

CICS® Transaction Server for VSE/ESA™



Report Controller Planning Guide

Release 1

CICS® Transaction Server for VSE/ESA™



Report Controller Planning Guide

Release 1

Note!

Before using this information and the product it supports, be sure to read the general information under "Notices" on page 87.

First Edition (June 1999)

This edition applies to Release 1 of CICS Transaction Server for VSE/ESA, program number 5648-054, and to all subsequent versions, releases, and modifications until otherwise indicated in new editions. Make sure you are using the correct edition for the level of the product.

The CICS for VSE/ESA Version 2.3 edition remains applicable and current for users of CICS for VSE/ESA Version 2.3.

Order publications through your IBM representative or the IBM branch office serving your locality.

At the back of this publication is a page entitled "Sending your comments to IBM". If you want to make any comments, please use one of the methods described there.

© **Copyright International Business Machines Corporation 1982, 1999. All rights reserved.**

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

Contents

Programming interface information	iv
Preface	vii
What this book is about	vii
Who should read this book	vii
What you need to know to understand this book	vii
How to use this book	vii
Notes on terminology	viii
Determining if a publication is current	x
Chapter 1. Introduction to the report controller	1
What the report controller is and what it does	1
The advantages of using the report controller	4
What hardware is suitable for the report controller	5
Creating reports with CICS transactions	5
Chapter 2. Report identification	9
Identifying a report	9
Chapter 3. Report creation	11
Creating a report	11
Opening a report	15
Writing data to a report	16
Closing a report	16
Chapter 4. Report formats	19
Formatting a report	19
Format conversion	19
NOCC, ASA, and MCC formats	20
SCS and T3270 formats	24
BMS-mapped reports	27
ESCAPE format	28
JCL format	29
Chapter 5. Managing reports and printers	31
Managing reports	31
Printing a report	32
Browsing reports	36
Shift-out/shift-in control data sequences	36
Managing printers	37
Sharing a report controller printer with another application	38
Managing input for batch jobs	39
Managing transient data queues	41
Managing reports created by batch programs	43
Audit trail	48
Chapter 6. The report controller and POWER	51
Operating the report controller and POWER	52
Destinations and printers	52
Starting a printer	53

Report controller report printing delay	55
Initial skip processing	56
Chapter 7. Report controller security	57
Security	57
What happens when the security check fails?	58
Report controller BROWSE security	59
Chapter 8. Report controller recovery	61
Report recovery levels	61
Syncpoint	62
PRINTFAIL	63
Chapter 9. Enabling the report controller and printers	65
Enabling the CICS spooler interface	66
Installing the DFHRCF group	66
Preparing the resource definitions	66
Defining VTAM configuration parameters	68
Preparing the CICS startup job	75
Other preparation work	76
Overriding the VTAM startup parameters	76
Submitting the CICS startup job	76
Chapter 10. Report controller sample programs	77
The low balance report sample program	77
Sample programs for use with the report controller tutorial	78
Bibliography	81
Books from VSE/ESA 2.4 base program libraries	82
Books from VSE/ESA 2.4 optional program libraries	84
Notices	87
Trademarks and service marks	88
Index	89
This book is intended to help you to migrate from an earlier release of CICS to CICS Transaction Server for VSE/ESA Release 1.	

This book also documents General-use Programming Interface and Associated Guidance Information and Product-sensitive Programming Interface and Associated Guidance Information provided by

Programming interface information

General-use programming interfaces allow the customer to write programs that obtain the services of CICS.

General-use Programming Interface and Associated Guidance Information is identified where it occurs by an introductory statement to a chapter or section.

Product-sensitive programming interfaces allow the customer installation to perform tasks such as diagnosing, modifying, monitoring, repairing, tailoring, or tuning of CICS. Use of such interfaces creates dependencies on the detailed design or implementation of the IBM software product. Product-sensitive programming

interfaces should be used only for these specialized purposes. Because of their dependencies on detailed design and implementation, it is to be expected that programs written to such interfaces may need to be changed in order to run with new product releases or versions, or as a result of service.

Product-sensitive Programming Interface and Associated Guidance Information is identified where it occurs by an introductory statement to a chapter or section.

Preface

What this book is about

This book is about the IBM CICS Transaction Server report controller facility. It provides guidance about report controller facilities and how to plan for them.

Who should read this book

This book is for application programmers, system programmers, planners, and installers of the report controller. It is mainly for system planners who decide what facilities and resources to provide for report controller users. It should also be of interest if you are planning the applications that are to use the report controller.

What you need to know to understand this book

You are assumed to have read the *CICS Report Controller User's Guide*, which introduces the report controller and describes it in general terms. You should have a basic knowledge of CICS Transaction Server.

How to use this book

You do not need to read sequentially through this book. Like the CICS product itself, this book is designed in a modular fashion—you select the parts that you need.

Notes on terminology

The terms listed in Table 1 are commonly used in the CICS Transaction Server for VSE/ESA Release 1 library. See the *CICS Glossary* for a comprehensive definition of terminology.

<i>Table 1 (Page 1 of 2). Commonly used words and abbreviations</i>	
Term	Definition (and abbreviation if appropriate)
\$(the dollar symbol)	In the character sets and programming examples given in this book, the dollar symbol (\$) is used as a national currency symbol and is assumed to be assigned the EBCDIC code point X'5B'. In some countries a different currency symbol, for example the pound symbol (£), or the yen symbol (¥), is assigned the same EBCDIC code point. In these countries, the appropriate currency symbol should be used instead of the dollar symbol.
BSM	BSM is used to indicate the basic security management supplied as part of the VSE/ESA product. It is RACROUTE-compliant, and provides the following functions: <ul style="list-style-type: none"> • Signon security • Transaction attach security
C	The C programming language
CICSplex	A CICSplex consists of two or more regions that are linked using CICS intercommunication facilities. Typically, a CICSplex has at least one terminal-owning region (TOR), more than one application-owning region (AOR), and may have one or more regions that own the resources accessed by the AORs
CICS Data Management Facility	The new facility to which all statistics and monitoring data is written, generally referred to as "DMF"
CICS/VSE	The CICS product running under the VSE/ESA operating system, frequently referred to as simply "CICS"
COBOL	The COBOL programming language
DB2 for VSE/ESA	Database 2 for VSE/ESA which was previously known as "SQL/DS".

Table 1 (Page 2 of 2). Commonly used words and abbreviations

Term	Definition (and abbreviation if appropriate)
ESM	<p>ESM is used to indicate a RACROUTE-compliant external security manager that supports some or all of the following functions:</p> <ul style="list-style-type: none"> • Signon security • Transaction attach security • Resource security • Command security • Non-terminal security • Surrogate user security • MRO/ISC security (MRO, LU6.1 or LU6.2) • FEPI security.
FOR (file-owning region)—also known as a DOR (data-owning region)	A CICS region whose primary purpose is to manage VSAM and DAM files, and VSAM data tables, through function provided by the CICS file control program.
IBM C for VSE/ESA	The Language Environment-conforming version of the C programming language compiler. Generally referred to as “C/VSE”.
IBM COBOL for VSE/ESA	The Language Environment-conforming version of the COBOL programming language compiler. Generally referred to as “COBOL/VSE”.
IBM PL/I for VSE/ESA	The Language Environment-conforming version of the PL/I programming language compiler. Generally referred to as “PL/I VSE”.
IBM Language Environment for VSE/ESA	The common runtime interface for all LE-conforming languages. Generally referred to as “LE/VSE”.
PL/I	The PL/I programming language
VSE/POWER	Priority Output Writers Execution processors and input Readers. The VSE/ESA spooling subsystem which is exploited by the report controller.
VSE/ESA System Authorization Facility	The new VSE facility which enables the new security mechanisms in CICS, generally referred to as “SAF”
VSE/ESA Central Functions component	The new name for the VSE Advanced Function (AF) component
VSE/VTAM	“VTAM”

Determining if a publication is current

IBM regularly updates its publications with new and changed information. When first published, both the printed hardcopy and the BookManager softcopy versions of a publication are in step, but subsequent updates are normally made available in softcopy before they appear in hardcopy.

For CICS Transaction Server for VSE/ESA Release 1 books, softcopy updates appear regularly on the *Transaction Processing and Data Collection Kit* CD-ROM, SK2T-0730-xx and on the *VSE/ESA Collection Kit* CD-ROM, SK2T-0060-xx. Each reissue of the collection kit is indicated by an updated order number suffix (the -xx part). For example, collection kit SK2T-0730-20 is more up-to-date than SK2T-0730-19. The collection kit is also clearly dated on the front cover.

For individual books, the suffix number is incremented each time it is updated, so a publication with order number SC33-0667-02 is more recent than one with order number SC33-0667-01. Updates in the softcopy are clearly marked by revision codes (usually a “#” character) to the left of the changes.

Note that book suffix numbers are updated as a product moves from release to release, as well as for updates within a given release. Also, the date in the edition notice is not changed until the hardcopy is reissued.

Chapter 1. Introduction to the report controller

This chapter contains the introduces the report controller in the following sections:

- “What the report controller is and what it does”
- “The advantages of using the report controller” on page 4
- “What hardware is suitable for the report controller” on page 5
- “Creating reports with CICS transactions” on page 5

The report controller provides distributed control over printing and printer devices. It is designed so that anyone, with little data processing skill and with no previous knowledge of CICS®, can control the printing of reports. (By **reports**, we mean any spool files managed by the report controller.)

End-user departments, operations staff, and programmers can use the report controller in different ways to manage report production. Control over when, where, and how reports are printed, and which printers are used, can be given to the people concerned with those reports and printers. Printers can be owned either by CICS, or by VSE/POWER.

The report controller is part of CICS for VSE/ESA.™ You need to have VSE/POWER installed, to be able to use the report controller. The report controller provides:

- Special options of the EXEC CICS SPOOL... commands, for writing programs to create reports
- An online tool for managing reports created either by using these commands, or by batch programs
- An interface that transfers reports to CICS printers.

All of these use an interface with VSE/POWER, which comes with CICS. For programming information about the CICS interface to POWER, see the *CICS Application Programming Guide* manual.

What the report controller is and what it does

The report controller is a suite of programs and transactions that interface with VSE/POWER, to assist in creating reports and printing them at distributed locations. The report controller provides **transactions** for end users to:

- Spool existing batch-created reports from the POWER LST queue to a CICS or POWER printer
- Spool CICS-created reports from the POWER LST queue to a CICS or POWER printer.
- Display information about reports, printers, jobs and transient data queues.
- Browse reports on the POWER LST queue.
- Control how, when, and where reports are printed.
- Control jobs in the POWER RDR queue.

- Create a report from the records in a transient data queue, and spool it to a CICS or POWER printer.
- Perform many of the tasks that your operators currently do with POWER commands.

Report controller transactions differ from other CICS-supplied transactions by being **panel-driven**, rather than command-driven. This makes them suitable for users who are less familiar with online systems. For these users, who are sometimes known as **end users**, there is a tutorial guide to using the report controller, see the *CICS Report Controller User's Guide* for details.

The report-controller transactions present the user with a series of panels on which they select listed options. Some of the panels are “form” or “data entry” type panels, on which the user can enter information. Others are “list” type panels, that allow options to be selected against the names of reports, jobs, queues, or printers. **Recovery** of reports, following a failure during creation or printing, is provided. **Security** of reports, printers, transient data queues and jobs is controlled by an external security manager. For more information, see Chapter 7, “Report controller security” on page 57.

Using the EXEC CICS SPOOL... commands, you can create reports, that can be managed using the report controller transactions. All reports are created as POWER spool files. Command options allow you to:

- Identify the report and specify how it is to be created.
- Control the formatting and appearance of the report.
- Influence how, when, and where it is to be printed.
- Provide for security and report recovery.

Some of these options can also be specified in the job control language (JCL) for reports created by batch programs. To handle nonstandard data streams of batch-generated reports, you can specify the name of an ESCAPE program in the JCL for the report. The named ESCAPE program is invoked when the report is printed.

If you use the POWER PNET feature, reports can be routed to and from other nodes using a network job entry (NJE) connection. The remote nodes can be VM/RSCS and MVS/JES systems. as well as other VSE/POWER systems.

Figure 1 on page 3 shows the components involved in creating and printing reports. Programs involving the EXEC CICS SPOOL... commands use the CICS system spooling interface to create reports on the POWER LST queue. The **REPORT option** and other options of the EXEC CICS SPOOL... commands are part of the report controller facility, as are the **printing component**, the **control component**, and the **CEMS and CEOS transactions**.

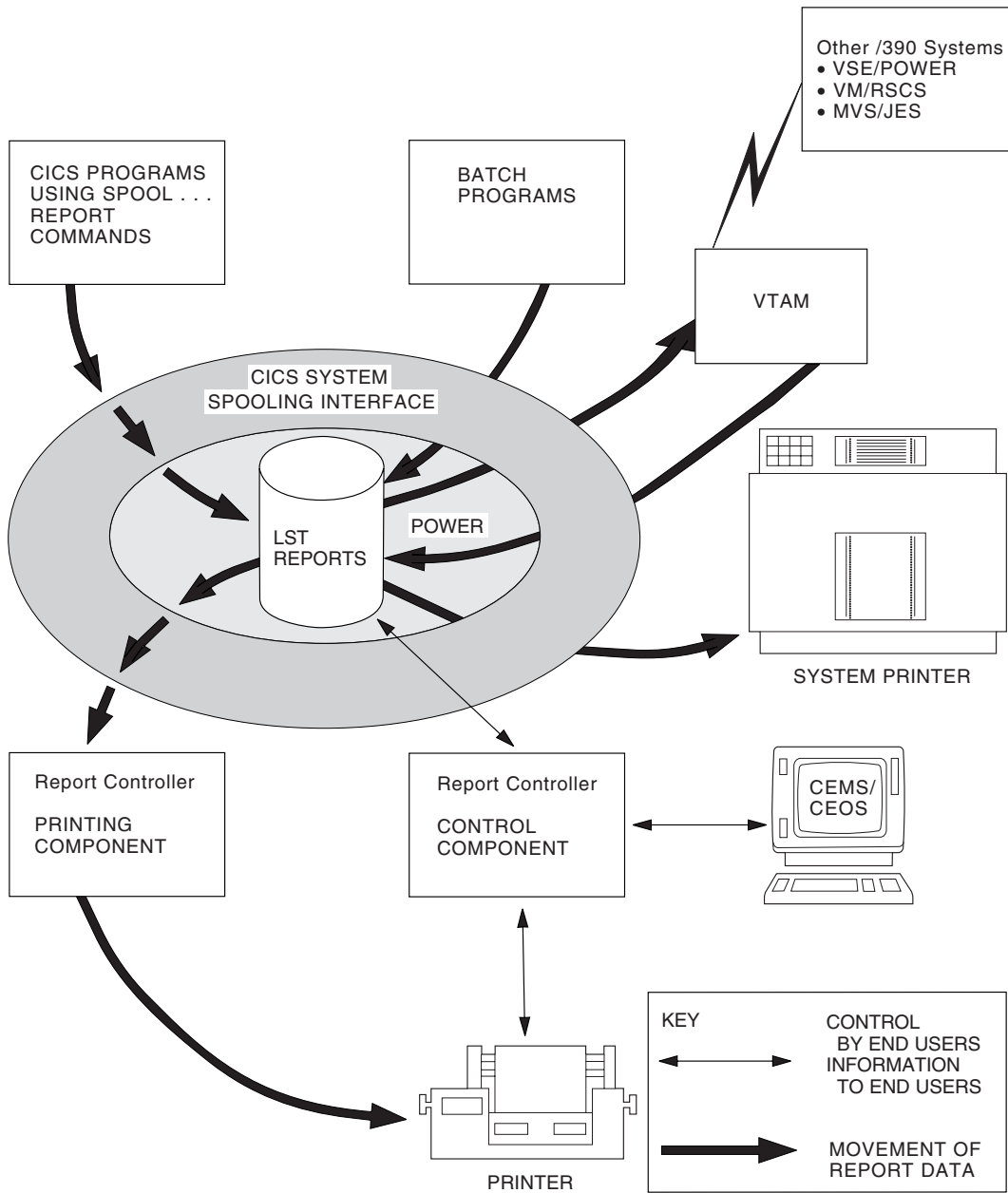


Figure 1. Report controller: reports and printers

Figure 3 on page 40 shows the components involved in creating and controlling input for batch jobs (JCL reports). Figure 4 on page 42 shows the components involved in creating a report from a transient data queue, and then using the report controller to control the printing of the report.

The advantages of using the report controller

The advantages of using the report controller are that:

- You can set up many of the characteristics of a report, and the end user can alter these characteristics if necessary, by using the CICS-supplied transactions CEMS and CEOS. This allows them to make decisions about how their reports are printed.
- You can create reports that can be printed with the same results on either POWER or CICS printers.

Without using the report controller, you can create and write to reports using EXEC CICS SPOOLOPEN REPORT and EXEC CICS SPOOLWRITE REPORT commands, and send them to a POWER printer, but not to a CICS printer.)

- You can leave the decision about which printer is to be used until the user is ready to print the report. This gives you more flexibility when formatting. It also gives you more flexibility for rerouting a report to another printer if a printer fails.

(If you use other methods, you can print data at a specific printer associated with the display device, but you have to format your data specifically for that printer.)

- You can specify physical or logical recovery for your reports in cases of failure during report *creation*.

(If you use only the CICS base spool interface, you cannot specify recovery for reports.)

- If you create reports with an appropriate *disposition*, your users can reprint a report, in cases of failure during report *printing*.

(Otherwise, your users might have to re-create the report, by rerunning the program that created it.)

- You are allowed more independence from the display screen than you may have currently: for instance, you can format a long report without tying up your display device for too long; you do not have to have a printer available at the same time as the display device; and your report format can be completely independent of the panel design.

- The end user can specify the forms type (stationery) that is loaded on a printer, and so avoid reports being printed on the wrong stationery.

(If you use the POWER PSTART command to start a printer, you cannot specify the forms type to be loaded on the printer.)

What hardware is suitable for the report controller

A minimum of an 80-character, 24-line display screen is necessary to run the report controller transactions. For the report browse function, wider and longer screens are exploited if defined appropriately.

Important

To use the report controller transactions CEMS and CEOS on a KANJI or CHINESE terminal, the TYPETERM definition of that terminal must be defined with SOSI=YES and EXTENDEDDES=YES in order to display the DBCS fields correctly.

You can print the reports on the following:

- A printer defined to CICS, that supports the 3270 data stream or the SNA character set (SCS). For example, the IBM® 3287 and IBM 3289 support both of these. (DBCS can be supported by both of these printers.) The printer can be connected using VTAM® or SNA.
- A batch printer owned by VSE/POWER. For example, an IBM 3211.

Creating reports with CICS transactions

Because a report is equivalent to a spool file, the EXEC CICS commands begin with SPOOL, to distinguish them from commands for creating ordinary files. Here they are referred to collectively as the EXEC CICS SPOOL... commands. They are:

EXEC CICS SPOOLOPEN REPORT

Used to open reports for the first time and reopen resumable reports.

EXEC CICS SPOOLWRITE REPORT

Used to write data to reports.

EXEC CICS SPOOLCLOSE REPORT

Used to close reports.

The REPORT option identifies the command as a report controller command.

You can use the report controller EXEC CICS SPOOL... commands in command-level assembler-language, COBOL, or PL/I programs.

You can use the command interpreter CECI to try out the EXEC CICS SPOOL... commands, but this is not recommended as a means of creating real reports. You can use CECI to view the commands even if the report controller is not installed. (You cannot execute the commands unless you have the report controller installed.)

You can use the execution diagnostic facility CEDF to debug application programs that use the EXEC CICS SPOOL... commands.

The report controller can also manage reports created by batch programs, and it can be used to create reports from transient data queues. See “Managing reports created by batch programs” on page 43, and “Managing transient data queues” on page 41.

Report types

The report controller recognizes three types of report:

A **standard** report is created by a single CICS task. This is the same as a normal VSE/POWER LST queue file.

A **resumable** report can be created by a series of CICS tasks, each of which can append one or more records to it. VSE/POWER refers to this type of report as appendable.

A **log** report may be created by several CICS tasks running concurrently. Records written by each CICS task are interleaved, but this type of report appears to VSE/POWER as a standard report.

You need to open a log report only once, and then close it when all records have been written. The log report can be closed either by the task that opened it or by another task.

