p> <p><b>Message Meaning:</b> I/O error while reading the DRSVLIB dataset.</p> <p><b>System Action:</b> DRS/VPI will abend with a U005 abend.</p> <p><b>Required Action:</b> Determine the cause of the I/O error and correct it.</p>

---

- 
- DRSV007E** DRS/VPI TERMINATED - ATTACH ERROR modulenm RC=rc  
modulenm: The name of the module DRS/VPI attempted to attach.  
rc: Return code from attach macro (see OS/VS2 MVS Supervisor Services).  
**Message Meaning:** DRS/VPI failed attempting to attach a support subtask.  
**System Action:** DRS/VPI will abend with a U005 abend.  
**Required Action:** Contact DRS technical support.
- DRSV008E** INVALID listtyp LIST MEMBER (mbrname) VALUE: keywordval  
listtyp: INCLUDE or EXCLUDE.  
mbrname: Name of inclusion or exclusion list member.  
keywordval: Invalid member name specification within member.  
**Message Meaning:** The contents of either the MLISTMEM or XLISTMEM member is invalid. An example would be a member name in the list that is longer than 8 characters.  
**System Action:** DRS/VPI will abend with a U005 abend.  
**Required Action:** Correct the member name in the MLISTMEM or XLISTMEM.
- DRSV009E** MUTUALLY EXCLUSIVE KEYWORDS SPECIFIED: MLISTMEM/XLISTMEM  
**Message Meaning:** Both MLISTMEM and XLISTMEM keywords were specified in the DRSSTART member. Since the keywords are mutually exclusive, only one of the keywords can be specified.  
**System Action:** DRS/VPI will abend with a U005 abend.  
**Required Action:** Specify either MLISTMEM or XLISTMEM as desired.
- DRSV010E** DRS/VPI INITIALIZATION FAILURE  
**Message Meaning:** DRS/VPI initialization has failed. This message will be preceded by a message giving the reason that initialization failed.  
**System Action:** DRS/VPI will terminate.  
**Required Action:** Refer to the preceding message(s) to determine the cause of the initialization failure.



---

<b>DRSV011N</b>	<p>*SYSTEM STORAGE(REGN(aaaaaaa,bbbbbb) (PRIV(ccccccc,ddddddd))  aaaaaaa:           Region requested below the 16M line.  bbbbbbb:           Region requested above the 16M line.  ccccccc:           Private area available below the 16M line.  ddddddd:           Private area available above the 16M line.  <b>Message Meaning:</b> This message gives information pertaining to the storage requested and available after DRS/VPI initialization is complete.    <b>System Action:</b>   None.  <b>Required Action:</b> None.</p>
<b>DRSV012E</b>	<p>SYSTEM TERMINATED - ERROR LOADING mmmmmmmm R1=xxxxxxxx  R15=r15  mmmmmmm:           The name of a DRS/VPI load module.  xxxxxxxx:           The contents of Register 1.  r15:                 The contents of Register 15.  <b>Message Meaning:</b> DRS/VPI attempted to LOAD a DRS/VPI module, but the LOAD macro gave a non-zero return code.    <b>System Action:</b>   DRS/VPI will terminate.  <b>Required Action:</b> Verify that all necessary DRS/VPI load modules are in either a linklist library or a library named in the STEPLIB DD in the DRS/VPI JCL.</p>
<b>DRSV018I</b>	<p>stcname SWAPABLE=N WAS SPECIFIED, BUT DRS/VPI LIBRARIES ARE NOT AUTHORIZED.  <b>Message Meaning:</b> DRS/VPI attempted to use SYSEVENT TRANSWAP to avoid being swapped out, but determined that the DRS/VPI system is not authorized.    <b>System Action:</b>   DRS/VPI will run swappable.  <b>Required Action:</b> Authorize the DRS/VPI libraries or set SWAPABLE=Y.</p>
<b>DRSV019I</b>	<p>aaaaaaa INITIALIZATION ERROR - LOAD FAILED FOR MODULE  bbbbbbb R1=ccccccc R15=ddddddd  aaaaaaa:           The name of the product for which the LOAD failed.  bbbbbbb:           The name of the load module for which the LOAD failed.  ccccccc:           The contents of Register 1.  ddddddd:           The contents of Register 15.  <b>Message Meaning:</b> DRS attempted to LOAD a module, but the LOAD macro gave an unexpected return code.    <b>System Action:</b>   The product will be disabled.  <b>Required Action:</b> Verify that the necessary load modules are in either a linklist library or a library named in the STEPLIB DD in the DRS JCL.</p>

---

---

**DRSV022E**      aaaaaaaa PRODUCT WILL EXPIRE IN nn DAYS  
aaaaaaa:            The name of the product which is about to  
                             expire (DRS/PC, DRS/TCPIP, or DRS/STI).  
nn:                    The number of days before the product will  
                             expire (1 to 30).  
**Message Meaning:**    The copy of the product which is being  
                             executed will reach its expiration date in “nn”  
                             days. At that time, the product will terminate,  
                             and attempts to start the product will fail. This  
                             message will begin to appear fourteen days  
                             before the expiration date.  
**System Action:**        DRS will continue processing.  
**Required Action:**      Contact LRS marketing personnel.

**DRSV023E**      aaaaaaaa PRODUCT HAS EXPIRED  
aaaaaaa:            The name of the product which has expired  
                             (DRS/PC, DRS/TCPIP, or DRS/STI).  
**Message Meaning:**    The copy of the product which is being  
                             executed has reached its expiration date.  
**System Action:**        The product will terminate. Attempts to start  
                             the product will fail.  
**Required Action:**      Contact LRS marketing personnel.

**DRSV025E**      aaaaaaaa PRODUCT KEY INVALID  
aaaaaaa:            The name of the product containing the invalid  
                             product key (DRS/PC, DRS/TCPIP, or  
                             DRS/STI).  
**Message Meaning:**    The product key, identified in the message, was  
                             either not specified in the DRS Initialization  
                             member or was specified incorrectly.  
                             The DRS product keys are supplied in file # 1 of  
                             the distribution cartridge.  
**System Action:**        DRS will disable the product. Attempts to  
                             activate printers that require the product will fail.  
**Required Action:**      Ensure that the product key has been entered  
                             correctly in the DRS initialization member. If  
                             you are unable to locate the product key, contact  
                             LRS marketing personnel.

---

**DRSV027E** req REQUEST FAILED FOR FILE file(DDNAME=ddname), RETCODE=rc  
req: The type of request that failed (OPEN or CLOSE).  
file: The name of the file on which the error occurred.  
ddname: The DDNAME of the file on which the error occurred.  
rc: Return code information.  
**Message Meaning:** An unexpected error occurred attempting to either OPEN or CLOSE the DRS/STI rules dataset.  
**System Action:** Processing continues, but the DRS/STI rules dataset is unavailable. As a result, the Smart Tag control buffer will not be recognized by the DRS/VPI virtual printers.  
**Required Action:** Ensure that the DRS/STI rules dataset is properly defined in the DRS/VPI startup JCL. If unable to resolve the problem, contact DRS technical support.

**DRSV099N** DRS/VPI TERMINATION COMPLETE  
**Message Meaning:** DRS/VPI has completed termination processing.  
**System Action:** The DRS/VPI address space will terminate.  
**Required Action:** None.

**DRSV100N** prtrid SYSOUT ALLOCATION SUCCESSFUL jobname ddname  
SYSOUT(c,ddddddd,ffff,wwwwwww)  
jobname: Job name, if specified using JOBNAME= keyword.  
ddname: DDNAME, if specified using DDNAME= keyword.  
c: SYSOUT class.  
ddddddd: SYSOUT destination.  
ffff: SYSOUT form.  
wwwwwww: SYSOUT writer name.  
**Message Meaning:** The DRS INIT call was successful to allocate the SYSOUT dataset.  
**System Action:** None.  
**Required Action:** None.

---

**DRSV101N** prtrid SYSOUT UNALLOCATION SUCCESSFUL jobname ddname  
SYSOUT(c,dddd,ffff,wwwwww)  
jobname: Job name, if specified using JOBNAME= keyword.  
ddname: DDNAME, if specified using DDNAME= keyword.  
c: SYSOUT class.  
dddd: SYSOUT destination.  
ffff: SYSOUT form.  
wwwwww: SYSOUT writer name.  
**Message Meaning:** The DRS TERM call was successful to unallocate the  
SYSOUT dataset.  
**System Action:** None.  
**Required Action:** None.

**DRSV104N** prtrid DASD ALLOCATION SUCCESSFUL ddname DSN=dsname  
ddname: DDNAME, if specified using DDNAME= keyword.  
dsname: The dataset name assigned to this dataset.  
**Message Meaning:** The DRS INIT call was successful to allocate the  
DASD dataset.  
**System Action:** None.  
**Required Action:** None.

**DRSV105N** prtrid DASD UNALLOCATION SUCCESSFUL ddname DSN=dsname  
ddname: DDNAME, if specified using DDNAME= keyword.  
dsname: The dataset name assigned to this dataset.  
**Message Meaning:** The DRS TERM call was successful to unallocate the  
DASD dataset.  
**System Action:** None.  
**Required Action:** None.

**DRSV200I** NOW INACTIVATED  
**Message Meaning:** Response to DRS/VPI INACTIVATE command.  
**System Action:** None.  
**Required Action:** None.

**DRSV201E** prtrid ATTACH FAILED FOR modulenm R15=r15  
modulenm: The name of the module DRS/VPI attempted to  
attach.  
r15: Return code from attach macro (see OS/VS2 MVS  
Supervisor Services).  
**Message Meaning:** DRS/VPI failed attempting to attach a support  
subtask for the specified printer.  
**System Action:** DRS/VPI will put this printer in EDRAINED status.  
**Required Action:** Contact DRS technical support.

---

<b>DRSV210E</b>	<p>prtrid LOGIC ERROR FORMATTING OUTPUT BUFFER - pgm  pgm: Name of DRS/VPI program encountering the error.</p> <p><b>Message Meaning:</b> DRS/VPI encountered a logic error while formatting the output buffer from the data received.</p> <p><b>System Action:</b> DRS/VPI will put this printer in EDRAINED status.</p> <p><b>Required Action:</b> Contact DRS technical support.</p>
<b>DRSV211E</b>	<p>prtrid OUTPUT BUFFER OVERRUN - pgm  pgm: Name of DRS/VPI program encountering the error.</p> <p><b>Message Meaning:</b> DRS/VPI received more data than would fit into the output buffer.</p> <p><b>System Action:</b> DRS/VPI will put this printer in EDRAINED status.</p> <p><b>Required Action:</b> Contact DRS technical support.</p>
<b>DRSV212E</b>	<p>prtrid DSC 3270 WRITE COMMAND IS MISSING</p> <p><b>Message Meaning:</b> DRS/VPI expected a 3270 WRITE command in the first byte of the buffer received, but it was not present in the buffer.</p> <p><b>System Action:</b> DRS/VPI will put this printer in EDRAINED status.</p> <p><b>Required Action:</b> Contact DRS technical support.</p>
<b>DRSV213E</b>	<p>prtrid ERROR PROCESSING SPANNED DSC 3270 COMMAND</p> <p><b>Message Meaning:</b> DRS/VPI encountered an error processing a 3270 command which spanned buffers received.</p> <p><b>System Action:</b> DRS/VPI will put this printer in EDRAINED status.</p> <p><b>Required Action:</b> Contact DRS technical support.</p>
<b>DRSV214E</b>	<p>prtrid WRITE STRUCTURED FIELD 3270 COMMAND NOT SUPPORTED</p> <p><b>Message Meaning:</b> DRS/VPI virtual printer received a Write Structured Field command from the sending application, but the command is not supported by DRS/VPI.</p> <p><b>System Action:</b> DRS/VPI will put this printer in EDRAINED status.</p> <p><b>Required Action:</b> Define MODETAB and DLOGMOD in the VTAM definition for the virtual printer, or define the printer in more detail to the sending application. If the error continues, contact DRS technical support.</p>

---

**DRSV217E** prtrid LOGIC ERROR IN DVSSVPRT - xxxxxxxx  
xxxxxxx The type of logic error detected.  
**Message Meaning:** DRS detected an invalid condition (e.g., invalid LUTYPE) while processing.  
**System Action:** DRS/VPI will put this printer in EDRAINED status.  
**Required Action:** Contact DRS technical support.

**DRSV218E** prtrid RECEIVED RECORD LENGTH llll INVALID - xxxxxxxx  
llll: Length of record received.  
xxxxxxx: The specific type of error or the name of the program which encountered the error.  
**Message Meaning:** A record was received by DRS/VPI which is zero or which exceeds the length of the receive buffer.  
**System Action:** DRS/VPI will put this printer in EDRAINED status.  
**Required Action:** Contact DRS technical support.

**DRSV219E** prtrid LOGIC ERROR FORMATTING PRINT BLOCK - MODULE  
pgm text  
pgm: Name of DRS/VPI program encountering the error.  
text: The type of logic error encountered.  
**Message Meaning:** DRS/VPI encountered an error while formatting the print line workarea.  
**System Action:** DRS/VPI will put this printer in EDRAINED status.  
**Required Action:** Contact DRS technical support.

**DRSV220E** prtrid SCS CONTROL CODE SPANS RECEIVE BUFFERS  
**Message Meaning:** DRS/VPI received part of the SCS control code in the buffer, but the SCS control code was not complete.  
**System Action:** DRS/VPI will put this printer in EDRAINED status.  
**Required Action:** Contact DRS technical support.

**DRSV221E** prtrid INVALID SCS CONTROL CODE  
**Message Meaning:** DRS/VPI received an SCS control code in the buffer which was invalid.  
**System Action:** DRS/VPI will put this printer in EDRAINED status.  
**Required Action:** Contact DRS technical support.

---

<b>DRSV227E</b>	<p>prtrid req REQUEST FAILED FOR FILE file, RETCODE=rc</p> <p>req: The type of request that failed (e.g., READ).</p> <p>file: The name of the file on which the error occurred.</p> <p>rc: Return code information.</p> <p><b>Message Meaning:</b> An unexpected error occurred attempting to access the DRS/STI rules dataset.</p> <p><b>System Action:</b> Processing continues, but the Smart Tag control buffer is not used to determine the attributes of the resulting spool dataset.</p> <p><b>Required Action:</b> Ensure that the DRS/STI rules dataset is properly defined in the DRS/VPI startup JCL. If unable to resolve the problem, contact DRS technical support.</p>
<b>DRSV230E</b>	<p>IPDS PRINT BLOCK IS INVALID</p> <p><b>Message Meaning:</b> DRS/VPI encountered a logic error processing the IPDS print line workarea.</p> <p><b>System Action:</b> DRS/VPI will put this printer in EDRAINED status.</p> <p><b>Required Action:</b> Contact DRS technical support.</p>
<b>DRSV231E</b>	<p>prtrid IPDS DATA LENGTH ERROR</p> <p><b>Message Meaning:</b> DRS/VPI encountered an error processing IPDS data because the length of the data was invalid.</p> <p><b>System Action:</b> DRS/VPI will put this printer in EDRAINED status.</p> <p><b>Required Action:</b> Contact DRS technical support.</p>
<b>DRSV232E</b>	<p>prtrid IPDS PRINT BLOCK OVERFLOW</p> <p><b>Message Meaning:</b> DRS/VPI received IPDS data which was longer than the workarea available for print lines to be stored.</p> <p><b>System Action:</b> DRS/VPI will put this printer in EDRAINED status.</p> <p><b>Required Action:</b> Contact DRS technical support.</p>
<b>DRSV233E</b>	<p>prtrid IPDS UNKNOWN COMMAND DETECTED CMD=cmd</p> <p>cmd: The value detected in the IPDS command field.</p> <p><b>Message Meaning:</b> DRS/VPI received an IPDS command which was not recognized.</p> <p><b>System Action:</b> DRS/VPI will put this printer in EDRAINED status.</p> <p><b>Required Action:</b> Contact DRS technical support.</p>

---

---

<b>DRSV240E</b>	prtrid HEADER IGNORED TYPE=type REASON=reason type: The type of header being processed; for example DRS/PC. reason: The reason the header is considered invalid. <b>Message Meaning:</b> An error was encountered while processing a header located in the input data. <b>System Action:</b> The header keywords will be ignored and the header will be retained in the print data. <b>Required Action:</b> Ensure that the header included in the data has the correct values.
<b>DRSV241E</b>	prtrid HEADER KEYWORD ERROR TYPE=type KWD=kwd reason type: The type of header being processed; for example, DRS/PC. kwd: The keyword which is in error. reason: The reason for the error. <b>Message Meaning:</b> An error was encountered while processing a keyword located in the header. The keyword is a valid keyword, but the value specified is in error. <b>System Action:</b> The header keywords will be ignored and the header will be retained in the print data. <b>Required Action:</b> Ensure that the header included in the data has the correct keyword values.
<b>DRSV250E</b>	prtrid ACB OPEN FAILED R15=r15 EC=ec ec: ACB error flag <b>Message Meaning:</b> DRS/VPI was unable to open the VTAM ACB for the specified printer. <b>System Action:</b> DRS/VPI will put this printer in EDRAINED status. <b>Required Action:</b> Correct the error based upon R15 and AC. See the appropriate VTAM programming manual.
<b>DRSV251E</b>	prtrid ACB OPEN FAILED,(5C) VTAM NOT ACTIVE. REPLY R TO RETRY, E TO END <b>Message Meaning:</b> VTAM is not active. <b>System Action:</b> If reply is R, DRS/VPI will retry the ACB open. If reply is E, DRS VPI will terminate. <b>Required Action:</b> Wait for VTAM and its application configurations to come active and reply R, or reply E to terminate DRS/VPI.



---

<b>DRSV252E</b>	ACB CLOSE FAILED R15=r15 EC=ec ec: <b>Message Meaning:</b> <b>System Action:</b> <b>Required Action:</b>	ACB error flag DRS/VPI had an error closing its VTAM ACB. Printer termination will continue. Correct the error based upon R15 and AC values. See the appropriate VTAM programming manual.
<b>DRSV253E</b>	prtrid OPNSEC FAILED R15=r15 R0=r0 RTNCD=rc FDBK2=f2 SENSE=sense OPTCD=xxxxxxx <b>Message Meaning:</b> <b>System Action:</b> <b>Required Action:</b>	VTAM OPNSEC for the specified printer failed. DRS/VPI will put this printer in EDRAINED status. Correct the error based upon R15, R0, RTNCD, FDBK2 and SENSE. See the appropriate VTAM programming manual.
<b>DRSV254E</b>	TERMSESS FAILED R15=r15 R0=r0 RTNCD=rc FDBK2=f2 SENSE=sense OPTCD=xxxxxxx <b>Message Meaning:</b> <b>System Action:</b> <b>Required Action:</b>	VTAM TERMSESS for the specified printer failed. DRS/VPI will put this printer in EDRAINED status. Correct the error based upon R15, R0, RTNCD, FDBK2 and SENSE. See the appropriate VTAM programming manual.
<b>DRSV255E</b>	prtrid SETLOGON FAILED R15=r15 R0=r0 RTNCD=rc FDBK2=f2 SENSE=sense OPTCD=START STOP HOLD <b>Message Meaning:</b> <b>System Action:</b> <b>Required Action:</b>	DRS/VPI was unable to inform VTAM that the printer is ready to accept logons. DRS/VPI will put this printer in EDRAINED status. Correct the error based upon R15, R0, RTNCD, FDBK2 and SENSE. See the appropriate VTAM programming manual.
<b>DRSV256E</b>	prtrid SESSIONC FAILED R15=r15 R0=r0 RTNCD=rc FDBK2=f2 SENSE=sense OPTCD=optcd CONTROL=cntl optcd: cntl: <b>Message Meaning:</b> <b>System Action:</b> <b>Required Action:</b>	Option code on SESSIONC VTAM macro. Control value from SESSIONC VTAM macro. DRS/VPI was unable to issue a session control macro for the designated printer. DRS/VPI will put this printer in EDRAINED status. Correct the error based upon R15, R0, RTNCD, FDBK2 and SENSE. See the appropriate VTAM programming manual.

---

---

**DRSV257E** prtrid RECEIVE FAILED R15=r15 R0=r0 RTNCD=rc FDBK2=f2  
SENSE=sense OPTCD=optcd  
optcd: Option code on RECEIVE VTAM macro.  
**Message Meaning:** VTAM RECEIVE for this printer failed.  
**System Action:** DRS/VPI will put this printer in EDRAINED status.  
**Required Action:** Correct the error based upon R15, R0, RTNCD, FDBK2 and SENSE. See the appropriate VTAM programming manual.

**DRSV258E** prtrid INQUIRE FAILED R15=r15 R0=r0 RTNCD=rc FDBK2=f2  
SENSE=sense OPTCD=optcd  
optcd: Option code on INQUIRE VTAM macro.  
**Message Meaning:** VTAM INQUIRE for this printer failed.  
**System Action:** DRS/VPI will put this printer in EDRAINED status.  
**Required Action:** Correct the error based upon R15, R0, RTNCD, FDBK2 and SENSE. See the appropriate VTAM programming manual.

**DRSV259E** SEND FAILED R15=r15 R0=r0 RTNCD=rc FDBK2=f2 SENSE=sense  
OPTCD=optcd STYPE=REQ|+RESP|-RESP|SHUTC  
optcd: Option code on SEND VTAM macro.  
**Message Meaning:** VTAM SEND for this printer failed.  
**System Action:** DRS/VPI will put this printer in EDRAINED status.  
**Required Action:** Correct the error based upon R15, R0, RTNCD, FDBK2 and SENSE. See the appropriate VTAM programming manual.

