Contacting BMC Software

You can access the BMC Software website at http://www.bmc.com. From this website, you can obtain information about the company, its products, corporate offices, special events, and career opportunities.

United States and Canada

<table>
<thead>
<tr>
<th>Address</th>
<th>Telephone</th>
<th>Fax</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC SOFTWARE INC</td>
<td>713 918 8800</td>
<td>713 918 8000</td>
</tr>
<tr>
<td>2101 CITYWEST BLVD</td>
<td>800 841 2031</td>
<td></td>
</tr>
<tr>
<td>HOUSTON TX</td>
<td></td>
<td></td>
</tr>
<tr>
<td>77042-2827</td>
<td></td>
<td></td>
</tr>
<tr>
<td>USA</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Outside United States and Canada

<table>
<thead>
<tr>
<th>Telephone</th>
<th>Fax</th>
</tr>
</thead>
<tbody>
<tr>
<td>(01) 713 918 8800</td>
<td>(01) 713 918 8000</td>
</tr>
</tbody>
</table>

© Copyright 1999-2015 BMC Software, Inc.

BMC, BMC Software, and the BMC Software logo are the exclusive properties of BMC Software, Inc., are registered with the U.S. Patent and Trademark Office, and may be registered or pending registration in other countries. All other BMC trademarks, service marks, and logos may be registered or pending registration in the U.S. or in other countries. All other trademarks or registered trademarks are the property of their respective owners. IT Infrastructure Library® is a registered trademark of the Office of Government Commerce and is used here by BMC Software, Inc., under license from and with the permission of OGC.

ITIL® is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office, and is used here by BMC Software, Inc., under license from and with the permission of OGC.

IBM® Tivoli® Business Service Manager, IBM Tivoli Workload Scheduler, IBM Cognos, IBM InfoSphere DataStage, IBM iSeries, IBM Websphere, and AIX® are the trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

UNIX® is the registered trademark of The Open Group in the US and other countries.

Linux is the registered trademark of Linus Torvalds.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

SAP® R/2 and SAP R/3, SAP Business Objects, and SAP NetWeaver are trademarks or registered trademarks of SAP AG in Germany and in several other countries.

BMC Software considers information included in this documentation to be proprietary and confidential. Your use of this information is subject to the terms and conditions of the applicable End User License Agreement for the product and the proprietary and restricted rights notices included in this documentation.
Restricted rights legend

U.S. Government Restricted Rights to Computer Software. UNPUBLISHED -- RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES. Use, duplication, or disclosure of any data and computer software by the U.S. Government is subject to restrictions, as applicable, set forth in FAR Section 52.227-14, DFARS 252.227-7013, DFARS 252.227-7014, DFARS 252.227-7015, and DFARS 252.227-7025, as amended from time to time. Contractor/Manufacturer is BMC SOFTWARE INC, 2101 CITYWEST BLVD, HOUSTON TX 77042-2827, USA. Any contract notices should be sent to this address.

Customer support

You can obtain technical support by using the BMC Software Customer Support website or by contacting Customer Support by telephone or e-mail. To expedite your inquiry, see “Before contacting BMC.”

Support website

You can obtain technical support from BMC 24 hours a day, 7 days a week at http://www.bmc.com/support. From this website, you can:

- Read overviews about support services and programs that BMC offers
- Find the most current information about BMC products
- Search a database for issues similar to yours and possible solutions
- Order or download product documentation
- Download products and maintenance
- Report an issue or ask a question
- Subscribe to receive proactive e-mail alerts when new product notices are released
- Find worldwide BMC support center locations and contact information, including e-mail addresses, fax numbers, and telephone numbers

Support by telephone or e-mail

In the United States and Canada, if you need technical support and do not have access to the web, call 800 537 1813 or send an e-mail message to customer_support@bmc.com. (In the subject line, enter SupID:<yourSupportContractID>, such as SupID:12345). Outside the United States and Canada, contact your local support center for assistance.

Before contacting BMC

Have the following information available so that Customer Support can begin working on your issue immediately:

- Product information
  - Product name
  - Product version (release number)
  - License number and password (trial or permanent)
- Operating system and environment information
  - Machine type
  - Operating system type, version, and service pack or other maintenance level such as PUT or PTF
• System hardware configuration
• Serial numbers
• Related software (database, application, and communication) including type, version, and service pack or maintenance level

□ Sequence of events leading to the issue
□ Commands and options that you used
□ Messages received (and the time and date that you received them)
  • Product error messages
  • Messages from the operating system, such as file system full
  • Messages from related software

License key and password information

If you have questions about your license key or password, contact BMC as follows:

□ (USA or Canada) Contact the Order Services Password Team at 800 841 2031, or send an e-mail message to ContractsPasswordAdministration@bmc.com.

□ (Europe, the Middle East, and Africa) Fax your questions to EMEA Contracts Administration at +31 20 354 8702, or send an e-mail message to password@bmc.com.

□ (Asia-Pacific) Contact your BMC sales representative or your local BMC office.

Third party Software

For the provisions described in the BMC License Agreement and Order related to third party products or technologies included in the BMC Product, see https://docs.bmc.com/docs/display/workloadautomation/Control-M+Workload+Automation+Documentation and click Third-party software (TPS).
Contents

About This Guide .............................................. 15
Format of This Guide ........................................ 15
Before You Begin ............................................. 17
Conventions Used in This Guide ............................. 17
Help Information ............................................. 20
Information New to This Version ......................... 20
Related Publications ........................................ 21

Chapter 1 Introduction to Control-M .......................... 23
Overview ....................................................... 24
Preparations .................................................... 24
Main Components of Control-M ............................ 24
Online Facility Primary Options ............................. 26
Introduction to the Scheduling Definition Facility ......... 26
Introduction to the Active Environment Screen .......... 26
  Toggling Between Display of Commands and Options 36
  Filtering the Active Environment Display ................ 37
  Confirming a Job Order .................................... 39
  Checking the Log for the Job from the Active Environment Screen 41
  Checking the Sysout for the Job ......................... 42
  Holding and Deleting Active Environment Screen Job Orders 44
Return to the Scheduling Definition Facility ............... 46
  Saving Job Scheduling Definitions ...................... 46
  Selecting a Table from the Table List Screen .......... 48
  Formats of the Job List Screen ......................... 50
Exiting the Online Facility .................................. 50
Review ........................................................ 51
  Recommended Reading .................................... 52

Chapter 2 Scheduling Definition and Manual Intervention .......................... 55
Overview ....................................................... 56
Preparations .................................................... 56
Sections of the Job Scheduling Definition ................ 57
  General Parameters Section .............................. 59
  Basis Scheduling Parameters Section ................... 60
  Runtime Scheduling Parameters Section ................. 64
  Post-Processing Parameters Section ...................... 67
Conditions, Date Qualifiers, and Job Dependencies ....... 71
Chapter 1  Additional Features 209

Overview ................................................................. 210
Preparations................................................................. 210
Defining Calendars .................................................... 211
  Structure of the IOA Calendar Definition Facility .............. 216
Responding to External Events through CMEM ..................... 217
  Event Types Handled by CMEM - Available ON Statements .... 223
Creating On Spool Jobs ................................................ 224
  Defining On Spool Jobs ............................................ 225
  Bringing the Job On Spool ........................................ 229
  Additional Points About On Spool Jobs ......................... 230
Batch Utility IOACND .................................................. 231
Miscellaneous Facilities .............................................. 234
  History Jobs File ................................................... 234
  Journaling and Restoration Capability .......................... 235
  Accumulating Statistics: Statistics Facility ..................... 235
  Automatic Tape Adjustment ...................................... 236
  Simulating Job Execution and Forecasting Resource Usage .... 236
  Control-M/Restart Dataset Cleanup Utility CTRCCLN (R2) .... 237
  Reporting Facility .................................................. 237
Exit the Online Session .............................................. 238
Review ................................................................. 238
  Recommended Reading ............................................. 239

Appendix A  Sample JCLs 241

Chapter 1 ................................................................. 241
  JOB1 ................................................................. 241
  JOB2 ................................................................. 241
Chapter 2 ................................................................. 242
  JOB3 ................................................................. 242
  JOB4 ................................................................. 242
Chapter 3 ................................................................. 243
  JOB5 ................................................................. 243
Chapter 4 ................................................................. 244
  JOB1 ................................................................. 244
  JOB2 ................................................................. 244
  JOB3 ................................................................. 244
  JOB4 ................................................................. 245
  JOB5 ................................................................. 246
Chapter 5 ................................................................. 247
  JOB6 ................................................................. 247
  JOB7 ................................................................. 247
Figures

Control-M Scheduling Definition Facility - Entry Panel ........................................ 26
Control-M Scheduling Definition Facility - Entry Panel ........................................ 27
Job Scheduling Definition Screen ........................................................................ 28
Job Scheduling Definition Screen Showing Completed Fields .............................. 30
Job List Screen ................................................................................................. 31
Job List Screen ................................................................................................. 34
Confirm Window ................................................................................................. 35
Job Order Messages Screen ................................................................................ 35
Control-M Active Environment Screen ............................................................... 36
Show Screen Filter Window ................................................................................ 38
Filtered Display in the Active Environment Screen .............................................. 39
Confirmation Window ........................................................................................... 40
Log Option in the Active Environment Screen ...................................................... 41
Control-M Log Screen ......................................................................................... 41
Job Order Execution History Screen ...................................................................... 43
Sysout Viewing Screen ......................................................................................... 44
Delete Option in the Active Environment Screen .................................................. 44
Job List Screen Exit Option Window ................................................................. 47
Control-M Table List Screen ............................................................................... 48
Job List Screen ................................................................................................. 49
Control-M Scheduling Definition Facility - Entry Panel ........................................ 57
Job Scheduling Definition Screen ........................................................................ 58
Option P (PLN) ................................................................................................. 73
Job Scheduling Plan Window ................................................................................ 73
Job Scheduling Screen ......................................................................................... 74
Job Scheduling Definition for job IDJOB4 .......................................................... 76
Job List Screen ................................................................................................. 77
Job List Screen ................................................................................................. 78
Why Screen .......................................................................................................... 80
Confirmation Window for Adding Condition in Why Screen .............................. 81
IOA Conditions/Resources Screen ....................................................................... 82
Add Resources Window in IOA Conditions/Resources Screen ............................ 83
Zoom Screen ........................................................................................................ 86
Job Scheduling Definition .................................................................................... 97
Confirm Restart Window ....................................................................................... 100
Restart Step List Window ..................................................................................... 102
SMART Table Entity Scheduling Definition ......................................................... 108
Basic Scheduling Parameter Section ................................................................ 111
Post-Processing Parameter Section ..................................................................... 113
Runtime Scheduling Parameter Section ............................................................. 117
Job Scheduling Definition for IDJOB4 ............................................. 123
Job List Screen for a SMART Table Entity .................................... 126
Sysout for Job IDJOB6 ................................................................. 136
AutoEdit Simulation Panel ........................................................... 149
AutoEdit Simulation Output .......................................................... 153
Show Screen Filter Window .......................................................... 173
Active Environment Screen Display Filters Window ......................... 176
Display Type A (All Fields) .......................................................... 178
Sysout Viewing Screen ............................................................... 187
Table Screen .............................................................................. 192
IOA Calendar Definition Facility - Entry Panel .................................. 212
IOA Calendar Definition Screen ..................................................... 213
Year List Screen .......................................................................... 215
CMEM Rule Definition Facility – Entry Panel .................................... 217
CMEM Rule Definition Screen ....................................................... 219
CMEM Rule List Screen ............................................................... 221
CMEM Exit Option Window .......................................................... 222
Control-M CMEM Order Messages Screen ...................................... 222
# Tables