Report options

Some of the options you can specify when you open a report for the first time provide very useful functions:

- USERDATA is a 16-character field that appears on the report list panel. You can use it to provide reference information for the end user.
- The DESTINATION of the report refers to the logical identifier for a printer device, or a group of printers. It can be defined as SPOOLDEST in the TERMINAL definition for each printer, but it can also be set when the printer is started, or changed by an authorized CEMS or CEOS transaction user. Using DESTINATION avoids coding a specific printer name in your program, or associating a particular printer with a display device. (However, you can set a default destination for a display device, using the SPOOLTO attribute in the TERMINAL definition for the device.)
- You can specify a resource security level, RSL. This can be either PUBLIC, or a value in the range 1 through 24. This value determines:
 - Which users can control the printing of the report
 - Which users can write to a resumable or log report
 - Which printers can print the report

Note: Although CICS internal security is not supported in CICS Transaction Server, the RSL specification is retained, for compatibility, for use by existing applications and the routing of reports to earlier releases of CICS/VSE®. For more information, see Chapter 7, “Report controller security” on page 57.

An additional security option is PRINTFAIL, which prevents a report from being reprinted automatically after a failure during printing.

- The HEAD and FOOT options allow you to specify a single line of text to be printed at the top or bottom, or at both the top and bottom, of each page.
- The report controller takes care of the page numbering as you require it. You can specify one of the following:
 - NONUM, if you do not want automatic page numbering
 - HEADNUM, if you want the page number in the top right-hand corner of each page

- FOOTNUM, if you want the page number in the bottom right-hand corner of each page
- BOTHNUM, if you want the page number in both top and bottom right-hand corners of each page.
- The report controller can also put the date and time that the report was created, on the top line of each report. You specify this with the DATETIME keyword.
- You can have separator pages, two before and two after the report. You use the SEP keyword or the NOSEP keyword as appropriate. If you print the report on a POWER system printer, you get the standard POWER separator pages. If you print the report on a CICS printer, you get separator pages that give much the same information as the POWER separator pages, but the width of the printing is tailored to fit the width of the paper that is loaded.
- You can specify the number of LINES for each page, instead of using carriage control characters.
- You can specify the format of the report, using the appropriate report format options (see Chapter 4, “Report formats” on page 19).
- Users of double byte character sets (DBCS) can specify whether shift-out/shift-in data sequences should take up a print position in a report (see “Shift-out/shift-in control data sequences” on page 36):
 - SOSI1 if you want shift-out/shift-in data sequences to take up a print position
 - SOSI2 if you *do not* want shift-out/shift-in data sequences to take up a print position
- The PWRSYSID option allows you to direct a report to a specific VSE system in a shared spool environment.

Chapter 2. Report identification

This chapter describes the options of the EXEC CICS SPOOL... commands that are used to identify reports. These are the REPORT, TITLE, USERDATA, and TOKEN options.

Identifying a report

The report name is the primary means of identifying the report. It is probably the name by which the end user knows the report. Your naming conventions for reports can influence the end users' efficiency, because they can use partial (or generic) names to request information about reports. If you allocate two or three characters to a user or group of users, and then start all their report names with those characters, it is easy for them to list all their reports by typing only those characters.

You identify the report by name in the REPORT option of the EXEC CICS SPOOL... commands. This option makes the report you are creating into a report that the end user can print using the report controller transactions CEMS or CEOS. You must use the REPORT option on each command; if you do not, the command is not recognized as a **report controller** command.

Report title

You can provide a 32-character **title** for the report. If you do not, the report name is used. The title appears on the CEMS and CEOS report characteristics panel.

Userdata

You can provide a 16-character description known as **userdata**. If you do not, the first 16 characters of the title are used. If you do not provide either title or userdata, the report name is used.

The userdata appears on the report controller report list panel. This is quite significant from the point of view of security, because any report controller user can see the contents of userdata for all reports. So, do not put any sensitive information in userdata. If you do not put anything in userdata, be careful about sensitive information in the title.

You can put whatever information you want in userdata. You may want to use it:

- To provide a reference number or other code, so that whoever created a report can recognize it on the report controller list panel. (You may have many users creating reports with the same name, or one user may create many versions of the same report. The reference number could be supplied by them to your application program, or generated by the application and displayed back to them.)
- To provide a reference number or other code, in the text of a report, so that whoever is controlling printing can relate the output back to the report controller list panel. (They may have several versions of the same report: this will help them to see which versions have been printed.)

- To provide a more meaningful name for the report. For example, if you code:
REPORT(TIM74MAY)
TITLE('DEPT74 TIMESHEET J.SMITH')

USERDATA defaults to the first 16 characters of TITLE, and DEPT74
TIMESHEET is visible to all report controller users. This simply elaborates a
little on the report name. J.SMITH is only visible to report controller users
authorized to see the characteristics of that report.

Token

A **token** is an 8-byte identifier that is used by each EXEC CICS SPOOL... command so that the CICS task can recognize which report is being created. When a report is first opened for input, CICS allocates a token to it. This token must be stored by the application program, to be used when the report is written to and when it is closed.

When a resumable or log report is closed, the token must be preserved and used when the report is opened again.

Report number

You have no control over the report number in your program. The report number is allocated by POWER. It is shown on the CEMS and CEOS report list and report characteristics panels, and so can be used to distinguish between reports of the same name that are waiting to print. It is also printed on the separator pages preceding and following each report, if you choose to have them.

Chapter 3. Report creation

This chapter contains information about creating reports in the following sections:

- “Creating a report”
- “Opening a report” on page 15
- “Writing data to a report” on page 16
- “Closing a report” on page 16

This chapter describes the types of report that you can create and the command options used to create them. For details of the EXEC CICS SPOOL... commands, refer to the *CICS Application Programming Reference* manual.

Creating a report

You can create three different types of report: standard, resumable, and log reports. They are described in detail here, together with examples of coding to create each type.

Standard report

A standard report is created by a single execution of a CICS transaction (a single task). A standard report is opened, written to, and then closed. If your program has all the data it needs to create the whole report before opening, the report can be standard.

The following coding shows how to create a standard report:

```
EXEC CICS SPOOLOPEN
        REPORT(N21001)
        TOKEN(TOK001)
        NOCC

EXEC CICS SPOOLWRITE
        REPORT(N21001)
        TOKEN(TOK001)
        FROM(I0001)

.
.
EXEC CICS SPOOLWRITE
        REPORT(N21001)
        TOKEN(TOK001)
        FROM(I0001)

.
.
EXEC CICS SPOOLWRITE
        REPORT(N21001)
        TOKEN(TOK001)
        FROM(I0001)

EXEC CICS SPOOLCLOSE
        REPORT(N21001)
        TOKEN(TOK001)
        HOLD
```

Resumable report

A resumable report is created by a series of separate CICS tasks, each of which may append data to the report. A resumable report is opened in exactly the same way as a standard report. At the end of the task it is closed with the RESUME option. At the beginning of the next task, it is opened with the RESUME option, and so on, until it is finally closed without the RESUME option.

The following coding shows how to create a resumable report:

TRN1:

```
EXEC CICS SPOOLOPEN
          REPORT(N21002)
          TOKEN(TOK002)
          ASA

EXEC CICS SPOOLWRITE
          REPORT(N21002)
          TOKEN(TOK002)
          FROM(I0002)
.
EXEC CICS SPOOLCLOSE RESUME
          REPORT(N21002)
          TOKEN(TOK002)

EXEC CICS RETURN
          TRANSID(TRN2)
          COMMAREA(TOK002)
```

TRN2:

```
EXEC CICS SPOOLOPEN RESUME
          REPORT(N21002)
          TOKEN(TOK002)

EXEC CICS SPOOLWRITE
          REPORT(N21002)
          TOKEN(TOK002)
          FROM(I0002)
.
EXEC CICS SPOOLCLOSE
          REPORT(N21002)
          TOKEN(TOK002)
          RELEASE
```

If your program is pseudoconversational, that is, if control is returned to CICS while the user is entering data on the screen, you should create a resumable report. You can gather some of the data, write it to the report, and then send another screen for the user to enter more data. You then close the report with the RESUME option, and store the token for use by the transaction that opens the report again. CICS uses the TOKEN to identify individual reports. You can use this technique for a series of different transactions, or for the same transaction invoked iteratively.

You must also create a resumable report if you want to issue a syncpoint during report creation. This is because a syncpoint closes any report that is open, and

sets it to READY(D) status (see “Closing a report” on page 16). To prevent this, you can code an EXEC CICS SPOOLCLOSE RESUME REPORT command before the syncpoint. You must specify LOGICAL recovery for the report if you are using syncpointing (see “Report recovery levels” on page 61).

The following coding shows how to create a resumable report, in a program that includes a syncpoint:

```
EXEC CICS SPOOLOPEN
        REPORT(N21003)
        TOKEN(TOK003)
        RSL(5)
        LOGICAL
        NOCC

EXEC CICS SPOOLWRITE
        REPORT(N21003)
        TOKEN(TOK003)
        FROM(I0003)

EXEC CICS SPOOLCLOSE RESUME
        REPORT(N21003)
        TOKEN(TOK003)

EXEC CICS SYNCPOINT

EXEC CICS SPOOLOPEN RESUME
        REPORT(N21003)
        TOKEN(TOK003)

EXEC CICS SPOOLWRITE
        REPORT(N21003)
        TOKEN(TOK003)
        FROM(I0003)

.
.

EXEC CICS SPOOLCLOSE
        REPORT(N21003)
        TOKEN(TOK003)
        RELEASE
```

Notes:

1. You cannot create a resumable report in JCL format; see “JCL format” on page 29.
2. You cannot direct a resumable report to a destination at a different node.
3. When the EXEC CICS SPOOLOPEN RESUME command is issued, a resource security level check is performed. The operator must have READ access to the FACILITY resource CICSRCF.RSL nn where nn is the RSL value that is specified on the first EXEC CICS SPOOLOPEN command.
4. Any number of resumable reports with the same name may be opened at the same time.

Log report

A log report when open can be written to by a number of tasks running concurrently.

You need to open a log report only once, and then simply close it when all records have been written. The log report can be closed either by the task that opened it or by another task.

A log report can be created in any format. For any particular report name, only one log report may be open at one time with that name. An attempt to open a log report that is already open results in the NAMEERROR condition being raised.

The following coding shows how to create a log report:

Task 1:

```
EXEC CICS SPOOLOPEN  
          REPORT(N21004)  
          TOKEN(TOK004)  
          LOG  
          ASA
```

```
EXEC CICS SPOOLWRITE  
          REPORT(N21004)  
          TOKEN(TOK004)  
          FROM(I0004)  
          LOG
```

.

Task 2:

```
EXEC CICS SPOOLWRITE  
          REPORT(N21004)  
          TOKEN(TOK004)  
          FROM(I0004)  
          LOG
```

.

Task 3:

```
EXEC CICS SPOOLWRITE  
          REPORT(N21004)  
          TOKEN(TOK004)  
          FROM(I0004)  
          LOG
```

.

Task n:

```
EXEC CICS SPOOLWRITE  
          REPORT(N21004)  
          TOKEN(TOK004)  
          FROM(I0004)  
          LOG
```

```
EXEC CICS SPOOLCLOSE  
          REPORT(N21004)  
          TOKEN(TOK004)  
          LOG
```

Opening a report

The EXEC CICS SPOOLOPEN REPORT command is used in two ways:

- To open a report for the first time
- To reopen a resumable report that has been closed

Opening a report for the first time

When you open a report for the first time, you specify, either explicitly or by default, many options that set the characteristics of the report.

In addition to specifying REPORT, TITLE, USERDATA, and TOKEN, you also specify:

- The report format, for example, ASA, MCC, NOCC, JCL
- Other formatting options
- Options that help control printing
- Options for security and report recovery

Here is an example of the command that opens a report for the first time:

```
EXEC CICS SPOOLOPEN
        REPORT('MKT01A')
        TOKEN(TOKENC)
        NOCC
        TITLE('MARKET ANALYSIS A')
        HEAD(HEADC)
        USERDATA(USERC)
        DATETIME
        SEP
        HEADNUM
```

Reopening a resumable report

The RESUME option on an EXEC CICS SPOOLOPEN REPORT command specifies that a resumable report is being reopened. The only other options you specify with this are REPORT and TOKEN, so the command looks like this:

```
EXEC CICS SPOOLOPEN
        RESUME
        REPORT('WAGEREPT')
        TOKEN(TOKEN1)
```

Here, WAGEREPT is a literal character string, and TOKEN1 is the name of a variable whose data value is the token preserved from the EXEC CICS SPOOLCLOSE RESUME REPORT command last executed for this report.

When you use EXEC CICS SPOOLOPEN RESUME REPORT, the next EXEC CICS SPOOLWRITE REPORT command appends a record to the report with the same token. (The report must previously have been closed with the RESUME option.)

Writing data to a report

You use the EXEC CICS SPOOLWRITE REPORT command to write data to a report. You usually specify the data to be written using the FROM option, except for BMS-mapped reports, when you can use the MAPONLY option instead. You can let CICS calculate the length of the data, or you can specify the length using the FLENGTH option. Note that, if you are writing a COBOL program to create a report that is not a BMS-mapped report, you **must** specify FLENGTH.

You can write any number of lines in one EXEC CICS SPOOLWRITE REPORT command. CICS calculates in different ways how many lines to write, depending on the report format. For more detail, see “Formatting a report” on page 19.

You can specify the length of each line using the LINELENGTH option of EXEC CICS SPOOLOPEN REPORT. If LINELENGTH is omitted, the line length defaults to 132 (NOCC, SCS, and T3270 formats) or 133 (ASA and MCC formats). The number of lines written by each SPOOLWRITE is calculated by dividing the FLENGTH value by the LINELENGTH value, any remainder being padded with blanks and considered as one line. This results in a report consisting of lines of fixed length. This applies to ASA, MCC, NOCC, JCL, and sometimes to SCS and T3270 report formats.

For ASA and MCC, the number of lines written also depends on any valid control characters found in column 1 **after** the FROM area has been split up into lines by dividing FLENGTH by LINELENGTH.

For JCL, the LINELENGTH is always 80. You must ensure that each JCL statement begins in column 1 and ends in column 80, after the FROM area has been split up into lines.

With other report formats, each EXEC CICS SPOOLWRITE REPORT command is assumed to be for a page. With BMS-mapped reports, the map determines how many lines are written. With SCS and T3270, all of the data can be considered as a single unit and written as one record. (Control orders within the data stream control the number of lines to be written.) With ESCAPE format, you must control the lines and pages in your escape program.

Here is an example of the command that writes a line of data to a report:

```
EXEC CICS SPOOLWRITE
      REPORT('MKT01A')
      TOKEN(TOKENC)
      FROM(PRINTLINEC)
      FLENGTH(132)
```

Closing a report

You use the EXEC CICS SPOOLCLOSE REPORT command to close a report. In addition to the mandatory options REPORT and TOKEN, there are some other options that you can use with the EXEC CICS SPOOLCLOSE REPORT command.

The LOG option identifies that a report is a log report.

When you close a report for the last time, you can specify DISP(data-value) DELETE, HOLD, KEEP, or RELEASE.

DISP(data-value) specifies the POWER disposition status of the report once it is closed:

- D** The report is created for immediate printing and is deleted after printing.
- H** The report is created in HELD(H) status. When released, it is automatically deleted, after printing.
- K** The report is created for immediate printing. After printing, it is retained on the POWER LST queue in HELD(L) status.
- L** The report is created in HELD(L) status. After printing, the report is returned to HELD(L) status.

Specifying HOLD is equivalent to specifying DISP('L'), RELEASE is equivalent to DISP('D') and KEEP is equivalent to DISP('K'). If you do not specify a disposition, the report is closed with a disposition of D. The CEMS and CEOS panels display the disposition of reports.

The DELETE option is used only when you do not, after all, want already-written report records to remain as a spool file on the POWER LST queue.

When you close a resumable report, you specify the RESUME option. This option indicates that the report has not been closed permanently: writing to it may be resumed by a later CICS task. You must preserve the token and use it when you code the EXEC CICS SPOOLCLOSE RESUME REPORT to reopen the report.

A syncpoint forces a report to be closed in READY(D) status, so to create a report across more than one logical unit of work, you must code an EXEC CICS SPOOLCLOSE RESUME REPORT command before the SYNCPOINT command. EXEC CICS SPOOLCLOSE RESUME REPORT puts the report in **resume** status (POWER DISP=A). You must then code an EXEC CICS SPOOLCLOSE RESUME REPORT command before you can write to the report again.

The following coding shows examples of the EXEC CICS SPOOLCLOSE REPORT command:

```
EXEC CICS SPOOLCLOSE
          REPORT(N21004)
          TOKEN(TOK004)
          DISP('L')
```

```
EXEC CICS SPOOLCLOSE
          REPORT(N21005)
          TOKEN(TOK005)
          HOLD
```

```
EXEC CICS SPOOLCLOSE
          RESUME
          REPORT(N21006)
          TOKEN(TOK006)
```

Chapter 4. Report formats

This chapter discusses report formats in the following sections:

- “Formatting a report”
- “Format conversion”
- “NOCC, ASA, and MCC formats” on page 20
- “SCS and T3270 formats” on page 24
- “BMS-mapped reports” on page 27
- “ESCAPE format” on page 28
- “JCL format” on page 29

This chapter describes the various formats for reports and the command options that determine the format of a report.

Formatting a report

You can create a report in a number of different formats, which you specify on an EXEC CICS SPOOLOPEN command using one of the following options:

NOCC No carriage control characters.

ASA ASA control characters.

MCC Machine code carriage control.

SCS SNA character string.

T3270 3270 data stream.

MAP BMS mapping.

ESC Escape mapping, for special things like graphics.

JCL Job control language format. The report is placed on the POWER RDR queue to be input to a batch job.

The format of a report dictates whether certain formatting options are valid; for example, headings or footings are not supported for a BMS map.

Format conversion

Whether you specify ASA, MCC, NOCC, SCS, or T3270, the report controller arranges that the report can be printed on a system printer, on an SCS printer, or on a 3270-type printer, **with the same results**.

It does this by converting all these formats to MCC. If you do not want the conversion, because you want to use the special features of a 3270-type or SCS printer, for instance, color, you can specify NOCONV (see page 27). If you do this, you get the special features if you use the appropriate printer. However, the report cannot be printed on a different kind of terminal printer, or directly through POWER to a channel-attached system printer.

Format conversion enables distributed report printing: each recipient receives an identical report, whether it was printed on a CICS printer or on a system printer.

NOCC and ASA reports created using an EXEC CICS SPOOLOPEN REPORT command are always converted to MCC. ASA reports created by batch programs can also be printed on SCS or 3270 printers.

ESC, MAP, and JCL reports are not converted, but remain fixed for the lifetime of the report.

NOCC, ASA, and MCC formats

No carriage control (**NOCC**) is the format to choose if you do not need to be able to insert unconditional page breaks and line skips. The LINES option specifies the maximum number of lines on each page, and a new page is started whenever the LINES value is exceeded by the line count.

The number of lines created by each EXEC CICS SPOOLWRITE REPORT command is calculated by dividing the LINELENGTH value into the FLENGTH value. The default LINELENGTH is 132. FLENGTH defaults to the length of the data area specified by the FROM option. Each line thus created counts as one line in the line count, and each heading and footing counts as two lines.

NOCC data is always converted to MCC format. The machine code characters are inserted as the lines are inserted. Do not allow for them in your LINELENGTH.

If you want to be able to specify unconditional line skips or page breaks in your program, you should choose **ASA** format. You can also make use of the LINES option to have an automatic page break wherever a page is filled.

Using ASA format all you do is code one of the following characters in the first position of each line that you want to print:

- blank** Space 1 line before writing the data.
- 0** Space 2 lines before writing the data.
- Space 3 lines before writing the data.
- +** Suppress space, that is, print on same line as previous line.
- 1** Skip to channel 1 (new page) before writing the data.

ASA data is always converted to MCC format. You can improve performance by writing several ASA lines in one EXEC CICS SPOOLWRITE REPORT command. The resulting increase in speed may, however, be offset by the increased storage needed for buffers.

If you write more than one ASA line in one EXEC CICS SPOOLWRITE REPORT command, the LINELENGTH value is divided into the FLENGTH value to calculate the number of lines. You must ensure that, after this calculation is made, the ASA control characters are in column 1. The total number of lines printed is FLENGTH divided by LINELENGTH adjusted to take account of the ASA characters. The default LINELENGTH is 133 (allowing for the ASA character and 132 characters of data).

If you specify `LINES(0)`, you can skip to channels¹ 2 through 12. `LINES(0)` means that no automatic page breaks are inserted. If you want automatic page breaks, you cannot skip to channels 2 through 12.

During report printing, invalid characters (in the range `X'00'` through `X'3F'`, or `X'FF'`) within ASA and MCC data records are converted to blanks to prevent terminal errors. However, if the printer is defined to support a double-byte character set (DBCS), records are printed as they are received from POWER, if the invalid characters are framed with shift-out/shift-in (SO/SI) control sequences.