**DRSV260E** prtrid RESETSR FAILED R15=r15 R0=r0 RTNCD=rc FDBK2=f2  
SENSE=sense OPTCD=optcd  
optcd: Option code on RESETSR VTAM macro.  
**Message Meaning:** VTAM RESETSR for this printer failed.  
**System Action:** DRS/VPI will put this printer in EDRAINED status.  
**Required Action:** Correct the error based upon R15, R0, RTNCD, FDBK2 and SENSE. See the appropriate VTAM programming manual.

**DRSV261E** prtrid CHECK FAILED R15=r15 R0=r0 RTNCD=rc FDBK2=f2  
SENSE=sense REQUEST=req  
req: RPL request type for which the CHECK failed.  
**Message Meaning:** VTAM CHECK for this printer failed.  
**System Action:** DRS/VPI will put this printer in EDRAINED status.  
**Required Action:** Correct the error based upon R15, R0, RTNCD, FDBK2 and SENSE. See the appropriate VTAM programming manual.

---

<b>DRSV262E</b>	<p>prtrid RECEIVE FAILED RTNCD=rc FDBK2=f2 SENSE=sense</p> <p><b>Message Meaning:</b> VTAM RECEIVE for this printer failed.</p> <p><b>System Action:</b> DRS/VPI will put this printer in EDRAINED status.</p> <p><b>Required Action:</b> Correct the error based upon R15, R0, RTNCD, FDBK2 and SENSE. See the appropriate VTAM programming manual.</p>
<b>DRSV263E</b>	<p>GENCB FAILED R15=r15 R0=r0 BLK=block</p> <p>block: VTAM control block which was being generated.</p> <p><b>Message Meaning:</b> GENCB macro for a VTAM control block failed.</p> <p><b>System Action:</b> DRS/VPI will terminate or DRS/VPI will put this printer in EDRAINED status.</p> <p><b>Required Action:</b> Correct the error based upon R15 and R0. See the appropriate VTAM programming manual.</p>
<b>DRSV264E</b>	<p>MODCB FAILED R15=r15 R0=r0 BLK=block FIELDS=(f1,f2,.....)</p> <p>block: VTAM control block which was being modified.</p> <p>f1,f2,... The fields in the control block which were to be modified.</p> <p><b>Message Meaning:</b> MODCB macro for a VTAM control block failed.</p> <p><b>System Action:</b> DRS/VPI will terminate or DRS/VPI will put this printer in EDRAINED status.</p> <p><b>Required Action:</b> Correct the error based upon R15 and R0. See the appropriate VTAM programming manual.</p>
<b>DRSV265E</b>	<p>SHOWCB FAILED R15=r15 R0=r0 BLK=block FIELDS=(f1,f2,.....)</p> <p>block: VTAM control block whose fields were being displayed.</p> <p>f1,f2,... The fields in the control block which were to be displayed.</p> <p><b>Message Meaning:</b> SHOWCB macro for VTAM control block failed.</p> <p><b>System Action:</b> DRS/VPI will terminate or DRS/VPI will put this printer in EDRAINED status.</p> <p><b>Required Action:</b> Correct the error based upon R15 and R0. See the appropriate VTAM programming manual.</p>
<b>DRSV266E</b>	<p>TESTCB FAILED R15=r15 R0=r0 BLK=block FIELDS=(f1,f2,.....)</p> <p>block: VTAM control block whose fields were being tested.</p> <p>f1,f2,... The fields in the control block which were to be tested.</p> <p><b>Message Meaning:</b> TESTCB macro for VTAM control block failed.</p> <p><b>System Action:</b> DRS/VPI will terminate or DRS/VPI will put this printer in EDRAINED status.</p> <p><b>Required Action:</b> Correct the error based upon R15 and R0. See the appropriate VTAM programming manual.</p>

---

---

**DRSV267E** prtrid CLSDST FAILED R15=r15 R0=r0 RTNCD=rc FDBK2=fb SENSE =sns

**Message Meaning:** CLSDST macro for this printer failed.

**System Action:** DRS/VPI will put this printer in EDRAINED status.

**Required Action:** Correct the error based upon R15, R0, RTNCD, FDBK2 and SENSE. See the appropriate VTAM programming manual.

**DRSV268E** prtrid SESSION REJECTED - SENSE=sense BECAUSE reason  
reason: Reason that DRS/VPI rejected the session.

**Message Meaning:** An application attempted to establish a session with DRS/VPI for the specified prtrid and the session was rejected because the session (BIND) parameters were incorrect for the printer or because the printer was not currently available for session establishment.

**System Action:** The session is rejected by DRS/VPI.

**Required Action:** If the printer is EDRAINED, issue a START command for the printer.

If the DRS/VPI is terminating, try to establish the session after DRS VPI is restarted and the printer is activated.

If a session is already active for the printer with another application, issue a command to release the session from that application.

If the session parameters are invalid, use a different logon mode table entry (DLOGMOD) for the virtual printer (on the APPL statement) or change the definition for the printer in the primary application to indicate different session parameters.

**Note:** Message DRSV269N showing the session parameters will also be issued if the session parameters are invalid.

<b>REASON:</b>	<b>COMMENTS:</b>
BRACKET FLAGS INVALID	For TERMRPT=BRACKET, the BIND parms must specify that brackets are used.

CHAINING FLAGS INVALID	For TERMRPT=CHAIN or TERMRPT=BRACKET, the BIND parms must specify that the session will use chain protocol.
------------------------------	---

DRS/VPI IS TERMINATING	If DRS/VPI is terminating, all session requests will be rejected.
---------------------------	---

---

IN SESSION W/pluname	If the virtual printer is already in session with another application, all additional BIND requests will be rejected. The message will display the name of the current session partner (pluname). This error will occur if SESSLIM=YES was not specified on the VTAM APPL definition for the printer.
LU TYPE IS INVALID	LU type in BIND is not 0, 1, 3, or 6.
PACING VALUE IS ZERO	Both primary send pacing and secondary receive pacing values are zero.
PRINTER IS EDRAINED	If the printer has been drained due to an error, any session requests will be rejected.
RUSIZE INVALID	Primary send RU size is greater than 3840 or secondary send RU size is smaller than 256.
LOGON NOT ALLOWED	A non-APPC device attempted to log on to DRS/VPI. Logon is only valid for LU type 6.
TERMRPT INVALID FOR LU	TERMRPT=APPC was specified for a virtual printer with LU type 0, 1, or 3. TERMRPT=APPC is only valid for COMMTYPE=(APPC,DRSPC)

**DRSV269N** prtrid SESSION PARAMETERS sess parms  
 sess parms: Session parameters in hex.  
**Message Meaning:** When message DRSV268E is issued for invalid BIND parms, the session parameters are displayed in this message.  
**System Action:** DRS/VPI will put this printer in EDRAINED status.  
**Required Action:** Correct the error based upon SENSE from corresponding DRSV268E message.

**DRSV270E**

prtrid APPCCMD FAILED CONT=aaaaaaaa QUAL=bbbbbbbb  
 RCPRI=cccc RCSEC=dddd RTNCD=rc FDBK2=f2 R15=r15 R0=r0

aaaaaaaa: CONTROL= value on VTAM APPCCMD macro.

bbbbbbbb: QUALIFY= value on VTAM APPCCMD macro.

cccc: Primary return code from RPL extension.

dddd: Secondary return code from RPL extension.

**Message Meaning:** VTAM APPCCMD macro for this printer failed.

**System Action:** DRS/VPI will put this printer in EDRAINED status.

**Required Action:** Correct the error based on the CONTROL, QUALIFY, RCPRI, RCSEC, R15, R0, RTNCD, and FDBK2. See the appropriate VTAM programming manual. If the problem cannot be resolved, contact DRS technical support.

Possible values for the error message variables are:

CONT=	QUAL=	Function=
CHECK	N/A	Check RPL after macro completion
OPRCNTL	ACTSESS	Accept session with APPC device
OPRCNTL	DACTSESS	Reject session with APPC device
RCVFMH5	N/A	Receive FMH5 to allocate conversation
RECEIVE	SPEC	Receive data on specific conversation
REJECT	N/A	Termination conversation and session
SEND	CONFRMD	Send confirmation/positive response
SEND	ERROR	Send error information to partner LU
SEND	FLUSH	Flush send buffer

Primary and Secondary return codes for the APPCCMD macros are described in the VTAM Programming for LU6.2 manual. The most common return codes received by DRS when operating with DRS/PC are described below:

RCPRI	RCSEC	Description/Possible Reasons/Action
0004	0000	ALLOCATION ERROR; NO RETRY
		Path error; check physical connections
		Device not defined or not active
		Device not ready; start DRS/PC
0004	0001	ALLOCATION ERROR; RETRY IS ALLOWED
		Device not ready; start DRS/PC
0010	0004	PARTNER LU STARTING SESSION
		DRS/PC has issued CNOS
0020	0000	CNOS FAILIURE, RETRY IS ALLOWED
		Device not ready; start DRS/PC
002C	0000	INVALID LU NAME
		LUNAME incorrect or not active

RCPRI	RCSEC	Description/Possible Reasons/Action
002C	0001	INVALID MODE
		Mode name unknown or misspelled
		Mode name not in logon mode table
002C	0002	INVALID CONVERSATION
		Conversation has terminated abnormally
002C	0003	INVALID LL
		Invalid record length (logic error)
002C	0013	NO CORRESPONDING MODE IN LM TABLE
		Mode name unknown or misspelled
002C	0014	INVALID BIND PARAMETERS
		Mode name does not specify LU6 session parameters
002C	001E	CID INVALID
		ACTSESS specified unknown session ID
		DACTSESS specified unknown session ID
0048	0000	RESOURCE FAILURE, NO RETRY
		DRS/PC was terminated; restart DRS/PC
0054	0000	UNRECOGNIZED MODE NAME
		Mode name not in logon mode table
0064	0000	ACTIVATION FAILURE
		APPC remote device not available
		DRS/PC APPC software not active; start DRS/PC
		DRS/PC using invalid mode name
0088	0000	CANCELLED BY REJECT/ABNORMAL DEALLOC
		DRS/PC terminated abnormally; restart DRS/PC
0090	0000	APPLICATION NOT APPC CAPABLE
		APPL definition does not allow APPCCMD; check DRS APPL

**DRSV281E**

prtrid TPEND type RECEIVED

type: The type of TPEND request received (QUICK or CANCEL).

**Message Meaning:** VTAM has been terminated with Z NET,QUICK/CANCEL, or the DRS virtual printer applid has been varies inactivate with immediate or force.

**System Action:** DRS/VPI will put this printer in EDRAINED status.

**Required Action:** Either vary the applid active or wait for VTAM to be restarted. Then start the DRS virtual printer.

---

<b>DRSV282E</b>	<p>prtrid UNSUPPORTED UNKNOWN REQUEST RECEIVED IN VTAM ATTN EXIT FOR REQUEST TYPE=aaaa</p> <p>aaaa:                   APPC request received in VTAM ATTN exit.</p> <p><b>Message Meaning:</b>   DRS/VPI encountered an unknown or unexpected request in the VTAM ATTN exit.</p> <p><b>System Action:</b>        DRS/VPI will ignore the request.</p> <p><b>Required Action:</b>     Contact DRS technical support.</p>
<b>DRSV283E</b>	<p>prtrid RPL EXIT UNKNOWN REQUEST=req R15=r15 R0=r0 RTNCD=rc FDBK2=f2 SENSE=sense</p> <p>req:                    Request code received in RPL EXIT from VTAM.</p> <p><b>Message Meaning:</b>    VTAM RPL exit was scheduled for an unexpected request type.</p> <p><b>System Action:</b>        DRS/VPI will put this printer in EDRAINED status.</p> <p><b>Required Action:</b>     Correct the error based upon R15, R0, RTNCD, FDBK2 and SENSE. See the appropriate VTAM programming manual.</p>
<b>DRSV284E</b>	<p>prtrid type FUNCTION MANAGEMENT HEADER FMH=fmh reason type:</p> <p>                          Type of FMH received (UNKNOWN or UNSUPPORTED).</p> <p>fmh:                    Function management header value.</p> <p>reason:                 Indicates the reason the FMH is not supported, as follows:</p> <ul style="list-style-type: none"> <li>• FMH LENGTH IS ZERO</li> <li>• STRUCTURED FIELD RECEIVED</li> <li>• NOT BEGIN OR END DEST</li> <li>• SESSION NOT LUTYPE 6</li> <li>• MULTIPLE FMHS CONCATENATED</li> </ul> <p><b>Message Meaning:</b>    DRS received an unknown or unsupported function management header.</p> <p><b>System Action:</b>        DRS/VPI will put this printer in EDRAINED status.</p> <p><b>Required Action:</b>     Correct the primary application, based on the reason listed in the message.</p>
<b>DRSV285E</b>	<p>prtrid INVALID comm REQUEST req</p> <p>comm:                   The communication protocol in use (TCPIP, VTAM or VTAM APPC).</p> <p>req:                    The type of request that failed.</p> <p><b>Message Meaning:</b>    DRS/VPI received an unrecognized request code.</p> <p><b>System Action:</b>        DRS/VPI will put this printer in EDRAINED status.</p> <p><b>Required Action:</b>     This is a logic error within the DRS/VPI programs. Contact DRS technical support.</p>



---

<b>DRSV286E</b>	UNKNOWN REQUEST RECEIVED IN NETWORK SERVICES EXIT <b>Message Meaning:</b> DRS/VPI encountered an unknown or invalid request in the VTAM network services exit. <b>System Action:</b> DRS/VPI will ignore the request. <b>Required Action:</b> Contact DRS technical support.
<b>DRSV287E</b>	prtrid UNEXPECTED DFSYN REQUEST=req CONTROL=cntl req: VTAM request code cntl: VTAM control code <b>Message Meaning:</b> DRS/VPI encountered an unexpected data flow control request. <b>System Action:</b> DRS/VPI will put this printer in EDRAINED status. <b>Required Action:</b> Contact DRS technical support.
<b>DRSV288E</b>	prtrid UNKNOWN UNSUPPORTED REQUEST RECEIVED IN VTAM SCIP EXIT WITH CONTROL=cntl cntl: VTAM control code <b>Message Meaning:</b> DRS/VPI encountered an unknown or unsupported control request in the VTAM SCIP exit. <b>System Action:</b> DRS/VPI will put this printer in EDRAINED status. <b>Required Action:</b> Contact DRS technical support.
<b>DRSV289E</b>	prtrid SNA CANCEL RECEIVED <b>Message Meaning:</b> DRS/VPI encountered the SNA CANCEL command to end the current report. <b>System Action:</b> The report will be terminated according to the ERRACTN parameter in the printer definition. <b>Required Action:</b> Contact DRS technical support.
<b>DRSV290E</b>	prtrid INVALID DRS CALL REQUEST req req: The request code used for the DRS call. <b>Message Meaning:</b> DRS call module received an unknown request code. <b>System Action:</b> DRS/VPI will put this printer in EDRAINED status. <b>Required Action:</b> There is a logic error within the DRS/VPI programs. Contact DRS technical support.

---

<b>DRSV291E</b>	<p>prtrid DRS CALL ERROR FUNC=func DRSRC=rc OSRC=osrc ATTR=attr text</p> <p>func:                   The DRS function code</p> <p>drsrc:                   The DRS return code</p> <p>osrc:                    The DRS OS return code</p> <p>attr:                    The DRS attribute number in error</p> <p>text:                    The reason for the failure, if available. For example:</p> <ul style="list-style-type: none"> <li>• DEVICE UNAVAILABLE</li> <li>• DDNAME NOT SPECIFIED</li> <li>• DDNAME WAS NOT FOUND</li> <li>• INVALID FUNCTION CODE</li> <li>• INVALID SYSOUT PROGRAM NAME</li> <li>• INVALID SYSOUT FORM NUMBER</li> <li>• INVALID SYSOUT CLASS</li> <li>• MUTUALLY EXCLUSIVE KEYWORDS</li> <li>• INVALID PARM IN TEXT UNIT</li> <li>• INVALID KEY IN TEXT UNIT</li> <li>• INVALID PARM LIST FORMAT</li> <li>• MUTUALLY EXCLUSIVE KEY</li> <li>• MUTUALLY INCLUSIVE KEY ERROR</li> <li>• REQUIRED KEY NOT FOUND</li> <li>• REACHED ALLOCATION LIMIT</li> <li>• JES DESTINATION INVALID</li> <li>• USER UNAUTHORIZED FOR JES</li> <li>• CANNOT PROCESS JES REQUEST</li> <li>• UNABLE TO ESTABLISH ESTAE</li> </ul> <p><b>Message Meaning:</b> DRS call module received an error processing a request.</p> <p><b>System Action:</b> DRS/VPI will put this printer in EDRAINED status.</p> <p><b>Required Action:</b> Determine the reason for the failure based on the DRS return code, OS return code, DRS attribute number, and error text in the message. If the reason cannot be determined, contact DRS technical support.</p>
-----------------	---

---

<b>DRSV292E</b>	prtrid DRS INIT CALL RE-ISSUED USING DEFAULT SPOOLING ATTRIBUTES
	<p><b>Message Meaning:</b> DRS/VPI will re-issue the DRS INIT call using the default spooling attributes (DRIB) specified in the virtual printer definition. This message will follow a DRSV291E message. If the default spooling attributes have not been modified, the second INIT call will not be made. Spooling attributes can be modified by information obtained from DRS/PC or by DRS/VPI User Exit 07.</p> <p><b>System Action:</b> DRS/VPI will attempt the DRS INIT call using the default spooling attributes defined at virtual printer activation.</p> <p><b>Required Action:</b> Correct the spooling attributes sent from DRS/PC or specified in DRS/VPI User Exit 07.</p> <p><b>Note:</b> It may be necessary to issue a JES command to re-route this report on the spool from the default spooling attributes (e.g., CLASS, DESTINATION, FORM, WRITER, etc.) to the correct spooling attributes.</p>
<b>DRSV380I</b>	TCPIP CONNECTION SUCCESSFUL HOST=host,PORT=port
	<p>host: Indicates the name of the local host for TCP/IP.</p> <p>port: Indicates the PORT obtained to connect to TCP/IP.</p> <p><b>Message Meaning:</b> DRS has successfully connected to TCP/IP.</p> <p><b>System Action:</b> None.</p> <p><b>Required Action:</b> None.</p>
<b>DRSV381E</b>	prtrid TCPIP IBM MACRO ERROR TYPE=aaaa R15=r15 SOCKET=bbbb FUNCTION=cccc
	<p>aaaa: Indicates the 16-byte descriptor provided for the IBM TCP/IP request.</p> <p>bbbb: Indicates the current socket value.</p> <p>cccc: Indicates the function requested or zero.</p> <p><b>Message Meaning:</b> An error occurred when a TCP/IP IUCV macro was issued.</p> <p><b>System Action:</b> DRS/VPI will put this printer in EDRAINED status.</p> <p><b>Required Action:</b> Contact DRS technical support if you are unable to determine the cause of the error.</p>

---

**DRSV382E** prtrid TCPIP ERR=reason TYPE=aaa..aaa FUNCTION=bbbb ANS=ans1 ans2

reason: The reason for the error, such as:

- CONNECTION TO TCPIP FAILED
- NO PORT CURRENTLY AVAILABLE
- PC NUMBER INIT FAILED
- REJECT ERROR DETECTED

aaa..aaa: Indicates the 16-byte descriptor provided for the IBM TCP/IP request.

bbbb: Indicates the function code requested or zero.

ans1: The first word of the answer back area.

ans2: The second word of the answer back area.

**Message Meaning:** A TCP/IP request failed due to an error condition.

**System Action:** DRS/VPI will put this printer in EDRAINED status.

**Required Action:** Contact DRS technical support if you are unable to determine the cause of the error.

**DRSV383E** prtrid TCPIP CONNECTION SERVERED - REASON=reason text

reason: The reason code supplied by MVS TCP/IP, if applicable.

text: A description of why the connection was severed.