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job Scheduling Definition Screen Field Descriptions</td>
<td>28</td>
</tr>
<tr>
<td>Values for Job Scheduling Definition Screen Fields</td>
<td>29</td>
</tr>
<tr>
<td>Commands to Display Job List Screen Formats</td>
<td>50</td>
</tr>
<tr>
<td>Job Scheduling Definition Screen General Parameters</td>
<td>60</td>
</tr>
<tr>
<td>Basic Job Scheduling Definition Parameters</td>
<td>61</td>
</tr>
<tr>
<td>Runtime Scheduling Parameters</td>
<td>64</td>
</tr>
<tr>
<td>Post-Processing Parameters ON and DO Statements</td>
<td>68</td>
</tr>
<tr>
<td>PF Keys and Screen Navigation Commands for Job Scheduling Plan Screens</td>
<td>74</td>
</tr>
<tr>
<td>DO IFRERUN: Selected FROM Keyword values</td>
<td>96</td>
</tr>
<tr>
<td>Valid Values For the ON TABLE-END Statement</td>
<td>112</td>
</tr>
<tr>
<td>Non-Date AutoEdit System Variables</td>
<td>140</td>
</tr>
<tr>
<td>Other AutoEdit Functions</td>
<td>144</td>
</tr>
<tr>
<td>Pointing to External Members</td>
<td>154</td>
</tr>
<tr>
<td>Additional AutoEdit Control Statements</td>
<td>156</td>
</tr>
<tr>
<td>Valid AutoEdit Operators</td>
<td>158</td>
</tr>
<tr>
<td>IDJOB8 SET VAR Statement Owners</td>
<td>162</td>
</tr>
<tr>
<td>Anticipated Results of Job</td>
<td>163</td>
</tr>
<tr>
<td>Job Dependency Fields</td>
<td>184</td>
</tr>
<tr>
<td>Job Information</td>
<td>186</td>
</tr>
<tr>
<td>Job Execution Information</td>
<td>186</td>
</tr>
<tr>
<td>Format of the Second Date Control Record</td>
<td>202</td>
</tr>
<tr>
<td>Non-Routine Job Ordering Methods</td>
<td>207</td>
</tr>
<tr>
<td>ON Statements Available to CMEM</td>
<td>223</td>
</tr>
<tr>
<td>CMEM Action Parameters</td>
<td>224</td>
</tr>
<tr>
<td>Job To Be Brought On Spool - Values for Exercise</td>
<td>225</td>
</tr>
<tr>
<td>CMEM Rule - Values for Exercise</td>
<td>225</td>
</tr>
<tr>
<td>Job Scheduling Definition - Values for Exercise</td>
<td>226</td>
</tr>
<tr>
<td>Qualifiers and Entity Types</td>
<td>233</td>
</tr>
<tr>
<td>Report Types</td>
<td>237</td>
</tr>
</tbody>
</table>
About This Guide

This guide describes how to begin using Control-M to actively control jobs in the production environment.

Format of This Guide

Each chapter in this guide contains

■ an introduction to the chapter and the material to be covered

■ a review of the preparations necessary for performance of the exercises in the chapter

■ an instruction set designed to explain the subject matter of the chapter

■ a review and summary of the principal lessons in the chapter

■ a list of recommended source material on the topics that were covered in the chapter

The exercises in this guide are designed to be progressive, with each exercise building on what was learned in a previous exercise. The chapters are organized the same way, with each building sequentially upon earlier material. A brief explanation of each chapter is shown below.

Chapter 1–Introduction to Control-M

This chapter introduces you to the Control-M Scheduling Definition facility and the Active Environment. It shows you the Scheduling Definition facility screens, the Active Environment screen, and some of the screens accessible from the Active Environment screen. In this chapter you will create two simple job scheduling definitions, and a dependency between them. You will then order these jobs, and observe and intervene in their processing.
Chapter 2–Scheduling Definition and Manual Intervention

This chapter provides a more detailed look at the parameters of the job scheduling definition, and introduces additional functionality in the Active Environment.

Chapter 3– Restarts under Control-M/Restart

This chapter shows you how to define Control-M/Restart parameters in the job scheduling definition, and how to intervene in the restart process when desired.

Chapter 4–SMART Table Scheduling

This chapter introduces you to SMART Table scheduling, and shows you how to define jobs in SMART Table.

Chapter 5–AutoEdit and JCL

This chapter introduces you to the AutoEdit facility, and has you use AutoEdit terms in both the JCL and in the job scheduling definition. In this chapter you will learn about and use system variables, local user defined variables, global variables, AutoEdit functions, and control statements.

Chapter 6–Navigating The Active Environment

This chapter takes you through features, facilities, and screens of the Active Environment that were not covered in the earlier chapters.

Chapter 7–Job Ordering and New Day Processing

This chapter instructs you in various ways of ordering jobs, and provides a detailed look at the use of the CTMJOB utility and User Daily jobs.

Chapter 8–Loose Ends

As the final chapter in this guide, this chapter describes various Control-M and IOA facilities that were not covered in previous chapters. Included are the IOA Calendar facility, CMEM Event Management facility, and several useful utilities.

Appendix A–Sample JCL

This appendix lists all preparations that are necessary before you can perform the exercises in each chapter of this guide. It provides sample JCLs for those preparations, many of which involve the creation of JCL members.
Before You Begin

Before proceeding with the Getting Started Guide, you should read the introductory chapter of the *Control-M for z/OS User Guide*.

You should perform the exercises in this guide in a test database.

It is entirely possible that other users will perform these exercises at the same time as you. To ensure that the someone else’s exercises do not impact your exercises, you should use your unique user identifier as a prefix to all named components. In the text and screen examples in this guide, *ID* or *id* are used to indicate where your unique user identifier should be entered.

While it is not absolutely essential that you use your user ID as the unique prefix, it is absolutely essential that you consistently use the same brief, unique prefix of some sort if others will perform these exercises within the same time frame as you.

Conventions Used in This Guide

Notational conventions that may be used in this guide are explained below.

**Standard Keyboard Keys**

Keys that appear on the standard keyboard are identified in boldface, for example, `Enter`, `Shift`, `Ctrl+S` (a key combination), or `Ctrl S` (a key sequence).

---

**WARNING**

The commands, instructions, procedures, and syntax illustrated in this guide presume that the keyboards at your site are mapped in accordance with the EBCDIC character set. Certain special characters are referred to in this documentation, and you must ensure that your keyboard enables you to generate accurate EBCDIC hex codes. This is particularly true on keyboards that have been adapted to show local or national symbols. You should verify that

- `$` is mapped to `x'5B'
- `#` is mapped to `x'7B'
- `@` is mapped to `x'7C'

If you have any questions about whether your keyboard is properly mapped, contact your system administrator.
Preconfigured PFKeys

Many commands are preconfigured to specific keys or key combinations. This is particularly true with regard to numbered PF keys, or pairs of numbered PFKeys. For example, the END command is preconfigured to, and indicated as, PF03/PF15. To execute the END command, press either the PF03 key or the PF15 key.

Instructions to enter commands may include

- only the name of the command, such as, enter the END command
- only the PF keys, such as, press PF03/PF15
- or both, such as, press PF03/PF15, or enter the END command

Command Lines and Option Fields

Most screens contain a command line, which is primarily used to identify a single field where commands, or options, or both, are to be entered. These fields are usually designated COMMAND, but they are occasionally identified as COMMAND/OPT or COMMAND/OPTION.

Option field headings appear in many screens. These headings sometimes appear in the screen examples as OPTION, or OPT, or O.

Names of Commands, Fields, Files, Functions, Jobs, Libraries, Members, Missions, Options, Parameters, Reports, Subparameters, and Users

The names of commands, fields, functions, jobs, libraries, members, missions, options, parameters, reports, subparameters, users, and most files, are shown in standard UPPERCASE font.

User Entries

In situations where you are instructed to enter characters using the keyboard, the specific characters to be entered are shown in this **UPPERCASE BOLD** text, for example, type EXITNAME.

Syntax statements

In syntax, the following additional conventions apply:

- A vertical bar ( | ) separating items indicates that you must choose one item. In the following example, you would choose a, b, or c:

  a | b | c
- An ellipsis (….) indicates that you can repeat the preceding item or items as many times as necessary.

- Square brackets ([ ]) around an item indicate that the item is optional. If square brackets ([ ]) are around a group of items, this indicates that the item is optional, and you may choose to implement any single item in the group. Square brackets can open ([ ]) and close (]) on the same line of text, or may begin on one line of text and end, with the choices being stacked, one or more lines later.

- Braces ({ }) around a group of items indicates that the item is mandatory, and you must choose to implement a single item in the group. Braces can open ({ }) and close (}) on the same line of text, or may begin on one line of text and end, with the choices being stacked, one or more lines later.

**Screen Characters**

All syntax, operating system terms, and literal examples are presented in this typeface. This includes JCL calls, code examples, control statements, and system messages. Examples of this are:

- calls, such as 

```
CALL 'CBLTDLI'
```

- code examples, such as 

```
FOR TABLE owner.name USE option, . . . ;
```

- control statements, such as 

```
//PRDSYSIN DD * USERLOAD PRD(2) PRINT
```

- system messages, both stand-alone, such as 

```
You are not logged on to database database_name,
```

and those embedded in text, such as the message 

```
You are not logged on to database database_name, are displayed on
```

the screen.
Variables

Variables are identified with italic text. Examples of this are:

- In syntax or message text, such as
  Specify database database_name
- In regular text, such as
  replace database database_name1 with database database_name2 for the current session
- In a version number, such as
  EXTENDED BUFFER MANAGER for IMS 4.1.xx

Special elements

This book includes special elements called notes and warnings:

--- NOTE ---

Notes provide additional information about the current subject.

--- WARNING ---

Warnings alert you to situations that can cause problems, such as loss of data, if you do not follow instructions carefully.

Help Information

Help information is readily available while using Control-M. Help screens can be accessed by using the HELP command by typing HELP in the COMMAND line or by pressing PF01/PF13 from any Control-M screen.

Use standard scrolling conventions (PF08/PF20 and PF07/PF19) to scroll forward and backward through the help information. To return to the original screen, use the END command (PF03/PF15).