Machine control character (**MCC**) format provides a similar function, but the order of the printing and paper movement is generally reversed; for example, an ASA character of `0` would first advance the paper position by two lines and then print the data line, the equivalent MCC control of `X'11'` would first print the data line and then advance the paper position by 2 lines. Further details about ASA and MCC (including the full range of codes) can be found in the *VSE/ESA System Macros Reference*.

Hex	Result
0B	Space 1 line immediately (no printing occurs).
13	Space 2 lines immediately (no printing occurs).
1B	Space 3 lines immediately (no printing occurs).
09	Space 1 line after writing the data.
11	Space 2 lines after writing the data.
19	Space 3 lines after writing the data.
89	Skip to channel 1 (new page) after writing the data.
8B	Skip to channel 1 (new page) immediately (no printing occurs).
01	Suppress space, that is, print on same line as previous line.

Note: There may be a slight performance improvement if you use MCC format, because no conversion takes place.

The options you can use for formatting NOCC, ASA, and MCC reports are as follows:

LINES

Specifies the page length in terms of number of lines on the page.

LINELENGTH

Specifies the page width in terms of number of characters in the line.

HEAD and FOOT

Specify heading and footing text.

NONUM, HEADNUM, FOOTNUM, and BOTHNUM

Specify page numbering.

DATETIME

Specifies date and time.

¹ The channel codes 1 through 12 each identify a position in a forms control buffer or in a page definition.

Page length

LINES (in the range 0 through 99) can be specified on the EXEC CICS SPOOLOPEN REPORT command for NOCC, ASA, or MCC format reports and for reports with SCS or T3270 format (without NOCONV). During creation, the report controller keeps a line count and, if the LINES value is exceeded, a page break is created using appropriate carriage control characters.

The number of lines per page is counted according to the report format, as follows:

ASA and MCC

The line count is based on the carriage control character; that is, the actual number of printed lines is calculated.

- A heading counts as two lines (that is, one header line and one blank line).
- A footing counts as two lines (that is, one blank line and one footing line).
- If the carriage control character specifies skip to line one on new page, the line count is reset.

NOCC

The line count is calculated as follows:

- Each line counts as one line.
- A heading counts as two lines.
- A footing counts as two lines.

If LINES is not specified, a system default is used. If LINES(0) is explicitly specified, no page formatting is attempted; DATETIME, HEAD, FOOT, HEADNUM, FOOTNUM, and BOTHNUM are invalid options with LINES(0).

Line length

LINELENGTH can be specified at EXEC CICS SPOOLOPEN for ASA, MCC, or NOCC format reports. LINELENGTH can also be specified at EXEC CICS SPOOLOPEN for reports with SCS or T3270 format if NOCONV is not specified. During creation, the report controller inserts new-line (NL) carriage control characters when the output of the EXEC CICS SPOOLWRITE command reaches the line length specified.

Headings

A heading (defined as the top line of each new page) is supported for reports of ASA, MCC, and NOCC format. The heading is supplied on the EXEC CICS SPOOLOPEN REPORT command and appears as the first line of each printed report page.

A heading is printed whenever:

- Printing of a report begins.
- A carriage control character specifying “skip to new page” is encountered.
- Lines per page (if specified) is exceeded.

A blank line always appears immediately after a heading line.

Footings

A footing (defined as the bottom line of each page) is supported for reports of ASA, MCC, and NOCC format. The footing is supplied on the EXEC CICS SPOOLOPEN REPORT command and appears at the bottom of each printed page.

The bottom line is calculated from the LINES option or by implying the system default.

A footing is printed whenever:

- A carriage control character specifying “skip to new page” is encountered (unless it appears on the very first line of the report).
- Lines per page or the system default is exceeded.
- The end of the report is reached.

A blank line always appears immediately before the footing.

Page numbering

Automated page numbering is requested by specifying the NONUM, HEADNUM, FOOTNUM, or BOTHNUM options of the EXEC CICS SPOOLOPEN REPORT command.

The page number appears as follows:

NONUM

No page number appears. This is the default.

HEADNUM

The page number appears as the last 6 characters (leading zeros suppressed) of the heading line. (If no heading line is supplied on the EXEC CICS SPOOLOPEN REPORT command, a blank heading line is produced.)

FOOTNUM

The page number appears as the last 6 characters (leading zeros suppressed) of the footing line. (If no footing line is supplied on the EXEC CICS SPOOLOPEN REPORT command, a blank footing line is produced.)

BOTHNUM

Page numbers appear as specified for both HEADNUM and FOOTNUM.

Date and time

On a report where a heading is specified on the EXEC CICS SPOOLOPEN REPORT command or a heading is implied by specifying HEADNUM or BOTHNUM, specifying DATETIME triggers the report controller to insert on each heading line the date and time of creating the report.

If no heading is specified, a blank heading line is produced.

The date and time appear 20 characters before the end of the line in one of the following system-defined formats:

DD/MM/YY HH:MN
MM/DD/YY HH:MN
YY/MM/DD HH:MN

where:

DD is the day of the month.
MM is the month.
YY is the year.
HH is the hour of the day.
MN is the minute within the hour.

The format used is as that defined with the DATFORM parameter of the system initialization table.

SCS and T3270 formats

You can use these formats to take advantage of SCS or T3270 printer options (with NOCONV), or to keep all reports using the same format. The T3270 format can be used to convert a report based on BMS mapping to print on a system printer. For more information, see “BMS-mapped reports” on page 27.

Usually, when you send SNA character string (SCS) data or a 3270 data stream to a printer, you do not send it as individual lines, but as one long string containing SCS or 3270 control orders as well as data. It is the printer that acts upon these control orders and formats the data accordingly. You can do this to create reports using the EXEC CICS SPOOL... commands, but the report controller provides another option: by default, the report controller converts SCS and T3270 reports into MCC format just as it does for NOCC reports, dividing LINELENGTH into FLENGTH. For example, if your report is based on a screen width of 80 characters, you specify LINELENGTH(80) to ensure correct format conversion.

In addition, the following SCS control orders are supported:

BS Backspace.
CR Carriage return.
FF Form feed.
LF Line feed.
NL New line.
TRN Transparent.

Any unsupported SCS orders are treated exactly as they are by printers that do not support them: if recognized as architected SCS orders, their default actions are taken as specified in *Systems Network Architecture: Sessions Between Logical Units*.

3270 to MCC conversion

The following buffer control orders are honored when carrying out format conversion:

SBA Set buffer address.
EUA Erase unprotected to address.
RTA Repeat to address.

Other 3270 orders are deleted from the data stream. Blanks are inserted where necessary to maintain correct format. The orders deleted from the data stream are:

SF Start field.
SFE Start field extended.
MF Modify field.
SA Set attribute.
IC Insert cursor.
PT Program tab.

The following format control orders are honored when carrying out format conversion:

CR Carriage return.
NL New line.
FF Form feed.
EM End message.
GRE Graphics escape (Defaulted to “_”).

Everything else is treated as data and included in the data part of the MCC report.

Conversion is carried out as if the EXEC CICS SPOOLWRITE REPORT command data stream were prefixed by a command byte and write control character (WCC) byte—you must **not** include these in the EXEC CICS SPOOLWRITE REPORT command. The command is EXEC CICS ERASE WRITE, and the WCC specifies both that the printout format is determined by NL and CR and that the buffer should be printed at the end of the write.

The 3270 data stream is converted to MCC in two stages, as shown in Figure 2.

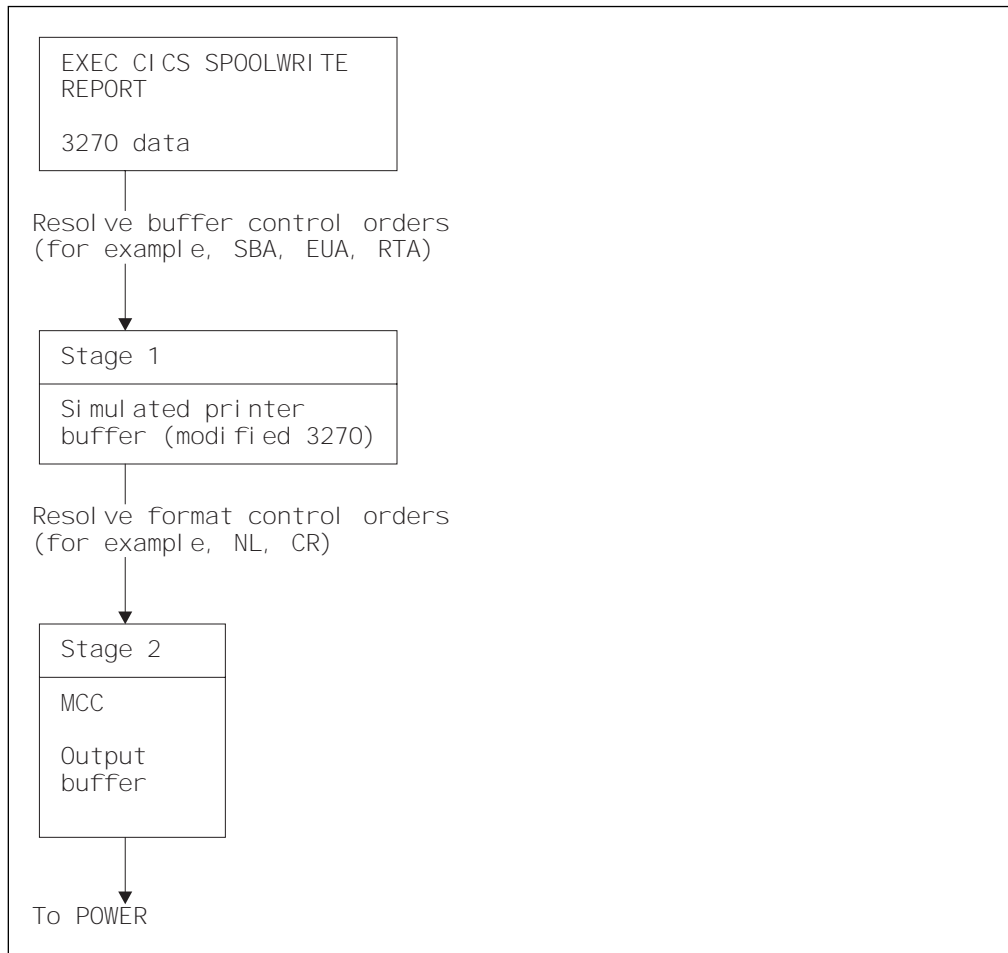


Figure 2. Conversion of 3270 data stream to MCC

The first stage emulates the data arriving at a printer buffer. Buffer control orders are resolved as the data is placed in the buffer.

The second stage creates an MCC report from this data. The format of this report is such that, if printed on a system printer, it would be as near as possible to the format of the original 3270 data stream as printed directly on a 3270 printer—that is, the format control orders are resolved.

Conversion starts from the start of the simulated print buffer, and continues until an EM order or the end of the buffer is reached. Note that, to prevent unwanted data being included in this conversion, you must include an EM order with **each** EXEC CICS SPOOLWRITE REPORT command.

Simulated buffer size

The simulated buffer size is 2KB. To accommodate large data streams this automatically expands to 4KB or 8KB. If an address translation (for example, SBA) is beyond the 8KB buffer, the task terminates with abend code APSG. If there are more than 8KB of modified 3270 data (that is, after stage 1 conversion), the additional data wraps to the start of the buffer.

Any unsupported 3270 orders are replaced by blanks where necessary, or are merely ignored. For example, start field (SF) occupies one character position in the print line, so it is replaced with one blank character. You can print reports in these

formats on any printer: a system printer, or a CICS printer, whether it is a 3270-type or an SCS printer. You can also use the LINES option to control automatic page breaks, just as you can for NOCC.

You cannot have automated headings and footings, page numbering, or the DATETIME option with SCS or T3270 formats.

The NOCONV option

For SCS and T3270 formats, you can specify the NOCONV option to prevent format conversion. If you do, these formats remain fixed for the lifetime of the report. Specifying NOCONV restricts a report created in T3270 format to printing on a CICS 3270 printer only, and a report created in SCS format to printing on a CICS SCS printer only. Use NOCONV if you want to use SCS or 3270 orders that are not supported by the conversion process.

If you specify a destination, it must be one that is ultimately a CICS printer, because NOCONV reports cannot be printed on a system printer. If you do not specify a destination, you should ensure that reports are created using a display device whose EXEC CICS SPOOLTO destination represents a CICS printer.

If you specify invalid control characters in a report that has the NOCONV option, the printer task abends.

You cannot use the LINES and LINELENGTH options when you use the NOCONV option.

BMS-mapped reports

If you have reports that include a lot of constant text, you can use BMS to format it. BMS-mapped reports are opened using the EXEC CICS SPOOLOPEN MAPNAME command. You could use BMS-mapped reports instead of preprinted forms. The data that is usually preprinted could be held as constants in the map.

You could use the MAPONLY option of the EXEC CICS SPOOLWRITE MAPNAME command to include a page of text that never varies, for example, a page of purchase conditions, or instructions.

There are two ways to use BMS:

1. By using BMS format. With this method, the report is formatted at print time. Only the variable data is held on the spool file, possibly saving space, but you can print the report on a CICS printer only.
2. By opening the report in T3270 format. This specifies that T3270 format is to be used to format the report. You then add the SET option to each SEND MAP command. This sets a pointer to an area containing the mapped data. You then insert an EXEC CICS SPOOLWRITE REPORT command, writing with a FROM value equal to the area pointed to by the SET option.

Method 2 allows you to print BMS-mapped reports on a CICS printer or on a system printer. It is the only way that BMS-mapped reports can be printed on a non-CICS printer. The T3270 format is described in “SCS and T3270 formats” on page 24.

BMS mapping format

You specify that the report is to be formatted using BMS maps, by opening the report with the MAPNAME option. You must also use the MAPNAME option on each EXEC CICS SPOOLWRITE command. You must name the map being used on the EXEC CICS SPOOLOPEN command, on the EXEC CICS SPOOLWRITE command, or on both. You can use more than one map in a report, but you cannot create a page from more than one map: one map is considered to be one page long. Thus, there is no ACCUM option, and you cannot specify headers, trailers, floating maps, and so on.

You can use all the options for controlling printing, and for security and recovery. However, the automated formatting features available for NOCC, ASA, and MCC formats are not available for BMS-mapped reports. This includes automatic page breaks, headings and footings, page numbering, and the DATETIME option. All your formatting must be specified within the map itself.

The length of the data is calculated from the length of the physical map. Remember that the map set and maps that you specify must be available to format the report at the time it is printed. When the report is being printed, however, you cannot disable a map until after the printer has been stopped.

ESCAPE format

To create reports in any format other than those described above, you must provide an **escape program** to be invoked at printing time to format the report. You can use this for graphics, or graphics and text on the same page, for instance. These reports cannot be printed on a system printer, but they can be printed on CICS printers.

You can specify the name of an escape program using the ESCAPE option on the EXEC CICS SPOOLOPEN REPORT command. You must ensure that this program is locatable at printing time. You can use several different escape programs for different reports.

An escape program must be defined with EXECKEY(CICS) because it runs under the control of transaction CEPW which is defined with TASKDATAKEY(CICS), otherwise you will receive an AEZD abend.

The data written by each EXEC CICS SPOOLWRITE command is considered to be one unit and is written as one record of a maximum record size of 32767 bytes. You must specify the length of the data as FLENGTH if you are using COBOL; if you are using assembler language, C for VSE, or PL/1 for VSE, the length is calculated from the length of the FROM area.

You can use all the options for controlling printing, and for security and recovery. However, the automated formatting features available for NOCC, ASA, and MCC formats are not available for ESCAPE format. This includes automatic page breaks, headings and footings, page numbering, and the DATETIME option. All your formatting must be specified within the escape program itself. You can also create ESCAPE reports in a VSE batch program by specifying an ESCAPE program name on the CICSDATA (or equivalent) keyword on the VSE POWER * \$\$ LST statement. For more information, see Chapter 5, "Managing reports and printers" on page 31.

When printing ESCAPE reports, CEMS or CEOS printer commands cannot be actioned until control is passed back from the user escape program. Control is passed back at end-of-copy and at end-of-report. This means that you cannot stop or pause a printer until the end of the copy or report is reached.

When the report is ready to be printed, it is written to temporary storage. At end-of-report, an EXEC CICS LINK is performed to the program named with the ESCAPE option. An 80-byte COMMAREA is passed to the escape routine and consists of the following:

On output link

The COMMAREA contains one or two fields. For **all** ESCAPE reports, a single temporary storage queue name is provided in the first 8 positions. The name is a concatenation of the hexadecimal characters X'FB00', the characters PS, and xxxx, where "xxxx" is the termid of the printer serving the destination. For ESCAPE reports which were created in batch, a second queue name is provided in the second 8 bytes of the COMMAREA. The first two characters of this queue are X'FB01'. The rest of the name is identical to the first queue. For each data record in the first queue there is a single byte record in the second queue, which is the MCC control character associated with the record on the spool. This queue can be used to determine how to format the data.

On return

A single byte return code followed by 79 bytes of user message text. If the return code is X'00', the escape report is set to its normal disposition (that is, it is held or deleted). If the return code is not X'00', message DFH5467 is sent to the CSPW transient data queue, containing the return code and any message text that you may insert in the trailing 79 bytes of the COMMAREA. The report on the POWER queue is then set to ERRPRT status.

When control is returned from the escape routine, the temporary storage queue is deleted.

JCL format

Using the EXEC CICS SPOOL... commands, you can create a file of input to a batch program. JCL format forces a LINELENGTH of 80, to create a file of 80-byte card images. These files are known as JCL reports, or jobs, and are written to the POWER RDR queue (not to the POWER LST queue). The options available for creating JCL reports are different from those available for creating printable reports. In fact, there are very few options.

A JCL report may be created as a standard report, or by many transactions as a log report. It may not be created as a resumable report.

A JCL report cannot be defined as physically recoverable: however, it can be defined as logically recoverable. And, because log reports cannot be logically recoverable, a log report in JCL format is not recoverable at all.

The command that opens a JCL report looks like this:

```
EXEC CICS SPOOLOPEN
          REPORT(REP1)
          TOKEN(TOKC)
          JCL
          [LOG]
          [LOGICAL]
```

There are no more options that you can specify on the EXEC CICS SPOOLOPEN REPORT JCL command. Some of the characteristics of JCL reports that CEMS users can use to control batch job submission can be specified on the JOB statement using the EXEC CICS SPOOLWRITE REPORT command. There are no options for formatting: a JCL report is written continuously and is always 80 characters wide. The resource security level value (RSL value) is set to 1.

Even the job status is derived from the JOB statement you write to the file, and not from the EXEC CICS SPOOLCLOSE REPORT command. The EXEC CICS SPOOLCLOSE REPORT command looks like this:

```
EXEC CICS SPOOLCLOSE
          REPORT(REP1)
          TOKEN(TOKC)
```

JOB statement parameters

The following parameters on the JOB statement written to a JCL report are of relevance to users of the report controller:

- CLASS
- PRIORITY
- DISP
- JOBNAME
- USER
- DEST or LDEST

The class, priority, and status of JCL reports can be changed by any authorized user of the CEMS transaction, if the user has the RSL value of 1 included in the RSLKEYs in their signon table entry. If the user is not signed on to CICS, the RSL value of 1 must be included in the OPERRSL keys defined for the display device.

If the JOB statement is missing, the report name is AUTONAME, the class is A, and the POWER defaults are used.

If a report submitted to POWER does not contain POWER JECL, the required JECL is inserted by POWER, and the characteristics of the JCL report default to those allocated by POWER. You must ensure that the syntax of the JCL is correct, because CICS does not check it.

When creating JCL reports, if a POWER EOJ statement is output, the POWER job is made available for running before the EXEC CICS SPOOLCLOSE is issued. This means that EXEC CICS SPOOLCLOSE REPORT DELETE only deletes the last POWER job and not the complete JCL report (job).

Chapter 5. Managing reports and printers

This chapter discusses the management of reports and printers in the following sections:

- “Managing reports”
- “Printing a report” on page 32
- “Browsing reports” on page 36
- “Shift-out/shift-in control data sequences” on page 36
- “Managing printers” on page 37
- “Sharing a report controller printer with another application” on page 38
- “Managing input for batch jobs” on page 39
- “Managing transient data queues” on page 41
- “Managing reports created by batch programs” on page 43
- “Audit trail” on page 48

This chapter describes how you use the report controller to control and print reports.

Managing reports

Using the CEMS and CEOS transactions, you can list reports and do various other things to them. You can list all reports, one report, or those reports that satisfy the selection criteria that you type on the selection panel. You can type a partial name as the report name, to list all those reports whose names begin with those characters.