- IUCVCHECKRC = IUCV error detected before initial send
- SHUTTINGDOWN = TCP/IP service is being shut down (during connect)
- KILL-9 = Fatal error in IUCV communications
- KILL-37 = A software error occurred in MVS TCP/IP
- KILL-38 = TCP/IP service is being shut down (after connect)

**Message Meaning:** The MVS TCP/IP address space has severed the connection with the DRS virtual printer.

**System Action:** DRS/VPI will put this printer in EDRAINED status.

**Required Action:** Contact DRS technical support if you are unable to determine the cause of the error.

---

<b>DRSV384E</b>	prtrid TCPIP PROCESSING LOGIC ERROR - reason reason:	<p>A description of the logic error, such as:</p> <ul style="list-style-type: none"> <li>• ALL TPLS IN USE</li> <li>• API OPTIONS LENGTH ZERO</li> <li>• CANNOT LOCATE TPL</li> <li>• CONFIRM/DISCONNECT MISSING</li> <li>• DATA INDICATION MISSING</li> <li>• DIRSRV - HOST NAME ZERO LENGTH</li> <li>• DIRSRV - INVALID ADDR LENGTH</li> <li>• ENDPOINT STATE ERROR</li> <li>• ERROR IN TCP SUPPORT MODULE</li> <li>• ERROR OBTAINING HOST ADDRESS</li> <li>• ERROR OBTAINING SOCKET NUMBER</li> <li>• LOCAL HOST ADDR LEN IS ZERO</li> <li>• MAXIMUM SOCKET NOT ALLOWED</li> <li>• PENDING CONNECTION MISSING</li> <li>• REQUESTED SOCKET NOT OBTAINED</li> <li>• SEND BUFFER NOT ALLOCATED</li> <li>• TINFO RETURNED ZERO LENGTH</li> <li>• TP OPTIONS LENGTH ZERO</li> <li>• TPL NOT READY FOR TCHECK</li> <li>• UNKNOWN TCPIP TYPE</li> </ul>
	<b>Message Meaning:</b>	A logic error has occurred when processing a dataset for a DRS/TCPIP virtual printer.
	<b>System Action:</b>	DRS/VPI will put this printer in EDRAINED status.
	<b>Required Action:</b>	Contact DRS technical support.
<b>DRSV385E</b>	UNKNOWN UNSUPPORTED INVALID INTERRUPT TYPE aa RECEIVED IN TCPIP IUCV EXIT UWORD=bbbbbbbb EXTB=ccccccc	
	aa:	The interrupt type received in the exit.
	bbbbbbbb:	The value of the user word passed to the exit.
	ccccccc:	The value of the first 8 bytes of the external interrupt buffer.
	<b>Message Meaning:</b>	The interrupt type received in the IUCV exit was unknown to DRS, was unsupported by DRS, or the external interrupt buffer contained invalid data.
	<b>System Action:</b>	The interrupt is ignored.
	<b>Required Action:</b>	Contact DRS technical support.

---

---

<b>DRSV386E</b>	DRS/TCPIP INTERFACE DISABLED - TCPIP type REQUEST TERMINATED ABNORMALLY
type:	“IBM IUCV” or “IBM V3 API” or “IBM HPNS” or “ICS API”
<b>Message Meaning:</b>	An abend occurred while a system request to TCP/IP was in progress. The TCP/IP interface will be disabled.
<b>System Action:</b>	All active TCP/IP virtual printers are EDRAINED.
<b>Required Action:</b>	Contact DRS technical support if you are unable to determine the cause of the error.
<b>DRSV387E</b>	TCPIP CONNECTION NOT AVAILABLE - REASON=aaa
reason:	The reason code supplied by MVS TCP/IP, such as: <ul style="list-style-type: none"> <li>• TIMEOUT ON SELECT CALL</li> <li>• EXCEPTION MASK NOT SET</li> </ul>
<b>Message Meaning:</b>	An error occurred during the connection process with the remote host.
<b>System Action:</b>	DRS/VPI will put this printer in EDRAINED status.
<b>Required Action:</b>	Contact DRS technical support.
<b>DRSV388I</b>	MAXIMUM TCPIP CONNECTIONS IN INITIALIZATION STATUS - ADDITIONAL CONNECTION REQUESTS DELAYED
<b>Message Meaning:</b>	DRS has reached the maximum number of concurrent initialization connection requests.
<b>System Action:</b>	DRS will queue further connection requests until a current pending connection request is completed.
<b>Required Action:</b>	None.
<b>DRSV389I</b>	prtrid LPD CONTROL RECORD IGNORED WARNING - TYPE=type REASON=reason VALUE=value
type:	Control record type.
reason:	The reason the control record is invalid, such as: <ul style="list-style-type: none"> <li>• RECORD LENGTH ERROR</li> <li>• VALUE EXCEEDS MAX</li> <li>• VALUE NOT NUMERIC</li> <li>• VALUE LESS THAN 0</li> </ul>
value:	The first 28 bytes of the control record.
<b>Message Meaning:</b>	DRS detected an invalid control record while processing the control file.
<b>System Action:</b>	DRS will either ignore the control record or adjust its value to the maximum allowed.
<b>Required Action:</b>	None.

---

<b>DRSV390I</b>	prtrid POSITIVE NEGATIVE ACK SENT - text text:	The text returned to the remote host.
	<b>Message Meaning:</b>	DRS/VPI has returned error text to a remote host. If a negative acknowledgement was sent, the file or command was rejected. If a positive acknowledgement was sent, error text was returned to the remote host to indicate that the command received is not supported.
	<b>System Action:</b>	The connection with the remote host will be terminated.
	<b>Required Action:</b>	The command or file being sent by the remote TCP/IP host should be corrected. If the problem cannot be eliminated, contact DRS technical support for assistance.
<b>DRSV391E</b>	TCPMRD INTERVAL EXPIRED - TCP/IP CONNECTION TIMED OUT	
	<b>Message Meaning:</b>	No response was returned to an outstanding TCP/IP request during the interval specified by the TCPMRD keyword. DRS/VPI terminates the connection with the TCP/IP remote host.
	<b>System Action:</b>	DRS/VPI will put this printer in EDRAINED status.
	<b>Required Action:</b>	Determine the reason the remote host stopped responding to DRS/VPI.
<b>DRSV392E</b>	DRS/TCPIP TIMER IRB EXIT UNABLE TO LOCATE DRS PRINTER CONTROL BLOCKS	
	<b>Message Meaning:</b>	The DRS/TCPIP timer IRB exit was unable to locate the DRS printer control blocks.
	<b>System Action:</b>	DRS/VPI will put this printer in EDRAINED status.
	<b>Required Action:</b>	This is a logic error within the DRS/VPI programs. Contact DRS technical support.

---

**DRSV393E**      aaaa EXIT PARAMETER RECEIVED IN TCPIP ASYNCHRONOUS  
EXIT - TYPE=type EVENT=event

aaaa:                      The type of asynchronous exit parameter  
received.

bbbbbbb:                The event for which the exit was scheduled,  
such as:

Type	Type Description	Event	Event Description
0001	Protocol event	00	Connect indication
0001	Protocol event	04	Confirm indication
0001	Protocol event	08	Data indication
0001	Protocol event	12	Expedited data indication
0001	Protocol event	16	Datagram error indication
0001	Protocol event	20	Disconnect indication
0001	Protocol event	24	Release indication
0002	Endpoint completion		
0003	TPEND (shutdown)	00	Operator drained subsystem
0003	TPEND (shutdown)	04	Operator stopped subsystem
0003	TPEND (shutdown)	08	Subsystem abended
0004	Synchronous error		
0005	APEND (shutdown)	04	Operator drained subsystem
0003	APEND (shutdown)	04	Operator stopped subsystem
0003	APEND (shutdown)	08	Subsystem abended

**Message Meaning:**      DRS/VPI has received an unknown request in  
the ICS asynchronous event exit.

**System Action:**        The request will be ignored.

**Required Action:**      Contact DRS technical support.



---

<b>DRSV394E</b>	<p>prtrid TCPIP ICS ERROR=reason TYPE=aaa..aaa R15=r15 R0=r0 RTNCD=rc</p> <p>reason:                   The reason for the error, such as:</p> <ul style="list-style-type: none"> <li>• API NOT CONFIGURED</li> <li>• API NOT ACTIVE</li> <li>• API NOT READY FOR SERVICE</li> <li>• API SUBSYSTEM STOPPED</li> <li>• API SUBSYSTEM DRAINING</li> <li>• TIME LIMIT EXCEEDED</li> <li>• REQUEST EXCEEDED TIME LIMIT</li> <li>• HOST DOES NOT EXIST</li> <li>• INSUFFICIENT DATA: NO HOST</li> <li>• SUBSYSTEM NOT INITIALIZED</li> <li>• DISCONNECT RECEIVED</li> <li>• TCONNECT: ADDRESS IN USE</li> </ul> <p>aaa..aaa:                The type of macro issued (AOPEN, TCLOSE, etc.)</p> <p><b>Message Meaning:</b>    An ICS macro failed due to an error condition.</p> <p><b>System Action:</b>      DRS/VPI will put this printer in EDRAINED status.</p> <p><b>Required Action:</b>    If unable to determine the cause of the error, contact DRS technical support.</p>
<b>DRSV395E</b>	<p>prtrid TCPIP ICS ERROR MESSAGE errmsg</p> <p>errmsg:                 Error text supplied by Interlink Computer Sciences.</p> <p><b>Message Meaning:</b>    An error was detected while attempting to use an ICS macro. Multiple DRSV395E messages will be issued to display the error text supplied by ICS. The DRSV394E message will also be issued for the error.</p> <p><b>System Action:</b>      DRS/VPI will put this printer in EDRAINED status.</p> <p><b>Required Action:</b>    If unable to determine the cause of the error, contact DRS technical support.</p>
<b>DRSV398I</b>	<p>prtrid TCPIP CONNECTION ACCEPTED FROM REMOTE HOST IP ADDRESS=addr PORT=port</p> <p>addr:                    IP address of the remote host</p> <p>port:                    The port number used by the remote host to connect to DRS/VPI</p> <p><b>Message Meaning:</b>    Indicates a connection has been established between the remote host and DRS/VPI.</p> <p><b>System Action:</b>      None.</p> <p><b>Required Action:</b>    None.</p>

---

---

**DRSV399I** prtrid TCPIP PRINT JOB REQUEST ACCEPTED FROM REMOTE  
HOST IP ADDRESS=addr PORT=port

addr: IP address of the remote host

port: The port number used by the remote host to  
connect to DRS/VPI

**Message Meaning:** DRS/VPI has accepted a “receive print job”  
request from a remote host.

**System Action:** None.

**Required Action:** None.

**DRSV520E** prtrid TCPIP QUEUE DYNAMIC ALLOCATION FAILURE RC=rc  
EC=eeee IC=iiii

rc: Dynamic allocation return code.

eeee: Dynamic allocation error code.

iiii: Dynamic allocation inf code.

**Message Meaning:** Dynamic allocation for the TCPIP staging  
dataset has failed. See OS/VS2 SPL: Job  
Management.

**System Action:** DRS will put this printer in EDRAINED status.

**Required Action:** Correct error causing the dynamic allocation  
failure.

**DRSV521E** prtrid TCPIP QUEUE DYNAMIC UNALLOCATION FAILURE  
RC=rc EC=eeee IC=iiii

rc: Dynamic allocation return code.

eeee: Dynamic allocation error code.

iiii: Dynamic allocation inf code.

**Message Meaning:** Dynamic unallocation of the TCPIP staging  
dataset has failed. See OS/VS2 SPL: Job  
Management.

**System Action:** DRS will put this printer in EDRAINED status.

**Required Action:** Correct error causing the dynamic unallocation  
failure.

**DRSV522E** prtrid TCPIP QUEUE OPEN|CLOSE FAILURE DDNAME=aaaaaaaa  
aaaaaaaa: DDNAME of the TCPIP staging dataset.

**Message Meaning:** OPEN or CLOSE of the TCPIP staging dataset  
has failed.

**System Action:** DRS will put this printer in EDRAINED status.

**Required Action:** Contact DRS technical support.

---

<b>DRSV523E</b>	<p>prtrid TCPIP QUEUE DCB ABEND CODE=sss-ss  sss-xx: System abend code - reason code.</p> <p><b>Message Meaning:</b> An abend occurred during OPEN/CLOSE/EOV processing for the TCPIP staging dataset.</p> <p><b>System Action:</b> DRS will put this printer in EDRAINED status.</p> <p><b>Required Action:</b> If the abend is an S213 or S214, the TCPIP staging dataset has been deleted while DRS is executing. This is generally caused when a "clean-up" job is run on a CPU other than the one on which DRS is running, in a shared DASD environment.</p>
<b>DRSV524E</b>	<p>prtrid I/O ERROR DURING TCPIP QUEUE PROCESSING</p> <p><b>Message Meaning:</b> An I/O error occurred while writing to the TCPIP staging dataset. Message DRSV525E will follow.</p> <p><b>System Action:</b> DRS will put this printer in EDRAINED status.</p> <p><b>Required Action:</b> If possible, correct the problem which caused the I/O error. Then issue a START command for the printer.</p>
<b>DRSV525E</b>	<p>prtrid i/o error synad message</p> <p><b>Message Meaning:</b> This is the I/O error synad message for DRSV524E.</p> <p><b>System Action:</b> See message DRSV524E.</p> <p><b>Required Action:</b> See message DRSV524E.</p>
<b>DRSV526E</b>	<p>INSUFFICIENT VIRTUAL STORAGE FOR TCPIP QUEUE BUFFERS</p> <p><b>Message Meaning:</b> There is insufficient virtual storage in the DRS address space for staging dataset buffers.</p> <p><b>System Action:</b> DRS will put this printer in EDRAINED status.</p> <p><b>Required Action:</b> Increase the REGION parameter in the DRS execution JCL and reinitiate DRS.</p>
<b>DRSV528E</b>	<p>prtrid TCPIP QUEUE INVALID DATASET DETECTED</p> <p><b>Message Meaning:</b> DRS/TCPIP detected that the staging dataset for the printer has been corrupted.</p> <p><b>System Action:</b> DRS will put this printer in EDRAINED status.</p> <p><b>Required Action:</b> Determine if the staging dataset was deleted or overwritten by another CPU in the configuration. If not, contact DRS technical support.</p>
<b>DRSV529E</b>	<p>prtrid TCPIP QUEUE OUTPUT BUFFER OVERRUN</p> <p><b>Message Meaning:</b> DRS/TCPIP determined that the queue output buffer was too small to contain a decompressed record.</p> <p><b>System Action:</b> DRS will put this printer in EDRAINED status.</p> <p><b>Required Action:</b> Contact DRS technical support.</p>

---

---

<b>DRSV600E</b>	LOG ALLOCATION FAILURE R15=r15 EC=eeee IC=iiii r15: Dynamic allocation return code. eeee: Dynamic allocation error code. iiii: Dynamic allocation info code (see OS/VS2 SPL: Job Management).
	<b>Message Meaning:</b> An error has occurred while allocating the SYSOUT dataset for the DRS/VPI log.
	<b>System Action:</b> Logging is disabled.
	<b>Required Action:</b> Determine the reason for the failure, based on the R15/EC/IC codes. If the reason cannot be determined, contact DRS technical support.
<b>DRSV601E</b>	LOG OPEN FAILURE <b>Message Meaning:</b> An error occurred opening the DRS/VPI log dataset.
	<b>System Action:</b> Logging is disabled.
	<b>Required Action:</b> If possible, correct the problem which caused the OPEN error. To enable logging the DRS/VPI address space must be restarted.
<b>DRSV602E</b>	LOG CLOSE FAILURE <b>Message Meaning:</b> An error has occurred on a close for the DRS/VPI log dataset.
	<b>System Action:</b> Logging is disabled.
	<b>Required Action:</b> Determine the reason for the failure. If the reason cannot be determined, contact DRS technical support.
<b>DRSV603R</b>	LOG CLOSE COMPLETE <b>Message Meaning:</b> A CLOSELOG command has been processed. <b>System Action:</b> DRS/VPI has responded to a CLOSELOG command by closing the DRS VPI log SYSOUT dataset and allocating a new SYSOUT dataset. Recording of log data will continue on the new dataset. The old SYSOUT dataset will be available for printing.
	<b>Required Action:</b> None.
<b>DRSV604R</b>	LOG DATASET IS FULL; LOGGING CONTINUES AT START OF DATASET <b>Message Meaning:</b> The DASD dataset being used for the DRS/VPI log is full.
	<b>System Action:</b> DRS/VPI will begin recording log data at the start of the dataset. Old data will be lost.
	<b>Required Action:</b> None.

---

<b>DRSV605E</b>	LOG DCB ABEND CODE=xxx-xx xxx-xx: Abend/return codes <b>Message Meaning:</b> An error occurred during OPEN/CLOSE/EOV processing for the DRS/VPI log dataset. <b>System Action:</b> Logging is disabled. <b>Required Action:</b> Determine the reason for the failure. If the reason cannot be determined, contact DRS technical support.
<b>DRSV606E</b>	LOG I/O ERROR xxxxxxxxxxxx xxxxxxxxxx: SYNAD message text. <b>Message Meaning:</b> An I/O error occurred while writing to the DRS/VPI log dataset. <b>System Action:</b> Logging continues. <b>Required Action:</b> Determine the cause of the I/O error and correct it.
<b>DRSV620E</b>	prtrid SNAP ALLOCATION FAILURE R15=r15 EC=eeee IC=iiii r15: Dynamic allocation return code. eeee: Dynamic allocation error code. iiii: Dynamic allocation info code (see OS/VS2 SPL: Job Management). <b>Message Meaning:</b> An error has occurred while attempting to allocate a snap dump dataset. <b>System Action:</b> The snap request is ignored. <b>Required Action:</b> Correct the error causing the allocation failure, based on the R15, EC, and IC values.
<b>DRSV621E</b>	prtrid SNAP OPEN FAILURE <b>Message Meaning:</b> An error occurred opening the snap dump dataset. <b>System Action:</b> DRS/VPI will put this printer in EDRAINED status. <b>Required Action:</b> If possible, correct the problem which caused the OPEN error. Then issue a START command for the printer.
<b>DRSV622E</b>	prtrid SNAP DCB ABEND CODE=xxx-xx xxx-xx: Abend/return codes. <b>Message Meaning:</b> An abend occurred during OPEN/CLOSE/EOV processing for the snap dump dataset. <b>System Action:</b> DRS/VPI will put this printer in EDRAINED status. <b>Required Action:</b> If possible, correct the problem which caused the abend. Then issue a START command for the printer.

---

---

**DRSV623E** prtrid SNAP I/O ERROR xxxxxxxxxxxxxxxx  
 xxxxxxxxxxxxxxxx SYNAD message text  
**Message Meaning:** An I/O error occurred while writing to the snap dataset.  
**System Action:** The snap request is ignored.  
**Required Action:** Contact DRS technical support.

**DRSV800E** DISPATCHER SUBTASK ABEND SYS=sss USR=uuu  
 sss: System abend code.  
 uuu: User abend code.  
**Message Meaning:** Dispatcher subtask has abended.  
**System Action:** DRS/VPI messages DRSV809E and DRSV810E will follow. DRS/VPI will abend with a U100 abend.  
**Required Action:** Contact DRS technical support.

**DRSV801E** COMMUNICATIONS SUBTASK ABEND SYS=sss USR=uuu  
 sss: System abend code.  
 uuu: User abend code.  
**Message Meaning:** Operator communications subtask has abended.  
**System Action:** DRS/VPI messages DRSV809E and DRSV810E will follow. DRS/VPI will abend with a U100 abend.  
**Required Action:** Contact DRS technical support.

**DRSV802E** LOG SUBTASK ABEND SYS=sss USR=uuu  
 sss: System abend code.  
 uuu: User abend code.  
**Message Meaning:** The DRS/VPI log subtask has abended.  
**System Action:** DRS/VPI messages DRSV809E and DRSV810E will follow. DRS/VPI logging will be disabled.  
**Required Action:** If logging is required, restart DRS/VPI.

**DRSV803E** DRIVER SUBTASK ABEND SYS=sss USR=uuu  
 sss: System abend code.  
 uuu: User abend code.  
**Message Meaning:** The DRS/VPI driver subtask has abended.  
**System Action:** DRS/VPI messages DRSV809E and DRSV810E will follow. DRS/VPI will abend with a U100 abend.  
**Required Action:** Contact DRS technical support.

---

<b>DRSV804E</b>	TCP/IP SUBTASK ABEND SYS=sss USR=uuu sss: System abend code. uuu: User abend code. <b>Message Meaning:</b> The DRS/TCPIP subtask has abended. <b>System Action:</b> Messages DRSV809E and DRSV810E will follow. The DRS/TCPIP product will be disabled. <b>Required Action:</b> Contact DRS technical support.
<b>DRSV805E</b>	SERVER SUBTASK ABEND SYS=sss USR=uuu sss: System abend code. uuu: User abend code. <b>Message Meaning:</b> The DRS/VPI Server subtask has abended. <b>System Action:</b> DRS/VPI messages DRSV809E and DRSV810E will follow. The SERVER connections will be lost. <b>Required Action:</b> Contact DRS technical support.
<b>DRSV807E</b>	prtrid VIRTUAL PRINTER SUBTASK ABEND SYS=sss USR=uuu sss: System abend code. uuu: User abend code. <b>Message Meaning:</b> The subtask for the specified virtual printer has abended. <b>System Action:</b> DRS/VPI messages DRSV809E and DRSV810E will follow. DRS/VPI will put the printer in EDRAINED status. <b>Required Action:</b> Contact DRS technical support.
<b>DRSV808E</b>	prtrid USER EXIT ABEND - EXITID=nn SYS=sss USR=uuu, EXIT DISABLED nn: User exit number. sss: System abend code. uuu: User abend code. <b>Message Meaning:</b> The DRS/VPI user exit module identified in the message has abended. <b>System Action:</b> DRS/VPI messages DRSV809E and DRSV810E will follow. DRS/VPI will disable the exit, and the printer will continue to process work, but without using this exit. <b>Required Action:</b> If the exit is required, correct the error in the exit and terminate and restart DRS/VPI to put the corrected version of the exit into effect. <b>Note:</b> This message will only appear if the exit has recovery on. If recovery is off, message DRSV807E will appear, and the printer will be put in EDRAINED status.

---

---

**DRSV809E** prtrid PSW=pppppppp pppppppp MODULE = mmmmmmmm + nnnn  
pppppppp pppppppp: PSW at time of abend.  
mmmmmmmm: Name of the abending module.  
nnnn: Hex offset into abending module.  
**Message Meaning:** A DRS/VPI module has abended. This message and message DRSV810E give debugging information about the abend.  
**System Action:** DRS/VPI will put this printer in EDRAINED status.  
**Required Action:** None.