Information New to This Version

Additional information that is new to this version is described in Appendix A of the INCONTROL for z/OS Upgrade Guide and What’s New section of the INCONTROL for z/OS Release Notes.
Related Publications

Control-M for z/OS User Guide

Detailed explanation of the base Control-M for z/OS product, an system that manages and automates the setup, scheduling and execution of jobs in the z/OS data center

INCONTROL for z/OS Administrator Guide

Information for system administrators about customizing and maintaining INCONTROL products

INCONTROL for z/OS Installation Guide

Step-by-step guide to installing INCONTROL™ products using the INCONTROL™ Installation and Customization Engine (ICE) application

INCONTROL for z/OS Messages Manual

Comprehensive listing and explanation of all INCONTROL and IOA messages and codes

INCONTROL for z/OS Security Guide

Step-by-step guide to implementing security in INCONTROL products

INCONTROL for z/OS Utilities Guide

Describes utilities designed to perform specific administrative tasks that are available to INCONTROL products
# Introduction to Control-M

This chapter includes the following topics:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overview</td>
<td>24</td>
</tr>
<tr>
<td>Preparations</td>
<td>24</td>
</tr>
<tr>
<td>Main Components of Control-M</td>
<td>24</td>
</tr>
<tr>
<td>Online Facility Primary Options</td>
<td>26</td>
</tr>
<tr>
<td>Introduction to the Scheduling Definition Facility</td>
<td>26</td>
</tr>
<tr>
<td>Toggling Between Display of Commands and Options</td>
<td>36</td>
</tr>
<tr>
<td>Filtering the Active Environment Display</td>
<td>37</td>
</tr>
<tr>
<td>Confirming a Job Order</td>
<td>39</td>
</tr>
<tr>
<td>Checking the Log for the Job from the Active Environment Screen</td>
<td>41</td>
</tr>
<tr>
<td>Checking the Sysout for the Job</td>
<td>42</td>
</tr>
<tr>
<td>Holding and Deleting Active Environment Screen Job Orders</td>
<td>44</td>
</tr>
<tr>
<td>Return to the Scheduling Definition Facility</td>
<td>46</td>
</tr>
<tr>
<td>Saving Job Scheduling Definitions</td>
<td>46</td>
</tr>
<tr>
<td>Selecting a Table from the Table List Screen</td>
<td>48</td>
</tr>
<tr>
<td>Formats of the Job List Screen</td>
<td>50</td>
</tr>
<tr>
<td>Exiting the Online Facility</td>
<td>50</td>
</tr>
<tr>
<td>Review</td>
<td>51</td>
</tr>
<tr>
<td>Recommended Reading</td>
<td>52</td>
</tr>
</tbody>
</table>
Overview

Control-M is an automated production control and scheduling system that manages and automates the setup, scheduling and execution of jobs in the z/OS data center.

The Control-M for z/OS Getting Started Guide provides you with an introduction to the functionality of Control-M, and teaches you how to use Control-M, one step at a time, with each step building upon what was learned in the previous step. Of necessity, the explanations of topics in this book are not comprehensive. For comprehensive explanations of topics, you should refer to the relevant guide, such as the Control-M for z/OS User Guide, INCONTROL for z/OS Administrator Guide, and so on.

The first chapter of the Control-M for z/OS User Guide provides an introduction to Control-M. You might find it useful to read that chapter before using this guide; however, you can still easily use this guide without having read that chapter in the User Guide.

This guide begins with a discussion of the main components of Control-M:

- Scheduling Definition facility
- Tracking and Control facility: Active Environment screen

Before you are introduced to the main components of Control-M, you should prepare the jobs that you will need in order to perform the exercises in this chapter.

Preparations

Create the JCL for two trivial jobs, identified throughout this guide as IDJOB1 and IDJOB2. Define the jobs to execute IEFBR14. For purposes of this guide, identify the JCL library as CTM.TEST.JCL.

Main Components of Control-M

Control-M automates the setup, scheduling, and execution of jobs in the z/OS environment. This is essentially a two step process:

1. You must tell Control-M how you want job processing handled.

   Job scheduling and processing instructions are provided through the Scheduling Definition facility, a main component of Control-M.
You specify these instructions in job scheduling definitions that you define in the Job Scheduling Definition screen. Each job scheduling definition must be defined only once; it can then be used as often as needed. Job scheduling definitions are stored in tables, or members, in a scheduling library.

2. Using the instructions you provided in the job scheduling definitions, Control-M automatically tracks and controls the setup, scheduling, and execution of the job. Control-M provides you with constant feedback on job processing, and enables you to manually intervene in the processing of jobs, through a series of screens.

“Tracking and Control facility” is a general term that applies to the screens that provide feedback on job processing, and enable manual intervention. The main screen in this facility is the Active Environment screen, which provides status and other information for currently scheduled and recently completed jobs, and accepts your commands and instructions. The Active Environment screen also acts as a gateway to other screens in the Tracking and Control facility.

You will take a first look at these components by doing the following in your first set of exercises:

■ You will use the Scheduling Definition facility to create job scheduling definitions for the two trivial jobs you just defined.

— You will make submission of the first job dependent on a manual intervention performed in the Active Environment screen.

— You will make submission of the second job dependent on successful completion of the first job.

■ You will order the jobs.

■ You will observe the results in the Active Environments screen. Along the way, you will become familiar with several commands, options and windows in the Active Environment screen. You will also become familiar with the Control-M Log screen.

To do this, enter the IOA Online facility, take a look at the main components of Control-M, and perform the exercises.
Online Facility Primary Options

The primary options of the Online facility are displayed when you enter the IOA Primary Option menu. Several of these options relate to Control-M. Through most of this guide, you will concentrate on two options:

- Option 2, which provides access to the Scheduling Definition facility
- Option 3, which provides access to the Active Environment screen, the main screen of the Tracking and Control facility.

1. Request Option 2 from the IOA Primary Option menu, to access the Scheduling Definition facility.

The Scheduling Definition Facility entry panel is displayed.

**Figure 1  Control-M Scheduling Definition Facility - Entry Panel**

<table>
<thead>
<tr>
<th>COMMAND ====&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPECIFY LIBRARY, TABLE, JOB</td>
</tr>
<tr>
<td>LIBRARY ===&gt; CTM.TEST.SCHEDULE</td>
</tr>
<tr>
<td>TABLE ===&gt; (Blank for table selection list)</td>
</tr>
<tr>
<td>JOB ===&gt; (Blank for job selection list)</td>
</tr>
<tr>
<td>SMART TABLE ===&gt; (Y - for new SMART Table)</td>
</tr>
<tr>
<td>SHOW JOB DOCUMENTATION ===&gt; N (Y/N)</td>
</tr>
<tr>
<td>AUTO-SAVE DOCUMENTATION ===&gt; N (Y/N)</td>
</tr>
</tbody>
</table>

You are about to create a Control-M job scheduling definition for each of the jobs, IDJOB1 and IDJOB2, that you just defined. The JCL member name of the job should be used as the name of its corresponding job scheduling definition.
Job scheduling definitions are stored in tables (members) in a scheduling library. A default scheduling library name automatically appears in the entry panel. Ensure that you use a test library or your own work library. You should not use the JCL library.

For purposes of illustration in this guide, the scheduling library is identified as CTM.TEST.SCHEDULE. In this library, you will be creating a table called IDGS1. Use your unique identifier as the first characters of the table name.

You can now begin defining the job scheduling definition for IDJOB1.

2 Fill in the LIBRARY, TABLE, and JOB name fields in the entry panel. The request appears similar to Figure 2.

Figure 2  Control-M Scheduling Definition Facility - Entry Panel

After entering the entry panel request, the Job Scheduling Definition screen is displayed. A full job scheduling definition, which contains more lines than your screen can display, is illustrated in Figure 3 on page 28.

Note that the first line in the screen, which is displayed above the COMMAND line, contains the same information about the job that you provided in the entry panel.
The Job Scheduling Definition screen is divided into sections, with each section separated from another by a line of equal signs (= =). Later, you will learn more about these sections and the fields they contain. The following table describes the fields of the Job Scheduling Definition screen that you will complete during these exercises.

Table 1  Job Scheduling Definition Screen Field Descriptions  (part 1 of 2)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEMNAME</td>
<td>Name of the member that contains the JCL of the job.</td>
</tr>
<tr>
<td>MEMLIB</td>
<td>Name of the library that contains the JCL of the job.</td>
</tr>
<tr>
<td>GROUP</td>
<td>Name of the group to which the job belongs.</td>
</tr>
<tr>
<td>DESC</td>
<td>Free text description of the job.</td>
</tr>
</tbody>
</table>
Table 1  Job Scheduling Definition Screen Field Descriptions (part 2 of 2)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DAYS</td>
<td>Days of the month to schedule the job.</td>
</tr>
<tr>
<td>CONFIRM</td>
<td>Yes/No indicator (Y/N) specifying whether manual confirmation is required before the job can be submitted.</td>
</tr>
<tr>
<td>OUT</td>
<td>Prerequisite conditions to be added and/or deleted when the job ends OK</td>
</tr>
</tbody>
</table>

3  Fill in the fields as indicated in the following table.

Table 2  Values for Job Scheduling Definition Screen Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Value to Enter (in Bold), and Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEMNAME</td>
<td>IDJOB1. This value, taken from the entry panel, already appears, because you specified the JCL member name in the JOB field of the entry panel.</td>
</tr>
<tr>
<td>MEMLIB</td>
<td>Name of the library containing JCL member idJOB1. For purposes of illustration, you should use CTM.TEST.JCL</td>
</tr>
<tr>
<td>GROUP</td>
<td>IDGRP1. This name will later be used to filter the display of jobs in the Active Environment screen by Group name.</td>
</tr>
<tr>
<td>DESC</td>
<td>THIS JOB MUST BE RUN BEFORE IDJOB2 (to remind you what this job does).</td>
</tr>
<tr>
<td>DAYS</td>
<td>ALL (Schedule the job every day.)</td>
</tr>
<tr>
<td>CONFIRM</td>
<td>Y (Yes - you want manual confirmation for the job.)</td>
</tr>
<tr>
<td>OUT</td>
<td>IDJOB1-ENDED-OK ODAT + (After specifying IDJOB1-ENDED-OK, press Tab to get to the field in which to enter ODAT.)</td>
</tr>
</tbody>
</table>

The Job Scheduling Definition screen is displayed as shown in Figure 4.
### Figure 4  Job Scheduling Definition Screen Showing Completed Fields