The report list panel shows you the following information about the reports:

- Name
- Userdata (which may contain a short description of the report)
- Number
- Current status (for example, HELD(L), READY(K), IN USE, RESUME, ERRCRT, ERRPRT)
- Forms type
- Class
- Number of pages
- Number of copies
- Priority
- Destination

The report number is allocated by POWER when the report is created. You can use it to distinguish between reports of the same name. All the other characteristics are allocated by the program that created the report.

The report characteristics panels show you more information about a single report.

Printing a report

Printing of a report begins when the following conditions are met:

- It has the READY(D) or READY(K) status, and
- A printer is **started**, and set up for the same CLASS, DESTINATION, FORMS, and RSL as the report, and
- There is no higher-priority report for the same CLASS and DESTINATION.

All these characteristics can be altered by the CEMS or CEOS transaction. You can set the number of copies and whether the report has separator pages (SEP or NOSEP), and the end user can also alter these. What the end user cannot alter is the RSL value. The RSL value also determines whether a report can be printed by a particular printer.

Destinations

The report controller uses logical printer destinations. The **destination** associated with a printer is the means by which you can direct reports to a printer without naming the printer explicitly in your program. A destination can have a number of printers associated with it. The destination for a printer can be specified as the SPOOLDEST in the TERMINAL definition (or as SPLDEST in the DFHTCT macro). This destination can be overridden when the printer is started (either by using CEMS or CEOS, or by using the POWER PSTART command). This destination can also be changed temporarily by using CEMS or CEOS.

You can specify the destination at which a report is to be printed, in the SPOOLOPEN REPORT command. However, you may not want to code a specific destination for the report in your program, in which case you can let the destination vary according to the display device used to create the report. Each display device can have a default destination specified in the SPOOLTO attribute of its TERMINAL definition (or in the DFHTCT macro). If SPOOLTO is not specified, the destination defaults to **SYSPRT, a special destination that indicates that a report is destined for the system printer.

You can specify **SYSPRT as a destination on the SPOOLOPEN REPORT command to direct the report explicitly to the system printer.

Note: A report controller destination is the same as a POWER USERID, but it is not the same as a CICS USERID. Note also that a report controller destination is not the same as a transient data destination.

In VSE/POWER terms, a destination is a user identifier (USERID). VSE/POWER uses user identifiers for a number of different purposes (see the *VSE/POWER Administration and Operations Guide*).

You must always ensure that the names you give to your CICS destinations are not reserved by POWER and are not being used by POWER for some other purpose.

Examples of destinations are as follows:

Display devices	DD01	SPOOLTO(DD01DEST)
	DD02	SPOOLTO(unspecified)
Printers defined to CICS	PR01	SPOOLDEST(DD01DEST)
	PR02	SPOOLDEST(DD01DEST)
	PR03	SPOOLDEST(XXXXDEST)

Reports created at DD01, with no DESTINATION specified on the SPOOLOPEN REPORT command, are directed to PR01 or PR02.

Reports created at DD02 with no DESTINATION specified are directed to a system printer.

Reports created at DD01 or DD02 with a destination of XXXXDEST specified are directed to PR03.

Reports created at DD01 or DD02 with a destination of **SYSPRT specified are directed to a system printer.

Routing reports to remote destinations

You can route a report to a destination associated with a printer at a different node in the network. You cannot do this with resumable reports, nor with JCL reports. Specifying the NODE option without DESTINATION means that the report is to be routed to a system printer. The SPOOLTO operand for the display device is ignored.

Report status

When you close a report for the last time, you specify a POWER disposition for the report with DISP(name), HOLD, KEEP, or RELEASE (see “Closing a report” on page 16).

This results in a change in the status of the report on the POWER LST queue.

A report closed with DISP('L') or HOLD appears to report controller end users as HELD(L) when displayed on the report controller panels. The end user has to reset this status to READY(K) before the report can be printed. When the report has been printed, the status is reset to HELD(L) again. This means that the report can be printed again, possibly by mistake. You may therefore prefer to use the DISP('H') option for some reports. On the other hand, DISP('L') and HOLD do have the advantage that a report lost because of a printing failure can be reprinted without being created again. This gives end users the most complete control over their reports, if that is what they want. They always have to intervene before a report is printed, and they always have to delete their reports from the queue when they no longer need them.

A report closed with DISP('H') appears to the end user as HELD(H) when displayed on the report controller panels. The end user has to reset this status to READY(D) before the report can be printed. When the report has been printed, it is deleted automatically.

A report closed with DISP('D') or RELEASE appears to the end user as READY(D) when displayed on the report controller panels. It is printed as soon as a printer is available for the appropriate destination and output class. After being printed, the report is deleted from the POWER LST queue and is not available for reprinting. If

you use this disposition, you should ensure that you can, if necessary, create again a report lost because of a failure when it was being printed. On the other hand, DISP('D') and RELEASE have the advantage that reports are printed and deleted from the queue with no operator intervention, beyond ensuring that a printer is started for the correct destination, class and security level.

A report closed with DISP('K') or KEEP appears to the end user as READY(K) when displayed on the report controller panels. It is printed as soon as a printer is available for the appropriate destination and output class. After being printed, the report is retained on the POWER LST queue in HELD(L) status, and is available for reprinting. This provides a means of reprinting a report lost because of a failure during printing.

If you decide to have all reports created so that they are ready to print, but still want to be able to control **when** they are printed, you have several options. You can use priorities as appropriate or, if this is not sufficient control, you can use a special destination name as a holding destination, or a special output class as a holding class.

Output class

Each report is assigned to one of 36 output classes. Each printer is also assigned to one of the 36 output classes. A report can only be printed by a printer assigned to the same output class. The default output class for reports is specified in the SPOOL system initialization parameter.

There is no restriction on the number of different destinations, forms types, and RSL values you can have for the same output class. However, you may find it helpful to use output classes to keep reports with different forms types separate. This is particularly helpful if most reports are closed with the DISP('D'), DISP('K'), or RELEASE option (ready to print), or if the user releases many reports all at once.

A printer prints reports according to priority for the destination, output class, and RSL values it is assigned to. If the ready-to-print reports all have equal priority, all those of one forms type are printed before going on to the next forms type. However, where different priorities are used, extra work for the end user who has to change the paper would result, unless the forms types are kept in separate output classes. The output class can be changed by using the CEMS or CEOS transaction.

Priority

Each report is assigned a priority value 1 through 9. For a particular destination and class, reports with priority 9 are printed first, followed by priority 8, and so on, down to 1. The default priority value is 3. The priority can be changed by using the CEMS or CEOS transaction.

SYSID

In a shared spooling environment, a report can have a VSE/POWER SYSID associated with it. This can be useful when a particular VSE/ESA system processes a particular file. PWRSYSID can be specified on the SPOOLOPEN command when creating a report. CEMS or CEOS support selection by SYSID, and allow it to be changed. Note, however, that the report controller print component does not use PWRSYSID as a selection criteria.

Forms type

The forms type is a 4-character name for a type of stationery. You may, for example, have different types of preprinted stationery, or different sizes of paper. It can be useful to separate different forms types by assigning them to different output classes; see “Output class” on page 34.

Before a report can be printed, the forms type specified for the report must match the forms type specified for the printer. The forms type for the printer can be altered by using the CEMS or CEOS transaction. However, no check is made on the physical size of the paper that is actually loaded. The report is printed when the names of the forms types match.

Number of copies

You can specify how many copies of a report are to be printed. The default number is 1, and can be changed by using the CEMS or CEOS transaction.

Separator pages

A separator page is a cover page to be printed with a report. It makes it easier to see where one report ends and the next begins, and also gives some information about the report. You can specify whether or not separator pages are to be printed with a report. If you specify SEP, two separator pages are printed preceding the body of the report, and two following it. If you specify NOSEP, no separator pages are printed.

The size and content of the separator pages are determined by the DESTINATION name of the printer on which the report is printed. If the DESTINATION is **SYSPRT, the separator pages are the standard POWER separator pages. If the DESTINATION is anything else, the separator pages are written by the report controller itself.

The width of the report controller separator pages depends on the width (LINELENGTH) of the report that they accompany (which, if the correct forms type is loaded, is less than or equal to the width of the paper).

Each report controller separator page may show the following information:

- The contents of the USERDATA field
- The CICS USERID of the user who created the report (AUTHOR)
- The date and time the report was printed
- The DESTINATION the report was printed at (DEST)
- The TERMINAL name (TRMIDNT) of the printer (PRT)
- The output CLASS (CL)
- The FORMS type (FNO)
- The FCB (forms control buffer) code
- The network NODE at which the report was printed
- The network NODE at which the report was created (O-NODE)
- The name of the CICS system that created the report (O-USR: always SYSCICS followed by the identifier specified in the SPOOL system initialization parameter)

- The report number (O-NBR).

Users of the CEMS or CEOS transactions can decide whether to have separator pages or not, and can override what you code. For more information about separator pages, see the *CICS Report Controller User's Guide*.

Printing multiformat reports

The report controller supports the printing of multiformat reports. Such reports consist of a number of data sets, each of which contains one type of format control character. ASA, MCC, and NOCC control characters are supported.

Multiformat reports cannot be generated under VSE/ESA, but can be imported from MVS systems.

When multiformat reports are printed on CICS printers, the same results are obtained as on system printers.

Browsing reports

You can view online certain types of report that you have created in batch or using the report controller.

The browse option is selected from the report list panel invoked by the CEMS and CEOS transactions. The disposition of the spool file to be browsed can be A, D, K, L, or H. The panel displays the name of the report you are browsing and the number of pages. Using PF keys, you can move backward and forward, page by page or within a single page. You can also go directly to a certain page in the report. If a report page is wider than the screen display area, you can use PF keys to scroll right and left. You can also search for strings within a report.

Reports of the following formats can be browsed online:

- MCC
- ASA
- NOCC
- T3270 or SCS reports that are convertible (that is, stored as MCC format)
- Multiformat reports
- Reports containing DBCS data

NOCONV and ESCAPE reports are **not** supported. Browsing of JCL reports is **not** supported. You can browse resumable reports. For information about BROWSE security, see "Report controller BROWSE security" on page 59, and for more information about browsing reports, see the *CICS Report Controller User's Guide*.

Shift-out/shift-in control data sequences

Shift-out/shift-in control data sequences cause hardware to shift between the EBCDIC single byte character set (SBCS) and double-byte character set (DBCS) for displaying and printing data. The DBCS is used, for example, in displaying the Kanji character set.

When shift-out/shift-in control data sequences appear in an SCS data stream, they do not occupy a display or print position, but in a 3270 data stream they do (a blank is inserted). This means that application programmers would need to know

the characteristics of the display or printer device when the report is being created. This is especially pertinent for example, where multiple destinations are involved.

To solve this problem, the shift-out/shift-in attribute can be indicated through two mutually exclusive options on the EXEC CICS SPOOLOPEN REPORT command:

1. SOSI1 indicates the application programmer's intent that the shift-out/shift-in sequences are to take up a print position.
2. SOSI2 indicates that the shift-out/shift-in sequences are **not** to take up a print position.

There is no default. The attribute is stored with the report and can be transmitted with the report within an Network Job Entry (NJE) job network.

The attribute is displayed as SOSI1 or SOSI2 on the report characteristics panels invoked through the CEMS and CEOS transactions. It is the responsibility of whoever displays or prints the report to ensure that the application programmer's intent is honored.

Managing printers

You can use the report controller for the following tasks:

- Listing the printers and getting information about their current usage. You see only the printers that you are authorized to control. You can select the ones you want to see on the list, by using a partial name on the Printer Selection panel.
- Changing the usage of a printer.

The characteristics you can change are:

- Destination.
 - Output class.
 - Forms type.
 - The number of pages that are to be used for alignment.
 - Whether printer messages are to be printed, as well as displayed on the printer list panel. (You might not want printer messages on your preprinted forms.) See "Printer status messages" on page 38.
 - Default forms control buffer.
 - The actions to be taken when a report does not start with a skip to channel 1.
- Starting report printing.
 - Pausing and resuming or restarting report printing.
 - Aligning forms on the printer.

The report controller stops printing when a report needs a different forms type, so you know when the paper needs changing. The forms type for a report can be specified when the report is created. The forms type for the printer can be specified when the printer is started. The printer prints only those reports with a matching forms type. This reduces the danger of printing a report on the wrong paper.

When you have changed the paper, you may want to align the forms, if you are using preprinted stationery. The report controller provides for this: you can specify the number of pages you want to use for lining up. The report controller prints the first part of the report text (replacing all letters with Xs and all numbers with 9s), so that you can see whether the paper is correctly aligned. See the *CICS Report Controller User's Guide* for more information.

- Stopping report printing.

Printer status messages

Printer status messages are displayed on the CEMS and CEOS printer list panels, and are optionally sent to report controller printers.

The printing of status messages is specified for individual printers using the PRTDMSG operand of the DFHTCT TYPE=TERMINAL or the PRINTEDMSG option of the TERMINAL resource definitions. The default value for both of these is NO.

The printing of status messages for the current CICS session can be enabled and disabled on the CEMS and CEOS printer characteristics panel.

Suppression of printer status messages is useful if you print on special forms, for example, receipts or invoices. It is essential that such forms are not misaligned because of status messages sent to the printer.

Sharing a report controller printer with another application

Normally, when a printer is started for the report controller, the printer remains attached to the print task (CEPW) until the operator stops the printer. In some environments, this could represent an inefficient use of the printer resource. Therefore, a printer sharing facility is provided to allow the printer to be shared with other CICS and VTAM applications.

Printer sharing is enabled by specifying a time-out period for the printer. If the printer is in WAITING status for longer than the specified period, it is released (RELEASED status) and becomes available for other applications. If more work becomes available for the printer after it has been released, it is restarted automatically as soon as it is free.

The time-out period is specified as the number of minutes (1 through 59) the printer can remain idle (that is, performing no report controller printing) before it is freed for use by other CICS or VTAM applications. The time-out is specified by defining a value for SPOOLPRTTO in the RDO TERMINAL definition. For more information see the *CICS Resource Definition Guide*.

If a value of zero is specified (the default), this means that the printer is not capable of being shared with other applications.

Note: Although the printer device is freed after the idle period has expired, the print task (CEPW) remains active in order to monitor the arrival of more work for the printer. When work arrives, the appropriate actions to reacquire and start printing again are taken.

For printer sharing to work, you must code RELREQ=YES and DISCREQ=YES in the RDO TYPETERM definition. This is necessary because an EXEC CICS ISSUE DISCONNECT command is issued when the printer is released.

Printer states

The following printer states are relevant to the printer sharing facility, and are displayed in the CEMS and CEOS Printer List panels.

RELEASED	The printer has remained idle for longer than the specified time-out period and has been released for other CICS or VTAM applications to use, but none is doing so at this time. If more work arrives for the task, printing resumes. The printer can be stopped at this point if required.
RELEASED (IN USE)	The printer has been released and another application has acquired the printer and is currently using it. If more work arrives for the print task, printing of reports resumes when the printer becomes free. The printer can be stopped at this point if required.
STARTING	More work has arrived for a printer that has been in the RELEASED state. CICS is in the process of reacquiring the printer and recommencing printing. No operator actions can be performed against a printer in this state.
STARTING (IN USE)	As with STARTING, this indicates that CICS is in the process of reacquiring the printer and recommencing printing, but another application has control of the printer. When the printer becomes free, report controller printing resumes. No operator actions can be performed against a printer in this state. Note: Typically, this state occurs only if the printer is in use by another VTAM application (which could be another CICS system) rather than another CICS application in the same CICS address space as the report controller The STARTING (IN USE) state is shown when the printer is started initially if, at that time, it is in use by another application.

Managing input for batch jobs

Input for batch jobs can be created using the EXEC CICS SPOOL... commands, by creating a report in JCL format. These **JCL reports**, like input for batch jobs that comes from other sources, are written to the POWER RDR queue, not to the POWER LST queue. Most of the characteristics of JCL reports are taken from the JCL itself, including whether the job is ready to run or held. All jobs waiting on the POWER RDR queue are accessible to users authorized to use CEMS (not CEOS), who have the value 1 included in their RSLKEYS.

The tasks involved in managing batch jobs are:

- Listing the waiting jobs and some information about them
- Deleting a job from the queue
- Changing the class or the priority of a job (or of many jobs)
- Releasing a job for running now

- Holding a job for running later

Figure 3 shows the components involved in creating and controlling input for batch jobs (JCL reports).

EXEC CICS SPOOL... commands and batch programs can be used to create input for batch jobs (JCL reports). The figure also shows that JCL reports are written to the POWER RDR queue and are managed by the report controller.

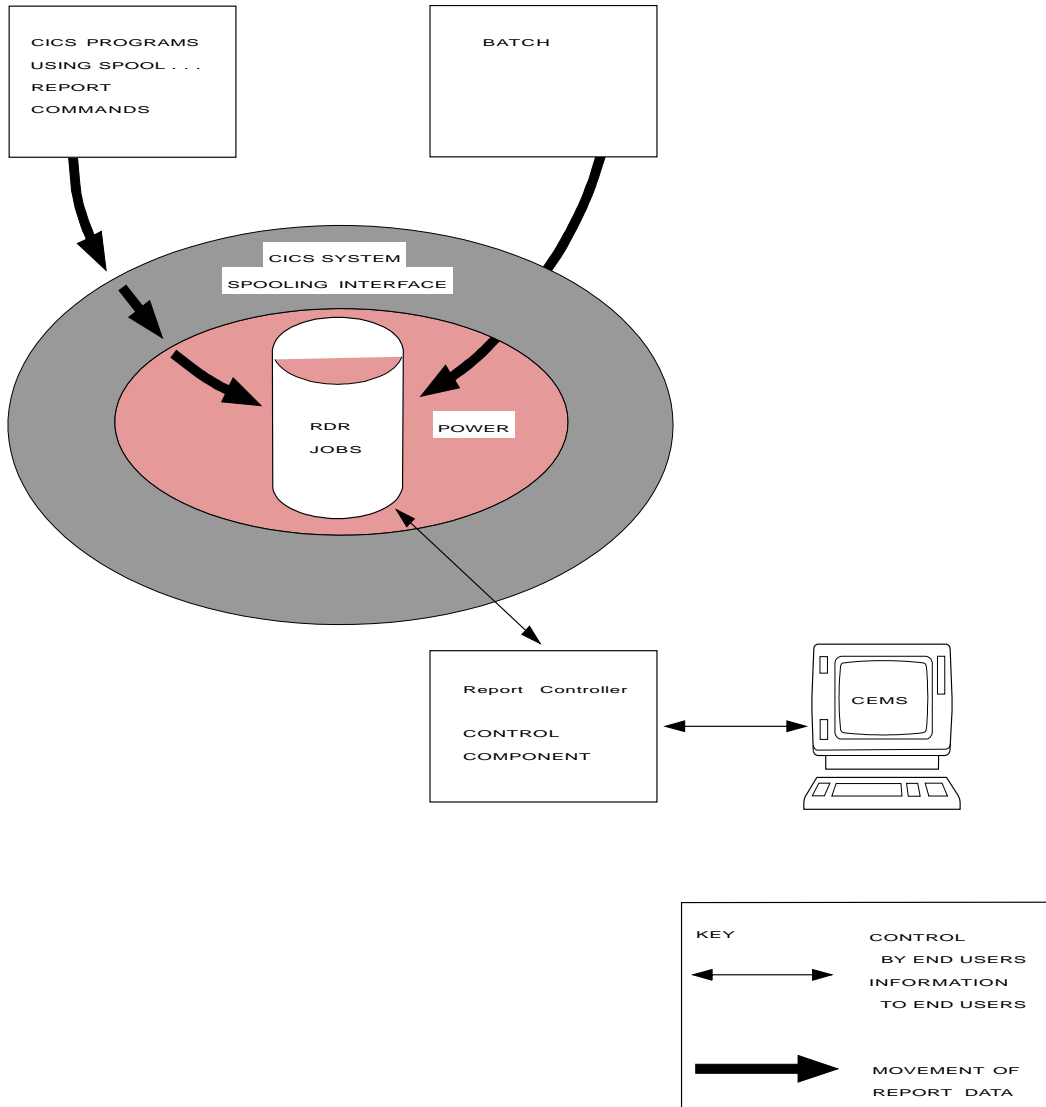


Figure 3. Report controller: jobs

Managing transient data queues

The report controller allows you to create a report from the messages, or records, written by CICS programs to a transient data queue. Reports can be created from intrapartition transient data queues, but not from extrapartition transient data queues. When you have created a report from a transient data queue, you can manage that report in the same way as other reports, using the report controller or using POWER commands.

The tasks involved in managing transient data queues are:

- Listing the waiting transient data queues and some information about them.
- Creating a report from a transient data queue. You can specify the following for the report:
 - Status
 - Class
 - Priority
 - Number of copies
 - Destination
 - Title
 - A field called USERDATA, which you can use as a description of the report to be created, or to hold some other information
 - Resource security level (RSL)
 - Whether there is to be a separator page or not
- Creating a report ready to print.
- Creating a report held for later printing.