**DRSV810E** prtrid R0-R7 xxxxxxxx xxxxxxxx (etc.)  
prtrid R8-R15 xxxxxxxx xxxxxxxx (etc.)  
xxxxxxx: The contents of a register at the time of an abend.  
**Message Meaning:** A DRS/VPI module has abended. This message and message DRSV809E give debugging information about the abend.  
**System Action:** None.  
**Required Action:** Contact DRS technical support.

**DRSV812E** UNABLE TO IDENTIFY ABENDED USER SUBTASK  
**Message Meaning:** The DRS/VPI was not able to find the printer control blocks associated with the task which is abending.  
**System Action:** None.  
**Required Action:** Contact DRS technical support.

**DRSV820N** INVALID CONTROL STATEMENT (mbrname) - error desc  
mbrname: The name of the member found to be in error.  
error desc: The description of the error.  
**Message Meaning:** An invalid control statement was found.  
**System Action:** The requested printer will not be activated. Message DRSV821N will follow.  
**Required Action:** Correct control statement error and activate the printer.



---

**DRSV821N** PRINTER ADD FOR mbrname FAILED - failure reason

mbrname: The name of the member that was not added.

failure reason: Reason for error. Examples of reasons include:  
CONTROL STATEMENT (nnn) STARTS AFTER  
POS. 16  
DRSVLIB DCB ABEND xxx-xx  
DRSVLIB DCB LRECL NOT 80  
DRSVLIB I/O ERROR  
DRSVLIB OPEN FAILURE  
ENDING QUOTE MISSING KWD=kwd  
EXPECTED CONTINUATION NOT RECEIVED  
INSUFFICIENT VIRTUAL STORAGE AVAILABLE  
MAXIMUM NUMBER OF PRINTERS EXCEEDED  
MEMBER ALREADY ACTIVATED  
MEMBER NOT FOUND  
MISSING COMMA AFTER KWD: keyword  
QUOTE NOT FOLLOWED BY COMMA OR SPACE  
KWD=kwd  
UNBALANCED PARENTHESES FOR KWD=kwd

**Message Meaning:** DRS/VPI printer activation has failed.

**System Action:** The requested printer will not be activated.

**Required Action:** Correct the error based upon the failure reason and activate the printer.

**Note:** Message DRSV820N precedes this message.

**DRSV822N** drsvlib i/o error synad message

**Message Meaning:** This is the I/O error SYNAD message for DRSV821N above if an I/O error occurred.

**System Action:** The requested printer will not be activated.

**Required Action:** Correct the error based upon the I/O error SYNAD message and activate the printer.

**DRSV823N** KEYWORD PARSE FAILURE MEMBER(mbrname) - xxxxxxxxxx

mbrname: The name of the member that was not added.

xxxxxxx: Keyword and keyword value in error.

**Message Meaning:** An invalid value has been specified for a printer keyword.

**System Action:** The printer will not be activated.

**Required Action:** Correct the keyword value and activate the printer.

---

---

**DRSV824N** stcname prtrid INVALID DSNAME=dsname REASON=reason  
dsname: Up to 44 characters of the dataset name.  
reason: Reason the dataset name is invalid.  
**Message Meaning:** DRS/VPI has encountered an error in the dataset name after symbolic substitution is complete.  
**System Action:** If the error occurs during activation, the printer will not be activated. If the error occurs during report initialization, the printer will be EDRAINED.  
**Required Action:** Correct the DSN keyword in the virtual printer definition and re-activate.

**DRSV825N** prtrid INVALID JOBNAME VALUE=name ACTION=ERROR|IGNORE REASON=rsn  
name: Value of JOBNAME keyword after substitution.  
rsn: The reason the JOBNAME is invalid; possible reasons include:  
INVALID SYMBOLIC VARIABLE  
NOT ALPHANUMERIC OR NATIONAL  
FIRST CHARACTER IS NUMERIC  
**Message Meaning:** The JOBNAME keyword value is invalid after symbolic substitution.  
**System Action:**

- If ACTION=ERROR occurs during printer activation, the activation will fail.
- If ACTION=ERROR occurs when receiving a print file, the printer will be EDRAINED.
- If ACTION=IGNORE occurs when receiving a print file, the JOBNAME will be ignored and the SYSOUT file will have the name of the DRS/VPI job or started task.

**Required Action:** Correct the JOBNAME keyword value and re-activate the printer definition.

**DRSV830E** INTERNAL LOGIC ERROR - MESSAGE ID ### IS UNDEFINED  
###: A DRS/VPI message number.  
**Message Meaning:** A DRS/VPI module has attempted to issue a DRS/VPI message with an undefined message number.  
**System Action:** The message is ignored.  
**Required Action:** Contact DRS technical support.

---

<b>DRSV840E</b>	LOGIC ERROR IN MODULE name desc name: The module name ( <b>DV33OLST</b> or <b>DV33LU60</b> ). desc: <b>If DV33OLST:</b> OUTPUT statement name. <b>If DV33LU60:</b> <ul style="list-style-type: none"> <li>• DECOMP ERROR - INPUT ADDR ZERO</li> <li>• DECOMP ERROR - INPUT LEN ZERO</li> <li>• DECOMP ERROR - REC RDW ZERO</li> <li>• DECOMP ERROR - REC RDW LONG</li> <li>• INVALID TYPE IN .CMP PREFIX</li> <li>• DECOMP BUFF LEN ERR - COMP REC</li> <li>• DECOMP BUFF LEN ERR - NORM REC</li> <li>• DV33COMP RETURN CODE NOT ZERO</li> <li>• .CMP DECOMPRESSION LEN IS ZERO</li> </ul>
	<b>Message Meaning:</b> <b>If DV33OLST:</b> DRS/VPI has attempted to locate an OUTPUT statement defined in either the DRS/VPI started task JCL, or as an OUTPUT reference member in the DRS/VPI control library. <b>If DV33LU60:</b> A decompression error occurred.
	<b>System Action:</b> DRS/VPI will put this printer in EDRAINED status.
	<b>Required Action:</b> Contact DRS technical support.
<b>DRSV899E</b>	XXXXXXXXXX XXXXXXXXXX: The SVC 99 text unit in error, including the DDNAME, the text unit key, the text unit number, and, depending on the error code, the text unit length and text unit parm.
	<b>Message Meaning:</b> An invalid text unit was specified when attempting to use SVC 99 to dynamically allocate a dataset.
	<b>System Action:</b> Dependent on the dataset for which allocation failed.
	<b>Required Action:</b> Refer to additional DRS/VPI messages to determine the appropriate action.
<b>DRSV900R</b>	COMMAND INVALID - command command: The DRS/VPI command that was issued. <b>Message Meaning:</b> An unknown DRS/VPI command was issued. <b>System Action:</b> The command is ignored. <b>Required Action:</b> Verify that the correct command was issued.

---

---

**DRSV901R**    COMMAND AMBIGUOUS - command  
command:                    The DRS/VPI command that was issued.  
**Message Meaning:**    Insufficient characters were entered to make  
the command unique to DRS/VPI.  
**System Action:**        The command is ignored.  
**Required Action:**      Enter additional command characters.

**DRSV902R**    PARAMETER LENGTH ERROR  
**Message Meaning:**    A parameter exceeded 50 characters.  
**System Action:**        The command is ignored.  
**Required Action:**      Shorten parameter as required.

**DRSV903R**    COMMAND HAS TOO MANY PARAMETERS  
**Message Meaning:**    Too many command parameters were specified.  
**System Action:**        The command is ignored.  
**Required Action:**      Remove unneeded parameters.

**DRSV904R**    PARAMETER INVALID - parm  
parm:                        the value of the parameter used on the command.  
**Message Meaning:**    A parameter on a command was invalid.  
**System Action:**        The command is ignored.  
**Required Action:**      Issue the command with valid parameters.

**DRSV905R**    NON-SPECIFIC PRINTER-ID NOT ALLOWED  
**Message Meaning:**    A generic prtrid was specified, but the command  
requires a specific prtrid.  
**System Action:**        The command is ignored.  
**Required Action:**      Specify a specific prtrid.

**DRSV906R**    REQUIRED PRINTER-ID|OUTREF NOT SPECIFIED  
**Message Meaning:**    Required prtrid or OUTREF not specified.  
**System Action:**        The command is ignored.  
**Required Action:**      Specify prtrid.

**DRSV907R**    SPECIFIED PRINTER-ID|OUTREF NOT FOUND - aaaaaaaa  
aaaaaaa:                    Printer ID or OUTREF name.  
**Message Meaning:**    Specified prtrid or output reference name was  
not found.  
**System Action:**        The command is ignored.  
**Required Action:**      Specify a valid prtrid.

---

<b>DRSV908R</b>	SPECIFIED PRINTER STATUS INVALID - status status: Status that was specified. <b>Message Meaning:</b> An invalid status request was made. <b>System Action:</b> The command is ignored. <b>Required Action:</b> Correct the status request.
<b>DRSV909R</b>	NO PRINTERS OUTREF CURRENTLY DEFINED IN SYSTEM <b>Message Meaning:</b> No printers or output reference statements currently exist in DRS/VPI. <b>System Action:</b> The command is ignored. <b>Required Action:</b> None.
<b>DRSV910R</b>	NO PRINTERS WITH REQUESTED STATUS FOUND - S=status status: Status that was specified. <b>Message Meaning:</b> No printers with the status specified in the command currently exist in DRS/VPI. <b>System Action:</b> The command is ignored. <b>Required Action:</b> None.
<b>DRSV911N</b>	prtrid PRINTER SUCCESSFULLY ACTIVATED REACTIVATED <b>Message Meaning:</b> Printer has been activated in response to an ACTIVATE command, or reactivated in response to a REACTIVATE command. <b>System Action:</b> None. <b>Required Action:</b> None.
<b>DRSV912R</b>	NO PRINTERS OUTREF WITH REQUESTED PARTIAL NAME FOUND - generic name generic name: Generic name that was specified. <b>Message Meaning:</b> No printers OUTREFs exist that match the partial name. <b>System Action:</b> The command is ignored. <b>Required Action:</b> None.
<b>DRSV913R</b>	prtrid INACTIVATION REQUEST SCHEDULED <b>Message Meaning:</b> Inactivation has been scheduled in response to an INACTIVATE command. <b>System Action:</b> None. <b>Required Action:</b> None.

---

<b>DRSV914R</b>	prtrid REQUIRED SET PARAMETER(S) NOT SPECIFIED
	<b>Message Meaning:</b> A SET command was issued, but no parameters were specified on the command.
	<b>System Action:</b> The command is ignored.
	<b>Required Action:</b> Issue the SET command with the required parameter(s).
<b>DRSV915R</b>	prtrid SET PARAMETER AMBIGUOUS - set type
	set type: The requested set parameter.
	<b>Message Meaning:</b> Insufficient characters were entered to uniquely identify the set command type.
	<b>System Action:</b> The command is ignored.
	<b>Required Action:</b> Enter more characters of the set type.
<b>DRSV917R</b>	SSET,TCPIP=D E F REJECTED, DRS/TCPIP status
	status: Status of the DRS/TCPIP system, such as:
	<ul style="list-style-type: none"> <li>• ABENDED</li> <li>• EXPIRED</li> <li>• FORCED TO END</li> <li>• NOT INSTALLED</li> </ul>
	<b>Message Meaning:</b> The SSET command for DRS/TCPIP was rejected due to the reason indicated in the message.
	<b>System Action:</b> The command is ignored.
	<b>Required Action:</b> If the status is ABENDED or FORCED TO END, restart DRS/VPI. If the status is EXPIRED or NOT INSTALLED, verify the KEYLPD keyword value and restart DRS/VPI.
<b>DRSV918R</b>	REQUIRED SSET PARAMETER(S) NOT SPECIFIED
	<b>Message Meaning:</b> The SSET command was issued without the required parameters.
	<b>System Action:</b> The command is ignored.
	<b>Required Action:</b> Enter the required parameters on the SSET command.
<b>DRSV919R</b>	SSET PARAMETER AMBIGUOUS - parameter
	<b>Message Meaning:</b> Insufficient characters were entered to uniquely identify the SSET parameter.
	<b>System Action:</b> The command is ignored.
	<b>Required Action:</b> Enter more characters of the SSET parameter.

---

<b>DRSV920R</b>	cmdtarget cmdtype COMMAND ACKNOWLEDGED
	cmdtarget: The "target" of the command, such as printer id, SYSTEM, a printer status, etc.
	cmdtype: The type of command, such as SNAP, DISPLAY, etc.
	<b>Message Meaning:</b> DRS/VPI has acknowledged the command.
	<b>System Action:</b> DRS/VPI will schedule and process the command.
	<b>Required Action:</b> None.
<b>DRSV921R</b>	DISPLAY PARAMETER AMBIGUOUS - parameter
	parameter: The requested display parameter.
	<b>Message Meaning:</b> Insufficient characters were entered to uniquely identify the display parameter.
	<b>System Action:</b> None.
	<b>Required Action:</b> Enter more characters of the display parameter.
<b>DRSV924R</b>	MESSAGE NOT LOGGED, LOGGING NOT ACTIVE
	<b>Message Meaning:</b> A log command was issued, but DRS/VPI logging is not currently active.
	<b>System Action:</b> The log command is ignored.
	<b>Required Action:</b> If desired, activate DRS/VPI logging.
<b>DRSV925R</b>	MESSAGE SUCCESSFULLY LOGGED
	<b>Message Meaning:</b> Requested message was successfully logged in response to a log command.
	<b>System Action:</b> None.
	<b>Required Action:</b> None.
<b>DRSV926R</b>	CLOSELOG COMMAND REJECTED, LOGGING NOT ACTIVE
	<b>Message Meaning:</b> A closelog command was issued, but logging is not currently active.
	<b>System Action:</b> The command is rejected.
	<b>Required Action:</b> None.
<b>DRSV927R</b>	CLOSELOG COMMAND REJECTED, LOG DATASET WAS PRE-ALLOCATED
	<b>Message Meaning:</b> A CLOSELOG command was issued, but the log dataset is a pre-allocated DASD dataset rather than a SYSOUT dataset.
	<b>System Action:</b> The CLOSELOG command is ignored.
	<b>Required Action:</b> None.

---

---

**DRSV928R** LOG=Y REJECTED, LOG SUBTASK HAS TERMINATED  
**Message Meaning:** An SSET command with LOG=Y was issued, but the log subtask has terminated.  
**System Action:** The LOG=Y option is ignored.  
**Required Action:** None.

**DRSV930R** prtrid START COMMAND REJECTED, reject reason  
reject reason: One of the following:  
DRS/VPI TERMINATION IN PROGRESS  
INACTIVATION PENDING  
**Message Meaning:** A START command was issued for a printer, but was rejected due to one of the reasons listed above.  
**System Action:** The START command is ignored.  
**Required Action:** None.

**DRSV931R** xxxxxxxx SNAP DUMP COMPLETE  
xxxxxxx: Printer name or "SYSTEM".  
**Message Meaning:** A snap dump has been successfully completed for the requested printer or for the DRS/VPI system in response to a SNAP command.  
**System Action:** None.  
**Required Action:** None.

**DRSV932R** xxxxxxxx SNAP DUMP NOT SUCCESSFUL RC=rc  
xxxxxxx: Printer name or "SYSTEM".  
rc: Snap dump return code (see OS/VS2 MVS Supervisor Services)  
**Message Meaning:** DRS/VPI attempted to create a snap dump in response to a SNAP command, but an error occurred in processing. The error occurred either allocating the snap dataset, opening the snap dataset, or issuing the SNAP macro. If allocation failed, message DRSV620E will precede this message.  
**System Action:** The SNAP command is ignored.  
**Required Action:** Contact DRS technical support.

**DRSV935R** DRS/VPI FAST TERMINATION REQUEST ACKNOWLEDGED  
**Message Meaning:** DRS/VPI has acknowledged a "P DRSV" command.  
**System Action:** All printers are drained via a "FORCE" command.  
**Required Action:** None.



---

<b>DRSV936R</b>	DRS/VPI TERMINATION REQUEST ACKNOWLEDGED <b>Message Meaning:</b> DRS/VPI has acknowledged an "END" command. <b>System Action:</b> DRS/VPI will terminate when all printers have finished printing their current report. <b>Required Action:</b> None.
<b>DRSV937R</b>	DRS/VPI ABNORMAL TERMINATION REQUEST ACKNOWLEDGED <b>Message Meaning:</b> DRS/VPI has acknowledged an "ABEND" command. <b>System Action:</b> DRS/VPI will terminate with a U001 abend. <b>Required Action:</b> None.
<b>DRSV940R</b>	*SYSTEM VERSION=V1R3.3.xxxxxx CUSTID=cccccc INIT(mm/dd-hh.mm) xxxxxx: DRS/VPI maintenance and fix level. cccccc: Customer ID. mm/dd-hh.mm: DRS/VPI initialization date and time. <b>Message Meaning:</b> Response to DISPLAY-SYSTEM. <b>System Action:</b> None. <b>Required Action:</b> None.
<b>DRSV941R</b>	*SYSTEM DRSVLIB(DEFLT=aaaaaaaa,MLIST=bbbbbbbb,MSG=ccccccc,START=ddddddd,XLIST=eeeeeee) aaaaaaaa: Default printer member name. bbbbbbbb: Include member list member name. ccccccc: Message modification member name. ddddddd: DRS/VPI initialization member name. eeeeeee: Exclude member list member name. <b>Message Meaning:</b> Response to DISPLAY-SYSTEM. <b>System Action:</b> None. <b>Required Action:</b> None.
<b>DRSV942R</b>	stcname AUTOACT(Y N,Y N) DESC(description) SWAPABLE(Y N,Y N) USEROPTS=X'nnnn' description: Description from DRSDDESC keyword. nnnn: User options specified in the USEROPTS keyword. <b>Message Meaning:</b> Response to DISPLAY-SYSTEM. <b>System Action:</b> None. <b>Required Action:</b> None.

---

**DRSV943R** \*SYSTEM LOG(c,ddddddd,ffffff,wwwwwww,ooooooo,  
HOLD|NOHOLD, ACTIVE|INACTIVE)

c: Log class.  
ddddddd: Log destination.  
ffffff: Log form.  
wwwwwww: Log writer.  
ooooooo: Log OUTPUT JCL statement name.  
**Message Meaning:** Response to DISPLAY-SYSTEM.  
**System Action:** None.  
**Required Action:** None.

**DRSV944R** \*SYSTEM MAXPRTS(VTAM(aaaaa) TCPIP(bbbbb) TOTAL(ccccc)  
CURR(ddddd) HIGH(eeeee))

aaaaa: Maximum number of active VTAM printers.  
bbbbbb: Maximum number of active TCPIP printers.  
ccccc: Maximum number of active printers (VTAM  
plus TCPIP).  
dddddd: Current active printers.  
eeeeee: Highest active printers.  
**Message Meaning:** Response to DISPLAY-SYSTEM.  
**System Action:** None.  
**Required Action:** None.

**DRSV945R** \*SYSTEM SNAP(c,ddddddd,ffffff,wwwwwww,ooooooo,  
HOLD|NOHOLD, ACTIVE|INACTIVE)

c: Snap class.  
ddddddd: Snap destination.  
ffffff: Snap form.  
wwwwwww: Snap writer.  
ooooooo: Snap OUTPUT JCL statement name.  
**Message Meaning:** Response to DISPLAY-SYSTEM.  
**System Action:** None.  
**Required Action:** None.

**DRSV946R** \*SYSTEM STORAGE LIMIT|VPI|DRS(REGN|CURR(aaaaa,bbbbbb)  
PRIV|MAC(aaaaa,bbbbbb))

aaaaa: Storage below the 16M line.  
bbbbbb: Storage above the 16M line.  
**Message Meaning:** Response to DISPLAY-SYSTEM.  
**System Action:** None.  
**Required Action:** None.

---

<b>DRSV947R</b>	*SYSTEM TCPIP(status, HOST=host,ID=id,PORT=port, TYPE=IBM221 IBM310 IBM320 ICS)
status:	ABENDED CONNECTED DISABLED INACTIVE SEVERED
host:	The local host name for TCPIP.
id:	The ID of the TCPIP address space.
port:	The port number used by DRS for TCPIP.
<b>Message Meaning:</b>	Response to DISPLAY-SYSTEM.
<b>System Action:</b>	None.
<b>Required Action:</b>	None.
<b>DRSV948R</b>	*SYSTEM TRACE(SIZE=sssK,types=X'tt',FMID=ff)
sss:	Size of the DRS/VPI trace table.
tt:	System trace types.
ff:	GTF format ID.
<b>Message Meaning:</b>	Response to DISPLAY-SYSTEM.
<b>System Action:</b>	None.
<b>Required Action:</b>	None.
<b>DRSV949R</b>	*SYSTEM WTO(RTCDE=aa,ACTN=Y N,INFO=Y N,STCPFX=Y N)
aa:	Route code to be used for WTOs.
<b>Message Meaning:</b>	Response to DISPLAY-SYSTEM.
<b>System Action:</b>	None.
<b>Required Action:</b>	None.
<b>DRSV955R</b>	*SYSTEM EXITnn NAME=name LENGTH=length EP=ep ACTIVE INACTIVE RECOVERY=ON OFF
nn:	Exit number.
name:	Exit module name.
length:	Exit module length.
ep:	Exit module entry point address.
<b>Message Meaning:</b>	Response to DISPLAY-SYSTEM.
<b>System Action:</b>	None.
<b>Required Action:</b>	None.