<table>
<thead>
<tr>
<th>JOB: IDJOB1</th>
<th>LIB: CTM.TEST.SCHEDULE</th>
<th>TABLE: IDGS1</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEMNAME</td>
<td>IDJOB1</td>
<td>MEMLIB: CTM.TEST.JCL</td>
</tr>
<tr>
<td>OWNER</td>
<td>ID</td>
<td>TASKTYPE: JOB</td>
</tr>
<tr>
<td>APPL</td>
<td></td>
<td>PREVENT-NCT2</td>
</tr>
<tr>
<td>DESC</td>
<td>THIS JOB MUST BE RUN BEFORE IDJOB2</td>
<td></td>
</tr>
<tr>
<td>WORKLIB</td>
<td></td>
<td>JOB: IDJOB1</td>
</tr>
<tr>
<td>SCHENV</td>
<td></td>
<td>ACTION: JOB</td>
</tr>
<tr>
<td>SET VAR</td>
<td></td>
<td>DFLT: N</td>
</tr>
<tr>
<td>CTB</td>
<td>STEP: AT</td>
<td>TYPE: JOB</td>
</tr>
<tr>
<td>DOCMEM</td>
<td>IDJOB1</td>
<td>DOCLIB: CTM.TEST.JCL</td>
</tr>
<tr>
<td>DAYS</td>
<td>ALL</td>
<td>DCAL: AND/OR</td>
</tr>
<tr>
<td>WDDAYS</td>
<td></td>
<td>WCAL: AND/OR</td>
</tr>
<tr>
<td>DATES</td>
<td></td>
<td>CONFCAL: RETRO N MAXWAIT 00 D-CAT</td>
</tr>
<tr>
<td>MINIMUM</td>
<td></td>
<td>PDS:</td>
</tr>
<tr>
<td>DEFINITION</td>
<td>ACTIVE FROM</td>
<td>UNTIL:</td>
</tr>
<tr>
<td>IN</td>
<td>CONTROL RESOURCE</td>
<td>PIPE:</td>
</tr>
<tr>
<td>FROM TIME</td>
<td>+ DAYS</td>
<td>UNTIL TIME</td>
</tr>
<tr>
<td>DUE OUT TIME</td>
<td>+ DAYS</td>
<td>PRIORITY: 00</td>
</tr>
<tr>
<td>TIME ZONE</td>
<td></td>
<td>SAC: CONFIRM Y</td>
</tr>
<tr>
<td>OUT</td>
<td>IDJOB1-ENDED-OK</td>
<td>ODAT: +</td>
</tr>
<tr>
<td>AUTO-ARCHIVE</td>
<td>SYSDB: MAXDAYS MAXRUNS</td>
<td></td>
</tr>
<tr>
<td>RETENTION</td>
<td># OF DAYS TO KEEP: # OF GENERATIONS TO KEEP</td>
<td></td>
</tr>
<tr>
<td>SYSDOUT OP</td>
<td>(C,D,F,N,R)</td>
<td>FROM:</td>
</tr>
<tr>
<td>MAXRUN</td>
<td>RERUNMEM</td>
<td>INTERVAL: FROM</td>
</tr>
<tr>
<td>STEP RANGE</td>
<td>FR (PGM.PROC)</td>
<td>TO:</td>
</tr>
<tr>
<td>ON PGMST</td>
<td>PROCST: CODES</td>
<td>A/O:</td>
</tr>
<tr>
<td>DO</td>
<td>SHOUT WHEN: TIME + DAYS TO URGN</td>
<td></td>
</tr>
<tr>
<td>MS</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

--- END OF SCHEDULING PARAMETERS ---

**COMMANDS:** EDIT, DOC, PLAN, JOBSTAT  11.17.00
4 Exit the Job Scheduling Definition screen. The Job List screen is displayed.

**Figure 5  Job List Screen**

<table>
<thead>
<tr>
<th>JOB LIST</th>
<th>LIB: C.TM.TEST.SCHEDULE</th>
<th>TABLE: IDGS1</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMAND</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OPT</td>
<td>NAME</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td></td>
<td>IDJOB1</td>
<td>THIS JOB MUST BE RUN BEFORE JOB2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The format of the Job List screen may be different than the format indicated in the screen example shown above. The different formats of the Job List screen will be discussed later in this chapter. The following procedure will ensure that you are working with the desired format.

5 Enter **DESC** in the COMMAND field of the Job List screen.

The Job List screen has the same format as the screen example shown above.

The Job List screen contains the list of jobs in the table. In this case, since you created a new table and only one job, there is only one job (IDJOB1) in the list. Notice that the description you provided in the job scheduling definition appears next to the job name in the Job List screen. Providing useful identifying information in the DESC field of the job scheduling definition can help you with job identification in the Job List screen.

You will now create the job scheduling definition for **IDJOB2**. You could do this much the same way that you created IDJOB1, that is, by returning to the entry panel and there specifying the name IDJOB2. But instead, you will create the job from the Job List screen.

6 Type **I** (Insert) in the OPT (Option) field to the left of the entry for IDJOB1, and press **Enter**.
The Job Scheduling Definition screen is displayed. When you create a new job using the I (Insert) option in the Job List screen, the newly created job contains most of the values defined in the job by which you specified the I option. Only the MEMNAME, DOCMEM, and DESCRIPTION values are not carried over; instead, these fields are left blank.

7 Type JOB2 in the MEMNAME field. The DOCMEM field will automatically be filled in with the same value. Leave the DESC field blank.
In the IN field, define the same value you had defined as the OUT value in JOB1 (IDJOB1-ENDED-OK ODAT). You should press Tab before typing ODAT, as you did in IDJOB1. The + from the OUT value is not specified. After defining the IN value, delete the Y from the CONFIRM field.

Change the plus sign (+) to a minus sign (-) in the OUT statement.

The IN and the OUT fields are used for specifying conditions. These conditions can be used to create job dependencies.

In this exercise, you used condition IDJOB1-ENDED-OK to ensure that IDJOB2 would not be submitted until IDJOB1 successfully completed:

- You made IDJOB1-ENDED-OK an IN condition for IDJOB2, which means that IDJOB2 cannot be run until this condition exists, that is, until IDJOB1-ENDED-OK has been created.
- You made IDJOB1-ENDED-OK an OUT condition of IDJOB1, specifying a plus sign in the appropriate place in the condition definition, which means that the condition will only be created after IDJOB1 completed successfully.

Therefore, IDJOB1 must complete successfully before IDJOB2 can be submitted.
Specifying the same condition as an OUT condition with a minus sign in IDJOB2 removes the condition once IDJOB2 has successfully ended. This ensures that an old occurrence of the condition will not satisfy the IN criteria.

Note also that every condition has an associated date value. This can be an explicitly defined date, such as 0909, or a valid keyword value that represents a date. ODAT is a keyword value that defaults to the current original scheduling date of the job. The original scheduling date, and other date concepts, are discussed in Chapter 5, “AutoEdit and JCL.”

10 Exit the job scheduling definition for IDJOB2. The Job List screen is displayed. IDJOB2 now appears in the list, but since you did not define a description for it, no description appears for the job.

Figure 6  Job List Screen

<table>
<thead>
<tr>
<th>OPT</th>
<th>NAME</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>IDJOB1</td>
<td>THIS JOB MUST BE RUN BEFORE IDJOB2</td>
</tr>
<tr>
<td></td>
<td>IDJOB2</td>
<td></td>
</tr>
</tbody>
</table>

Notice the options at the bottom of the Job List screen. You will now use one of them, the O (Order) option, to order these two jobs.

11 Request option O (Order) in the OPT field for both IDJOB1 and IDJOB2.

12 If a confirm window is displayed, confirm the job order by entering Y in the CONFIRM field.
For each job ordered, the Job Order Messages screen with messages similar to the following will be displayed:

**Figure 7 Confirm Window**

<table>
<thead>
<tr>
<th>JOB LIST</th>
<th>LIB: CTM.TEST.SCHEDULE</th>
<th>TABLE: IDGS1</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMAND</td>
<td></td>
<td>SCROLL====&gt; CRSR</td>
</tr>
<tr>
<td>OPT NAME</td>
<td>CONFIRM Y</td>
<td>ODATE 020201</td>
</tr>
<tr>
<td>0 IDJOB1</td>
<td>ASK FOR EACH ONE Y</td>
<td></td>
</tr>
<tr>
<td>0 IDJOB2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>=========</td>
<td></td>
<td>--------------</td>
</tr>
<tr>
<td>NO MORE JOBS IN TABLE</td>
<td>&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt; ====</td>
<td></td>
</tr>
</tbody>
</table>
**Introduction to the Active Environment Screen**

The Active Environment screen is your main interface to the Tracking and Control facility. It lists currently scheduled jobs. These jobs may or may not have already been submitted, and may or may not have already completed their executions.

**Figure 9  Control-M Active Environment Screen**

<table>
<thead>
<tr>
<th>Filter:</th>
<th>Control-M  Active  Environment ------ UP    &lt;D&gt;   (3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMAND ===&gt;</td>
<td>SCROLL ==&gt; CRSR</td>
</tr>
<tr>
<td>O Name</td>
<td>Owner</td>
</tr>
<tr>
<td>CICSPROD M22</td>
<td>020201</td>
</tr>
<tr>
<td>CICSTEST M22</td>
<td>020201</td>
</tr>
<tr>
<td>BRIVPCC IVP</td>
<td>020201</td>
</tr>
<tr>
<td>BRCC0001 IVP</td>
<td>020201</td>
</tr>
<tr>
<td>BRCC0002 IVP</td>
<td>020201</td>
</tr>
<tr>
<td>BRCC0003 IVP</td>
<td>020201</td>
</tr>
<tr>
<td>BRCC1ND IVP</td>
<td>020201</td>
</tr>
<tr>
<td>BRUPDT02 IVP</td>
<td>020201</td>
</tr>
<tr>
<td>BRREPO01 IVP</td>
<td>020201</td>
</tr>
<tr>
<td>BRREPO02 IVP</td>
<td>020201</td>
</tr>
<tr>
<td>BRIVPCE IVP</td>
<td>020201</td>
</tr>
<tr>
<td>CRREPO01 IVP</td>
<td>020201</td>
</tr>
<tr>
<td>INTRO004 M22</td>
<td>020201</td>
</tr>
<tr>
<td>INTRO002 M22</td>
<td>020201</td>
</tr>
<tr>
<td>INTRO003 M22</td>
<td>020201</td>
</tr>
<tr>
<td>INTRO004 M22</td>
<td>020201</td>
</tr>
</tbody>
</table>

Commands: Opt Display Show History RBal REFresh Auto Jobstat SHPF Note Table

Opt command toggles between Commands and Options display  15.15.48

This screen is most probably displaying a number of jobs from your site that are not relevant to these exercises. Soon, you will filter the display so that only the relevant jobs are displayed. But first, look at the list of available commands or options at the bottom of the screen.

**Toggling Between Display of Commands and Options**

The bottom two lines of the Active Environment screen list either the commands or options available in the screen. Upon entry to the screen, the list of commands is displayed.

**Commands: Opt Display Show History RBal REFresh Auto Jobstat SHPF Note Table
Opt command toggles between Commands and Options display  15.15.**
The first command listed is `OPt`, and the bottom line actually explains that command `OPt` can be used to toggle between the command display and the option display.