Figure 4 on page 42 shows the components involved in creating a report from a transient data queue, and the use of the report controller to control the printing of the report. The report is created by a CEMS transaction, as a report on the POWER LST queue. It can then be controlled in the same way as any other report on the POWER LST queue.

Figure 1 on page 3 shows that a report created from a transient data queue can be sent to a system printer. The figure also shows the components involved in creating a report from a transient data queue, and the role of the report controller in controlling the printing of the report.

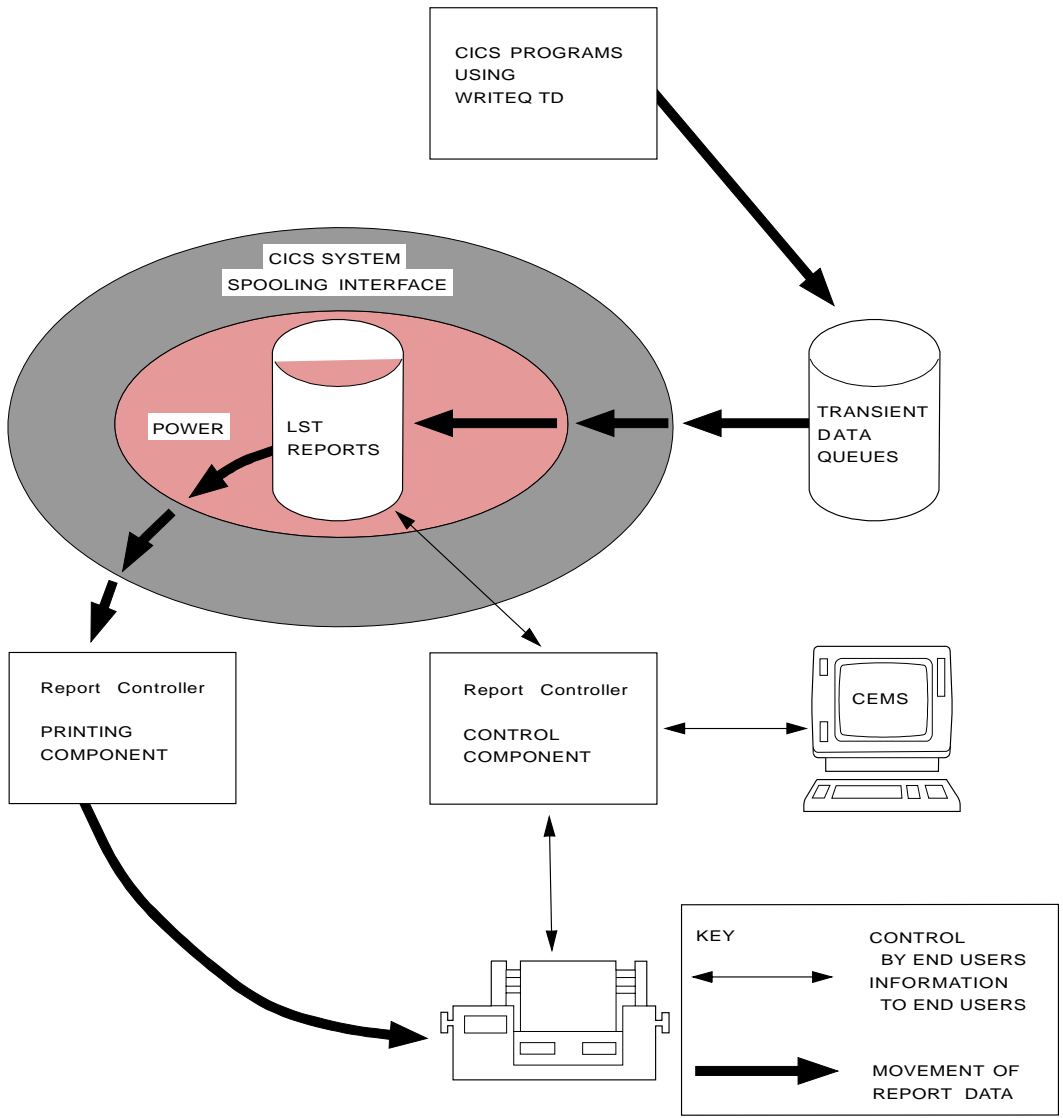


Figure 4. Report controller: transient data queues

Managing reports created by batch programs

You can use the report controller to manage the printing of reports created by batch programs. As Figure 3 on page 40 shows, reports created by batch programs do not use the CICS system spooling interface with POWER. They can, however be printed either on a system printer or on a CICS printer. This enables distributed printing of non-CICS reports at different locations. From the end user's view, managing reports not created by CICS is much the same as managing CICS-created reports.

From the report-creation view, you can use your existing application programs, but you may have to modify the JCL slightly to enable CEMS or CEOS users to control printing. Table 2 on page 44 shows the report characteristics that correspond to keywords, which come from the VSE/POWER JECL, or from * \$\$ JOB and \$\$ LST statements (from POWER JECL or VSE SEGMENT segmentation macros). The START and END pages of a report are always the first and the last; you cannot change them using CEMS or CEOS. No other report characteristics exist for reports created by batch programs.

Options for reports created by batch programs

<i>Table 2. Options for reports created by batch programs</i>			
Report characteristic	EXEC CICS Option	CEMS/CEOS List Panel	POWER Keyword
Report name	REPORT	NAME	JOBNAME
Report number (see note 1)	–	NUMBER	–
Description	USERDATA	USERDATA	USER
Stationery type	FORMS	FORMS	FNO
Output class	CLASS	CLS	CLASS
Number of copies	COPIES	CPY	CPY
Priority	PRIORITY	PRI	PRI
Printing destination	DESTINATION	DEST	DEST=(,USERID)
RSL value (see note 2)	RSL	–	(see note 2)
Escape program name	ESC	–	see “Naming an escape program in a batch job” on page 46
Separator pages (see note 3)	SEP NOSEP	–	SEP
Forms control buffer (see note 4)	FCB	–	FCB
STATUS (see note 5)			
Waiting	DISP('L') or HOLD	HELD(L)	DISP=L
Waiting	DISP('H')	HELD(H)	DISP=H
Ready to print	DISP('D') or RELEASE	READY(D)	DISP=D
Ready to print	DISP('K') or KEEP	READY(K)	DISP=K
Unfinished	RESUME	RESUME	DISP=A
Being printed	–	IN USE	DISP=*
Error during creation	–	ERRCRT	DISP=X
Error during printing	–	ERRPRT	DISP=Y

Notes for Table 2:

1. The report number, for reports created by batch programs and those created by CICS, is allocated by POWER.
2. The default RSL value is 1 for a report not created by CICS. However, you may set an RSL value for each report you create, in the range 1 through 24, or PUBLIC. To do this you must include a DEFINE statement in the POWER AUTOSTART deck, as follows:


```
// EXEC POWER
:
DEFINE L,keyword,3F00,2,255,*
```

The *keyword* can be any 8 characters. You are recommended to use CICSDATA. For example:

```
// EXEC POWER
:
DEFINE L,CICSDATA,3F00,2,255,*
```

You refer to the keyword on the POWER LST statement, as follows:

```
* $$ LST CLASS=a,keyword=(rslvalue),...
```

For example:

```
* $$ LST CLASS=C,CICSDATA=(RSL=5),...
```

You can also use the CICSDATA keyword to name an escape program.

If you use the VSE SEGMENT segmentation macro, you can change the RSL for each segment of a LST report. In this way you can use segmentation to split a report for multiple destinations (by using multiple USERID values), with a different RSL value for each destination.

For more information about CICS resource security level (RSL) checking, see Chapter 7, "Report controller security" on page 57.

3. Using CEMS or CEOS, you cannot change SEP to NOSEP, or NOSEP to SEP, for reports not created by CICS.
4. For more information, see "Forms control buffers (FCBs) and printing" on page 47
5. When a DISP=D report has been printed using the report controller, it is deleted. When a DISP=K report has been printed, it is held on the POWER LST queue as DISP=L.

If you use CEMS and CEOS to hold DISP=D or DISP=K reports before printing, then DISP=D reports are set to H, and DISP=K reports are set to L.

If you use CEMS and CEOS to PRINT DISP=H or DISP=L reports, they are set to READY: DISP=H reports are set to D, and DISP=L reports are set to K.

DISP=4, DISP=X, and DISP=Y are set internally. They cannot be specified on POWER JECL.

Naming an escape program in a batch job

With the ESCAPE option of the EXEC CICS SPOOLOPEN REPORT command you can specify the name of an escape program to be invoked at **printing** time to handle data streams that the report controller does not support. You can also specify an escape program for reports not generated by CICS, using POWER JECL.

When a report is created with the report controller, CICS-specific attributes of that report are recorded in an extension to the spool file header called the output parameter text block (OPTB). (This is maintained by VSE/POWER, according to Network Job Entry protocols.) The OPTB is made up of a number of text units (TUs). CICS has a TU called CICS DATA (TU key X'3F00'), in which the CICS-specific attributes are saved. VSE/POWER provides services to build and retrieve TUs through the Spool Access Services (SAS) interface.

In addition, POWER allows you, using the autostart DEFINE statement, to specify your own keywords to be used on JECL statements in a batch job. You can use such a keyword to build the CICS DATA TU with the data specified on the keyword. The data can contain the name of an escape program to be invoked when the report is ready for printing. To do this you must include a DEFINE statement in the POWER AUTOSTART deck, as follows:

```
// EXEC POWER
:
DEFINE L,keyword,3F00,2,255,*
```

The *keyword* can be any eight characters. However, you are recommended to use CICS DATA. (Note that X'3F' is the keyword ID for CICS DATA.) For example:

```
// EXEC POWER
:
DEFINE L,CICSDATA,3F00,2,255,*
```

You refer to the keyword on the POWER LST statement in the batch job, as follows:

```
* $$ LST CLASS=a,keyword=(cicslist)
```

Where *cicslist* contains strings that are passed directly to CICS. CICS recognizes two strings: RSL=*nn* for the resource level, and ESCAPE=*progrname* for the name of the ESCAPE program (*progrname* can be up to 8 characters in length). For example:

```
* $$ LST CLASS=C,CICSDATA=(ESCAPE=BATREP,RSL=5)
```

When the batch-generated report is directed to the report controller print component, the escape program name is retrieved and the program invoked as if it had been specified on the EXEC CICS SPOOL OPEN REPORT command. For more information about the DEFINE statement see the *VSE/POWER Administration and Operation* manual.

Forms control buffers (FCBs) and printing

A forms control buffer is a buffer for controlling the vertical format of printed output. By default, when printing using the report controller, a pseudo-FCB is used. When using this FCB, the FCB field on the report separator page is left blank. This FCB defines channel 1 as the first (top) line that can be printed on a page. All other channels (2 through 12) are undefined.

If a line in a report requests a skip to an undefined channel, a blank line followed by that line is printed.

Note: When you make any paper realignments using printer hardware facilities such as form feed, you must do one of the following:

- Stop the printer first using CEMS or CEOS.
- Issue an alignment request following the paper adjustment, to ensure that alignment of FCB-controlled reports is maintained by the report controller.

User-defined default installation FCB

You can specify an alternative to the default pseudo-FCB by cataloging an FCB phase to a library available to CICS at startup. The name of this phase must be DFHPSFCB and it should be structured as described in the *VSE/ESA System Control Statements* manual. If CICS finds this phase it will be used as the default instead of the pseudo-FCB.

Printer specific FCBs

Each printer can have a specific FCB associated with it which overrides any other default. The FCB phase name is specified on the RDO terminal definition. This specification can be changed while CICS is running from the report controller CEOS and CEMS control transactions.

FCBs for reports created using CICS

You can specify an FCB with the FCB option of the EXEC CICS SPOOL OPEN REPORT command.

If an FCB is specified in this way, the name is checked for the presence of \$\$\$\$ as the first four characters, indicating a generic FCB (for example, \$\$\$RC01). If the name is a generic name, the \$\$\$\$ characters are converted to the default, FCB3.

FCBs for reports created by batch programs

Just as you can specify an FCB on reports created using CICS, a batch-generated report on the POWER LST queue may have an FCB name associated with it, if FCB is specified on the POWER * \$\$ LST statement. The report controller provides support for such reports in the following manner.

At report print time, the report controller tries to load the named FCB. If the load is successful, it is used during printing (as described), and its name appears on the separator pages. If a named FCB is not loaded, or the FCB name is allowed to

default, the customer-defined default FCB, DFHPSFCB, is used if it is available. The CICS pseudo-FCB is used if no other image is successfully loaded.

Note that, at the time the output is created in batch, POWER tries to load the FCB to simulate channel 9 and 12 posting. If the FCB is not found, POWER nullifies the name and it is not passed to the report controller at print time. It is, therefore, essential that FCB phases reside in a sublibrary that is accessible to both POWER and CICS.

FCB hierarchy

Table 3 shows the sequence in which the FCB is used.

<i>Table 3. FCB hierarchy</i>			
FCB on report (POWER JECL or SPOOLOPEN)	FCB on printer definition	Installation default (DFHPSFCB)	FCB which will be used
Yes	Yes	Yes	Report definition
Yes	Yes	No	Report definition
Yes	No	Yes	Report definition
Yes	No	No	Report definition
No	Yes	Yes	Printer definition
No	Yes	No	Printer definition
No	No	Yes	DFHPSFCB
No	No	No	Pseudo-FCB

Device-independent FCBs

If the FCB name on the POWER * \$\$ LST statement is in the form \$\$\$\$xxxx (where xxxx is a string of characters of your choosing), the \$\$\$\$ will be replaced by a device-dependent string. When print output is being created in batch, POWER replaces the \$\$\$\$ with a string representing the printer device that the program is assigned to by the VSE/ESA // ASSGN command at that time. When printing the data, the replacement reflects the physical output device to which POWER is printing. When a report with a device-independent FCB is passed to the report controller for printing, \$\$\$\$ is always replaced with FCB3.

Multi-segment reports

A report on the POWER LST queue might contain segments. These segments appear as separate reports on the report selection panel. Segments are viewed and managed as separate reports by the report controller feature.

Audit trail

Changes to the characteristics or status of reports and printers, and the reports printed on each printer, are recorded in transient data queue CSPA.

When the characteristics or status of a report are changed, the following information is recorded:

- Report number
- Report name

- User ID of the operator
- Date and time
- Before-and after-images of the report characteristics and status

When the characteristics or status of a printer are changed, the following information is recorded:

- Printer name
- User ID of the operator
- Date and time
- Before-and after-images of the printer characteristics and status

For each report printed, the following information is recorded:

- Date and time the printing of the report started
- Date and time the printing of the report ended

Action messages for the print task CEPW and severe error messages are written to transient data queue CSPW. The following events are recorded:

- Request for change of forms
- Completed change of forms
- Printer interruption
- Abend when a report in escape format is being printed

You can use CEMS to create a printable report out of either of the transient data queues. For instance, the system administrator or supervisor can create a report from CSPW each day. Following a printing failure, they could create and print a report from CSPW, to find out which reports might need reprinting.

To use the audit trail facility you must include entries for CSPW and CSPW in the destination control table (DCT).

The report controller error messages for English, Kanji, German, and Chinese are contained in the load modules DFHPSEE, DFHPSEK, DFHPSEG, and DFHPSEC defined to CICS as programs. This and changes to the report controller mapsets are supplied in the CICS upgrade resource definition list DFHCU23F.

Chapter 6. The report controller and POWER

This chapter contains information about the report controller and POWER in the following sections:

- “Operating the report controller and POWER” on page 52
- “Destinations and printers” on page 52
- “Starting a printer” on page 53
- “Report controller report printing delay” on page 55

If printing in your organization is distributed among locations, users at a remote location can control printing only by using the report controller. Operations staff at the central computer location may use the report controller, or they may choose to continue to use POWER commands to control printing. The computer operations staff also become involved in tasks related to the “end use” of the report controller, such as:

- Ensuring that the POWER LST queue does not fill up
- Recovering a report after a system failure
- Rerouting reports (both locally and to other nodes)
- Resolving messages sent to the system console
- Resolving end-user problems
- Coordinating security

Also, the report controller panels provide many of the functions currently invoked by operations staff using POWER commands at the console. Operations staff can be authorized to use the report controller for these functions to give centralized control over printing devices owned by CICS. In this way, operations workload can be reduced. The report controller functions that can be used instead of equivalent POWER commands are summarized in Table 4 on page 52.

<i>Table 4. Report controller functions and POWER commands</i>	
Function provided by report controller	Equivalent POWER command
Delete a report or job from a queue.	PDELETE
Change the class or priority of a report or job.	PALTER
List printers owned by CICS. (The list shows the status and other information about the printer.)	PINQUIRE
List reports or jobs for POWER. (The list shows the status and other information about each report or job.)	PDISPLAY
Stop the printing of a report immediately.	PFLUSH
Reactivate a device that is waiting for operator action.	PGO
Change the status of a report from dispatchable to held.	PHOLD
Restart an interrupted printing session. (See note.)	PRESTART
Align forms.	PSETUP
Connect a terminal printer to POWER printing. (This allows the relationship between destinations and printers to be altered.)	PSTART
Disconnect a terminal printer from POWER printing.	PSTOP
Note: With CEMS and CEOS transactions, the restart page cannot be set to a value greater than 65535. This restriction does not apply to the equivalent POWER command.	

Operating the report controller and POWER

There are several different ways of operating a combination of the report controller and POWER commands, for example:

- Using the report controller to control printers and to release held reports, and delete them after printing
- Using the report controller to control printers, but creating reports ready to print, so that no intervention is required
- Controlling CICS printers using POWER, but using the report controller to manage reports. There are two ways of starting a printer:
 - Entering the PSTART command at a console
 - Including a PSTART command and a PRINTERS command in a POWER autostart procedure.

For more information about POWER commands, see the *VSE/POWER Administration and Operations Guide*.

Destinations and printers

In VSE/POWER terms, a destination is a user identifier (user ID). POWER uses user IDs for a number of different purposes (see the *VSE/POWER Administration and Operations Guide* for more details). You must always ensure that the names you give to your CICS destinations are not reserved by POWER, and are not being used by POWER for some other purpose.

Starting a printer

Your end users can use CEMS or CEOS to start a printer for a particular destination. Also, an operator can use the PSTART command at a central computer location, to start a printer. To specify the destination to be associated with the printer, the operator must specify:

```
PSTART DEV,name,SYSCICSa,b,PARM='destname'
```

Where:

- *destname* is the name of the destination
- *name* is the TERMINAL name or TRMIDNT of the printer
- SYSCICS identifies the CICS system to POWER
- *a* is the specification in the SPOOL system initialization parameter
- *b* is the class of reports that the printer is to print

By using the ddsname of the CICS system (SYSCICS*a*), and by using PARM='*destname*' to name the destination, the operator is telling POWER that the CICS report controller controls printing.

If PSTART is used to start a printer, you cannot specify the forms type to be loaded on the printer. Users of the CEMS or CEOS transactions can print only those reports for which no particular forms type has been specified.

If your CICS system is running as a guest system under VM, do not use the same output class identifiers for reports destined for CICS printers and for reports destined for VM. If you do, then when you start output for VM, any reports for the same class, created by CICS programs and in READY status, are routed to the VM output class. If VM can find a destination of the same name as the report destination, it sends the report there. If VM cannot find a destination of the same name, it puts the report back on the POWER LST file, in HELD status, and writes a message to the console.

System printers

You can use the report controller to control fast, locally attached system printers. You must define the printer to CICS, using RDO. There is a special destination name ****SYSPRT**, which you must use. This identifies the printer as a system printer, and allocates a larger buffer (2KB) to be used to get data from POWER through the XPCC calls, giving better performance for fast printers.

You can use the destination name ****SYSPRT** for CICS terminal printers and POWER system printers. You still get the 2KB buffer. This is intended for fast system printers attached to CICS.

****SYSPRT** is equivalent to a printer with the POWER USERID=LOCAL. If you start a printer with a destination (that is, a user ID) called LOCAL, this is translated by the report controller, for the benefit of users of the CEMS or CEOS transactions, into ****SYSPRT**. This is because, to the end user at a remote location, the system printer may be anything but local.

If you start a POWER system printer with PSTART, without using PARM=destname, it ignores all reports created by CICS programs, unless they have a blank destination specified, or a destination of **SYSPRT.

The first report printed on a non-SCS printer that has been started to the report controller is preceded by a formfeed (blank page) when the first control character of the report is a formfeed. This is a precautionary measure against overtyping in particular circumstances.