---

**DRSV960R** prtrid OUTREF(name STATUS=ACTIVE|INACTIVE,ID=id, REFERENCES=nnnnn)  
name: OUTREF name.  
id: The system-generated name assigned to this OUTREF.  
nnnnn: Number of virtual printers referencing this OUTREF.  
**Message Meaning:** Response to DISPLAY-OUTREF.  
**System Action:** None.  
**Required Action:** None.

**DRSV960R** prtrid OUTREF(name @OUTDESC=desc)  
name: OUTREF name.  
desc: OUTREF description.  
**Message Meaning:** Response to DISPLAY-OUTREF.  
**System Action:** None.  
**Required Action:** None.

**DRSV960R** prtrid OUTREF(name ADDRESS=addr)  
name: OUTREF name.  
desc: ADDR1, ADDR2, ADDR3, and ADDR4 keyword values.  
**Message Meaning:** Response to DISPLAY-OUTREF.  
**System Action:** None.  
**Required Action:** None.

**DRSV960R** prtrid OUTREF(name BUILDING=bldg,BURST=(Y|N)  
name: OUTREF name.  
bldg: BUILDING keyword value.  
**Message Meaning:** Response to DISPLAY-OUTREF.  
**System Action:** None.  
**Required Action:** None.

---

**DRSV960R** prtrid OUTREF(name CHARS(char,char,char,char),  
CKPTLINE=ckptline, CKPTPAGE=ckptpage,CKPTSEC=ckptsec)  
name: OUTREF name.  
char: CHARS keyword value(s).  
ckptline: CKPTLINE keyword value.  
ckptpage: CKPTPAGE keyword value.  
ckptsec: CKPTSEC keyword value.  
**Message Meaning:** Response to DISPLAY-OUTREF.  
**System Action:** None.  
**Required Action:** None.

**DRSV960R** prtrid OUTREF(name CLASS=c,COLORMAP=colormap,  
COMPACT=compact, COMSETUP=comsetup,CONTROL=control)  
name: OUTREF name.  
c: CLASS keyword value.  
colormap: COLORMAP keyword value.  
compact: COMPACT keyword value.  
comsetup: COMSETUP keyword value.  
control: CONTROL keyword value.  
**Message Meaning:** Response to DISPLAY-OUTREF.  
**System Action:** None.  
**Required Action:** None.

**DRSV960R** prtrid OUTREF(name COPIES=nnn,COPYGRP=(ccc,ccc,....,ccc,ccc),  
DATAACK=dataack)  
name: OUTREF name.  
nnn: COPIES keyword value.  
ccc: COPYGRP keyword value.  
dataack: DATAACK keyword value.  
**Message Meaning:** Response to DISPLAY-OUTREF.  
**System Action:** None.  
**Required Action:** None.

**DRSV960R** prtrid OUTREF(name DEPARTMENT=department)  
name: OUTREF name.  
department: DEPARTMENT keyword value.  
**Message Meaning:** Response to DISPLAY-OUTREF.  
**System Action:** None.  
**Required Action:** None.

---

**DRSV960R** prtrid OUTREF(name DEST=destination)  
name: OUTREF name.  
destination: DEST keyword value.  
**Message Meaning:** Response to DISPLAY-OUTREF.  
**System Action:** None.  
**Required Action:** None.

**DRSV960R** prtrid OUTREF(name DPAGELBL=Y|N,DUPLEX=NO|NORMAL|TUMBLE,  
FCB=fcb, FLASH(flsh,flshcnt), FORMDEF=formdef)  
name: OUTREF name.  
fcb: FCB keyword value.  
flsh: FLASH name.  
flshcnt: FLASH copy count.  
formdef: FORMDEF keyword value.  
**Message Meaning:** Response to DISPLAY-OUTREF.  
**System Action:** None.  
**Required Action:** None.

**DRSV960R** prtrid OUTREF(name FORMLEN=formlen,FORMS=form,  
GROUPID=groupid,INDEX=index,INTRAY=intray)  
name: OUTREF name.  
formlen: FORMLEN keyword value.  
form: FORMS keyword value.  
groupid: GROUPID keyword value.  
index: INDEX keyword value.  
intray: INTRAY keyword value.  
**Message Meaning:** Response to DISPLAY-OUTREF.  
**System Action:** None.  
**Required Action:** None.

**DRSV960R** prtrid OUTREF(name LINDEX=linex,LINECT=linect,  
MODIFY(modify,trc))  
name: OUTREF name.  
linex: LINDEX keyword value.  
linect: LINECT keyword value.  
modify: MODIFY copy modification module name.  
trc: MODIFY copy modification TRC value.  
**Message Meaning:** Response to DISPLAY-OUTREF.  
**System Action:** None.  
**Required Action:** None.

---

<b>DRSV960R</b>	prtrid OUTREF(name NAME=namek) name: OUTREF name. namek: NAME keyword value. <b>Message Meaning:</b> Response to DISPLAY-OUTREF. <b>System Action:</b> None. <b>Required Action:</b> None.
<b>DRSV960R</b>	prtrid OUTREF(name NOTIFY(notify1,notify2,notify3,notify4) name: OUTREF name. notify1: NOTIFY keyword value. notify2: NOTIFY2 keyword value. notify3: NOTIFY3 keyword value. notify4: NOTIFY4 keyword value. <b>Message Meaning:</b> Response to DISPLAY-OUTREF. <b>System Action:</b> None. <b>Required Action:</b> None.
<b>DRSV960R</b>	prtrid OUTREF(name OFFSETXB=offsetxb,OFFSETXF=offsetxf, OFFSETYB=offsetyb) name: OUTREF name. offsetxb: OFFSETXB keyword value. offsetxf: OFFSETXF keyword value. offsetyb: OFFSETYB keyword value. <b>Message Meaning:</b> Response to DISPLAY-OUTREF. <b>System Action:</b> None. <b>Required Action:</b> None.
<b>DRSV960R</b>	prtrid OUTREF(name OFFSETYF=offsetyf,OUTBIN=outbin, OUTDISP(displ,disp2)) name: OUTREF name. offsetyf: OFFSETYF keyword value. outbin: OUTBIN keyword value. disp1: OUTDISP normal completion value. disp2: OUTDISP abnormal completion value. <b>Message Meaning:</b> Response to DISPLAY-OUTREF. <b>System Action:</b> None. <b>Required Action:</b> None.

---

**DRSV960R** prtrid OUTREF(name OVERLAYB=overlayb,OVERLAYF=overlayf,OVFL=ON|OFF,PAGEDEF=pagedef,PIMSG=(Y|N,pimsgn))

name: OUTREF name.

overlayb: OVERLAYB keyword value.

overlayf: OVERLAYF keyword value.

pagedef: PAGEDEF keyword value.

pimsgn: PIMSG number of errors.

**Message Meaning:** Response to DISPLAY-OUTREF.

**System Action:** None.

**Required Action:** None.

**DRSV960R** prtrid OUTREF(name PORTNO=portno,PRMODE=prmode,PRERROR=prerror,PRTOPTNS=prtoptns)

name: OUTREF name.

portno: PORTNO keyword value.

prmode: PRMODE keyword value.

prerror: PRERROR keyword value.

prtoptns: PRTOPTNS keyword value.

**Message Meaning:** Response to DISPLAY-OUTREF.

**System Action:** None.

**Required Action:** None.

**DRSV960R** prtrid OUTREF(name PRTY=prty,RESFMT=resfmt)

name: OUTREF name.

prty: PRTY keyword value.

resfmt: RESFMT keyword value.

**Message Meaning:** Response to DISPLAY-OUTREF.

**System Action:** None.

**Required Action:** None.

**DRSV960R** prtrid OUTREF(name RETAINF=retainf,RETAINS=retains,RETRYL=retryl,RETRYT=retryt)

name: OUTREF name.

retainf: RETAINF keyword value.

retains: RETAINS keyword value.

retryl: RETRYL keyword value.

retryt: RETRYT keyword value.

**Message Meaning:** Response to DISPLAY-OUTREF.

**System Action:** None.

**Required Action:** None.



---

<b>DRSV960R</b>	prtrid OUTREF(name ROOM=room,SYSAREA=sysarea) name: OUTREF name. room: ROOM keyword value. sysarea: SYSAREA keyword value. <b>Message Meaning:</b> Response to DISPLAY-OUTREF. <b>System Action:</b> None. <b>Required Action:</b> None.
<b>DRSV960R</b>	prtrid OUTREF(name TITL=title,TRC=trc,UCS=ucs) name: OUTREF name. title: TITLE keyword value. trc: TRC keyword value. ucs: UCS keyword value. <b>Message Meaning:</b> Response to DISPLAY-OUTREF. <b>System Action:</b> None. <b>Required Action:</b> None.
<b>DRSV960R</b>	prtrid OUTREF(name USERDATA=userdata) name: OUTREF name. userdata: USERDATA keyword value. <b>Message Meaning:</b> Response to DISPLAY-OUTREF. <b>System Action:</b> None. <b>Required Action:</b> None.
<b>DRSV960R</b>	prtrid OUTREF(name USERLIB=userlib) name: OUTREF name. userlib: USERLIB keyword value. <b>Message Meaning:</b> Response to DISPLAY-OUTREF. <b>System Action:</b> None. <b>Required Action:</b> None.
<b>DRSV960R</b>	prtrid OUTREF(name WRITER=writer) name: OUTREF name. writer: WRITER keyword value. <b>Message Meaning:</b> Response to DISPLAY-OUTREF. <b>System Action:</b> None. <b>Required Action:</b> None.
<b>DRSV961R</b>	prtrid NO OUTREF MEMBERS DEFINED FOR THIS PRINTER <b>Message Meaning:</b> Response to DISPLAY-OUTREF. <b>System Action:</b> The DISPLAY command is ignored. <b>Required Action:</b> None.

---

<b>DRSV980R</b>	prtrid status information status information:	<p>The information displayed here will vary, depending on the status of the printer. The first portion of the message will contain the status of the printer (PRINTING, FORCED, IDLE, EDRAINED, CONNECTED).</p> <p>An EDRAINED printer will show an error message. An IDLE or FORCED printer will show the spooling or DASD attributes. A PRINTING printer will show the DRS report ID and the number of lines which have been placed on the spool for this report. A CONNECTED printer will display the partner LU (for VTAM) or remote host IP address (for TCP/IP).</p> <p>A list of the possible error messages for EDRAINED printers and the associated DRS/VPI message number which was displayed at the time of the error appears later in this section, as <a href="#">“EDRAINED Printer Error Messages”</a> on page 35.91.</p>
	<b>Message Meaning:</b>	Response to DISPLAY-STATUS or ACTIVATE.
	<b>System Action:</b>	None.
	<b>Required Action:</b>	None.
<b>DRSV981R</b>	prtrid INIT(mm/dd-hh.mm) LINES=lines D/S=datasets mm/dd-hh.mm: lines: datasets:	<p>Printer initialization date and time.</p> <p>Total number of lines spooled since initialization.</p> <p>Total number of datasets spooled since initialization.</p>
	<b>Message Meaning:</b>	Response to DISPLAY-PROCESSING.
	<b>System Action:</b>	None.
	<b>Required Action:</b>	None.
<b>DRSV982R</b>	prtrid CURRENT DEFAULT SPOOL ATTR(CLASS=class, DDNAME=ddname,DEST=dest,FCB=fcb,FORM=form) class: ddname: dest: fcb: form:	<p>SYSOUT class.</p> <p>DDNAME assigned to SYSOUT dataset.</p> <p>JES node name and destination userid.</p> <p>FCB name.</p> <p>Form name.</p>
	<b>Message Meaning:</b>	Response to DISPLAY-ATTRIBUTES.
	<b>System Action:</b>	None.
	<b>Required Action:</b>	None.

---

**DRSV983R** prtrid CURRENT|DEFAULT SPOOL ATTR(HOLD=Y|N,  
JOBNAME=jobname,OUTREF=outref,UCS=ucs,WRITER=writer)  
jobname: Name of job as specified using JOBNAME=  
keyword.  
outref: OUTPUT JCL statement name.  
ucs: UCS name.  
writer: Writer name.  
**Message Meaning:** Response to DISPLAY-ATTRIBUTES.  
**System Action:** None.  
**Required Action:** None.

**DRSV984R** prtrid RECORD ATTR(BLKSIZE=blksize,LRECL=lrecl,  
RECFM=recfm)  
blksize: Dataset block size.  
lrecl: Dataset record size.  
recfm: Dataset record format.  
**Message Meaning:** Response to DISPLAY-ATTRIBUTES.  
**System Action:** None.  
**Required Action:** None.

**DRSV985R** prtrid DASD ATTR(DSN=dsname,MEMBER=member)  
dsname: DASD dataset name.  
member: PDS member name (if dataset is partitioned).  
**Message Meaning:** Response to DISPLAY-ATTRIBUTES.  
**System Action:** None.  
**Required Action:** None.

**DRSV986R** prtrid DASD ATTR(AVGREC(avgrec,u),DATACLAS=class,  
DISP=(disp1,disp2),DSNTYPE=PDS|LIBRARY)  
avgrec: Average record size.  
u: Multiplier for record allocation.  
class: SMS data class.  
disp1: Initial disposition.  
disp2: Normal disposition.  
**Message Meaning:** Response to DISPLAY-ATTRIBUTES.  
**System Action:** None.  
**Required Action:** None.

---

**DRSV987R** prtrid DASD ATTR(EXPDT=yyyyddd,MGMTCLAS=class, RETPD=rrrr,SPACE=(B|C|E|R, pri,sec,dirblks,RLSE))  
yyyyddd: Julian date for expiration.  
class: SMS management class.  
rrrr: Number of days to retain.  
pri: Primary allocation units.  
sec: Secondary allocation units.  
dirblks: Number of directory blocks.  
**Message Meaning:** Response to DISPLAY-ATTRIBUTES.  
**System Action:** None.  
**Required Action:** None.

**DRSV988R** prtrid DASD ATTR(STORCLAS=class,UNIT(unitname,unitcnt), VOLCNT=volcnt,VOLUME=volume)  
class: SMS storage class.  
unitname: Unit name.  
unitcnt: Unit count.  
volcnt: Volume count.  
volume: Volume name.  
**Message Meaning:** Response to DISPLAY-ATTRIBUTES.  
**System Action:** None.  
**Required Action:** None.

**DRSV989R** prtrid MISC(DEFMTMEM=deflmem,DRAINED=Y|N, ERRACTN=erractn, IPDCLASS=ipdsclass,IPDSOPTS=X'ff', MAXLPG=(current,startup))  
deflmem: Default member name.  
erractn: Action to be taken when DRS/VPI encounters an error while processing a report.  
ipdsclass: SYSOUT class for IPDS output.  
ff: IPDS options.  
current: Maximum lines/page currently in effect.  
startup: Maximum lines/page specified at printer activation.  
**Message Meaning:** Response to DISPLAY-MISC.  
**System Action:** None.  
**Required Action:** None.

---

**DRSV990R** prtrid MISC(MPP(current,startup),PRTROPTS=x'ffffff',  
PRTXDBCS(Y|N,gggggggg,hhhh,iiii,jjjj))

current: Maximum print positions in each print line  
which is currently in effect.

startup: Maximum print position value specified at  
printer activation.

ffffff: Printer options in effect.

gggggggg: Name of the translation table.

hhhh: Action to be taken for shift-out and shift-in  
characters.

iiii: The shift-out (SO) replacement character(s).

jjjj: The shift-in (SI) replacement character(s).

**Message Meaning:** Response to DISPLAY-MISC.

**System Action:** None.

**Required Action:** None.

**DRSV991R** prtrid MISC(PRTLATE(Y|N,xlatetab,Y|N,trntab,DELT|KEEP),  
SEPAR=(N|S|E|B,separ),SERVER=aaaaaaaa,SNAP=Y|N)

xlatetab: Name of the translation table for normal SBCS data.

trntab: Name of the translation table for transparent data.

separ: Name of the separator routine.

aaaaaaaa: Name of the LRS/MVS Server.

**Message Meaning:** Response to DISPLAY-MISC.

**System Action:** None.

**Required Action:** None.

**DRSV992R** prtrid MISC(SUBSYS(aaaa,bbb...bbb),TAB=(ccc,ddd),  
TERMRPT(eeeeeee,ffffff,ggggggg))

aaaa: Name of the subsystem

bbb...bbb: The parameter to be passed to the subsystem.

ccc: The TAB value currently in use.

ddd: The TAB value specified at printer activation.

eeeeeee: The TERMRPT type currently in use.

ffffff: The TERMRPT type specified at printer  
activation.

gggggggg: The TERMRPT table name specified.

**Message Meaning:** Response to DISPLAY-MISC.

**System Action:** None.

**Required Action:** None.

---

**DRSV993R** prtrid MISC(TRACE(Y|N,x'tt'),TRN(C|B|S|T,trnbeg,trnend),  
UDATA=udata)  
tt: Indicates the trace options for the printer.  
trnbeg: Indicates the character(s) which will begin  
transparency for the printer.  
trnend: Indicates the character(s) which will end  
transparency for the printer.  
uata: Indicates the value of the UDATA keyword.  
**Message Meaning:** Response to DISPLAY-MISC.  
**System Action:** None.  
**Required Action:** None.

**DRSV994R** prtrid STAGING(BUFSIZE=bufsize,I/O=iocnt,  
SPACE=(BLK|CYL|TRK,pri,sec),UNIT=unit,VOLUME=volume  
bufsize: Buffer size to be used for the staging dataset.  
dsname: Staging dataset name.  
iocnt: I/O count for the staging dataset.  
pri: Primary allocation quantity for the staging dataset.  
sec: Secondary allocation quantity for the staging  
dataset.  
unit: Unit name for staging dataset.  
volume: Volume serial number for staging dataset.  
**Message Meaning:** Response to DISPLAY-QUEUE  
**System Action:** None.  
**Required Action:** None.

**DRSV995R** prtrid TCPIP(CONNECT=ACTIVE|INACTIVE,HOST=host, PORT=port)  
host: The local host name for TCPIP.  
port: The port number used by DRS for TCPIP.  
**Message Meaning:** Response to DISPLAY-TCPIP.  
**System Action:** None.  
**Required Action:** None.

---

<b>DRSV996R</b>	<p>prtrid TCPIP(LIMIT=limit,MRD=mrd,OPTS=X'opts', RUSIZES(send,receive),TYPE(type,type))</p> <p>limit: Indicates the maximum number of bytes that will be accepted from a remote TCPIP host.</p> <p>mrd: Indicates the TCPIP MRD interval, in minutes.</p> <p>opts: Indicates the TCPIP options for the printer.</p> <p>send: TCPIP send buffer size.</p> <p>receive: TCPIP receive buffer size.</p> <p>type: Indicates the value of the COMMTYPE keyword.</p> <p><b>Message Meaning:</b> Response to DISPLAY-TCPIP.</p> <p><b>System Action:</b> None.</p> <p><b>Required Action:</b> None.</p>
<b>DRSV997R</b>	<p>prtrid VTAM(ACB(OPEN CLOSED) SLU(ENABLED DISABLED) SESSION(ACTIVE INACTIVE) PLU(pluname) TYPE(type,type))</p> <p>pluname: Indicates the name of the application (partner logic unit) which is currently in session with this virtual printer; if no session is active, pluname will be NONE.</p> <p>type: Indicates the value of the COMMTYPE keyword.</p> <p><b>Message Meaning:</b> Response to DISPLAY-VTAM.</p> <p><b>System Action:</b> None.</p> <p><b>Required Action:</b> None.</p>
<b>DRSV998R</b>	<p>prtrid VTAM(LUa,RUSIZES(bbbbb,cccc),BIND=ddddddd)</p> <p>a: Indicates the LU type of the current session (0, 1 or 3); if there is no active session, a '?' will be displayed.</p> <p>bbbb: Indicates the maximum RU size to be used when this DRS virtual printer is sending data.</p> <p>cccc: Indicates the maximum RU size to be used when this DRS virtual printer is receiving data.</p> <p>ddddddd: Indicates the session (BIND) parameters in effect for the current session.</p> <p><b>Message Meaning:</b> Response to DISPLAY-VTAM.</p> <p><b>System Action:</b> None.</p> <p><b>Required Action:</b> None.</p>
<b>DRSV999R</b>	<p>prtrid VPB(xx-xx-xx-xx xx-xx-xx-xx) VPBV(vv-vv-vv-vv vv-vv-vv-vv)</p> <p>xx-xx-xx-xx: Virtual printer status bytes (8 bytes).</p> <p>vv-vv-vv-vv: Virtual printer VTAM status bytes (8 bytes).</p> <p><b>Message Meaning:</b> Response to DISPLAY-DIAG.</p> <p><b>System Action:</b> None.</p> <p><b>Required Action:</b> None.</p>

---

---

## DRS/SAPR2 Messages

The DRS/SAPR2 interface writes messages to the CICS queues defined by the TDQMSG and TDQTRC keywords in the DRS/SAPR2 defaults module (DRSSSDEF). Normal messages will be written to the file represented by the TDQMSG keyword. DRSP070I messages are special trace messages which will be written to the message file represented by the TDQTRC keyword. It is recommended that both queues be defined to CICS as JES SYSOUT datasets. For more information, see [“Updating the CICS Tables” on page 20.14](#).