15 Enter `OPT` in the COMMAND field to toggle to the option display. The list of available options is displayed at the bottom of the screen.

```
     N Net D Del F Free S Stat G Group U Undelete J JCL Edit C Confirm 15.46.0
```

16 Again, enter `OPT` in the COMMAND field to redisplay the list of available commands at the bottom of the screen.

```
Commands: OPt Display Show History RBal Refresh Auto Jobstat SHPF Note Table
           OPt command toggles between Commands and Options display 15.15.48
```

One or the other of the two lists will be particularly relevant for different steps, and you will be instructed, as necessary, to display the appropriate screen. Now that the Command display appears, notice the SHOW command. You will be using the SHOW command in the next set of steps.

### Filtering the Active Environment Display

As mentioned above, when you enter the active environment the screen is probably displaying a number of jobs from your site that are not relevant to these exercises. The following steps show how you can filter the display.

17 Enter `SHOW` in the COMMAND field. The Show Screen Filter window is displayed.
Figure 10  Show Screen Filter Window

The Show Screen Filter window enables you to filter the display based on certain fields and values. The following steps explain what you need to know in order to create the filter for use in these exercises. Filtering is explained in greater detail in “A Closer Look At Filters” on page 171.

18 Specify that you want to create and save a filter called IDGS by entering **IDGS** in the Filter field, and **Y** in the Save field. Specify a description for the filter by typing **GS-EXERCISES** in the Desc field.

19 Define the filtering criteria by specifying **IDGRP** in the Group field. The filter appears as shown below:

```
+------------------- Show Screen Filter -----------(3.SHOW)+
| Filter             | Save  Y (Y/N) Desc: GS-EXERCISES |
| Memname            |       |                                   |
| Group              |       |                                   |
| Sch Lib            |       |                                   |
| Memname            |       |                                   |
| Group              |       |                                   |
| Sch Lib            |       |                                   |
| Memname            |       |                                   |
| Group              |       |                                   |
| Sch Lib            |       |                                   |
| Memname            |       |                                   |
| Group              |       |                                   |
| Sch Lib            |       |                                   |
| Memname            |       |                                   |
| Group              |       |                                   |
| Sch Lib            |       |                                   |
| Memname            |       |                                   |
| Group              |       |                                   |
| Sch Lib            |       |                                   |
| Memname            |       |                                   |
| Group              |       |                                   |
| Sch Lib            |       |                                   |
```

The Group field works on a prefix basis. By assigning the IDGRP prefix value, the filter will only display jobs whose Group name begins with IDGRP. This means only the current jobs in group IDGRP1. In later chapters, it will also display jobs belonging to groups idGRP2, idGRP3, and so on.

20 Press **Enter** to exit the window. The Active Environment now displays only IDJOB1 and IDJOB2.
Confirming a Job Order

When you examine the Active Environment screen you will see the following:

- A data line for each job order displays information about the job. This information includes the job name, job owner and the job order date (ODATE), as well as other information.

- The status of each job order is displayed in the Status field for each data line.

When you examine the status of these jobs you will see the following:

- The status of IDJOB1 is **Wait Confirmation (for schedule)**.

  **Wait Confirmation (for schedule)** status is displayed because you defined this job to require job confirmation.

- The status of IDJOB2 is **Wait Schedule**.

  **Wait Schedule** status is displayed because IDJOB2 cannot be scheduled, or submitted, until its prerequisite IN condition, **IDJOB1-ENDED-OK**, has been satisfied, and this can only occur after IDJOB1 successfully ends.

To confirm that JOB1 should be submitted you need to enter the appropriate option in the **O (option)** field by the job name.

21. Enter **OPT** in the COMMAND field to toggle the command/option display.

   The list of available options is displayed at the bottom of the screen.

   

   You can now see that to confirm the scheduling of a job, you should enter line command **C (Confirm)**.

22. Enter **C** in the **O (option)** field, to the left of IDJOB1.
If a confirmation window is displayed, that is, if double confirmation is required, specify Y (Yes) in the confirmation window.

Figure 12 Confirmation Window

The Status field display changes for each of the two jobs, as they are executed.

Keep pressing Enter in the Active Environment screen.

If you press Enter quickly enough, you will note the following progression:

- IDJOB1 changes status to Wait Submission.
- IDJOB1 changes status to Submitted.
- IDJOB1 changes status to Ended “OK”.
- IDJOB2 changes status to Wait Submission.
- IDJOB2 changes status to Submitted.
- IDJOB2 changes status to Ended “OK”.

At the end of the process, the screen is displayed as follows:
Checking the Log for the Job from the Active Environment Screen

You can use the L (Log) option of the Active Environment screen to check the job log for these jobs. The log is then displayed in the Control-M Log screen. If this option is specified for multiple jobs in the Active Environment screen, the log displays will be stacked. Each time PF03/PF15 is pressed, the next log in the stack is displayed, until all logs have been displayed.

**Figure 13 Log Option in the Active Environment Screen**

![Image of the Active Environment screen with log option highlighted]

25 Type L (Log) in the O (Option) field to the left of IDJOB1 and IDJOB2, and then press Enter. The log for IDJOB1 is displayed.

**Figure 14 Control-M Log Screen**

<table>
<thead>
<tr>
<th>SHOW LIMIT ON =&gt; USERID GROUP MEM/MIS</th>
<th>DATE 020201 - 020201</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATE</td>
<td>TIME</td>
</tr>
<tr>
<td>020201</td>
<td>151226</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>020201</td>
<td>151908</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>020201</td>
<td>151911</td>
</tr>
<tr>
<td>020201</td>
<td>151912</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>020201</td>
<td>151917</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>020201</td>
<td>151917</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>020201</td>
<td>151917</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>020201</td>
<td>151917</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>020201</td>
<td>151918</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

You can see the clear progression of messages that were generated as the job executed under the control of Control-M. You can scroll the display forward to see additional lines.

26 Press PF03/PF15.
The log for IDJOB2 is displayed.

27 Press PF03/PF15 again.

The Active Environment screen is redisplayed.

As an alternative to using the Log option in the Active Environment screen to request the log display for specific jobs, you can use Option 5 (LOG) of the IOA Primary Option menu to request display of the IOA Log screen. This screen displays the log for all INCONTROL for z/OS products. You can then filter the display in much the same way that you filtered the display of the Active Environment screen.

### Checking the Sysout for the Job

If Control-M/Restart is installed at your site, and the SYSDATA for a job is automatically archived, you can view the Sysout for that job.

To do so, you must first specify option V (View Sysout) for the job, in the Active Environment screen. This option displays the Job Order Execution History screen, which, as its name indicates, displays the execution history of the job. From the list of job executions, you can decide the job for which you want to view the Sysout.

To see the Sysout of a particular run of the job, specify option S (Select) next to the specific job execution in the Job Order Execution History screen. The sysout for the job is then displayed in the Sysout Viewing screen.

---

**NOTE**

If Control-M/Restart is not operational at your site, or if SYSDATA is not archived, skip the remaining exercises in this section, and proceed to the topic “Holding and Deleting Active Environment Screen Job Orders” below.

---

Enter option V for job IDJOB1 to check the sysout of job IDJOB1.
The Job Order Execution History screen for IDJOB1 is displayed.

**Figure 15  Job Order Execution History Screen**

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>SCROLL</th>
<th>MEMNAME</th>
<th>OWNER</th>
<th>ORDERID</th>
<th>ODATE</th>
<th>O JOBNAME</th>
<th>JOBID</th>
<th>DATE</th>
<th>START</th>
<th>ELAPSED</th>
<th>PAGES</th>
<th>MAX</th>
<th>RC</th>
<th>STATUS</th>
<th>ODATE</th>
<th>ORDERID</th>
<th>OWNER</th>
<th>MEMNAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>JOB ORDER EXECUTION HISTORY</td>
<td>CRSR</td>
<td>IDJOB1</td>
<td>ID</td>
<td>001S3</td>
<td>020201</td>
<td>JOBID</td>
<td>32166</td>
<td>020201</td>
<td>18:05</td>
<td>0:00</td>
<td>00003</td>
<td>ENDED</td>
<td>&quot;OK&quot;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The Job Order Execution History screen, and the Sysout Viewing screen shown below, are discussed in Chapter 6, “Navigating The Active Environment.”

28 Enter option S for the job run of IDJOB1 to request the desired Sysout.

The Sysout Viewing screen is displayed. You can scroll down and up, and right and left, through the Sysout.
29 Exit the Sysout Viewing screen, and then exit the Job Order Execution History screen, to the Active Environment screen.

Holding and Deleting Active Environment Screen Job Orders

Although it is unnecessary to delete these job orders from the Active Environment screen—they will probably be automatically deleted during New Day processing, which is discussed in Chapter 7, “Job Ordering and New Day Processing”—you should do so anyway, as part of the lesson.

30 Type D (Delete) in the O (Option) field for both jobs, as shown below, and press Enter.

Figure 17 Delete Option in the Active Environment Screen
The jobs are not deleted. Instead, a message appears at the top of the screen, as shown below:

```
CTM653E CANNOT DELETE TASK IDJOB1 ODATE 020201 - NOT HELD  UP  <D> - (3)
COMMAND ===>
SCROLL ===> CRSR
O Name  Owner  Odate  Jobname  JobID   Typ ---------- Status ----------
D IDJOB1 ID  020201 M21  /24861  JOB  Ended "OK"
D IDJOB2 ID  020201 M21  /24866  JOB  Ended "OK"
```

The message, CTM653E CANNOT DELETE TASK IDJOB1 ODATE 020201 - NOT HELD, indicates that the first requested deletion, that of job IDJOB1, could not be performed because the job is not Held. In fact, any action that affects a job order, or the status of a job order, cannot be performed until the job order is placed in Held status. To hold a job order, specify option H (Hold).

31 Replace the D option by each of the jobs with an H option and press Enter to place the jobs in Held status.

The status of each job order changes to Requested Held Ended “OK”, and messages at the top of the screen indicate the change. The last issued message for IDJOB2 remains displayed at the top of the screen.

```
CTM659I HOLD OF TASK IDJOB2 GROUP "IDGRP1 " ODATE 020201 PERFORMED - (3)
COMMAND ===> SCROLL ==> CRSR
O Name  Owner  Odate  Jobname  JobID   Typ ---------- Status ----------
IDJOB1  ID  020201 M21  /24861  JOB  Requested Held Ended "OK"
IDJOB2  ID  020201 M21  /24866  JOB  Requested Held Ended "OK"
```

32 Press Enter again.

The status of each job now changes to Held Ended “OK”, and the message disappears.

```
Filter: D      ------- Control-M  Active  Environment ------ UP  <D> - (3)
COMMAND ===>
SCROLL ===> CRSR
O Name  Owner  Odate  Jobname  JobID   Typ ---------- Status ----------
IDJOB1  ID  020201 M21  /24861  JOB  Held Ended "OK"
IDJOB2  ID  020201 M21  /24866  JOB  Held Ended "OK"
```

33 Once again, specify the D option by each of the jobs.

34 If a Delete Confirmation window is displayed, confirm the Delete request.

Messages at the top of the screen indicate that the job orders are deleted, and the last issued message for IDJOB2 remains displayed. The job orders no longer appear in the screen.
Do not confuse deleting a job order from the Active Environment screen with deleting a job scheduling definition from the Job List screen. Deleting the job order impacts only that order. The job scheduling definition itself remains unaffected and can be ordered again. If you delete a job scheduling definition from the Job List screen, the job scheduling definition itself is gone, and the job can no longer be ordered from the table.