You can suppress the production of a formfeed by changing the trailing YES|NO option on the SPOOL system initialization parameter. Coding NO means that a formfeed does not precede the first report printed on that printer.

However, you should be aware that coding NO (and therefore suppressing the blank page) means that there may be a risk of the last line of the previous output being overtyped when the printer is started to the report controller.

Printers with the early print complete (EPC) feature

Printers with the early print complete (EPC) feature (for example, the IBM 4245), offer higher efficiency because host processing is overlapped with physical I/O.

EPC works by signaling end of print before the data is actually committed to paper. However, this can introduce an integrity exposure because errors may be indicated after the data (which may have failed to print) has been discarded by the report controller.

Report integrity can be improved by an architected method that allows CICS program control to turn off the EPC feature. For the final transmission to an EPC-capable printer, the report controller turns off EPC in order to synchronize the logical end of data with the physical printing of that data. If an error is returned at end-of-report synchronization, and correct repositioning cannot be determined, the spool file is placed back on the POWER LST queue in ERRPRT status (DISP=Y) for manual repositioning by the operator.

When printers defined as EPC-capable are started, a test is made of whether the device and control unit support the architected method of EPC control. If this capability is present, then dynamic enabling and disabling of the feature is performed to ensure report integrity. If the architected method is not supported, a dummy block (consisting of carriage return only) is sent to the printer at the end of each copy, and at the end of each report. This ensures that most print errors during the printing of a particular copy or report are detected before that copy or report is deleted. If an error occurs during printing to an EPC-capable printer, the last block successfully printed is not known, so the default restart page is page 1.

EPC-capable printers are defined to CICS with PRINTERTYPE(EPC) in the RDO TYPETERM definition. If the printer is not specified as EPC-capable in the TYPETERM resource definition for the printer, but it does in fact support the EPC feature, and if full integrity is required, you must ensure that the printer is manually disabled.

Using printers for both report controller and other methods of printing

If you want to use the same printers for printing using the report controller and for printing using other methods, you should note the following points:

- If printer sharing is defined for the printer (see page 38), the printer is released after a time-out period, and becomes available to other applications.
- Screen-copying, using the hardware copy feature, a CICS local copy key, or the EXEC CICS ISSUE PRINT command, does not work if the printer named as the PRINTER or ALTPRINTER in the TERMINAL definition is not in use. In report controller terms, this means that the printer must be stopped. If the printer is in WAITING or IN USE status, it cannot be used for screen-copying. Screen-copying cannot wait for the printer to be stopped.
- If the printer is in use by another application, or by screen-copying, it is unavailable to the report controller. Normally, you cannot use the CEMS or CEOS START option for a printer that is unavailable. However, you might enter the START option, and then find that another transaction has just started to use the printer (the STARTING(IN USE) status). The report controller waits until the printer becomes free again.
- The report controller cannot be used to print reports on remotely defined printers.

Report controller report printing delay

The CICS Report Controller print task is a long running task designed to deliver reports to CICS terminal printers as they become available on the VSE/POWER spool. If the terminal printer is, in fact, an intelligent workstation which simulates printing operations, problems can occur because there is no indication to the simulation software of an 'end of report' condition. For the communication protocols in such simulation environments, this condition would, typically, use the SNA 'end-bracket' indicator for this purpose.

Because the CICS/VSE print task is long running, the end-bracket does not occur between reports but at end of task. The effect of this is that the terminal printer simulation software builds up long queues of reports that will not be printed until the RCF print task is stopped.

To overcome this, the RDO TERMINAL definition provides a parameter, PRINTIMMED(**NO**|YES) (PRINTIM operand in DFHTCT TYPE=TERMINAL macro). If YES is specified, the RCF print task ensures that end bracket flows after each report. **NO** is the default. For more information about the numeric values that applied to the PRINTIMMED parameter, see the *CICS Resource Definition Guide* manual.

Note: It is important to understand that this is implemented in such a way that there is a new print task created for every report sent to the terminal. This results in some additional overhead, and users should use the IMMEDIATE keyword only for the type of printer simulation environments described and not for real terminal printers.

Initial skip processing

Usually, a report will start by skipping to the top of a new page. However, in some cases this may not be true, and this can result in overprinting and other undesirable effects. To allow the maximum flexibility for these cases, VSE/POWER provides the facility to specify what should happen when a report without an initial skip is to be printed *on a POWER-controlled printer*. It does this at 2 levels:

- A default can be set in the VSE/POWER autostart job with the SKIP parameter of the SET command. If SKIP=YES is specified, VSE/POWER inserts a page skip if the report does not contain one at the start.
- When starting an individual printer with PSTART, you can specify SKIP/NOSKIP to override the global setting.

The report controller supports this for *CICS-controlled printers* as follows:

- The default setting is taken from the VSE/POWER specification.
- An individual CICS controlled printer can be started from the CICS CExS printer panel specifying SKIP(YES|NO). Starting a CICS printer from the VSE/ESA console with a PSTART DEV,... command is also supported.

Chapter 7. Report controller security

This chapter contains information about report controller security in the following sections:

- “Security”
- “What happens when the security check fails?” on page 58
- “Report controller BROWSE security” on page 59

This chapter describes the security aspects of the report controller and how security checking is implemented.

Security

The report controller provides two possible levels of operator control. There are two transactions:

1. CEMS, which provides functions suitable for the “computer operator”
2. CEOS, which provides functions suitable for the “end user.”

By protecting the use of these transactions with an external security manager, you can control the range of report controller functions available to the user.

The report controller also provides control over the following:

- Which printers can be used to print a report
- Which users can write to a resumable or log report
- Which users can change the characteristics of a printer or a report.
- Whether a report is reprinted automatically, following a failure during printing (PRINTFAIL option)
- Which users can browse a report

Resource security checking is used to control the first three. However, because compatibility with previous releases of the report controller must be maintained, the concept of an RSL value (in the range 1 through 24, or PUBLIC (127)) is retained. This allows existing applications to continue to execute correctly and ensures that reports routed to and from earlier releases of CICS/VSE can be correctly processed on all systems.

To provide resource security level (RSL) compatibility in an environment managed by external security, the report controller requires three sets of resource names to be defined to the FACILITY class of the ESM. These are as follows:

For **report** security: DFHRCF.RSL01 through DFHRCF.RSL24 and DFHRCF.RSLPU

For **printer** security: DFHRCF.PRSL01 through DFHRCF.PRSL24 and DFHRCF.PRSLPU

For **browse** security: DFHRCF.BRSL01 through DFHRCF.BRSL24 and DFHRCF.BRSLPU

In each case, 01-24 and PU correspond to an RSL specification. For more information about resource security, see the *CICS Security Guide*.

Users have access only to those reports and printers for which they are thus authorized, so there is no danger to other departments' or branches' reports and printers. Also, they may use their printer to print only the reports for which it is authorized. The security values and keys need some careful planning to be used effectively, however. For instance, you need to avoid giving users control over the characteristics of a printer that cannot be used to print their reports.

Reports are secured with the DFHRCF.RSL nn resource name where nn is the RSL value of the report. Whenever a program attempts to append to a report log (log and resumable reports) or a CEMS/CEOS user attempts to access a report, the ESM is called to ensure the user has the necessary UPDATE authority.

The report RSL is also used when printing a report on a CICS terminal printer. In this case, the userid representing the terminal printer must have READ authority for DFHRCF.RSL nn .

The report RSL is specified on the EXEC CICS SPOOLOPEN REPORT command. For more information about the EXEC CICS SPOOLOPEN REPORT command, see the *CICS Application Programming Reference* manual. You can also specify an RSL value for a report created by a batch program (see "Managing reports created by batch programs" on page 43), and when creating a report from a transient data queue (CEMS option 4). For information about defining resources for the report controller, see the *CICS Resource Definition Guide* manual.

Note that jobs and transient data queues are not accessible to anyone who is not authorized to use CEMS. In addition, if resource checking of transient data queues is active, (DFHSIT) parameter XDCT=YES, and RESSEC=YES is specified on transaction CEMS) the signed-on user must be authorized to access the transient data queue. Any report created without specifying an RSL value is assumed to have an RSL value of 1, and are therefore accessible to any user of the CEMS transaction who is authorized to DFHRCF.RSL01.

In addition to RSL, a report created by a transaction can be given the PRINTFAIL option, which prevents it from being reprinted without operator intervention, following a printer failure or a security check failure. This avoids the possibility of sensitive information being printed twice, or being printed while the printer is accessible to unauthorized persons.

The USERDATA on a report is visible to all report controller users who look at the report list panel, so take care how you use this field when creating reports.

What happens when the security check fails?

A printer intended for RCF use can be given an RSL value for control purposes. This is specified with the SPOOLPRTRSL keyword in the terminal definition for the printer. Users of the CEMS/CEOS transactions must have UPDATE authority to DFHRCF.PRSL nn to be able to control the printer

When an attempt is made to print a report on a printer whose userid does not have READ access to DFHRCF.PRSL nn where nn is the RSL value of the report, the report is set to ERRPRT status.

If it is necessary to have reports with different RSL values printed at the same physical location (on the same printer or group of printers), you are recommended to use several destination names at that location. You should allocate one RSL value to each destination.

You are not recommended to allocate RSL values to output classes within destination, because this is more complicated and error-prone: reports are more likely to be set to ERRPRT status through user error.

If a user accidentally starts a printer for reports for which the USERID of the printer is not authorized to the appropriate DFHRCF.RSL nn resource, the reports are sent to ERRPRT status. A security administrator can change the authority of the printer's USERID to include access to the required DFHRCF.RSL nn resource.

The security administrator might need to find out the RSL value required for the report from the application program, because this value is not recorded anywhere else. Even when the USERID associated with the printer is given access to the required DFHRCF.RSL nn resource to print the report, the user may still have a problem. The reports may be reports to which the user has no access, and so someone else with access to the appropriate DFHRCF.RSL nn resource must reset the status of the reports so that they can be printed.

Report controller **BROWSE** security

The implementation of the Report Controller Browse function has been changed to use the facilities of an external security manager. In previous releases of the report controller, authority to browse the contents of a report was controlled using the RSL of a dummy program, DFHPSBRS, which had to match one of the user's RSL keys.

In CICS Transaction Server for VSE/ESA Release 1 the user attempting to browse a report must have READ access to the resource DFHRCF.BRSL nn , where nn is the RSL value of the report.

Chapter 8. Report controller recovery

This chapter contains report controller recovery information in the following:

- “Report recovery levels”
- “Syncpoint” on page 62
- “PRINTFAIL” on page 63.

This chapter describes the recovery processing that CICS provides for the report controller.

Report recovery levels

You can specify the level of recovery required for reports with the EXEC CICS SPOOLOPEN REPORT command. The recovery level remains fixed for the lifetime of the report. Two types can be specified as follows:

PHYSICAL

A physically recoverable report is restored to the state it was in immediately before an abend. Each line is written directly to disk. This type of recovery makes heavy use of input/output, and its use should be carefully considered before it is selected.

You must specify PHYSICAL for log reports, if you want them to be recoverable. When log reports abend they are put in ERRCRT status. This means that after restart any attempt to write to or close the report results in the NAMEERROR condition.

Physical recovery is not allowed for JCL reports, and if coded is ignored.

LOGICAL

A logically recoverable report is restored, following an abend, to the state it was in at the start of the CICS task or at the last syncpoint. In this way you can ensure that an unfinished report is not created when an abend occurs. LOGICAL is the default.

You make choices concerning recovery in the EXEC CICS SPOOL... command options and by operator actions (switching from a failed printer, for example), but there are no special recovery programming requirements.

If you are using the CICS interface to POWER without the report controller, only facilities for logical recovery are provided. There is a replaceable terminal error program for use with the report controller. (For programming information about the terminal error program, see the *CICS Customization Guide*.)

Report controller recovery and restart falls into two main areas. Most of these failures are detected and may be recoverable and restartable:

- **Failures during report creation**—for example, a report writing transaction, on transaction abend, may be backed out to the last CICS syncpoint using dynamic transaction backout. The transaction might then be restarted from a known safe point using transaction restart.

In this example, the recovery action can be automatically invoked.

- **Failures during report printing and processing (reading)**—for example, a printer failure during report printing may be circumvented by redirecting the report to a serviceable printer.

In this example, operator intervention is required (CEMS or CEOS).

Types of report controller failure

There are five main types of report controller failure:

1. VSE system abend
2. POWER abend
3. CICS abend
4. CICS transaction abend
5. Printer failures (CICS and system local)

See the *CICS Recovery and Restart Guide* for more information about these types of failure.

Extended records

Records retrieved from POWER which are longer than the carriage width of the printer wrap onto the next line. Because report formatting is based on counting input lines, this may cause corruption. Care must be taken not to create spool records that exceed carriage width if the report controller is to be used to print a formatted report.

Output to a printer may require up to 10 bytes for carriage control or EOM type control information, so large POWER spool records are truncated to ensure that this data can be sent to the printer. The limiting factor is the size of the printer's buffer, as defined to CICS in the TCT. If truncation takes place, message DFH5493 will appear in the CSPW log. Report output continues by printing the maximum data that can be handled, but formatting may have been corrupted.

Syncpoint

Issuing an EXEC CICS SYNCPOINT command in an application program closes all reports.

All standard and resumable reports open at the time the syncpoint is issued are closed and their status set to "ready to print".

Standard reports with logical recovery for which an EXEC CICS SPOOLCLOSE REPORT command has been issued before the EXEC CICS SYNCPOINT command is issued are committed; that is, they are released for print. Only reports opened after the syncpoint are affected by abnormal termination of the task.

Log reports can be recovered at the physical level only. As a result, each record is committed to disk.

PRINTFAIL

Normally, a report that is being printed when the writer task CEPW abends, is reprinted automatically when CEPW is recovered. This could result in, for example, two copies of an invoice being printed, or in sensitive information being printed while unauthorized people are present. To prevent this situation, you can specify the PRINTFAIL option on the EXEC CICS SPOOL OPEN REPORT command for any report. A report with PRINTFAIL specified will not be reprinted without operator intervention.

Chapter 9. Enabling the report controller and printers

This chapter discusses the enablement of the report controller and printers in the following:

- “Enabling the CICS spooler interface” on page 66
- “Installing the DFHRCF group” on page 66
- “Preparing the resource definitions” on page 66
- “Defining VTAM configuration parameters” on page 68
- “Preparing the CICS startup job” on page 75
- “Other preparation work” on page 76
- “Overriding the VTAM startup parameters” on page 76
- “Submitting the CICS startup job” on page 76

This chapter describes what you have to do to set up the report controller and printers to work with the report controller in a CICS Transaction Server for VSE/ESA system. You may be enabling the report controller for the first time or you may simply want to add some new printers. References are made to CICS Transaction Server for VSE/ESA and VTAM books as necessary.

The following steps are necessary to set up CICS Transaction Server for VSE/ESA to use the report controller:

1. Enabling the CICS spooler interface.
2. Installing the group DFHRCF.
3. Preparing the resource definitions.

You can code macros to set up the terminal control table (TCT), program control table (PCT), and other tables with the appropriate definitions. Alternatively, you can use resource definition online to create a CICS system definition file (CSD) containing the necessary definitions.

4. Defining the VTAM configuration parameters. This is necessary to define the printers to VTAM.
5. Preparing the startup job stream. You must ensure that the CICS startup job specifies the correct system initialization parameters for the report controller and printers.
6. Other preparation work. The signon entries for CICS users must be set up.
7. Overriding the VTAM startup defaults with your own values. This is how the printers are defined to VTAM.
8. Submitting the CICS startup job. The correct system initialization parameters must be specified.

Enabling the CICS spooler interface

This section describes the steps you must take to enable the CICS spooler interface before you can use it with VSE/POWER and the report controller. You must:

1. Have the VSE/POWER product delivered with VSE/ESA 2.4 or a later release (note that this will normally be pre-installed as part of VSE/ESA installation) initialized in one of your CICS Transaction Server for VSE/ESA partitions (usually by means of an automated system initialization (ASI) JCL procedure).
2. Activate the system spooler interface as described in the *CICS System Definition Guide*.

Installing the DFHRCF group

You must install the DFHRCF group, which contains the TRANSACTION, PROGRAM, and MAPSET definitions for the report controller. You should include group DFHRCF in the group list you intend to use for CICS startup (see “Using resource definition online to define a terminal definition group” on page 67).

Preparing the resource definitions

This section describes the resources you must define to use the report controller in CICS for Transaction Server. You can define some resources by coding macros and most of them by using resource definition online. Refer to the *CICS Resource Definition Guide* manuals for full details.

Program and transaction resource definitions

You define the program and transaction resources by using RDO. Under normal conditions all that is required is to include the DFHRCF group in your group list at startup. However, if you are performing a non-standard installation/migration, you can upgrade an existing DFHRCF group using the DFHCSDUP utility program. Irrespective of the release you are migrating from, use

- UPGRADE USING(DFHCURCF)

DCT resource definitions

You can only define DCT resources by coding macros. Include the CSPA and CSPW entries in your destination control table (DCT). These two transient data queues provide an audit trail.

Figure 5 on page 67 is an example of DCT macro coding.

```

*****
*   $MOD(DFHDCT1$),COMP(STARTER),PROD(CICS/VSE) :   *
*****
      TITLE 'DFHDCTRC - CICS SAMPLE DESTINATION CONTROL TABLE'
*
      DFHDCT TYPE=INITIAL,                                X
              STARTER=YES,          ALLOWS      SUFFIX      X
              SUFFIX=RC
*
      COPY   DFH$DCT1          TABLE ENTRIES FOR:
      COPY   DFH$DCT           - SDSCI ENTRY
      COPY   DFH$DCTI          - BASIC CICS FACILITIES
      COPY   DFH$DCTS          - ICCF ENTRIES
      COPY   DFH$DCTS          - SAMPLE APPLICATIONS
*
      IESN   DFHDCT TYPE=INTRA,                                X
              DESTID=IESN,
      CSPA   DFHDCT TYPE=INDIRECT,                              X
              DESTID=CSPA,
              INDDEST=IESN
      CSPW   DFHDCT TYPE=INDIRECT,                              X
              DESTID=CSPW,
              INDDEST=IEP1
      IEP1   DFHDCT TYPE=INTRA,                                X
              DESTID=IEP1,
              DESTFAC=(TERMINAL,CNSL),
      DFHDCT TYPE=FINAL
      END

```

Figure 5. Example of DCT macro coding

TCT resource definitions

You can define TCT resources for VTAM-connected terminals only using resource definition online. If you have a terminal control table macro source deck from a previous release, you can use this by reassembling it using CICS Transaction Server macros. However, you must then use the assembler output to migrate all VTAM resources to RDO. For more information, see the *CICS Migration Guide*.

Using resource definition online to define a terminal definition group

If you are using resource definition online, you must define a group (for example, TCTRC) that contains the necessary TYPETERM and TERMINAL definitions for your terminals and printers. You must include this group in a list, (for example, RCFLIST) that includes all other necessary group definitions. The list is specified using the GRPLIST system initialization parameter.

TYPETERM definitions

The TYPETERM definition must correspond to the VTAM logon characteristics that appear in the VTAM logon mode table (see “The logon mode tables” on page 73 and Figure 8 on page 74).

The following definitions are of particular importance:

ALTSCREEN Defines the 3270 screen size.

DEFSCREEN	Defines the 3270 screen size or 3270 printer page size.
DEVICE	Specifies the device type.
DISCREQ	Indicates whether disconnect requests are to be honored. Code DISCREQ=YES if you want other applications to share the printer with the report controller.
FORMFEED	Indicates whether the device supports form feed (FF).
GROUP	Specifies the group name.
NETNAME	Specifies the name used to identify the terminal to VTAM, see page 72.
PAGESIZE	Specifies the default page size and size of the print buffer.
PRINTERTYPE	Specifies the type of printer. Specify PRINTERTYPE=EPC if the printer has an early print complete (EPC) feature.
TTI	Indicates whether CICS can RECEIVE data from the terminal. For EPC printers you should specify TTI=YES. We recommend that you specify this parameter value for all printers.
TYPETERM	Specifies the name referred to in all the terminal definitions.

TERMINAL definitions

Each TERMINAL definition refers to a TYPETERM definition, therefore TYPETERM and GROUP must match those in the TYPETERM definition.

The following TERMINAL definitions are also of particular importance:

AUTINSTMODEL	Indicates whether the TERMINAL definition can be used as a model terminal definition for autoinstall.
PRINTEDMSG	Indicates whether printer status messages are sent to the printer.
SPOOLDEST	Specifies the default destination for the printer.
SPOOLPRTRSL	Controls whether an operator is authorized to use the report controller.
SPOOLPRTTO	Specifies the time for which the printer can remain idle before it is freed for use by other CICS or VTAM applications.