If the TD queues defined in DRSSSDEF are not available, the messages will be written to the CSMT log, which is normally a JES SYSOUT dataset with a DD name of MSGUSR.

### DRS/SAPR2 Message Format

Each DRS/SAPR2 message begins with an 8-byte message identifier formatted as follows:

**Characters 1-4 - Always DRSP**

**Characters 5-7 - Message number uniquely identifying this message**

**Character 8 - One character identifying the message level**

The possible message importance levels (the last character of the message identifier) are as follows:

**I - Informational message**

**E - Error message**

It is possible to eliminate all informational messages by setting USEROPTS=80 in the DRS/SAPR2 defaults module (DPSSSDEF). See [“Building the DRS/SAPR2 Defaults Module” on page 27.4](#).

### DRS/SAPR2 Message Text

The date and time will be inserted into the message. In addition, the SAP R2 TD queue name will be moved into the message where ‘prtld’ appears in each message.



---

<b>DRSP000I</b>	<p>prtid DRS SYSTEM QUERY SUCCESSFUL – VERSION=V1rel.fix CUSTID=cust</p> <p>rel: DRS Release number (R3.3) fix: DRS Fix number cust: Customer ID</p> <p><b>Message Meaning:</b> The DRS/SAPR2 interface has successfully issued a system query to locate the DRS/API modules and obtain DRS system information.</p> <p><b>System Action:</b> None</p> <p><b>Required Action:</b> None.</p>
<b>DRSP001I</b>	<p>prtid DRS REPORT INIT SUCCESSFUL C=class D=dest F=form W=wtr PD=pagedef FD=formdef</p> <p>class: SYSOUT class dest: Destination form: Form name wtr: Writer name pagedef: PAGEDEF name formdef: FORMDEF name</p> <p><b>Message Meaning:</b> The DRS/SAPR2 interface has allocated a new SYSOUT dataset with the indicated attributes.</p> <p><b>System Action:</b> None</p> <p><b>Required Action:</b> None.</p>
<b>DRSP002I</b>	<p>prtid REPORT COMPLETE - nnnnn LINES</p> <p>nnnnn: Number of lines</p> <p><b>Message Meaning:</b> The DRS/SAPR2 interface has successfully written a report to the JES spool.</p> <p><b>System Action:</b> None</p> <p><b>Required Action:</b> None.</p>
<b>DRSP003E</b>	<p>prtid DRS/SAPR2 PROCESSING TERMINATED DUE TO ERROR</p> <p><b>Message Meaning:</b> The DRS/SAPR2 interface encountered an unexpected error.</p> <p><b>System Action:</b> Processing terminates.</p> <p><b>Required Action:</b> Check for other error messages for a description of the error. If the reason for the failure cannot be determined, contact LRS technical support.</p>

---

**DRSP004E** prtld PROCESSING TERMINATED - GETMAIN FAILED FOR aaaaa  
bbbb

aaaaa: Area for which storage was requested.  
bbbb: Amount of storage requested.

**Message Meaning:** The DRS/SAPR2 interface was unable to obtain the storage requested. DRS/SAPR2 will abend with DR00 abend.

**System Action:** DRS/SAPR2 will abend with DR00 abend.

**Required Action:** Determine why storage is not currently available.

**DRSP010E** prtld INCOMPLETE REPORT action - nnn LINES CLASS=class  
DEST=dest

action: Action based on ERRACTN in DPSSSDEF

- DELETED
- HELD
- KEPT

nnn,nnn,nnn: Number of lines in report  
class: SYSOUT class based on ERRCLASS  
dest: Destination based on ERRCLASS

**Message Meaning:** The DRS/SAPR2 interface has successfully written a report to JES, but the print data passed from SAP R2 was incomplete. This does not necessarily indicate an error as the creating ABAP may have been terminated by the user. If the report was not terminated by the user, then this can indicate that there was an excessive delay in SAP R2 writing the print information to the TD queue and the report was closed after a time-out period. For more information on the WAIT time and RETRY attempts to obtain the additional print lines, see [“Building the DRS/SAPR2 Defaults Module”](#) on page 27.4.

**System Action:** None  
**Required Action:** None

---

<b>DRSP011E</b>	<p>prtid REPORT action AFTER ERROR - nnn,nnn,nnn LINES CLASS=class DEST=dest</p> <p>action: Action based on ERRACTN in DPSSSDEF</p> <ul style="list-style-type: none"> <li>• DELETED</li> <li>• HELD</li> <li>• KEPT</li> </ul> <p>nnn,nnn,nnn: Number of lines in report</p> <p>class: SYSOUT class based on ERRCLASS</p> <p>dest: Destination based on ERRCLASS</p> <p><b>Message Meaning:</b> An unrecoverable error has occurred while processing a report. Output processing for this report has terminated and the SYSOUT file in error has been held, deleted or kept based on the ERRACTN keyword in the DPSSSDEF defaults module.</p> <p><b>System Action:</b> Processing terminates.</p> <p><b>Required Action:</b> Check for other error messages for a description of the error. If the reason for the failure cannot be determined, contact LRS technical support.</p>
<b>DRSP022I</b>	<p>prtid DRS/SAPR2 PRODUCT WILL EXPIRE IN nnn DAYS</p> <p>nnn: Number of days remaining in trial period</p> <p><b>Message Meaning:</b> The DRS/SAPR2 interface trial will expire within 14 days. The number of days is indicated in the message. This message will be issued in each CICS system once per day.</p> <p><b>System Action:</b> Processing continues.</p> <p><b>Required Action:</b> Contact LRS marketing personnel.</p>
<b>DRSP023E</b>	<p>prtid DRS/SAPR2 PRODUCT HAS EXPIRED</p> <p><b>Message Meaning:</b> The DRS/SAPR2 interface trial has expired.</p> <p><b>System Action:</b> Processing terminates.</p> <p><b>Required Action:</b> Contact LRS marketing personnel.</p>
<b>DRSP025E</b>	<p>prtid DRS/SAPR2 PRODUCT KEY INVALID</p> <p><b>Message Meaning:</b> The DRS/SAPR2 interface product key was either not specified or was specified incorrectly.</p> <p><b>System Action:</b> Processing terminates.</p> <p><b>Required Action:</b> Contact LRS marketing personnel.</p>
<b>DRSP026E</b>	<p>prtid DRS PRODUCT KEY INVALID OR EXPIRED</p> <p><b>Message Meaning:</b> The DRS product key was either not specified or was specified incorrectly. DRS/SAPR2 processing cannot continue without DRS/API.</p> <p><b>System Action:</b> Processing terminates.</p> <p><b>Required Action:</b> Contact LRS marketing personnel.</p>

---

---

**DRSP030E** prtld DRS/SAPR2 INTERFACE REQUIRES MVS/ESA  
**Message Meaning:** The DRS/SAPR2 interface has detected that it is not running on an MVS/ESA system. The DRS/SAPR2 interface requires MVS/ESA to create dynamic output JCL statements.  
**System Action:** Processing terminates.  
**Required Action:** Move DRS/SAPR2 to a system running MVS/ESA or contact LRS technical support.

**DRSP031E** fac DRS/SAPR2 PRINT INTERFACE NOT INITIATED BY ATI  
fac: Facility name (Terminal ID)  
**Message Meaning:** The DRS/SAPR2 print interface transaction was initiated incorrectly and was not started by a TD queue initiate.  
**System Action:** The print transaction terminates.  
**Required Action:** Verify the TD definition is correct.

**DRSP032E** prtld LOAD FAILED for modname MODULE RC=nnnnnnnn  
modname: Name of module.  
nnnnnnnn: EIBRESP value.  
**Message Meaning:** The load for failed for the named module.  
**System Action:** The print transaction terminates.  
**Required Action:** Check that the module named is defined to CICS and is available in the DFHRPL library concatenation.

**DRSP033E** prtld INVALID DEFAULTS MODULE (DPSSSDEF)  
**Message Meaning:** The DRS/SAPR2 interface output defaults module (DPSSSDEF) is invalid.  
**System Action:** Processing terminates.  
**Required Action:** Check to be sure that the correct module is loaded by CICS from the DFHRPL concatenation. If the reason for the failure cannot be determined, contact LRS technical support.

**DRSP034E** prtld INVALID PRINTER ALIAS TABLE (DPSSPTAB)  
**Message Meaning:** The DRS/SAPR2 interface printer alias module (DPSSPTAB) is invalid.  
**System Action:** Processing continues without printer alias support.  
**Required Action:** Check to be sure that the correct module is loaded by CICS from the DFHRPL concatenation. If the reason for the failure cannot be determined, contact LRS technical support.

---

<b>DRSP035E</b>	<p>prtid *DRS* PARAMETERS NOT TERMINATED OR EXCEED MAX LENGTH - IGNORED</p> <p><b>Message Meaning:</b> An error occurred while parsing a SYSOUT parameter options string from the SAPR2 formats table 022D. The DRS/SAPR2 print interface parameter options must be terminated by a less than (&lt;) character and must not contain more than 1024 characters.</p> <p><b>System Action:</b> Processing continues using the default output attributes from the DPSSSDEF module, and the parameter string is printed as part of the report.</p> <p><b>Required Action:</b> Ensure that the *DRS* parameter options string in the SAP R2 formats table 022D is correctly terminated.</p>
<b>DRSP036I</b>	<p>prtid DRS/SAPR2 PARAMETER PARSING SUCCESSFUL</p> <p><b>Message Meaning:</b> The *DRS* parameters were valid and will be used for the SYSOUT attributes to create the report.</p> <p><b>System Action:</b> None.</p> <p><b>Required Action:</b> None.</p>
<b>DRSP037E</b>	<p>prtid DRS/SAPR2 PARAMETER PARSING COMPLETED WITH WARNING</p> <p><b>Message Meaning:</b> An error occurred while parsing a DRS SYSOUT parameter options string from the SAPR2 formats table 022D. The warning error message text will appear in one or more DRSP039E messages.</p> <p><b>System Action:</b> Processing continues using the default SYSOUT parameter options string, plus any valid values from the DRS parameters. If USEROPTS=40 is not specified, the report will be DELETED, HELD or KEPT based on the ERRACTN keyword, SYSOUT class will be set to ERRCLASS and destination will be set to ERRDEST. If USEROPTS=40 is specified, the ERRACTN, ERRDEST and ERRCLASS will not be used.</p> <p><b>Required Action:</b> Check the parameter options string in the SAP R2 formats table 022D for incorrect keyword values.</p>

---

<b>DRSP038E</b>	<p>prtid DRS/SAPR2 PARAMETER PARSING UNSUCCESSFUL</p> <p><b>Message Meaning:</b> A severe error occurred while parsing a DRS SYSOUT parameter options string from the SAPR2 formats table 022D. The error text will appear in the DRSP039E message.</p> <p><b>System Action:</b> Processing continues using the default SYSOUT parameter options string. The report will be DELETED, HELD or KEPT based on the ERRACTN keyword, SYSOUT class will be set to ERRCLASS and destination will be set to ERRDEST.</p> <p><b>Required Action:</b> Check the parameter options string in the SAP R2 formats table 022D for incorrect keywords or keyword values.</p>
<b>DRSP039E</b>	<p>prtid ERROR: Parse_error</p> <p>Parse_error: Error description</p> <p><b>Message Meaning:</b> An error occurred while parsing a DRS SYSOUT parameter options string. The error text indicates the reason for the error and the parameter keyword in error.</p> <p><b>System Action:</b> Processing continues based on the type of error or warning message text and the options specified in the defaults module (DPSSSDEF). See messages DRSP037E or DRSP038P for more information.</p> <p><b>Required Action:</b> Check the syntax of the parameter options string in the SAP R2 formats table 022D.</p>
<b>DRSP040E</b>	<p>prtid ERROR READING TD TS QUEUE EIBRESP=nnnnnnnn</p> <p>nnnnnnnn: EIBRESP value</p> <p><b>Message Meaning:</b> An error occurred trying to read the print data transient data (TD) queue or a temporary storage (TS) queue.</p> <p><b>System Action:</b> The print transaction terminates.</p> <p><b>Required Action:</b> Determine the reason for the failure. If the reason cannot be determined, contact LRS technical support.</p>
<b>DRSP041E</b>	<p>prtid ERROR WRITING TD TS QUEUE EIBRESP=nnnnnnnn</p> <p>nnnnnnnn: EIBRESP value</p> <p><b>Message Meaning:</b> An error occurred trying to write to a CICS TS or TD queue.</p> <p><b>System Action:</b> The print transaction terminates.</p> <p><b>Required Action:</b> Determine the reason for the failure. If the reason cannot be determined, contact LRS technical support.</p>

---

<b>DRSP042E</b>	<p>prtid INVALID SPOOL BLOCK LENGTH nnnnn</p> <p>nnnnn: Length value from SAP R2 spool block header</p> <p><b>Message Meaning:</b> The DRS/SAPR2 print transaction encountered an error processing a SAP R2 spool block. The length field in the spool block is less than the minimum allowed.</p> <p><b>System Action:</b> The print transaction terminates</p> <p><b>Required Action:</b> Determine the reason for the failure. If the reason cannot be determined, contact LRS technical support.</p>
<b>DRSP043E</b>	<p>prtid INVALID SPOOL BLOCK IDENTIFIER</p> <p><b>Message Meaning:</b> The DRS/SAPR2 print transaction encountered an error processing a spool block. The data contained in the TD queue does not contain a valid SAP R2 spool block identifier.</p> <p><b>System Action:</b> The print transaction terminates</p> <p><b>Required Action:</b> Determine the reason for the failure. If the reason cannot be determined, contact LRS technical support.</p>
<b>DRSP044E</b>	<p>prtid SPOOL BLOCK LENGTH aaaaa NOT EQUAL TO TD QUEUE RECORD LENGTH bbbbb</p> <p>aaaaa: SAP R2 spool block length</p> <p>bbbbbb: TD queue record length</p> <p><b>Message Meaning:</b> The DRS/SAPR2 print transaction encountered an error processing a spool block. The length field in the SAP R2 spool block does not match the record length of the TD queue record.</p> <p><b>System Action:</b> The print transaction terminates</p> <p><b>Required Action:</b> Determine the reason for the failure. If the reason cannot be determined, contact LRS technical support.</p>

---

**DRSP045E** prtld RECORD LENGTH OF aaaaa EXCEEDS MAXIMUM LRECL  
 bbbbb - action

aaaaa: Length of print record  
 bbbbb Maximum LRECL specified in defaults module  
 action One of the following actions taken due to the error:  
 RECORD TRUNCATED  
 PROCESS STOPPED

**Message Meaning:** The DRS/SAPR2 print transaction encountered an error processing a spool block. The length field in the SAP R2 print record is longer than the maximum allowed in the DRS/SAPR2 defaults (DPSSSDEF).

**System Action:** The record is truncated and processing continues. If USEROPTS=20 is set, then processing will terminate.

**Required Action:** Verify the MAXLRECL is set correctly.

**DRSP046E** prtld INVALID RECORD IN SPOOL BLOCK

**Message Meaning:** The DRS/SAPR2 print transaction encountered an error processing a record in the SAP R2 spool block. The length is too short to contain the record header.

**System Action:** The print transaction terminates

**Required Action:** Determine the reason for the failure. If the reason cannot be determined, contact LRS technical support.

**DRSP047E** prtld CICS REQUEST UNSUCCESSFUL – EIBRESP=aaaaaaa  
 REQUEST=bbbbbbb

aaaaaaa: EIBRESP value  
 bbbbbbbb: Type of CICS request made

**Message Meaning:** An error occurred during an EXEC CICS call.

**System Action:** The print transaction continues.

**Required Action:** Determine the reason for the failure. If the reason cannot be determined, call LRS technical support.

**DRSP050E** prtld LINK TO DRSSINTC FAILED RC=nnnnnnnn  
 nnnnnnn: EIBRESP value.

**Message Meaning:** An error occurred during an EXEC CICS LINK to the DRS/API interface module DRSSINTC.

**System Action:** The print transaction terminates

**Required Action:** Determine the reason for the failure. If the reason cannot be determined, contact LRS technical support.



---

<b>DRSP051E</b>	<p>prtid DRS INIT REQUEST FAILED RC=nnn  nnn: DRS/API return code in decimal  <b>Message Meaning:</b> An error occurred during a DRS INIT request.  <b>System Action:</b> The print is generated using the default SYSOUT attributes and ERRACTN defined in DPSSSDEF.  <b>Required Action:</b> Determine the reason for the failure. If the reason cannot be determined, contact LRS technical support. The DRS/API return codes are documented in <a href="#">“DRS/API Return Codes”</a> on page 24.1.</p>
<b>DRSP052E</b>	<p>prtid DRS PUT REQUEST FAILED RC=nnn REPORT ID=repid  nnn: DRS/API return code in decimal  repid: Report ID associated with request  <b>Message Meaning:</b> An error occurred during a DRS PUT request.  <b>System Action:</b> The print transaction terminates  <b>Required Action:</b> Determine the reason for the failure. If the reason cannot be determined, contact LRS technical support. The DRS/API return codes are documented in <a href="#">“DRS/API Return Codes”</a> on page 24.1.</p>
<b>DRSP053E</b>	<p>prtid DRS TERM REQUEST FAILED RC=nnn REPORT ID=repid  nnn: DRS/API return code in decimal  repid: Report ID associated with request  <b>System Action:</b> The print transaction terminates.  <b>Required Action:</b> Determine the reason for the failure. If the reason cannot be determined, contact LRS technical support. The DRS/API return codes are documented in <a href="#">“DRS/API Return Codes”</a> on page 24.1.</p>
<b>DRSP054E</b>	<p>prtid DRS OUTPUT JCL STATEMENT ADD FAILED  <b>Message Meaning:</b> An error occurred during a DRS/API OUTP request to dynamically add an OUTPUT card. This message is followed by message DRSP055E which further describes the error.  <b>System Action:</b> The print is generated using the default SYSOUT attributes and ERRACTN defined in DPSSSDEF.  <b>Required Action:</b> Determine the reason for the failure. If the reason cannot be determined, contact LRS technical support.</p>

---

**DRSP055E** prtld DRRB RC=aaa OUTP RC=bbb REASON CODE=cccc TEXT  
UNIT=X'ddddddd'

aaa: DRS/API DRRB return code in decimal  
bbb: DRS/API DROB return code in decimal  
cccc: DRS/API DROB reason code in decimal  
ddddddd: Text unit number in hex

**Message Meaning:** An error occurred during a DRS/API OUTP request to dynamically add an OUTPUT card.

**System Action:** The print is generated using the default SYSOUT attributes and ERRACTN defined in DPSSSDEF.

**Required Action:** Determine the reason for the failure. If the reason cannot be determined, contact LRS technical support. The DRS/API return codes are documented in [“DRS/API Return Codes” on page 24.1.](#)

**DRSP056E** prtld OUTPUT JCL STATEMENT DELETE FAILED RC=nnn

nnn: DRS/API return code in decimal

**Message Meaning:** An error occurred during a DRS/API OUTP request to dynamically delete an OUTPUT card.

**System Action:** Processing continues.

**Required Action:** Determine the reason for the failure. If the reason cannot be determined, contact LRS technical support. The DRS/API return codes are documented in [“DRS/API Return Codes” on page 24.1.](#)

**DRSP057E** prtld DRS CMND REQUEST FAILED RC=nnn TYPE=type

nnn: DRS/API return code in decimal  
type: The type of CMND request made:  
SNAP DRS BLOCKS

**Message Meaning:** An error occurred during a DRS CMND request.

**System Action:** Processing terminates.

**Required Action:** Determine the reason for the failure. If the reason cannot be determined, contact LRS technical support. The DRS/API return codes are documented in [“DRS/API Return Codes” on page 24.1.](#)

---

<b>DRSP058E</b>	<p>prtid DRS QURY REQUEST FAILED RC=nnn TYPE=type  nnn: DRS/API return code in decimal  type: The type of QURY request made:  SYSTEM QUERY</p> <p><b>Message Meaning:</b> An error occurred during a DRS QURY request.  <b>System Action:</b> Processing terminates.  <b>Required Action:</b> Determine the reason for the failure. If the reason cannot be determined, contact LRS technical support. The DRS/API return codes are documented in <a href="#">“DRS/API Return Codes” on page 24.1.</a></p>
<b>DRSP059E</b>	<p>prtid OUTPUT *DRS* PARAMETERS FAILED - DEFAULT  SYSOUT PARAMETERS WILL BE USED</p> <p><b>Message Meaning:</b> An error occurred during a DRS/API OUTP request to initialize the report. A DRSP037 or DRSP038 message, along with one or more DRSP039 messages, will be issued to describe the error in the *DRS* parameters.</p> <p><b>System Action:</b> The print is generated using the default attributes and ERRACTN as defined in DPSSSDEF.</p> <p><b>Required Action:</b> Determine the reason for the failure. If the reason cannot be determined, contact LRS technical support.</p>
<b>DRSP060E</b>	<p>prtid DRS/SAPR2 - LOGIC ERROR CALLING module REQUEST=request  module: DRS/SAPR2 module name.  request: Request type.</p> <p><b>Message Meaning:</b> An logic error occurred during a DRS/SAPR2 processing.</p> <p><b>System Action:</b> Processing terminates.  <b>Required Action:</b> Contact LRS technical support.</p>
<b>DRSP070I</b>	<p>prtid trace_data  trace_data</p> <p><b>Message Meaning:</b> Entry from DRS/SAPR2 internal trace table. Trace data has been requested from the internal DRS/SAPR2 trace table.</p> <p><b>System Action:</b> Entries from the DRS/SAPR2 trace table are moved to the file defined by the TDQTRC keyword in the defaults module. Entries are created based on the last byte of the USEROPTS value; the entries should only be created upon the request of LRS technical support.</p> <p><b>Required Action:</b> If tracing was requested by LRS technical support, forward the data in the trace file to LRS.</p>

---

---

**DRSP099E** prtld INVALID MESSAGE NUMBER - CONTACT LRS TECHNICAL SUPPORT

**Message Meaning:** An logic error occurred during a DRS/SAPR2 message processing.

**System Action:** Processing terminates.

**Required Action:** Contact LRS technical support.

## EDRAINED Printer Error Messages

When DRS/VPI puts a virtual printer into an EDRAINED status, it issues a message which indicates the type of error encountered. When message DRSV990R is displayed for an EDRAINED printer, the message will include an abbreviated version of the original error message.