Return to the Scheduling Definition Facility

In this chapter, you will do a little more work in the Scheduling Definition facility.

Saving Job Scheduling Definitions

So far, you have created and exited two job scheduling definitions—for IDJOB1 and IDJOB2. But they are currently only in memory; you have not saved them.

You can now save the job scheduling definitions. To do that you must first return to the Scheduling Definition facility.

35 Enter =2 in the COMMAND field.

The Job List screen is displayed.

36 Exit the Job List screen.

The Exit option window is displayed. This window prompts whether you want to save the job scheduling definitions you created.
The LIBRARY and TABLE fields indicate the library and table in which the job scheduling definitions were defined. The specified values can be modified, for example, to save the job scheduling definitions in a different table.

The window also contains SAVE and CREATE fields. The Exit Option window enables you to perform many actions.

- To create a new table in a library and save the job scheduling definitions in that table, specify Y (Yes) in the CREATE field of the window, leaving the SAVE field blank, and exit the window.

  Because you specified a new table name in the entry panel—one that had not previously existed—you are now going to permanently create that table as well. Therefore, the values you entered in this chapter are the values that you will use.

  If you changed the library and table fields of the window to reflect a currently non-existing table, you would also use these values.

- To save the job scheduling definitions in an existing table in a library, specify Y (Yes) in the SAVE field of the window, leaving the CREATE field blank, and exit the window. This procedure should be used even if you change the library and or table fields to a different, but existing, table.

- To cancel the changes you made, specify N (No) in the both the SAVE and CREATE fields of the window, and exit the window.
To close the Exit Option window and remain in the Job List screen, with the changes remaining in memory, press **PF04/PF16** to execute the **RESET** command.

37 Specify **Y** in the **CREATE** field and leave the **SAVE** field blank.

The new table is created and the job scheduling definitions are saved in the table. The **Entry Panel** is displayed.

---

**NOTE**

Normally, when you exit the Job List screen, the Table List screen is displayed. However, if the Table List screen was bypassed when you entered the Scheduling Definition facility, that is, if you specified a **TABLE** value in the entry panel, which you did, the Table List screen is not displayed upon exiting the Job List screen; instead, the **Entry Panel** is displayed.

---

## Selecting a Table from the Table List Screen

You can now enter the Table List screen so that it displays your newly created table.

38 Ensure that correct library name is entered in the **LIBRARY** field of the entry panel and that the **TABLE** field is blank.

The Table List screen is displayed. Unless you are using a library dedicated to this set of exercises, other table names appear in the Table List screen.

### Figure 19  Control-M Table List Screen

<table>
<thead>
<tr>
<th>OPT</th>
<th>NAME</th>
<th>VV.MM</th>
<th>CREATED</th>
<th>CHANGED</th>
<th>SIZE</th>
<th>INIT</th>
<th>MOD</th>
<th>ID</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ADABAS</td>
<td>01.00</td>
<td>98/09/09 00/05/09</td>
<td>00:50</td>
<td>70</td>
<td>70</td>
<td>0</td>
<td>O01</td>
</tr>
<tr>
<td></td>
<td>APPLTN</td>
<td>01.00</td>
<td>98/09/09 00/05/09</td>
<td>00:50</td>
<td>180</td>
<td>180</td>
<td>0</td>
<td>O01</td>
</tr>
<tr>
<td></td>
<td>APP001</td>
<td>01.00</td>
<td>98/09/09 00/05/09</td>
<td>00:50</td>
<td>41</td>
<td>41</td>
<td>0</td>
<td>O01</td>
</tr>
<tr>
<td></td>
<td>ARCNIGHT</td>
<td>01.00</td>
<td>98/09/09 00/05/09</td>
<td>00:50</td>
<td>5</td>
<td>5</td>
<td>0</td>
<td>S07</td>
</tr>
<tr>
<td></td>
<td>ASM01</td>
<td>01.00</td>
<td>98/09/09 00/05/09</td>
<td>00:50</td>
<td>9</td>
<td>9</td>
<td>0</td>
<td>S07</td>
</tr>
<tr>
<td></td>
<td>ASM02</td>
<td>01.00</td>
<td>98/09/09 00/05/09</td>
<td>00:50</td>
<td>14</td>
<td>14</td>
<td>0</td>
<td>S07</td>
</tr>
<tr>
<td></td>
<td>BACKUP</td>
<td>01.00</td>
<td>98/09/09 00/05/09</td>
<td>00:50</td>
<td>5</td>
<td>5</td>
<td>0</td>
<td>S07</td>
</tr>
<tr>
<td></td>
<td>CICSJOB</td>
<td>01.00</td>
<td>98/09/09 00/05/09</td>
<td>00:50</td>
<td>70</td>
<td>70</td>
<td>0</td>
<td>O01</td>
</tr>
<tr>
<td></td>
<td>CICS001</td>
<td>01.00</td>
<td>98/09/09 00/05/09</td>
<td>00:50</td>
<td>180</td>
<td>180</td>
<td>0</td>
<td>O01</td>
</tr>
<tr>
<td></td>
<td>CICSTEST</td>
<td>01.00</td>
<td>98/09/09 00/05/09</td>
<td>00:50</td>
<td>41</td>
<td>41</td>
<td>0</td>
<td>O01</td>
</tr>
<tr>
<td></td>
<td>CIC5000</td>
<td>01.00</td>
<td>98/09/09 00/05/09</td>
<td>00:50</td>
<td>5</td>
<td>5</td>
<td>0</td>
<td>S07</td>
</tr>
<tr>
<td></td>
<td>CLIENTS</td>
<td>01.00</td>
<td>98/09/09 00/05/09</td>
<td>00:50</td>
<td>9</td>
<td>9</td>
<td>0</td>
<td>S07</td>
</tr>
<tr>
<td></td>
<td>DB2EXE</td>
<td>01.00</td>
<td>98/09/09 00/05/09</td>
<td>00:50</td>
<td>14</td>
<td>14</td>
<td>0</td>
<td>S07</td>
</tr>
<tr>
<td></td>
<td>DLOAD</td>
<td>01.00</td>
<td>98/09/09 00/05/09</td>
<td>00:50</td>
<td>5</td>
<td>5</td>
<td>0</td>
<td>S07</td>
</tr>
<tr>
<td></td>
<td>MAINDAY</td>
<td>01.00</td>
<td>98/09/09 00/05/09</td>
<td>00:50</td>
<td>180</td>
<td>180</td>
<td>0</td>
<td>O01</td>
</tr>
<tr>
<td></td>
<td>MAINT</td>
<td>01.00</td>
<td>98/09/09 00/05/09</td>
<td>00:50</td>
<td>41</td>
<td>41</td>
<td>0</td>
<td>O01</td>
</tr>
<tr>
<td></td>
<td>MAINTPL</td>
<td>01.00</td>
<td>98/09/09 00/05/09</td>
<td>00:50</td>
<td>5</td>
<td>5</td>
<td>0</td>
<td>S07</td>
</tr>
<tr>
<td></td>
<td>ONSPOOL</td>
<td>01.00</td>
<td>98/09/09 00/05/09</td>
<td>00:50</td>
<td>9</td>
<td>9</td>
<td>0</td>
<td>S07</td>
</tr>
<tr>
<td></td>
<td>ONS001</td>
<td>01.00</td>
<td>98/09/09 00/05/09</td>
<td>00:50</td>
<td>14</td>
<td>14</td>
<td>0</td>
<td>S07</td>
</tr>
<tr>
<td></td>
<td>OPERCLN</td>
<td>01.00</td>
<td>98/09/09 00/05/09</td>
<td>00:50</td>
<td>5</td>
<td>5</td>
<td>0</td>
<td>S07</td>
</tr>
</tbody>
</table>

OPTIONS S SELECT  O ORDER  F FORCE  G GRAPHIC  FLOW  B BROWSE  D DELETE 15.38.37
Among the options of the Table List screen are the options S (Select) and B (Browse). Both these options bring you to the Job List screen, where the list of jobs in the table are displayed, but each displays the Job List screen in a different mode:

- The Select option displays the Job List screen in Edit mode. When you then select a job from the Job List screen, you can edit and update the job scheduling definition.

- The Browse option displays the Job List screen in Browse mode. When you then select a job from the Job List screen, the job scheduling definition can be viewed, but it cannot be edited or updated.

You will not edit the job scheduling definition at this point, so use the B (Browse) option.

39 Either page down until the table name is visible, or perform an ISPF LOCATE to locate your table. Ensure that the cursor is in the OPTION field for your table.

40 Specify option B (Browse) in the OPTION field next to your table.

The Job List screen is displayed in Browse mode. The screen displays the two defined jobs in your table.

Figure 20  Job List Screen

<table>
<thead>
<tr>
<th>OPT</th>
<th>NAME</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>IDJOB1</td>
<td>THIS JOB MUST BE RUN BEFORE IDJOB2</td>
</tr>
<tr>
<td></td>
<td>IDJOB2</td>
<td></td>
</tr>
</tbody>
</table>

From the Job List screen, you can now use the S (Select) option to browse either the IDJOB1 or the IDJOB2 job scheduling definition in the table.

In addition to the screen shown above, there are other possible formats of the Job List screen.
Formats of the Job List Screen

You had previously used the DESC command to ensure the desired format of the screen. This command confirmed that you wanted the display type that shows the description of the job, as taken from the DESC field in the job scheduling definition, to be displayed next to the job name.

Table 3 describes the commands used to display other information.

Table 3  Commands to Display Job List Screen Formats

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATA</td>
<td>Command DATA displays the Application name and Group name of the job next to the job name. The Application name and Group name are taken from the corresponding fields in the job scheduling definition.</td>
</tr>
<tr>
<td>STAT</td>
<td>Command STAT displays, next to the job name, the following ISPF-like statistical information about the job: Version and modification numbers, creation date, last modification date, and user ID.</td>
</tr>
</tbody>
</table>

Alternate between specifying the DATA, STAT, and DESC commands in the COMMAND field of the Job List screen, and note the differences in the display following the entry of each command. Ensure that the last command you enter is DESC, because that is the format you are using in the exercises in this guide.

Note that whichever format you use, the actual jobs listed in the screen, that is, jobs IDJOB1 and IDJOB2, remain the same. The format of the Job List screen does not affect which jobs are listed. It only affects the accompanying data that is displayed for those jobs.

Exiting the Online Facility

You have now finished the most of the exercises in this chapter of the guide. All that remains is to learn to exit the IOA Online facility.

Leave the online facility by exiting one screen at a time, until you have exited the entire online facility.

41  Press PF03/PF15 or enter the END command.

The Table List screen is displayed.