For each of the terminal display units using the report controller, include the **SPOOLTO** operand. (This names a destination at which reports created at the terminal are to be printed.) Refer to the *CICS Resource Definition Guide* manual for details of using the CEDA transaction to define resources.

Defining VTAM configuration parameters

To define your printers to VTAM, you must specify a number of parameters. Default parameters are filed in the VTAM definition library, but you can override these with definitions from a private sublibrary.

In the example in Figure 6 on page 70, the job VTAMSET is used to catalog the following members into a sublibrary of PRIV02:

- ATCSTR.xx.B** A start option list used to supplement the default start option list ATCSTR00.B. The suffix *xx* is the same as supplied by the LIST start option in your TCT (see “Overriding the VTAM startup parameters” on page 76).
- ATCCON.xx.B** A configuration list used to override the default configuration list ATCCON00.B.
- APPL.xx.B** The member defining the application program major node.
- LOC77.xx.B** A member defining a group of terminals.
- LOC78.xx.B** A member defining a group of terminals.
- PATH.xx.B** A member defining the routes VTAM takes to communicate with other nodes.

```

// JOB VTAMSET
// EXEC LIBR
ACCESS SUBLIB=PRIV02.COWAN
CATALOG ATCSTRRC.B,REPLACE=YES
* * * * *
*           ATCSTRRC (Start option list
* (FROM ATCSTR00 - SSCPID=10,NETID=GBIBMIYA,
*           HOSTPU=ISTPUS,IOINT=180)
* * * * *
CONFIG=RC      *POINTER TO CONFIGURATION LIST ATCCONRC
/+
CATALOG ATCCONRC.B,REPLACE=YES
* * * * *
*           ATCCONRC (CONFIGURATION LIST OF MAJOR NODES)
* * * * *
APPLRC,LOC77RC,LOC78RC
/+
CATALOG APPLRC.B,REPLACE=YES
* * * * *
*           APPLDC (APPLICATION PROGRAM MAJOR NODE)
* * * * *
           VBUILD TYPE=APPL
*
CICSRC  APPL  AUTH=(ACQ),           C
           VPACING=4,               C
           EAS=16,                  C
           PARSESS=YES,             C
           SONSCIP=YES
/+
CATALOG LOC77RC.B,REPLACE=YES
* * * * *
*           LOC77RC
* * * * *
LOC77RC  LBUILD
* ACTIVE SCREENS
L77A    LOCAL TERM=3277,CUADDR=0C0, ISTATUS=ACTIVE,MODETAB=MT3278B,  C
           FEATUR2=(MODEL2),USSTAB=BSCUSSTB,LOGTAB=INTABAB
L77B    LOCAL TERM=3277,CUADDR=0C1, ISTATUS=ACTIVE,MODETAB=MT3278B,  C
           FEATUR2=(MODEL2),USSTAB=BSCUSSTB,LOGTAB=INTABAB
L77C    LOCAL TERM=3277,CUADDR=0C2, ISTATUS=ACTIVE,MODETAB=MT3278B,  C
           FEATUR2=(MODEL2),USSTAB=BSCUSSTB,LOGTAB=INTABAB
L77D    LOCAL TERM=3277,CUADDR=0C3, ISTATUS=ACTIVE,MODETAB=MT3278B,  C
           FEATUR2=(MODEL2),USSTAB=BSCUSSTB,LOGTAB=INTABAB
L77E    LOCAL TERM=3277,CUADDR=0C4, ISTATUS=ACTIVE,MODETAB=MT3278B,  C
           FEATUR2=(MODEL2),USSTAB=BSCUSSTB,LOGTAB=INTABAB
/+

```

Figure 6 (Part 1 of 2). Job for cataloging VTAM definitions

```

CATALOG LOC78RC.B,REPLACE=YES
* * * * *
*       LOC78RC
* * * * *
LOC78RC  LBUILD
* ACTIVE SCREENS
L78A    LOCAL TERM=3277,CUADDR=0D0, ISTATUS=ACTIVE,MODETAB=MT3278B,   C
        FEATUR2=(EDATS),USSTAB=BSCUSSTB,LOGTAB=INTABAB
L78B    LOCAL TERM=3277,CUADDR=0D1, ISTATUS=ACTIVE,MODETAB=MT3278B,   C
        FEATUR2=(MODEL2,EDATS),USSTAB=BSCUSSTB,LOGTAB=INTABAB
L78C    LOCAL TERM=3277,CUADDR=0D2, ISTATUS=ACTIVE,MODETAB=MT3278B,   C
        FEATUR2=(MODEL2,EDATS),USSTAB=BSCUSSTB,LOGTAB=INTABAB
L78D    LOCAL TERM=3277,CUADDR=0D3, ISTATUS=ACTIVE,MODETAB=MT3278B,   C
        FEATUR2=(MODEL2,EDATS),USSTAB=BSCUSSTB,LOGTAB=INTABAB
L78E    LOCAL TERM=3277,CUADDR=0D4, ISTATUS=ACTIVE,MODETAB=MT3278B,   C
        FEATUR2=(MODEL2,EDATS),USSTAB=BSCUSSTB,LOGTAB=INTABAB
* ACTIVE PRINTERS
L870    LOCAL TERM=3286,CUADDR=0D6, ISTATUS=ACTIVE,MODETAB=MT3287B,   C
        FEATUR2=(MODEL2,PRINTR)
L871    LOCAL TERM=3284,CUADDR=0D7, ISTATUS=ACTIVE,MODETAB=MT3287B,   C
        FEATUR2=(MODEL2,PRINTR)

/+
CATALOG PATHRC.B,REPLACE=YES
* * * * *
*       PATHRC
* * * * *
PATH0   PATH  DESTSA=20,ER0=(20,1),VR0=0
/+
/*
/&

```

Figure 6 (Part 2 of 2). Job for cataloging VTAM definitions

For full details of how to code a job such as VTAMSET, see the *VTAM Installation and Resource Definition Reference* manual.

The following sections describe the various start options and statements in the example in Figure 6 on page 70.

The CONFIG start option

The CONFIG start option names a list of major nodes to be activated when VTAM is started. If this is coded as CONFIG=xx, then the file ATCCONxx.B is the list of major nodes. The xx suffix is the same as used in the LIST start option at VTAM startup (see “Overriding the VTAM startup parameters” on page 76).

Note: The configuration list named by CONFIG=xx replaces the default configuration list. This differs from start option list processing where the xx list is merged with the 00 list.

The VBUILD statement

You define an application major node by filing one VBUILD statement for the major node, and an APPL definition statement for each application program in the major node. In the example, CICSRC is the name of the application program and this defines CICS as a VTAM application. You must use the same application name with the APPLID system initialization parameter.

The LBUILD statement

You must code one LBUILD statement for each logical group of channel-attached non-SNA terminals. You define each terminal in the group by using a LOCAL definition statement.

The LOCAL statement

You code one LOCAL statement for each terminal. The name applying to the statement (for example, L77A) is the name assigned to the device whose address appears in the CUADDR operand in this local definition statement. This name must also be used in the NETNAME operand in the CICS TERMINAL resource definitions (see "Using resource definition online to define a terminal definition group" on page 67). The meanings of the operands of the LOCAL statement are as follows:

- CUADDR** Three hexadecimal digits (X'000'–X'FFF') that identify the channel device address for the terminal. You must not enclose the address within quotation marks or apostrophes.
- FEATUR2** The machine features for a specific terminal:
- MODEL2** Identifies Model 2 as the model of printer.
 - PRINTR** Identifies the terminal as a printer.
- ISTATUS** Whether the terminal should be automatically activated when the local non-SNA major node to which it belongs is first activated.
- LOGTAB** The name of an interpret table to be used by VTAM when processing logon requests that come from the local LU. The interpret table is used to interpret the name received from an initiation request (see "Interpret tables" on page 74).
- MODETAB** The name of a logon mode table to be used.
- TERM** The specific channel-attached non-SNA terminal. You can code 3277, 3284, or 3286 only.
- USSTAB** The name of a USS definition table created as described in the *VTAM Installation and Resource Definition Reference* manual.

The PATH statement

The PATH statement defines the routes VTAM takes to communicate with other subarea nodes. One or more PATH statements are filed in the VTAM definition library. See the *VTAM Installation and Resource Definition Reference* manual for further details.

The logon mode tables

Logon mode tables (see Figure 7) are lists of session protocols that SNA logical units may use during a session between a terminal and a VTAM application program such as CICS. A logon mode table entry is a set of rules that specify how the requested session is to be conducted. The tables are defined by specifying a MODETAB macro, one MODEENT macro per device, and one MODEEND macro.

VTAM has a logon mode table named ISTINCLM, which provides generally accepted session protocols for a basic list of IBM device types. You must write your own logon mode table to accommodate other device types. The *VTAM Installation and Resource Definition Reference* manual describes how to write your own logon mode tables and include them in VTAM. Each logon mode table must be assembled and link-edited into the private definition library referenced in the VTAM startup procedure. Figure 7 contains definitions for non-SNA 3270 devices.

```

MT3287B  MODETAB
*-----*
*
* MODE TABLE FOR BSC 3287 AND 3289.
*
*-----*
*
MT32872  MODEENT LOGMODE=MT32872,  3287/3289  MODEL 2
          TYPE=1,                    ONLY TYPE RECOGNIZED
          FMPROF=X'02',              NON-SNA
          TSPROF=X'02',              NON-SNA
          PRIPROT=X'71',              PRIMARY PROTOCOL
          SECPROT=X'40',              SECONDARY PROTOCOL
          COMPROT=X'2000',            COMMON PROTOCOL
          PSERVIC=X'00800000000000000000200'

*-----
MT32871  MODEENT LOGMODE=MT32871,  3287/3289  MODEL 1
          TYPE=1,                    ONLY TYPE RECOGNIZED
          FMPROF=X'02',              NON-SNA
          TSPROF=X'02',              NON-SNA
          PRIPROT=X'71',              PRIMARY PROTOCOL
          SECPROT=X'40',              SECONDARY PROTOCOL
          COMPROT=X'2000',            COMMON PROTOCOL
          PSERVIC=X'00800000000000000000100'

*-----
KANPRT   MODEENT LOGMODE=KANPRT,    5553 KANJI PRINTER (NON-SCS)
          TYPE=1,                    ONLY TYPE RECOGNIZED
          FMPROF=X'02',              NON-SNA
          TSPROF=X'02',              NON-SNA
          PRIPROT=X'71',              PRIMARY PROTOCOL
          SECPROT=X'40',              SECONDARY PROTOCOL
          COMPROT=X'2000',            COMMON PROTOCOL
          PSERVIC=X'008000000000185018507F00'

*-----
          MODEEND
          END

```

Figure 7. Example of a logon mode table

The MODEENT macro associates a logon mode table entry with a set of session parameters. The PSERVIC operand contains such information as the primary and alternate screen sizes.

Figure 8 shows the relationship between the PSERVIC operand in the MODEENT macro and the TYPETERM definition. It also shows the relationship between the TERMINAL and TYPETERM definitions.

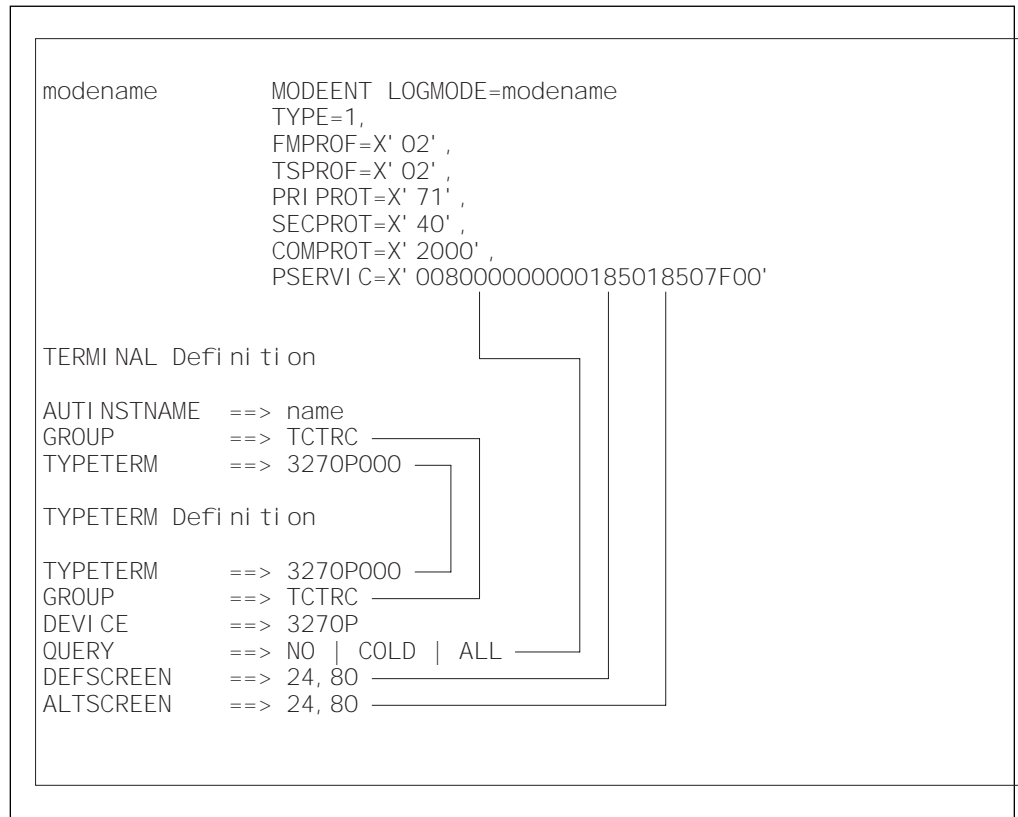


Figure 8. MODEENT and TYPETERM relationship

Interpret tables

VTAM uses the interpret table to determine which application program is to be notified when a session-establishment request is received. The name (in the example in Figure 9, INTABAB) must be the same as the name used in the LOGTAB operand in the VTAM definition file.

In the example, the two LOGCHAR macros allow the logon message to be in uppercase or lowercase letters. The APPLID operand identifies CICS1 as the application with which VTAM establishes a session.

```

INTABAB  INTAB
SEQ1     LOGCHAR APPLID=(USERVAR,CICSR1),SEQNCE='CICSR1'
SEQ2     LOGCHAR APPLID=(USERVAR,CICSR1),SEQNCE='cicsr1'
          ENDINTAB
          END
  
```

Figure 9. Example of an interpret table

Refer to the *VTAM Resource Definition Reference* manual for details of coding interpret tables. Each interpret table must be assembled and link-edited into the private definition library referenced in the VTAM startup procedure.

Preparing the CICS startup job

Figure 10 shows an extract from a CICS startup job, showing the statements that are necessary for adding the report controller.

```
LOG
// LIBDEF      *,SEARCH=(PRIV02.COWAN,BSFV02.BSF,PRD2.PROD,          *
                  COBOL2.CICS,COBOL2.RUN,COBOL2.DEB,              *
                  C370.RUN)
// UPSI 101
* // PAUSE  ENTER LIBDEFS ETC. HERE, EXEC CICS NEXT
// EXEC DFHSIP,SIZE=2M
SIT=1$,GRPLIST=RCFLIST
APPLID=CICSRC
START=COLD,MSGVLV=2
MCT=1$,JCT=1$
PLI=YES,COBOL2=YES
TCT=RC
PCT=1$
PPT=1$
DCT=1$
ISC=NO,DIP=YES
SPOOL=(YES,S,S)
$END
/*
// PAUSE EOJ HERE
// EXEC LISTLOG
NOLOG
/&
* $$ EOJ
```

Figure 10. Extract from a CICS startup job

Notes:

1. The LIBDEF statement must include the private sublibrary that contains your resource definitions and other files (PRIV02.COWAN in this example).
2. The GRPLIST system initialization parameter specifies that the list RCFLIST contains your resource definitions.
3. When you run CICS with VTAM, you must ensure that your CICS system application identifier (applid) is also defined to VTAM in an APPL statement (see “The VBUILD statement” on page 72).
4. The system initialization parameter TCT=RC specifies that DFHTCTRC contains the terminal definitions.
5. The system initialization parameter SPOOL=YES specifies the system spooler interface, which is necessary for the report controller. The second value specified by SPOOL is a 1-character identifier for the CICS system. If you plan to run CICS systems in different VSE partitions, you must use a different 1-character identifier for each CICS system. Otherwise, the VSE/POWER

interface is not initialized and the message DFH5301I is issued for one of the systems at CICS system startup.

System initialization parameters can be specified in the system initialization table (SIT) or as a system initialization override.

Other preparation work

Other things you must do are:

- Set up the signon entries for CICS users. Include READ or UPDATE access to the FACILITY resources for RSL compatibility. Also, include the language to be used by the users (English, Kanji, or German).
- Make sure that you have a nucleus load table (NLT) entry for DFHSP, if required.

Overriding the VTAM startup parameters

The VTAM startup job is run after VSE and POWER have started. (This might, depending on the partition, be called F1VTAM.JOB). When VTAM is started, it reads from the VTAM definition library a list of user-defined default values named ATCSTR00.B. You can enter the LIST=xx start option however, which names another list, ATCSTRxx.B to supplement ATCSTR00.B. (The suffix xx must be the same as used for the APPLID.) The supplemental list can override options in the default list, as well as the IBM-supplied internal default values.

If you enter LIST=RC at the ENTER VTAM START PARAMETERS prompt, the supplemental list is ATCSTRRC.B. This allows you to override the default VTAM configuration parameters, and define your printers to VTAM.

Submitting the CICS startup job

The CICS startup job is submitted after the VTAM startup job. The system initialization overrides then take effect (see “Preparing the CICS startup job” on page 75).

Chapter 10. Report controller sample programs

This chapter contains report controller sample programs information as follows:

- “The low balance report sample program”
- “Sample programs for use with the report controller tutorial” on page 78

This chapter describes the sample programs that generate reports. These sample programs are included in source form on the report controller feature tape, and are executable only when the report controller is installed.

The low balance report sample program

The low balance report sample program is presented in assembler language, C for VSE/ESA, COBOL for LE, and PL/I.

You can invoke each of the five versions of the sample program by entering its transaction identifier (ARPS, DRPS, 2RPS, and PRPS respectively) on to a clear screen. The low balance report sample program creates a report called DFH\$ARPS, DFH\$DRPS, or DFH\$PRPS, that lists all entries in the dataset FILEA for which the amount is less than or equal to \$50.00.

The program shows page building techniques and the use of the terminal paging facilities of BMS. The report is written to the POWER spool queue, and you can then control the output of this report using the CEMS or CEOS transactions.

The report names, mapset, and transaction identifiers for this function are:

Language	Program	Mapset	Trans. ID
Assembler	DFH\$ARPS.A	DFH\$AGD	ARPS
C for VSE/ESA	DFH\$DRPS.H	DFH\$DGD	DRPS
COBOL for VSE/ESA	DFH0CRPS.C	DFH0CGD	2RPS
PL/I for VSE/ESA	DFH\$PRPS.P	DFH\$PGD	PRPS

The program does a sequential scan through the file selecting each entry that obeys the search criterion. The pages are built from four maps that comprise mapset DFH\$AGD, using the paging option so that the data is not displayed immediately but is stored instead for later retrieval. The HEADING map is inserted at the head of each page. The detail map (DFH\$AGD) is written repeatedly until the OVERFLOW condition occurs. The FOOTING map is then written at the foot of the page and the HEADING map written at the top of the next page. The command to write the detail map that caused overflow is then repeated. When all the data has been written, the FINAL map is written at the bottom of the last page and the transaction is terminated.

The terminal operator then enters paging commands to display the data, clearing the screen before entering each paging command.

Before you can run the low balance report sample program, you must initialize CICS with the SPOOL=YES and PGRET=P/ SI parameters. You should install the Report Controller definitions GROUP(DFHRCF) and the FILEA sample applications group for your chosen language (DFHcAFLA, DFHcDFLA, DFHcCFL or DFHcPFLA). You should also create the sample file FILEA. For more details about this, see the *CICS System Definition Guide*.

Sample programs for use with the report controller tutorial

The sample transaction SREP generates the set of reports used in the tutorial section of the *CICS Report Controller User's Guide*. The transaction is provided as an aid to end-user training. If you make the transaction available, either on your production CICS system or on a test CICS system, the user can sign on to CICS and type in the transaction code.

SREP generates a set of reports with the user's user ID as the report controller destination. End users can process these reports as they work through the tutorial, without affecting other users of the report controller, or other users of the tutorial.

The program names and transaction identifiers for this function are:

<i>Table 6. Programs, mapsets, and transaction identifiers</i>			
Language	Program	Mapset	Trans. ID
Assembler	DFH\$ASRE	None	SREP
COBOL for VSE/ESA	DFH0CSRE	None	Not defined
PL/I for VSE/ESA	DFH\$PSRE	None	Not defined

The TRANSACTION definition and DFHPCT entry supplied for SREP both have the assembler-language version of the program specified. Therefore, if you want to use PL/I or COBOL, you need to copy the definition or entry and change it.