The abbreviated error messages and their associated DRS message numbers are:

Associated Message Number	Abbreviated Error Message
DRSV201E	ATTACH FAILED FOR xxxxxxxx R15=xx
DRSV210E	LOGIC ERROR FORMATTING OUTPUT BUFFER - xxxx
DRSV211E	OUTPUT BUFFER OVERRUN - xxxx
DRSV212E	DSC 3270 WRITE COMMAND IS MISSING
DRSV213E	ERROR PROCESSING SPANNED DSC 3270 COMMAND
DRSV214E	WRITE STRUCTURED FIELD 3270 COMMAND NOT SUPPORTED
DRSV217E	LOGIC ERROR DVSSVPRT - xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
DRSV218E	RECV LEN xxxx INVALID xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
DRSV219E	LOGIC ERROR PBLK xxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
DRSV220E	SCS CONTROL CODE SPANS RECEIVE BUFFERS
DRSV221E	INVALID SCS CONTROL CODE
DRSV230E	IPDS PRINT BLOCK IS INVALID
DRSV231E	IPDS DATA LENGTH ERROR
DRSV232E	IPDS PRINT BLOCK OVERFLOW
DRSV233E	IPDS UNKNOWN COMMAND DETECTED CMC=xxxx
DRSV250E	ACB OPEN FAILED R15=xx EC=xx
DRSV252E	ACB CLOSE FAILED R15=xx EC=xx
DRSV253E	OPNSEC ERROR R15=xx R0=xx RC=xx F2=xx SNS=xxxxxxxx
DRSV254E	TERMSESS ERROR R15=xx R0=xx RC=xx F2=xx SNS=xxxxxxxx
DRSV255E	SETLOGON ERROR R15=xx R0=xx RC=xx F2=xx SNS=xxxxxxxx
DRSV256E	SESSIONC ERROR R15=xx R0=xx RC=xx F2=xx SNS=xxxxxxxx
DRSV257E	RECEIVE ERROR R15=xx R0=xx RC=xx F2=xx SNS=xxxxxxxx
DRSV258E	INQUIRE ERROR R15=xx R0=xx RC=xx F2=xx SNS=xxxxxxxx
DRSV259E	SEND ERROR R15=xx R0=xx RC=xx F2=xx SNS=xxxxxxxx
DRSV260E	RESETSR ERROR R15=xx R0=xx RC=xx F2=xx SNS=xxxxxxxx
DRSV261E	CHECK ERROR R15=xx R0=xx RC=xx F2=xx SNS=xxxxxxxx
DRSV262E	RECEIVE FAILED RTNCD=xx FDBK2=xx SENSE=xxxxxxxx
DRSV264E	MODCB ERROR R15=xx R0=xx BLK=xxxxx
DRSV265E	SHOWCB ERROR R15=xx R0=xx BLK=xxxxx
DRSV266E	TESTCB ERROR R15=xx R0=xx BLK=xxxxx

Associated Message Number	Abbreviated Error Message
DRSV267E	CLSDST ERR R15=xx R0=xx RC=xx F2=xx SNS=xxxxxxxx
DRSV268E	SESSION REJECT SNS=xxxxxxxx xxxxxxxxxxxxxxxxxxxxxxx
DRSV270E	APPCCMD ERR xxxxxxxxxxx xxxxxxxx RCPR=xxxx RCSC=xxxx
DRSV281E	TPEND xxxxxx RECEIVED
DRSV282E	xxxxxxxxxxx REQ RECEIVED in ATTn EXIT xxxx
DRSV283E	RPLEXIT UNKNOWN REQ=xx R15=xx R0=xx RC=xx F2=xx
DRSV283E	TCPIP ERROR xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
DRSV285E	INVALID xxxxxxxx REQUEST xxxxxxxx
DRSV286E	UNKNOWN REQUEST RECEIVED IN NETWORK SERVICES EXIT
DRSV287E	UNEXPECTED DFSYN REQUEST=xx CONTROL=xxxxxx
DRSV288E	xxxxxxxxxxx REQ RECEIVED IN SCIP EXIT xxxxxx
DRSV290E	INVALID DRS CALL REQUEST xxxxxxxx
DRSV291E	DRS CALL ERROR FUNC=xxxx RC=xxxxxxxx OSRC=xxxxxxxx
DRSV381E	xxxxxxxxxxxxxxxxxxxxxxxx ERROR R15=xxxxxxxx
DRSV382E	TCPIP ERR=xxxxxxxx TYPE=xxxxxxxx FUNCTION=xxxx
DRSV383E	TCPIP CONNECTION SEVERED
DRSV384E	TCPIP LOGIC ERR xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
DRSV387E	TCPIP CONNECTION NOT AVAILABLE
DRSV391E	TCPMRD EXPIRED - CONNECTION TIMED OUT
DRSV394E	TCPIP ERR=xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
DRSV520E	TCP Q ALLOC ERR RC=xx EC=xxxx IC=xxxx
DRSV521E	TCP Q UNALLOC ERR RC=xx EC=xxxx IC=xxxx
DRSV522E	TCP QUEUE xxxxx FAILURE DDNAME=xxxxxxxx
DRSV523E	TCPIP QUEUE DCB ABEND CODE=xxx-xx
DRSV525E	TCP QUEUE I/O ERR xxxxxxxxxxxxxxxxxxxxxxxxxxx
DRSV526E	TCP INSUFFICIENT STORAGE - QUEUE BUFFER
DRSV528E	TCPIP QUEUE INVALID DATASET DETECTED
DRSV529E	TCPIP QUEUE OUTPUT BUFFER OVERRUN
DRSV807E	VIRTUAL PRINTER SUBTASK ABEND

**Table 35.1: EDRAINED Printer Error Messages**

---

## DRS/STI Messages

The following messages can result from using program DS33LOAD to populate the DRS/STI rules dataset.

<b>DSST010E</b>	type REQUEST FAILED FOR FILE DSSTRLIB, RC=X'retcode'
type:	The file access request type.
retcode:	The file access request return code.
<b>Message Meaning:</b>	The rules file access request shown in the message failed with the return code shown in the message.
<b>System Action:</b>	Processing terminates.
<b>Required Action:</b>	Ensure that the DRS/STI rules dataset name is properly defined in the CLIST. If unable to resolve the problem, notify DRS technical support.
<b>DSST020E</b>	type REQUEST FAILED FOR FILE DSSTRLIB, RETCODE=X'rc', ERRCODE=X'ec', INFCODE=X'ic'
type:	The file allocation request type.
rc:	Dynamic allocation return code.
ec:	Dynamic allocation error code.
ic:	Dynamic allocation info code.
<b>Message Meaning:</b>	The rules file dynamic allocation request shown in the message failed with the error feedback information shown in the message.
<b>System Action:</b>	Processing terminates.
<b>Required Action:</b>	Ensure that the DRS/STI rules dataset name is properly defined in the CLIST. If unable to resolve the problem, notify DRS technical support.
<b>DSST900I</b>	INITIALIZING DRS PRINT RULES DATA SET
<b>Message Meaning:</b>	The DRS/STI rules data set has been opened in VSAM load mode and is being loaded with the initial control data.
<b>System Action:</b>	None.
<b>Required Action:</b>	None.

---

<b>DSST904I</b>	ddname STATEMENT NOT PRESENT-PRINT RULES type DATA BYPASSED
	ddname: The JCL DDNAME of the file which is not present. This will be either VPSLIB or DRSVLIB.
	type: The type of print rules data for which processing was bypassed. This will be either PRINTER or OUTPUT.
	<b>Message Meaning:</b> The DD statement shown in the message is not present in the job control language. As a result, processing for the print rules data type shown in the message will be bypassed.
	<b>System Action:</b> None.
	<b>Required Action:</b> None.
<b>DSST908I</b>	UPDATING PRINT RULES type DATA USING SYSTEM INITIALIZATION MEMBER member
	type: The type of print rules data for which processing is being performed. This will be either PRINTER or OUTPUT.
	member: The name of the VPS or DRS system initialization member used to determine the printer and output statement definitions to process.
	<b>Message Meaning:</b> Processing for the print rules data type shown in the message is beginning using the system initialization member shown in the message.
	<b>System Action:</b> None.
	<b>Required Action:</b> None.
<b>DSST920I</b>	PRINT RULES action FOR PRINTER printer
	action: Indicates the action taken against the named printer definition (CREATED, REPLACED, or RETAINED).
	printer: Indicates the name of the print rules printer definition.
	<b>Message Meaning:</b> Print rules processing took the action shown in the message for the printer shown definition in the message.
	<b>System Action:</b> None.
	<b>Required Action:</b> None.



---

<b>DSST924I</b>	PRINT RULES action FOR OUTPUT STATEMENT outref
action:	Indicates the action taken against the named output definition (CREATED, REPLACED, or RETAINED).
printer:	Indicates the name of the print rules output definition.
<b>Message Meaning:</b>	Print rules processing took the action shown in the message for the output definition shown in the message.
<b>System Action:</b>	None.
<b>Required Action:</b>	None.
<b>DSST960W</b>	EXEC STATEMENT PARAMETER parm IS NOT RECOGNIZED
parm:	Indicates the JCL EXEC statement parameter that is not recognized.
<b>Message Meaning:</b>	A JCL EXEC statement parameter was specified that is not recognized.
<b>System Action:</b>	Processing continues, ignoring the unrecognized parameter.
<b>Required Action:</b>	None.
<b>DSST970E</b>	SYNTAX ERROR IN STMT stmt OF MEMBER member-desc
stmt:	Indicates the statement number of the VPS or DRS control library member that contains the syntax error.
member:	Indicates the VPS or DRS control library member name that contains the syntax error.
desc:	The description of the syntax error.
<b>Message Meaning:</b>	A syntax error was identified in the VPS or DRS control library member shown in the message.
<b>System Action:</b>	If the syntax error was identified in the system initialization member or the inclusion/exclusion list member, processing terminates. Otherwise, processing continues with the next printer or output statement definition.
<b>Required Action:</b>	Correct the syntax error and retry the process.

---

---

<b>DSST972E</b>	SYNTAX ERROR IN KEYWORD keyword OF MEMBER member- desc
	keyword: Indicates the keyword statement of the VPS or DRS control library member that contains the syntax error.
	member: Indicates the VPS or DRS control library member name that contains the syntax error.
	desc: The description of the syntax error.
	<b>Message Meaning:</b> A syntax error was identified in the VPS or DRS control library member shown in the message.
	<b>System Action:</b> If the syntax error was identified in the system initialization member or the inclusion/exclusion list member, processing terminates. Otherwise, processing continues with the next printer or output statement definition.
	<b>Required Action:</b> Correct the syntax error and retry the process.
<b>DSST976E</b>	MEMBER member IS NOT A VALID PRINTER DEFINITION
	member: Indicates the VPS control library member name.
	<b>Message Meaning:</b> The member name shown in the message was identified by the member inclusion or exclusion list as a VPS printer definition. However, the member does not contain any statement specifying the printer CLASS, DEST, FORM, or WRITER.
	<b>System Action:</b> Processing continues with the next printer definition.
	<b>Required Action:</b> None.
<b>DSST978E</b>	MLISTMEM/XLISTMEM NOT SPECIFIED IN SYSTEM INITIALIZATION MEMBER member
	member: Indicates the VPS or DRS control library system initialization member name.
	<b>Message Meaning:</b> The system initialization member shown in the message does not specify either a member inclusion list (MLISTMEM) or a member exclusion list (XLISTMEM).
	<b>System Action:</b> Processing terminates.
	<b>Required Action:</b> Correct the system initialization member and retry the process.

---

<b>DSST980E</b>	<p>OPEN FAILED FOR FILE ddname</p> <p>ddname:                   The JCL DDNAME of the file for which the OPEN failed.</p> <p><b>Message Meaning:</b>    The attempt to OPEN the file shown in the message failed.</p> <p><b>System Action:</b>       Processing terminates.</p> <p><b>Required Action:</b>     Correct the JCL and retry the process.</p>
<b>DSST984E</b>	<p>request REQUEST FAILED FOR FILE DRSRULES, RETCODE=X'retcode'</p> <p>request:                   The file access request type.</p> <p>retcode:                   The file access request return code.</p> <p><b>Message Meaning:</b>    The print rules file access request shown in the message failed with the return code shown in the message.</p> <p><b>System Action:</b>       Processing terminates.</p> <p><b>Required Action:</b>     Ensure that the DRS/STI print rules data set is properly defined in the job control language. If unable to resolve the problem, notify DRS technical support.</p>
<b>DSST988E</b>	<p>FIND FOR LIBRARY MEMBER member FAILED, RETCODE=X'retcode', RETCODE=X'retcode', RSNCODE=X'rsncode'</p> <p>member:                   The VPS or DRS control library member for which the FIND request failed.</p> <p>retcode:                   The FIND return code.</p> <p>rsncode:                   The FIND reason code.</p> <p><b>Message Meaning:</b>    The attempt to locate the VPS or DRS control library member shown in the message failed with the return code and reason code shown in the message.</p> <p><b>System Action:</b>       If the locate failure occurred while processing the system initialization member or the inclusion/exclusion list member, processing terminates. Otherwise, processing continues with the next printer or output statement definition.</p> <p><b>Required Action:</b>     Correct the VPS or DRS control library definitions and retry the process.</p>

---

## DRS/STI CICS Messages

**DSSC005E** TRANSACTION INVOCATION ERROR – COMMAREA LENGTH INVALID

**Message Meaning:** The commarea length is invalid.

**System Action:** Processing terminates.

**Required Action:** Correct the programming error and retry the process.

**DSSC010E** TRANSACTION INVOCATION ERROR – MUST BE LINKED TO FROM A PROGRAM PASSING A COMMEAREA

**Message Meaning:** A transaction invocation error has occurred as indicated in the message.

**System Action:** Processing terminates.

**Required Action:** Correct the programming error and retry the process.

---

## DRS/API Abend Codes

### System Abend Codes

#### SA03 Abend

An SA03 abend occurs whenever a task terminates, but all of its subtasks (i.e., DRSSMAIN) have not yet terminated.

#### CICS Mode

An SA03 abend will occur during CICS termination if DRSSSHTC is not included in the CICS PLT. The DRSSSHTC program is used to initiate DRS termination. DRS termination consists of terminating all active report(s) and detaching the DRSSMAIN subtask(s). See [“Updating the CICS Tables” on page 20.14](#) for a description of the PLT entry that is used to invoke DRSSSHTC.

DRS 1.0 users MUST replace PLT entry DRS1SHUT with DRSSSHTC when migrating to DRS Version 1 Release 3.3. Both programs perform the same function. The difference is that DRS1SHUT is a CICS macro level program and DRSSSHTC is a CICS command level program. The macro level version will not work with DRS Version 1 Release 3.3. This change was made because IBM will be dropping support of the CICS macro level interface in future releases of CICS.

#### Non-CICS mode (TP Monitor)

An SA03 abend will occur if DRSSSHTB is not executed prior to the termination of the TP monitor. The DRSSSHTB program is used to initiate DRS termination. DRS termination consists of terminating all active report(s) and detaching the DRSSMAIN subtask(s).

**Non-CICS mode (batch program)** An SA03 abend will occur during termination of the batch program under the following conditions:

- The batch program initiated one or more reports but failed to terminate all of them. The program should be corrected to terminate all reports prior to program termination.
- The DRSSOPTS module specified a non-zero value for the TCBBTCH keyword. A non-zero value for the TCBBTCH keyword should only be specified when executing DRS in an environment that allows execution of program DRSSSHTB or DRSSSHTC at shutdown. Change the TCBBTCH keyword value to zero in the DRSSOPTS module. Re-assemble and relink the DRSSOPTS module and rerun the program.
- The DRSSOPTS module specified LOG=YES. LOG=YES should only be specified when executing DRS in an environment that allows execution of program DRSSSHTB or DRSSSHTC at shutdown. Change the LOG keyword value to NO in the DRSSOPTS module. Re-assemble and relink the DRSSOPTS module and rerun the program.

---

## All Other System Abends

All other system abends should be reported to DRS technical support.

## DRS/API Abend Codes

Both the CICS and non-CICS versions of the DRS/API issue certain user abends when critical errors occur. In a CICS environment, the transaction which issued the DRS call that caused the critical error will be abended. In a non-CICS environment, an MVS abend will be issued.

In a non-CICS environment, you can choose to use DRS User Exit 06 (OS Services Exit) to change the MVS abend to some other type of abend which will not abend the entire address space.

The DRS log will contain one or more error messages indicating the exact cause of the abend.

The **non-CICS DRS/API user abend codes** and their meanings are:

- U0700**     The parameter list passed to DRS was invalid.
- U0701**     A GETMAIN for a critical DRS control block failed.

The **CICS DRS/API user abend codes** and their meanings are:

- DR00**     No DRRB or invalid DRRB
- DR01**     System Record TS-READQ error
- DR02**     System Record TS-WRITEQ error
- DR03**     Invalid DRS System Record
- DR04**     Error allocating RQE
- DR05**     Error allocating DRS System Area (SYS)
- DR06**     Error allocating SCA table
- DR07**     Error loading support modules
- DR08**     Error attaching support subtask

---

## **DRS/SAPR2 Abend Codes**

The CICS programs for DRS/SAPR2 issue certain abends when critical errors occur. The DRS/SAPR2 log will contain one or more error messages indicating the cause of the abend.

The **CICS DRS/SAPR2 user abend codes** and their meanings are:

- DP00**      Error allocating DRS control blocks.
- DP01**      Error writing a DRS/SAPR2 message to the message queue.

---

## DRS/VPI Abend Codes

### User Abend Codes

<b>Code</b>	<b>Dump?</b>	<b>Reason for Abend</b>	<b>Corresponding Message</b>
U001	YES	Requested by ABEND command.	DRSV937R
U002	YES	VTAM is inactive. Operator response was to end DRS/VPI	DRSV251E
U005	YES	A critical error has occurred during DRS/VPI initialization. Various messages may be issued to describe the error(s) detected.	DRSV010E
U006	YES	A critical error has occurred in the TCP/IP interface; unable to locate DRS/VPI control blocks.	(none)
U100	NO	A required subtask has abended. The user abend will not create a dump, but the original subtask abend would create a dump.	DRSV800E DRSV801E



---

# Appendix A

## DRS Documentation

The most recent version of this manual can be downloaded from the LRS Web site ([www.lrs.com](http://www.lrs.com)).

As a licensed user of this product, you may print the PDF file on the *Enterprise Output Management Product Documentation* CD for use within your company as allowed by your license.