42  Press PF03/PF15 or enter the END command.
The Entry panel is displayed.

43 Press **PF03/PF15** or enter the **END** command.

The IOA Primary Option Menu is displayed.

44 Enter X in the **COMMAND** field of the IOA Primary Option menu.

The IOA Online facility is exited.

---

**Review**

In this chapter you

- accessed the following screens of the scheduling definition facility
  - Entry Panel
  - Table List screen
  - Job List screen
  - Job Scheduling Definition screen

- created a job scheduling definition directly from the Entry panel, and then created a second job scheduling definition by using the I (Insert) option in the Job List screen

- learned about and utilized the following job scheduling definition fields while creating your job scheduling definitions
  - MEMNAME
  - MEMLIB
  - GROUP
  - DESC
  - DAYS
  - IN
  - CONFIRM
  - OUT

- defined one of the jobs to require manual confirmation before submission

- used IN and OUT conditions to create a dependency between the two jobs

- ordered the two jobs from the Job List screen
viewed messages in the Job Order Messages screen

moved between facilities using an =x command instead of returning to the IOA Primary Option menu

displayed the Active Environment screen, and toggled between display of the list of commands and the list of options at the bottom of the screen

defined a filter and filtered the Active Environment display

learned about statuses in the Active Environment screen

used the C (Confirm) option in the Active Environment screen to manually confirm a job

observed changes to job statuses in the Active Environment screen

used the L (Log) option in the Active Environment screen to display the message log for the job orders in the Control-M Log screen

used the V option to display the Job Order Execution History screen and from there displayed the Sysout of a job execution in the Sysout Viewing screen

used the H (Hold) option so that you could delete job orders from the Active Environment screen

used the D (Delete) option in the Active Environment screen to delete the job orders when they were no longer needed

used the Exit Option window of the Job List screen to create a new table and save your two job scheduling definitions in the table

learned the difference between the S (Select) and B (Browse) options of the Table List screen

changed the format of the Job List screen

exited the Online facility by working your way up the screen hierarchy as you exited individual screens

---

**Recommended Reading**

Before continuing with the next chapter, you should read the following in the *Control-M for z/OS User Guide*
In Chapter 1

■ Introduction to Control-M components
■ Prerequisite condition concept
■ Date concepts

In Chapter 2

■ Introduction to the Scheduling Definition Facility
■ Description of the following Scheduling Definition Facility screens including their formats, options and commands
  — Entry Panel
  — Table List screen
  — Job List screen
■ Description of the layout of the Job Scheduling Definition screen and the fields in the screen
■ Topics “Exiting the Scheduling Definition Facility” and “Ordering (Scheduling) Jobs” in the Scheduling Definition facility description
■ Introduction to the Active Environment screen and the description of its fields
■ Descriptions of the Control-M and IOA Log screens

In Chapter 3

Detailed descriptions of the parameters you have already used: CONFIRM, DAYS, DESC, GROUP, IN, MEMLIB, MEMNAME, OUT

NOTE

The description of the parameter DAYS is quite detailed. Read it for a general understanding of parameter usage and possible parameter formats, without trying to memorize all the detail. For now, you can skip the information related to subparameters DCAL and AND/OR.
Scheduling Definition and Manual Intervention

This chapter includes the following topics:

Overview ................................................................. 56
Preparations ............................................................. 56
Sections of the Job Scheduling Definition. ............................ 57
  General Parameters Section ........................................... 59
  Basis Scheduling Parameters Section ................................. 60
  Runtime Scheduling Parameters Section ............................. 64
  Post-Processing Parameters Section ................................. 67
Conditions, Date Qualifiers, and Job Dependencies .................. 71
Displaying the Scheduling Plan for the Job ............................. 72
Defining JOB4 .......................................................... 75
Job Ordering and Job Forcing ............................................. 77
Why Screen ............................................................. 79
  Why the Job Did Not Run .............................................. 80
  Adding a Missing Condition in the Why Screen ...................... 81
IOA Conditions/Resources Screen ...................................... 82
  Filtering the IOA Conditions/Resources Screen .................... 83
  Adding a Condition or Resource ....................................... 83
  A Second Look at the Why Screen ................................... 84
Zoom Screen ............................................................ 85
  Freeing a Held Job ...................................................... 87
Review ................................................................. 89
  Recommended Reading ............................................... 90
Overview

In the previous chapter, you were introduced to the Scheduling Definition Facility and the Active Environment screen. In this chapter, you will

■ take a much closer look at the job scheduling definition screen, the sections into which it is divided, and the many of the fields of those sections, including a focus on
  — Basic Scheduling parameters
  — Runtime Scheduling parameters
  — Post-Processing parameters

■ view the scheduling plan of a job in a graphic calendar display

■ force a job whose basic scheduling criteria are not satisfied

■ use the Why screen, IOA Conditions/Resources screen, and the Zoom screen to intervene when a job is not submitted

Preparations

In this chapter, you are going to create two new jobs and their job scheduling definitions. In the last chapter, you used the names IDJOB1 and IDJOB2. In this chapter, you will use the names IDJOB3 and IDJOB4, respectively. (You will use the same library and table that you used in the last chapter.)

For the JCL of IDJOB4, use a copy of the same trivial JCL that you used for IDJOB1 & IDJOB2—that is, a job that does basically nothing, but ends successfully.

For IDJOB3, you need a JCL that will result in an error condition. For this purpose, you will use a test utility in which you indicate the desired error return code.

You can now create the JCL for those jobs.

■ Create trivial job IDJOB3 to run utility TESTUTIL and define it to return a condition code of C0008.

■ Create trivial job IDJOB4 and define it to run IEFBR14.
Sections of the Job Scheduling Definition

1. Enter the IOA Online Facility and request the Scheduling Definition facility.

   The Scheduling Definition entry panel is displayed.

2. Enter your library and table names in the LIBRARY and TABLE fields, and IDJOB3 in the JOB field. The request appears as shown below:

   **Figure 21 Control-M Scheduling Definition Facility - Entry Panel**

   | COMMAND ====> |
   | SPECIFY LIBRARY, TABLE, JOB |
   | LIBRARY ===> CTM.TEST.SCHEDULE |
   | TABLE ===> IDGS1 (Blank for table selection list) |
   | JOB ===> IDJOB3 (Blank for job selection list) |
   | SMART TABLE ===> (Y - for new SMART Table) |
   | SHOW JOB DOCUMENTATION ===> N (Y/N) |
   | AUTO-SAVE DOCUMENTATION ===> N (Y/N) |

   USE THE COMMAND SHPF TO SEE PFK ASSIGNMENT 23.00.04
The Job Scheduling Definition screen for IDJOB3 is displayed.

**Figure 22  Job Scheduling Definition Screen**

<table>
<thead>
<tr>
<th>JOB: IDJOB3</th>
<th>LIB CTM.TEST.SCHEDULE</th>
<th>TABLE: IDGS1</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEMNAME</td>
<td>IDJOB3</td>
<td>MEMLIB</td>
</tr>
<tr>
<td>OWNER</td>
<td>ID</td>
<td>TASKTYPE JOB</td>
</tr>
<tr>
<td>APPL</td>
<td></td>
<td>PREVENT-NCT2</td>
</tr>
<tr>
<td>DESC</td>
<td></td>
<td>DFLT N</td>
</tr>
<tr>
<td>OVERLIB</td>
<td></td>
<td>STAT CAL</td>
</tr>
<tr>
<td>SCHENV</td>
<td></td>
<td>SYSTEM ID</td>
</tr>
<tr>
<td>SET VAR</td>
<td></td>
<td>NJE NODE</td>
</tr>
<tr>
<td>CTB</td>
<td>IDJOB3</td>
<td>DOCLIB</td>
</tr>
<tr>
<td>STEP AT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DOCMEM</td>
<td>IDJOB3</td>
<td>DOCLIB</td>
</tr>
<tr>
<td>DATES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CONFCAL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MINIMUM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEFINITION</td>
<td>ACTIVE FROM</td>
<td></td>
</tr>
<tr>
<td>RESOURCE</td>
<td>COMMANDS: EDIT, DOC, PLAN, JOBSTAT</td>
<td>11.17.00</td>
</tr>
</tbody>
</table>

In Chapter 1, “Introduction to Control-M.”, you learned that the job scheduling definition is partitioned into sections, each section separated from the preceding section by a line of equal signs.

To familiarize yourself with several more fields in these sections, you will fill in the job scheduling definition for IDJOB3. To do this, you must first outline the type of processing you expect for IDJOB3, which requires that you assume the following:

- IDJOB3 is an application that tabulates certain fiscal results once a month.
- IDJOB3 should always be scheduled on the last Friday of the month.
- IDJOB3 should be submitted only if the previous execution of the job (during the previous month) ended OK.
- IDJOB3 should be submitted only if the following resources are available:
  - 1 tape drive
  - 5 units of a CPU resource called IDCPOWER:
- IDJOB3 should be submitted from sometime between the hours of 11:00 p.m. on Friday night and 4:00 a.m. on Saturday Morning.
- IDJOB3 should be considered a critical path job with a priority of 5.
- IDJOB3 should be automatically submitted, no requirement for manual confirmation.

- If IDJOB3 ends OK, the appropriate reports are generated by the application.

- If the budget for IDJOB3, as defined in the application, was exceeded, the job returns a condition code of C0008 or above, depending on the amount by which the budget was exceeded, and the job should end NOTOK.

- If IDJOB3 ends OK, a prerequisite condition should be created indicating that it ended OK and the corresponding condition from the previous month’s run should be deleted.

- If IDJOB3 ends NOTOK, that is, if it has a condition code of C0008 or higher, the following should occur:
  - A condition should be created stating that the job ended NOTOK.
  - An appropriate message should be sent, or shouted, to user OPER1.
  - An auxiliary job, called IDJOB4, should be run to perform additional processing.

You can now fill in the job scheduling definition of IDJOB3. Note that fields are described only briefly in the following sections, and that not all fields are discussed. Detailed descriptions for all fields of the Job Scheduling Definition screen are discussed in the *Control-M for z/OS User Guide*.

### General Parameters Section

The fields in the first section of the Job Scheduling Definition screen correspond to the Job Scheduling Definition General parameters.
The General parameters, some of which you have seen earlier, provide basic information about the job, and are shown in the following table.