Using the report controller sample programs

Before you make report controller sample programs available to your users, you should carry out the following checks:

1. The report controller sample programs take the user ID from the CICS signon as the destination identifier for the set of reports generated. If your users do not sign on to CICS using one of the signon transactions, the signon user ID is not available, so you must provide an alternative method for obtaining a report destination. You can do this by modifying the appropriate program, either to pick up a user ID from some other source, or use some convention of your own to generate a destination identifier.
2. Make sure that there are no conflicts between the characteristics generated on the sample reports and your normal installation standards.

The reports are generated with the following:

- Class—the sample program does not generate a report class and therefore the report controller takes whatever you have specified as the default spool class in the SIT.
- Report names.

- Userdata—a variety of examples are used; see the source listings for details of the data.
- Priority—this has effect within the destination only, but you may need to check this if you modify the way in which the destination identifier is derived (see item 1 on page 78 above).
- Forms—the sample programs use either SAMP or leave blank.
- Size of reports—these have been kept small, and should be accommodated on all standard paper sizes.

If the destination identifier and sample report characteristics are satisfactory for your installation, you can use the pregenerated assembler-language version, DFH\$ASRE.

However, if you need to modify the sample program, edit the source program of whichever language you prefer and then translate, compile (or assemble), and link-edit a new version. (Some guidance on modifying the sample programs is given in the comments at the beginning of the sample source members.)

If you install an amended version, you should check the resource definitions for the transaction and program. Note that only the assembler-language version has a transaction code defined in the CSD by the samples upgrade module DFHCU22F.

Note: If you change any of the report characteristics, remember to note the changes in the *CICS Report Controller User's Guide* tutorial text before you make them available to any user of the report controller.

The tutorial advises users that they should delete the sample reports when they have finished.

Using the tutorial

To implement the tutorial in the *CICS Report Controller User's Guide*, your users will need a printer that is:

- Defined to CICS
- Stopped, but with power turned on and ready to print
- Loaded with standard paper

They need to know the 4-character name of this printer, and they need to be able to sign on to CICS using CSSN, with a user ID and password. They should sign on to a 3270-type terminal that is defined to CICS.

Bibliography

CICS Transaction Server for VSE/ESA Release 1 library

Evaluation and planning	
<i>Release Guide</i>	GC33-1645
<i>Migration Guide</i>	GC33-1646
<i>Report Controller Planning Guide</i>	GC33-1941
General	
<i>Master Index</i>	SC33-1648
<i>Trace Entries</i>	SC34-5556
<i>User's Handbook</i>	SC34-5555
<i>Glossary (softcopy only)</i>	GC33-1649
Administration	
<i>System Definition Guide</i>	SC33-1651
<i>Customization Guide</i>	SC33-1652
<i>Resource Definition Guide</i>	SC33-1653
<i>Operations and Utilities Guide</i>	SC33-1654
<i>CICS-Supplied Transactions</i>	SC33-1655
Programming	
<i>Application Programming Guide</i>	SC33-1657
<i>Application Programming Reference</i>	SC33-1658
<i>Sample Applications Guide</i>	SC33-1713
<i>Application Migration Aid Guide</i>	SC33-1943
<i>System Programming Reference</i>	SC33-1659
<i>Distributed Transaction Programming Guide</i>	SC33-1661
<i>Front End Programming Interface User's Guide</i>	SC33-1662
Diagnosis	
<i>Problem Determination Guide</i>	GC33-1663
<i>Messages and Codes Vol 3 (softcopy only)</i>	SC33-6799
<i>Diagnosis Reference</i>	LY33-6085
<i>Data Areas</i>	LY33-6086
<i>Supplementary Data Areas</i>	LY33-6087
Communication	
<i>Intercommunication Guide</i>	SC33-1665
<i>CICS Family: Interproduct Communication</i>	SC33-0824
<i>CICS Family: Communicating from CICS on System/390</i>	SC33-1697
Special topics	
<i>Recovery and Restart Guide</i>	SC33-1666
<i>Performance Guide</i>	SC33-1667
<i>Shared Data Tables Guide</i>	SC33-1668
<i>Security Guide</i>	SC33-1942
<i>External CICS Interface</i>	SC33-1669
<i>XRF Guide</i>	SC33-1671
<i>Report Controller User's Guide</i>	GC33-1940
CICS Clients	
<i>CICS Clients: Administration</i>	SC33-1792
<i>CICS Universal Clients Version 3 for OS/2: Administration</i>	SC34-5450
<i>CICS Universal Clients Version 3 for Windows: Administration</i>	SC34-5449
<i>CICS Universal Clients Version 3 for AIX: Administration</i>	SC34-5348
<i>CICS Universal Clients Version 3 for Solaris: Administration</i>	SC34-5451
<i>CICS Family: OO programming in C++ for CICS Clients</i>	SC33-1923
<i>CICS Family: OO programming in BASIC for CICS Clients</i>	SC33-1671
<i>CICS Family: Client/Server Programming</i>	SC33-1435
<i>CICS Transaction Gateway Version 3: Administration</i>	SC34-5448

Books from VSE/ESA 2.4 base program libraries

VSE/ESA Version 2 Release 4

Book title	Order number
Administration	SC33-6705
Diagnosis Tools	SC33-6614
Extended Addressability	SC33-6621
Guide for Solving Problems	SC33-6710
Guide to System Functions	SC33-6711
Installation	SC33-6704
Licensed Program Specification	GC33-6700
Messages and Codes Volume 1	SC33-6796
Messages and Codes Volume 2	SC33-6798
Messages and Codes Volume 3	SC33-6799
Networking Support	SC33-6708
Operation	SC33-6706
Planning	SC33-6703
Programming and Workstation Guide	SC33-6709
System Control Statements	SC33-6713
System Macro Reference	SC33-6716
System Macro User's Guide	SC33-6715
System Upgrade and Service	SC33-6702
System Utilities	SC33-6717
TCP/IP User's Guide	SC33-6601
Turbo Dispatcher Guide and Reference	SC33-6797
Unattended Node Support	SC33-6712

High-Level Assembler Language (HLASM)

Book title	Order number
General Information	GC26-8261
Installation and Customization Guide	SC26-8263
Language Reference	SC26-8265
Programmer's Guide	SC26-8264

Language Environment for VSE/ESA (LE/VSE)

Book title	Order number
C Run-Time Library Reference	SC33-6689
C Run-Time Programming Guide	SC33-6688
Concepts Guide	GC33-6680
Debug Tool for VSE/ESA Fact Sheet	GC26-8925
Debug Tool for VSE/ESA Installation and Customization Guide	SC26-8798
Debug Tool for VSE/ESA User's Guide and Reference	SC26-8797
Debugging Guide and Run-Time Messages	SC33-6681
Diagnosis Guide	SC26-8060
Fact Sheet	GC33-6679
Installation and Customization Guide	SC33-6682
LE/VSE Enhancements	SC33-6778
Licensed Program Specification	GC33-6683
Programming Guide	SC33-6684
Programming Reference	SC33-6685
Run-Time Migration Guide	SC33-6687
Writing Interlanguage Communication Applications	SC33-6686

VSE/ICCF

Book title	Order number
Administration and Operations	SC33-6738
User's Guide	SC33-6739

VSE/POWER

Book title	Order number
Administration and Operation	SC33-6733
Application Programming	SC33-6736
Networking Guide	SC33-6735
Remote Job Entry User's Guide	SC33-6734

VSE/VSAM

Book title	Order number
Commands	SC33-6731
User's Guide and Application Programming	SC33-6732

VTAM for VSE/ESA

Book title	Order number
Customization	LY43-0063
Diagnosis	LY43-0065
Data Areas	LY43-0104
Messages and Codes	SC31-6493
Migration Guide	GC31-8072
Network Implementation Guide	SC31-6494
Operation	SC31-6495
Overview	GC31-8114
Programming	SC31-6496
Programming for LU6.2	SC31-6497
Release Guide	GC31-8090
Resource Definition Reference	SC31-6498

Books from VSE/ESA 2.4 optional program libraries

C for VSE/ESA (C/VSE)

Book title	Order number
C Run-Time Library Reference	SC33-6689
C Run-Time Programming Guide	SC33-6688
Diagnosis Guide	GC09-2426
Installation and Customization Guide	GC09-2422
Language Reference	SC09-2425
Licensed Program Specification	GC09-2421
Migration Guide	SC09-2423
User's Guide	SC09-2424

COBOL for VSE/ESA (COBOL/VSE)

Book title	Order number
Debug Tool for VSE/ESA Fact Sheet	GC26-8925
Debug Tool for VSE/ESA Installation and Customization Guide	SC26-8798
Debug Tool for VSE/ESA User's Guide and Reference	SC26-8797
Diagnosis Guide	SC26-8528
General Information	GC26-8068
Installation and Customization Guide	SC26-8071
Language Reference	SC26-8073
Licensed Program Specifications	GC26-8069
Migration Guide	GC26-8070
Migrating VSE Applications To Advanced COBOL	GC26-8349
Programming Guide	SC26-8072

DB2 Server for VSE

Book title	Order number
Application Programming	SC09-2393
Database Administration	GC09-2389
Installation	GC09-2391
Interactive SQL Guide and Reference	SC09-2410
Operation	SC09-2401
Overview	GC08-2386
System Administration	GC09-2406

DL/I VSE

Book title	Order number
Application and Database Design	SH24-5022
Application Programming: CALL and RQDLI Interface	SH12-5411
Application Programming: High-Level Programming Interface	SH24-5009
Database Administration	SH24-5011
Diagnostic Guide	SH24-5002
General Information	GH20-1246
Guide for New Users	SH24-5001
Interactive Resource Definition and Utilities	SH24-5029
Library Guide and Master Index	GH24-5008
Licensed Program Specifications	GH24-5031
Low-level Code and Continuity Check Feature	SH20-9046
Library Guide and Master Index	GH24-5008
Messages and Codes	SH12-5414
Recovery and Restart Guide	SH24-5030
Reference Summary: CALL Program Interface	SX24-5103
Reference Summary: System Programming	SX24-5104
Reference Summary: HLPI Interface	SX24-5120
Release Guide	SC33-6211

PL/I for VSE/ESA (PL/I VSE)

Book title	Order number
Compile Time Messages and Codes	SC26-8059
Debug Tool For VSE/ESA User's Guide and Reference	SC26-8797
Diagnosis Guide	SC26-8058
Installation and Customization Guide	SC26-8057
Language Reference	SC26-8054
Licensed Program Specifications	GC26-8055
Migration Guide	SC26-8056
Programming Guide	SC26-8053
Reference Summary	SX26-3836

Screen Definition Facility II (SDF II)

Book title	Order number
VSE Administrator's Guide	SH12-6311
VSE General Introduction	SH12-6315
VSE Primer for CICS/BMS Programs	SH12-6313
VSE Run-Time Services	SH12-6312

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

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

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply in the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore this statement may not apply to you.

This publication could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM United Kingdom Laboratories, MP151, Hursley Park, Winchester, Hampshire, England, SO21 2JN. Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Programming License Agreement, or any equivalent agreement between us.

Trademarks and service marks

The following terms, used in this publication, are trademarks or service marks of IBM Corporation in the United States or other countries:

CICS, CICS/VSE, IBM, VSE/ESA, VTAM, CICS Transaction Server for VSE/ESA.

Other company, products, and service names may be the trademarks or service marks of others.

Index

A

aligning forms on a printer 37
appendable reports 6
APPL definition statement 72
application programming for report controller 5
ASA report format 20, 36
audit trail 48

B

batch programs, reports created by 43
BMS mapping 27
BOTHNUM option, SPOOLOPEN REPORT
command 6
BROWSE security 59
browsing reports 36

C

CECI transaction 5
CEDF transaction (CICS execution diagnostic
facility) 5
CEMS transaction 57
CEOS transaction 57
CEPW transaction (writer task) 63
changing characteristics of a report 31
changing job characteristics 39
changing usage of printer with report controller 37
CICS execution diagnostic facility (CEDF) 5
CICS spooler interface 66
CICS startup job 75
CICS DATA 45, 46
closing a report 16
commands 5
CONFIG start option 71
controlling
jobs 39
printers 37
reports 31
transient data queues 41
creating
report from transient data queue 41
reports 5
CSPA report controller audit trail 48, 66
CSPW report controller print task audit trail 48, 66

D

date, report 23
DATETIME option, SPOOLOPEN REPORT
command 7

DBCS (double-byte character set) 36
DCT (destination control table) 66
debugging the SPOOL... commands 5
DEFINE statement 46
DELETE option, SPOOLCLOSE REPORT
command 16
deleting
a job 39
a transient data queue 41
destination control table (DCT) 66
destination for a report 32
DESTINATION option, SPOOLOPEN REPORT
command 6
devices suitable for report controller 5
DFHRCF group 66
DISP option, SPOOLCLOSE REPORT command 16
dispositions 45
distributed report printing 19
double-byte character set (DBCS) 36

E

early print complete (EPC) feature 54
equipment suitable for report controller 5
ESCAPE option in VSE batch program
ESCAPE option, SPOOLOPEN REPORT
command 28
escape program 46
ESCAPE report format 28, 36
EXEC-level commands 5

F

FCB (forms control buffer) 47
FCB hierarchy
FLENGTH option, SPOOLWRITE REPORT
command 16
FOOT option, SPOOLOPEN REPORT command 6
footings, report 23
FOOTNUM option, SPOOLOPEN REPORT
command 6
format conversion 19
format of a report 19
formfeeds 54
forms control buffer (FCB) 47
forms type 35
FROM option, SPOOLWRITE REPORT command 16

H

hardware suitable for report controller 5
HEAD option, SPOOLOPEN REPORT command 6

headings, report 22
HEADNUM option, SPOOLOPEN REPORT
command 6
HOLD option, SPOOLCLOSE REPORT command 16
holding a job 39

I

identifying reports 9
installing the report controller 65
interpret table 74
introduction to the report controller 1
invalid characters 21

J

JCL report format 29
JOB statement parameters 29

K

KEEP option, SPOOLCLOSE REPORT command 16

L

LBUILD statement 72
line length of reports 22
LINELENGTH option, SPOOLOPEN REPORT
command 16
lines per page in reports 22
LIST start option 76
listing
 printers 37
 the job queue 39
 transient data queues 41
LOCAL statement 72
LOG option, SPOOLCLOSE REPORT command 16
log reports 6, 14
 See also audit trail
 opening and closing 6
logical recovery 61
logon mode table 73

M

managing
 jobs 39
 printers 37
 reports 31
 transient data queues 41
mapping, BMS 27
MCC report format 21, 36
multiformat reports 36

N

NOCC report format 20, 36
NOCONV option, EXEC CICS SPOOLOPEN REPORT
command 27
NOCONV option, SPOOLOPEN REPORT
command 19
NOCONV report format 36
NONUM option, SPOOLOPEN REPORT command 6
NOSEP option, SPOOLOPEN REPORT command 7
number of copies 35

O

opening a report 15
OPTB (output parameter text block) 46
output class 34
output parameter text block (OPTB) 46

P

page headings and footings 6
page numbering, report 6, 23
PATH statement 72
PCT (program control table) 66
physical recovery 61
POWER
 commands provided by report controller
 functions 51
 destinations and printers 52
 dispositions 45
 XMT queue 2
PPT (processing program table) 66
printer destination 32
printer status messages 68
printers
 controlling with the report controller 37
 format conversion 19
 security characteristics 57
 suitable for report controller 5
 use of PRINTFAIL option 58
PRINTFAIL option, SPOOLOPEN REPORT
command 6, 58, 63
printing reports 32
processing program table (PPT) 66
program control table (PCT) 66
programming the report controller 5
PWRSYSID option 7

R

recovery level of report
 logical recovery 61
 no recovery 61
 physical recovery 61
RELEASE option, SPOOLCLOSE REPORT
command 16

- releasing a job to run 39
- reopening a resumable report 15
- report controller
 - audit trail 48
 - comparison with POWER commands 51
 - debugging the SPOOL... commands 5
 - introduction 1
 - level of operator control 57
 - operations staff role 51
 - options 6
 - page headings and footings 6
 - page numbering 6
 - planning information 1
 - printers suitable 5
 - producing reports 5
 - recovery 61
 - report name 9
 - report title 9
 - report types 6
 - security facilities 57
 - separator pages 7
 - terminals suitable 5
 - time and date on reports 7
 - transactions (CEMS, CEOS) 57
 - trying out the EXEC CICS SPOOL... commands 5
 - tutorial 78
 - types of failure 62
 - userdata 9
- report controller failure, types 62
- report formats 19, 36
- report name 9
- report number 10
- REPORT option, SPOOL commands 9
- REPORT option, SPOOLCLOSE REPORT command 16
- Report printing delay 55
- report priority 34
- report recovery level 61
- report status 33
- report title 9
- report types
 - log 6, 11
 - resumable (appendable) 6, 11
 - standard 6, 11
- reports
 - controlling printing 31
 - created by batch programs 43
 - creating, using SPOOL... commands 5
 - date 23
 - footings 23
 - headings 22
 - line length 22
 - multiformat 36
 - page length 22
 - page numbering 23
 - security 57

- reports (*continued*)
 - time 23
- resource definitions 66
- resource security level checking (RSLC) 57
- resumable reports 6, 12
- RESUME option, SPOOLCLOSE REPORT command 16
- RESUME option, SPOOLOPEN REPORT command 15
- resuming printing with the report controller 37
- RSL option, SPOOLOPEN REPORT command 6
- RSLC (resource security level checking) 57
- RSLKEY 30, 39

S

- sample programs
 - for report controller tutorial 78
 - low balance report 77
- SCS report format 24, 36
- security 57
- SEP option, SPOOLOPEN REPORT command 7
- separator pages 7, 35
- shared spool environment 7
- shift-out/shift-in control data sequences (SO/SI) 36
- SO/SI (shift-out/shift-in) control data sequences 36
- SOSI1 option, SPOOLOPEN REPORT command 7, 36
- SOSI2 option, SPOOLOPEN REPORT command 7, 36
- SPOOL commands
 - REPORT option 9
 - TOKEN option 10
- SPOOLCLOSE REPORT command 16
- SPOOLOPEN REPORT command 6, 11, 15
- SPOOLWRITE REPORT command 16
- standard reports 6, 11
- starting a job 39
- starting a printer with the report controller 37
- startup job, CICS 75
- status messages, printer 68
- status, report 33
- stopping a printer with the report controller 37
- syncpoint and the report controller 62
- syncpoints in reports 16
- SYSID 34
- system printers 53

T

- T3270 report format 24, 36
- terminal control table (TCT) 67
- TERMINAL resource definitions 68
- terminals suitable for report controller 5
- time and date on reports 7

- time, report 23
- TITLE option, SPOOL OPEN REPORT command 9
- TOKEN option, SPOOL commands 10
- TOKEN option, SPOOL CLOSE REPORT command 16
- TOKEN option, SPOOL OPEN REPORT command 15
- tokens, in report creation 10
- transient data
 - reports from queues with the report controller 41
- TYPETERM resource definitions 67

U

- userdata 9
- USERDATA option, SPOOL OPEN REPORT command 6, 9

V

- VBUILD statement 72
- viewing reports online 36
- VSE/POWER 51
- VTAM
 - configuration parameters 68
 - interpret table 74
 - logon mode table 73
 - overriding startup parameters 76

W

- writer task (CEPW transaction) 63
- writing to a report 16

Sending your comments to IBM

CICS® Transaction Server for VSE/ESA™

Report Controller Planning Guide

GC33-1941-00

If you want to send to IBM any comments you have about this book, please use one of the methods listed below. Feel free to comment on anything you regard as a specific error or omission in the subject matter, and on the clarity, organization or completeness of the book itself.

To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

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

You can send your comments to IBM in any of the following ways:

- By mail:
 - IBM UK Laboratories
 - Information Development
 - Mail Point 095
 - Hursley Park
 - Winchester, SO21 2JN
 - England
- By fax:
 - From outside the U.K., after your international access code use 44 1962 870229
 - From within the U.K., use 01962 870229
- Electronically, use the appropriate network ID:
 - IBM Mail Exchange: GBIBM2Q9 at IBMMAIL
 - IBMLink: HURSLEY(IDRCF)
 - Email: idrcf@hursley.ibm.com

Whichever method you use, ensure that you include:

- The publication number and title
- The page number or topic to which your comment applies
- Your name and address/telephone number/fax number/network ID.



Program Number: 5648-054



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

GC33-1941-00



Spine information:



CICS TS for VSE/ESA

Report Controller Planning Guide

Release 1