## #

#LRTCCNT macro, 9.3

## \$

\$DRSPTAB, 27.14

\$DRSSDEF

parameters, 27.8

@OUTDESC, 3.108

## O

022B table, 26.1, 27.2, 27.18, 27.19

## A

ABAP Program, 28.5

Abends, 35.99, 35.100, 35.101, 35.102

Add lines exit for DRS/API, 22.8

ADDRn, 3.109

ALIAS, 27.14

ASCII, 6.1, 6.2, 11.7, 12.2, 13.5

Assembler coding with DRS/API, 21.3

AUTOACT, 3.16, 3.42

AVGREC, 3.45

## B

Banner pages, 9.2

BLKSIZE, 3.46

BUILDING, 3.110

Building the DRS/SAPR2 Defaults Module, 27.4

Building the DRS/SAPR2 Printer Alias Table,  
27.13

BURST, 3.111

## C

CHARS, 3.112

CICS JCL, 29.2

CICS mode with DRS/API, 21.2

CICS table entries

For DRS/API, 20.14

for DRS/NATURAL, 31.6

For DRS/SAPR2, 27.15

For DRS/STI, 20.15

CKPTLINE, 3.113

CKPTPAGE, 3.114

CKPTSEC, 3.115

CLASS, 3.47, 3.116

CMND call, 21.1, 21.6

COBOL coding with DRS/API, 21.4

COLORMAP, 3.117

COMMAND block (DRCB), 21.39

Commands for DRS/VPI, 8.4

COMMTYPE, 3.48

COMPACT, 3.118

COMSETUP, 3.119

CONTROL, 3.120

COPIES, 3.121

Copies of SAP R2 Spool Record Data, 29.2

COPYGRP, 3.122

Copying the DRS/SAPR2 Modules for CICS, 27.17

Creating a Report Termination Table, 10.6

Creating a User Report Termination Table, 10.7

## D

DATAACK, 3.123

DATACLAS, 3.49

DCT, 27.15

DCT Entry, 29.2

DDNAME, 3.50

Defining DRS/SAPR2 Command Strings, 28.9

Defining the Printer TD Queue, 27.18

Defining the Printer to SAP R2, 27.19

Defining the Report Termination Table, 10.6

DEFLTMEM, 3.17, 3.51

DEPT, 3.124

DEST, 3.52, 3.125

DFHRPL, 27.17, 31.7

DISP, 3.53

DN33MAIN, 31.6, 31.7

Documentation, A.1

DP00 abend, 35.101

DP01 abend, 35.101

DP32SAPP, 27.15, 27.17

DPAGELBL, 3.126

DPSSPTAB, 27.17

DPSSSDEF, 27.4, 27.15, 27.17

DR00 abend code, 35.100

DR01 abend code, 35.100

DR02 abend code, 35.100

DR03 abend code, 35.100

DR04 abend code, 35.100

DR05 abend code, 35.100

DR06 abend code, 35.100

DR07 abend code, 35.100

DR08 abend code, 35.100

DRAINED, 3.54

DRCB, 21.39

DRIB, 21.9

DROB, 21.40

DRQB, 21.21

DRRB, 21.7

DRS, 1.7

---

- DRS calls
  - QUERY call, 20.3
- DRS documentation, A.1
- DRS release 1.0, 20.3
- DRS release 2.0, 20.3
- DRS release 3.0, 20.3
- DRS release 3.1, 20.3
- DRS system options for DRS/API, 20.9
- DRS system options for DRS/VPI, 3.8
- DRS/API
  - Assembler coding, 21.3
  - Calls, 21.1
  - CICS mode, 21.2
  - CICS tables, 20.14
  - CMND call, 21.1
  - COBOL coding, 21.4
  - Control blocks, 21.5
  - DRS system options, 20.9
  - INIT call, 21.1
  - Installation, 20.1
  - Introduction, 19.1
  - JES considerations, 23.1
  - Messages, 35.1
  - NonCICS mode, 21.2
  - OUTP call, 21.1
  - PUT call, 21.1
  - QUERY call, 21.1
  - TERM call, 21.1
  - User exits, 22.1
- DRS/API abend codes, 35.100
- DRS/API calls, 21.1
  - CMND Call, 21.6
  - General information, 21.2
  - INIT call, 21.5
  - OUTP call, 21.6
  - Parameters, 21.5
  - PUT call, 21.5
  - QUERY call, 21.6
  - TERM call, 21.6
- DRS/API Control blocks
  - COMMAND block (DRCB), 21.39
  - INIT block (DRIB), 21.9
  - OUTPUT block (DROB), 21.40
  - QUERY block (DRQB), 21.21
  - REQUEST block (DRRB), 21.7
  - TERMINATION block (DRTB), 21.19
- DRS/API control blocks, 21.5
- DRS/API installation steps, 20.4
- DRS/API modules
  - DRSSINTB, 19.1
  - DRSSINTC, 19.1
- DRS/API Return Codes, 29.1
- DRS/API return codes, 24.1
- DRS/API Snap dumps, 29.1
- DRS/API User exits
  - Add lines exit, 22.8
  - Execution environment, 22.2
  - Exit selection table, 22.4
  - Exit tracing, 22.3
- Exit types, 22.2
- Exittoexit communication, 22.2
- Initialization exit, 22.5
- Language, 22.1
- Linkage conventions, 22.2
- OS services exit, 22.11
- Parameters passed, 22.2
- Recovery considerations, 22.2
- Reentrancy considerations, 22.1
- Request exit, 22.7
- Return codes, 22.2
- SNAP dump exit, 22.9
- Termination exit, 22.6
- WTO exit, 22.10
- DRS/NATURAL, 1.7
  - Building the DRS/NATURAL Output Characteristics Table, 32.9
  - Controlling Hardcopy Attributes, 32.8
  - DRS/API Snap dumps, 33.2
  - Dynamic Control of Natural Print Attributes, 32.3
  - Installation, 31.1
  - Installing the DRS/API Product, 31.3
  - Introduction, 1.7, 30.1
  - Mapping Natural Attributes to JES SYSOUT Attributes, 32.4
  - messages, 33.1
  - Messages and Codes, 34.1
  - Natural fix numbers, 31.4
  - Natural Print File Attributes, 32.1
  - Output Characteristics Selection Table, 32.11
  - Output Characteristics Table Header, 32.10
  - Output Characteristics Table Output Options, 32.12
  - SYSUDUMP from ABEND, 33.2
  - Updating the CICS Tables, 31.6
- DRS/NATURAL and DRS/API Return Codes, 33.1
- DRS/Natural Variable Substitution, 32.7
- DRS/PC, 5.1
- DRS/SAPR2
  - Installation, 27.1
  - Introduction, 26.1
- DRS/SAPR2 abend codes, 35.101
- DRS/SAPR2 Printer Alias Table, 27.14
- DRS/SAPR2 problem determination, 29.1
- DRS/SAPR2 SYSOUT parameters, 28.10
- DRS/SAPR2 System Defaults, 28.2
- DRS/STI, 7.1
  - CICS tables, 20.15
- DRS/TCPIP
  - Installation, 13.2
  - Introduction, 12.1
  - MVS requirements, 13.3
- DRS/VPI
  - Banner pages, 9.2
  - Control library members, 3.9
  - DRS system options, 3.8
  - EDRAINED printer messages, 35.91
  - Installation, 3.1

---

---

- Installation of DRS/PC, 5.1
- Introduction, 2.1
- JCL, 3.7
- Messages, 35.20
- Operation, 8.1
- Separator routine, 9.2
- User exits, 9.1, 9.5
- VTAM applications, 4.1
- VTAM definition requirements, 3.5
- With DRS/TCPIP, 6.1
- DRS/VPI Control library members
  - Exclusion list member, 3.11, 3.40
  - Inclusion list member, 3.10, 3.39
  - Message modification member, 3.10
  - Output reference members, 3.107
  - Printer default member, 3.41
  - Printer default members, 3.11
  - Printer definition members, 3.12, 3.42
  - Syntax, 3.9
  - System initialization member, 3.10, 3.13
  - System initialization parameters, 3.15
- DRS/VPI Operation
  - Command summary, 8.3
  - Commands, 8.4
- DRS/VPI User exits
  - User exit 01, 9.6
  - User exit 02, 9.8
  - User exit 03, 9.9
  - User exit 04, 9.11
  - User exit 05, 9.13
  - User exit 06, 9.15
  - User exit 07, 9.16
  - User exit 08, 9.18
  - User exit 09, 9.20
  - User exit 10, 9.22
- DRS/VPI With DRS/STI, 7.1
- DRSDESC, 3.18
- DRSDRIB macro, 9.3
- DRSLANA macro, 9.3
- DRSNSDEF, 31.6, 31.7
- DRSSINTB, 19.1, 20.3
- DRSSINTC, 19.1, 20.3
- DRSSKEY macro, 20.13
- DRSSKEY module, 20.13
- DRSSPTAB, 27.13
- DRSSHTB, 35.99
- DRSSHTC, 20.3, 20.16
- DRSSUE00, 22.5
- DRSSUE01, 22.6
- DRSSUE02, 22.7
- DRSSUE03, 22.8
- DRSSUE04, 22.9
- DRSSUE05, 22.10
- DRSSUE06, 22.11
- DRSVPRAT macro, 9.3
- DRSVSYAT macro, 9.3
- DRTB, 21.19
- DSN, 3.55
- DSNTYPE, 3.57

- DUPLEX, 3.127
- Dynamic output, 19.1

## E

- EBCDIC, 6.1, 6.2, 12.2, 13.5
- ERRACTN, 3.58
- Error codes, 35.20, 35.91, 35.99, 35.100, 35.101, 35.102
- Establishing User Defaults, 28.7
- EXITnn, 3.19
- Exits, 1.9
  - DRS/API, 20.4, 20.17, 21.15, 22.1
  - DRS/VPI, 3.4, 3.14, 3.19, 3.102, 8.5, 8.8, 9.1, 11.2, 12.1, 12.2, 13.2, 13.11
- EXPDT, 3.59

## F

- FCB, 3.60, 3.128
- FLASH, 3.129
- FORM, 3.61
- FORMDEF, 3.130
- FORMLEN, 3.131
- FORMS, 3.132

## G

- GROUPID, 3.133
- GTF tracing, 3.35, 3.98, 11.1, 11.7, 14.1, 20.10, 20.12, 21.29, 22.3

## H

- HOLD, 3.62

## I

- IBM TCPIP
  - With DRS/TCPIP, 13.6
- Including DRS/Natural in the Natural Nucleus, 32.20
- INDEX, 3.134
- INIT block (DRIB), 21.9
- INIT call, 21.1, 21.5
- Initialization exit for DRS/API, 22.5
- Installation of DRS/API, 20.1
- Installation of DRS/TCPIP, 13.2
- Installation of DRS/VPI, 3.1
- Installing the DRS/API Product, 27.3
- Interface modules
  - DRSSINTB, 20.3
  - DRSSINTC, 20.3
- Interlink TCPIP
  - With DRS, 13.3, 13.4
- INTRAY, 3.135
- Introduction to DRS/API, 19.1
- Introduction to DRS/TCPIP, 12.1

---

Introduction to DRS/VPI, 2.1  
IPDCLASS, 3.63  
IPDSOPTS, 3.64

## J

JCL for DRS/VPI, 3.7  
JES considerations, 23.1  
JES spool, 21.1  
JES2, 23.1  
JES3, 23.2  
JOBNAME, 3.65

## K

### Key

DRSSKEY module, 20.3, 20.4, 20.13  
KEYDRS, 20.13, 35.6  
KEYDRSPC, 5.1  
KEYLPD, 13.4, 35.60  
KEYNAT, 20.13  
KEYSAPR2, 20.13  
KEYSTI, 7.1, 16.1  
KEYDRSPC, 3.20  
KEYLPD, 3.21  
KEYSTI, 3.22

### Keywords - in Output Reference Member

@OUTDESC, 3.108  
ADDRn, 3.109  
BUILDING, 3.110  
BURST, 3.111  
CHARS, 3.112  
CKPTLINE, 3.113  
CKPTPAGE, 3.114  
CKPTSEC, 3.115  
CLASS, 3.116  
COLORMAP, 3.117  
COMPACT, 3.118  
COMSETUP, 3.119  
CONTROL, 3.120  
COPIES, 3.121  
COPYGRP, 3.122  
DATAACK, 3.123  
DEPT, 3.124  
DEST, 3.125  
DPAGELBL, 3.126  
DUPLEX, 3.127  
FCB, 3.128  
FLASH, 3.129  
FORMDEF, 3.130  
FORMLEN, 3.131  
FORMS, 3.132  
GROUPLD, 3.133  
INDEX, 3.134  
INTRAY, 3.135  
LINDEX, 3.136  
LINECT, 3.137  
MODIFY, 3.138

NAME, 3.139  
NOTIFY, 3.140  
NOTIFY2, 3.141  
NOTIFY3, 3.142  
NOTIFY4, 3.143  
OFFSETXB, 3.144  
OFFSETXF, 3.145  
OFFSETYB, 3.146  
OFFSETYF, 3.147  
OUTBIN, 3.148  
OUTDISP, 3.149  
OVERLAYB, 3.150  
OVERLAYF, 3.151  
OVFL, 3.152  
PAGEDEF, 3.153  
PIMSG, 3.154  
PORTNO, 3.155  
PRMODE, 3.156  
PRERROR, 3.157  
PRTOPTNS, 3.158  
PRTQUEUE, 3.159  
PRTY, 3.160  
RESFMT, 3.161  
RETAINF, 3.162  
RETAINS, 3.163  
RETRYL, 3.164  
RETRYT, 3.165  
ROOM, 3.166  
SYSAREA, 3.167  
TITLE, 3.168  
TRC, 3.169  
UCS, 3.170  
USRDTAnn, 3.171  
USRLIBnn, 3.172  
WRITER, 3.173

### Keywords - in Printer Member

AVGREC, 3.45  
BLKSIZE, 3.46  
CLASS, 3.47  
COMMTYPE, 3.48  
DATACLAS, 3.49  
DDNAME, 3.50  
DEFLTMEM, 3.51  
DEST, 3.52  
DISP, 3.53  
DRAINED, 3.54  
DSN, 3.55  
DSNTYPE, 3.57  
ERRACTN, 3.58  
EXPDT, 3.59  
FCB, 3.60  
FORM, 3.61  
HOLD, 3.62  
IPDCLASS, 3.63  
IPDSOPTS, 3.64  
JOBNAME, 3.65  
LRECL, 3.68  
MAXLPG, 3.69  
MEMBER, 3.70

---

MGMTCLAS, 3.71  
MPP, 3.72  
NODE, 3.73  
OUTREF, 3.74  
PRTROPTS, 3.75  
PRTXDBCS, 3.77  
PRTXLATE, 3.79  
QBUFSIZE, 3.80  
QSPACE, 3.81  
QUNIT, 3.82  
QVOLUME, 3.83  
RECFM, 3.84  
RETPD, 3.85  
SEPAR, 3.86  
SNAP, 3.87  
SPACE, 3.88  
STORCLAS, 3.89  
SUBSYS, 3.90  
TAB, 3.91  
TCPLIMIT, 3.92  
TCPMRD, 3.93  
TCPOPTS, 3.94  
TERMRPT, 3.96  
TRACE, 3.98  
TRN, 3.99  
UCS, 3.101  
UDATA, 3.102  
UNIT, 3.103  
VOLCNT, 3.104  
VOLUME, 3.105  
WRITER, 3.106

Keywords – in System Initialization Member

AUTOACT, 3.16  
DEFLTMEM, 3.17  
DRSDESC, 3.18  
EXITnn, 3.19  
KEYDRSPC, 3.20  
KEYLPD, 3.21  
KEYSTI, 3.22  
LOG, 3.23  
LOGHOLD, 3.24  
LOGOUTP, 3.25  
MAXPRTS, 3.26  
MLISTMEM, 3.27  
MSMODMEM, 3.28  
SNAPHOLD, 3.29  
SNAPOUTP, 3.30  
SWAPABLE, 3.31  
TCPIPID, 3.32  
TCPPORT, 3.33  
TCPTYPE, 3.34  
TRACE, 3.35  
USEROPTS, 3.36  
WTO, 3.37  
XLISTMEM, 3.38

## L

LINDEX, 3.136  
LINECT, 3.137  
LOG, 3.23  
LOGHOLD, 3.24  
LOGOUTP, 3.25  
LPD  
    Converting LPD configuration SERVICE definitions, 13.6  
    In COMMTYPE keyword, 13.5  
    With DRS/TCPIP, 12.1, 13.4  
LRECL, 3.68

## M

Macros DSECT mapping  
    #LRTCCNT macro, 9.3  
    DRSDRIB macro, 9.3  
    DRSLANA macro, 9.3  
    DRSVPRAT macro, 9.3  
    DRSVSYAT macro, 9.3  
MAXLPG, 3.69  
MAXPRTS, 3.26  
MEMBER, 3.70  
Messages for DRS/API, 35.1  
Messages from DRS/VPI, 35.20  
MGMTCLAS, 3.71  
Migrating to DRS R3.3, 20.3  
Migrating to DRS/VPI R3.3, 3.3  
MLISTMEM, 3.27, 3.42  
MODIFY, 3.138  
MPP, 3.72  
MSMODMEM, 3.28  
MVS requirements for DRS/TCPIP, 13.3

## N

NAME, 3.139  
NATRDO, 31.6  
NDEFASMH, 32.9  
NDEFASML, 32.9  
NODE, 3.73  
NonCICS mode with DRS/API, 21.2  
NOTIFY, 3.140  
NOTIFY2, 3.141  
NOTIFY3, 3.142  
NOTIFY4, 3.143

## O

OFFSETXB, 3.144  
OFFSETXF, 3.145  
OFFSETYB, 3.146  
OFFSETYF, 3.147  
Operation of DRS/VPI, 8.1  
Options module, 3.4, 20.9  
OS services exit for DRS/API, 22.11

---

OUTBIN, 3.148  
OUTDISP, 3.149  
OUTP, 19.1  
OUTP call, 21.1, 21.6  
OUTPUT block (DROB), 21.40  
OUTPUT reference members, 3.4, 3.9, 3.12, 3.107  
OUTREF, 3.74  
OVERLAYB, 3.150  
OVERLAYF, 3.151  
OVFL, 3.152

## P

PAGEDEF, 3.153  
Parameter options module, 3.4, 20.9  
Parameters for DRS/API calls, 21.5  
PCT, 27.15  
PIMSG, 3.154  
PORTNO, 3.155  
PPT, 27.15, 31.6  
PRINT-CONTROL statement, 28.5  
Printer Alias Table  
    using, 28.8  
Printer Defaults, 28.4  
PRMODE, 3.156  
Problem determination  
    DRS/TCPIP, 14.1  
    DRS/VPI, 11.1  
problem determination  
    DRS/SAPR2, 29.1  
PRTERORR, 3.157  
PRTOPTNS, 3.158  
PRTQUEUE, 3.159  
PRTROPTS, 3.3, 3.75  
PRTXDBCS, 3.77  
PRTXDBCS keyword, 10.2  
PRTXLATE, 3.79  
PRTXLATE keyword, 10.2  
PRTY, 3.160  
PTABASMH, 27.13  
PTABASML, 27.13  
PUT call, 21.1, 21.5

## Q

QBUFSIZE, 3.80  
QOR (Queue Owning Region), 20.16  
QSPACE, 3.81  
QUERY block (DRQB), 21.21  
QUNIT, 3.82  
QURY call, 20.3, 21.1, 21.6  
QVOLUME, 3.83

## R

RDO, 27.15, 31.6  
RECFM, 3.84  
Report SYSOUT Attributes, 28.5

REQUEST block (DRRB), 21.7  
Request exit for DRS/API, 22.7  
RESFMT, 3.161  
Restoring the DRS Distribution Libraries, 20.6  
RETAINF, 3.162  
RETAINS, 3.163  
RETPD, 3.85  
RETRYL, 3.164  
RETRYT, 3.165  
Return codes, 24.1  
ROOM, 3.166

## S

SA03 abend, 35.99  
Sample DRS Programs  
    DRS Call Utility, 25.5  
    Sample COBOL and assembler programs, 25.2  
SAP R2 external spool interface, 1.7, 26.1  
SAPID, 27.14  
SAPRDO, 27.15  
SDEFASMH, 27.4  
SDEFASML, 27.4  
SEPAR, 3.86  
SEPAR keyword, 9.2  
Separators, 3.86, 9.2, 21.9, 22.8  
Setting SYSOUT Attributes with DRS/SAPR2,  
28.1  
SNAP, 3.87  
SNAP dump exit for DRS/API, 22.9  
SNAPHOLD, 3.29  
SNAPOUTP, 3.30  
SPACE, 3.88  
Specifying Printer and Report SYSOUT Attributes,  
28.3  
STORCLAS, 3.89  
SUBSYS, 3.90  
SWAPABLE, 3.31  
SYSAREA, 3.167  
System abend codes, 35.99  
SYSUDUMP from ABEND, 29.3

## T

TAB, 3.91  
TCPIP  
    Converting LPD configuration SERVICE defini-  
    tions, 13.6  
TCPIPID, 3.32  
TCPLIMIT, 3.92  
TCPMRD, 3.93  
TCPOPTS, 3.94  
TCPPORT, 3.33  
TCPTYPE, 3.34  
TERM call, 21.1, 21.6  
TERMINATION block (DRTB), 21.19  
Termination exit for DRS/API, 22.6  
TERMRPT, 3.96



---

TITLE, 3.168  
TRACE, 3.35, 3.98  
Tracing  
    DRS/VPI internal, 11.6  
    GTF, 11.1, 14.1, 20.10  
    User exits, 22.3  
TRANID, 27.15  
TRANSID, 27.18  
Transient Data (TD) queue, 27.18  
Translation  
    DRS/TCPIP, 9.21, 13.5, 13.11  
    DRS/VPI, 3.79, 6.1  
    Effect of filter records, 12.2  
TRC, 3.169  
TRN, 3.99  
TS queue, 20.16

## U

U0700 abend, 35.100  
U0701 abend, 35.100  
UCS, 3.101, 3.170  
UDATA, 3.102  
UNIT, 3.103  
Updating the CICS Tables, 27.15  
User abend codes, 35.102  
User abends  
    CICS DRS/SAPR2 environment  
        DP00, 35.101  
        DP01, 35.101  
    CICS environment  
        DR00, 35.100  
        DR01, 35.100  
        DR02, 35.100  
        DR03, 35.100  
        DR04, 35.100  
        DR05, 35.100  
        DR06, 35.100  
        DR07, 35.100  
        DR08, 35.100  
    NonCICS environment  
        U0700, 35.100  
        U0701, 35.100  
User exits for DRS/API, 22.1  
User exits for DRS/VPI, 9.1, 9.5  
USEROPTS, 3.36  
USRDTAnn, 3.171  
USRLIBnn, 3.172

## V

VOLCNT, 3.104  
VOLUME, 3.105  
VTAM definition requirements for DRS/VPI, 3.5

## W

WRITER, 3.106, 3.173  
WTO, 3.37  
WTO exit for DRS/API, 22.10

## X

XLISTMEM, 3.38