Table 4  Job Scheduling Definition Screen General Parameters

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Type of Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEMNAME, MEMLIB</td>
<td>Location of the JCL</td>
</tr>
<tr>
<td>OWNER</td>
<td>Owner of the job</td>
</tr>
<tr>
<td>TASKTYPE</td>
<td>Type of job</td>
</tr>
<tr>
<td>DESC</td>
<td>Job description</td>
</tr>
<tr>
<td>APPL, GROUP</td>
<td>Application and group to which the job belongs</td>
</tr>
<tr>
<td>DOCMEM, DOCLIB</td>
<td>Location of documentation about the job</td>
</tr>
</tbody>
</table>

3 Specify the name of your test JCL library in the MEMLIB field

4 Specify IDGRP2 in the GROUP field.

5 Specify FISCAL RESULTS ON LAST FRIDAY OF MONTH in the DESC field. The section looks as follows:

Basis Scheduling Parameters Section

The fields in the second section of the Job Scheduling Definition screen correspond to the Job Scheduling Definition Basic parameters.
These basic scheduling parameters determine on what day the job is scheduled, specifically, on what days the job becomes a candidate for execution. Scheduling days can be expressed in many ways, according to need. Several of the basic scheduling parameters are explained in the following table:

**Table 5  Basic Job Scheduling Definition Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Explanation</th>
</tr>
</thead>
</table>
| DAYS      | Days within the specified months on which to schedule the job. For more information, see MONTHS in this table.  
Values can be expressed in various formats, the simplest of which is “straight values.” For example, values 01 and 15 would mean the first and fifteenth day of the month, respectively.  
Value ALL means all days of the month. |
| WDAYS     | Days of the week within the specified months on which to schedule the job. For more information, see MONTHS in this table.  
Values can be expressed in various formats.  
Scheduling is relative to the day of the week that is defined as the start of the week at your site. For example, if Monday is defined as the start, or first day of the week at your site, a value of 1 means that the job is scheduled on Mondays, and 4 means that the job is scheduled on Thursdays. |
| AND/OR    | Conjunctonal parameter that determines the relationship between the DAYS and WDAYS parameters. An A (And) value requires that both sets of criteria be satisfied. An O (Or) value means that the criteria are satisfied if the DAYS criteria and/or the WDAYS criteria are satisfied. |
| MONTHS    | Months in which to schedule the job.  
All twelve months are listed as 01 through 12. A Y (Yes) or N (No) value can be specified for each month.  
For months in which a job should be scheduled, that is, months with a Y value, the job is scheduled on those days determined by the DAYS and/or WDAYS values. |
| DATES     | Specific dates on which to schedule the job. Format is mmdd or ddmm, depending on the site standard. For example, a DATES value of 0505 schedules the job on the 5th of May. |
Several examples of how these parameters might be used are shown below.

- To schedule a job on the 1st and 15th of every month

```plaintext
DAYS  01,15
DCAL
AND/OR
WDAYS  W3CAL
DATES
CONF N MAXWAIT 00  D-CAT
MINIMUM  PDS
DEFINITION ACTIVE FROM  UNTIL
```

- To schedule a quarterly job on 31 March, 30 June, 30 September and 31 December (assuming a date format of mmdd)

```plaintext
DAYS
DCAL
AND/OR
WDAYS  W3CAL
DATES  0331 0630 0930 1231
CONF CAL  RETRO N MAXWAIT 00  D-CAT
MINIMUM  PDS
DEFINITION ACTIVE FROM  UNTIL
```

- To schedule a job on every Wednesday in June (assuming start of week is Monday)

```plaintext
DAYS
DCAL
AND/OR
WDAYS  W3CAL
MONTHS  1- N 2- N 3- N 4- N 5- N 6- Y 7- N 8- N 9- N 10- N 11- N 12- N
DATES
CONF CAL  RETRO N MAXWAIT 00  D-CAT
MINIMUM  PDS
DEFINITION ACTIVE FROM  UNTIL
```

In your job scheduling definition, you want to schedule the job on the last Friday of the month. For this you need to know that in the DAYS parameter, a value of L1 means the last day of the month, L2 is the second to the last day of the month, and so on.

6 In the DAYS field, specify the following values: L1,L2,L3,L4,L5,L6,L7. Be sure to separate values by a comma, and leave no embedded spaces.

7 In the AND/OR field, specify A (And).

8 In the WDAYS field, specify 5.
NOTE
This example assumes your site-defined start of week is Monday. If the start of
week is defined as Sunday, use a WDAYS value of 6 instead of 5. If you do not know
the site-defined start of week, it does not really matter which value you use in this
exercise.
But to use this parameter successfully in your real job scheduling definitions, you
need to know, and should find out from your INCONTROL administrator, your
site-defined start of week.

9 Make no changes to the MONTHS values, because they are correctly set to Y (Yes)
for all months.

This section of your screen appears as shown below:

<table>
<thead>
<tr>
<th>DAYS</th>
<th>L1,L2,L3,L4,L5,L6,L7</th>
<th>DCAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>WDAYS</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>DATES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CONFCAL</td>
<td>RETRO N MAXWAIT OO D-CAT</td>
<td></td>
</tr>
<tr>
<td>MINIMUM</td>
<td>PDS</td>
<td></td>
</tr>
<tr>
<td>DEFINITION</td>
<td>ACTIVE FROM UNTIL</td>
<td></td>
</tr>
</tbody>
</table>

Note the following points:

■ The values of the DAYS parameter limits the scheduling to the last week of the
  month.

■ The value of the WDAYS parameter limits the scheduling to Fridays.

■ The A (And) value in the AND/OR field requires that both sets of criteria be
  satisfied. This limits the scheduling to the last Friday in the last week of the month.

■ The Y (Yes) value for all months ensures that the scheduling on the last Friday of
  the month will occur every month.
The fields in the third section of the Job Scheduling Definition screen correspond to the Job Scheduling Definition Runtime Scheduling parameters.

---

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONTROL</td>
<td>Control Resource. Each Control resource specification consists of two values: Resource name and the type of control required. Valid values for the type of control are:</td>
</tr>
<tr>
<td></td>
<td>- E – Exclusive</td>
</tr>
<tr>
<td></td>
<td>- S – Shared</td>
</tr>
<tr>
<td></td>
<td>For example, this parameter can be used to specify that exclusive control of a particular file is required. Control-M does not submit the job until it has exclusive control of that file.</td>
</tr>
<tr>
<td>RESOURCE</td>
<td>Quantitative Resource. Each Quantitative resource specification consists of two values: Resource name and the quantity required.</td>
</tr>
<tr>
<td></td>
<td>For example, this parameter can be used to specify that two tape drives are required by the job. Control-M does not submit that job until two tape drives are available and reserved for that job.</td>
</tr>
</tbody>
</table>

---

**NOTE**

The PIPE parameter is only displayed if MAINVIEW Batch Optimizer (MVBO) is installed at your site.
You had previously determined the following submission criteria for the job:

- The job can be submitted only if the previous month’s execution of the job ended OK.
- The job requires the following resources:
  - 1 tape drive
  - 5 units of CPU
- The job must be submitted between the hours of 11:00 p.m. (Friday) and 4:00 a.m. (Saturday).
- The job is a critical path job with a priority of 5.
- The job must be submitted automatically without manual confirmation.

You should now define these criteria.

### Table 6  Runtime Scheduling Parameters (part 2 of 2)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIME FROM</td>
<td>TIME FROM indicates the earliest time that Control-M can submit the job. If specified, the job is not submitted before this time of the scheduling day.</td>
</tr>
<tr>
<td>TIME UNTIL</td>
<td>TIME UNTIL indicates the latest time by which a job can be submitted. If Control-M is unable to submit the job before this time (for example, if a required resource is not available before the UNTIL time), the job is not submitted. Either value or both, or neither, can be specified. No value means that a time factor is irrelevant for job submission.</td>
</tr>
<tr>
<td><strong>Note</strong>: It is possible to specify the FROM/TO offset in days in addition to the FROM/TO time. For details, see the Control-M for z/OS User Guide.</td>
<td></td>
</tr>
<tr>
<td>PRIORITY</td>
<td>Job priority. If two or more jobs awaiting submission are competing for resources, this value enables Control-M to determine which job to submit, based on priority.</td>
</tr>
<tr>
<td></td>
<td>Alphabetic or numeric values can be specified, where A &lt; Z &lt; 1 &lt; 9.</td>
</tr>
<tr>
<td></td>
<td>Critical path priority can be indicated by specifying an asterisk (*) before the priority value (for example, *5). A critical path job always takes precedence over a non-critical path job.</td>
</tr>
</tbody>
</table>
10. In the IN fields, specify **IDJOB3-ENDED-OK** as the condition name.

11. Press **Tab** after specifying the condition name, to place the cursor in the DATE field.

12. Specify **PREV** as the value for the DATE field.

   When you define post-processing parameters, you will see that when the job ends OK this same IN condition is added, but with a date of **ODAT**.

13. In the RESOURCE field, specify **IDTAPE-DRIVE** as the resource name.

14. Press **Tab** after specifying the resource name to place the cursor in the QUANTITY field.

15. Specify **1** as the value for the QUANTITY field.

   Two resources can be specified on a single line. Once you specify the quantity of the first resource, the cursor automatically tabs over to the RESOURCE-NAMESPACE field for the second resource.

16. Specify **IDCPOWER** as the second resource name, and **5** as the quantity.

   Once you have filled in two resources on the resource line, an additional resource line is opened to enable you to specify more resources.

17. In the TIME FROM field, specify **2300**. In the UNTIL field specify **0400**.

18. In the PRIORITY field, specify ***5**.

19. In the CONFIRM field, which you defined in the previous exercise, delete the **Y** value.

   This section of the job scheduling definition appears as shown below:

   ```plaintext
   IN       IDJOB3-ENDED-OK      PREV
   CONTROL
   RESOURCE IDTAPE-DRIVE         0001      IDCPOWER             0005
   PIPE
   FROM TIME 2300 + DAYS        UNTIL TIME 0400 + DAYS
   DUE OUT TIME + DAYS           PRIORITY *5      SAC      CONFIRM
   TIME ZONE:                   
   ```
Post-Processing Parameters Section

The fields in the fourth section of the Job Scheduling Definition screen correspond to the Job Scheduling Definition Post-Processing parameters.

<table>
<thead>
<tr>
<th>OUT</th>
<th>AUTO-ARCHIVE</th>
<th>SYSDB</th>
<th>MAXDAYS</th>
<th>MAXRUNS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RETENTION:</td>
<td># OF DAYS TO KEEP</td>
<td># OF GENERATIONS TO KEEP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SYSOUT OP</td>
<td>(C.D.F.N.R)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MAXRERUN</td>
<td>RERUNMEM</td>
<td>INTERVAL</td>
<td>FROM</td>
</tr>
<tr>
<td></td>
<td>STEP RANGE</td>
<td>FR (PGM.PROC)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ON PGMST</td>
<td>PROCST</td>
<td>CODES</td>
<td>A/O</td>
</tr>
<tr>
<td></td>
<td>DO</td>
<td>SHOUT WHEN</td>
<td>TIME</td>
<td>+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MS</td>
<td>DAYS</td>
<td>TO</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>URGN</td>
<td></td>
</tr>
</tbody>
</table>

These Post-Processing parameters define actions to be performed by Control-M after the job ends. Basically, these can be divided into three main groups—those relevant if the job ends OK, those relevant under specified circumstances, and those that are always relevant.

The OUT parameter has already been discussed, and those discussions are relevant only if the job ends OK. Table 7 explains several other Post-Processing parameters.
### Table 7  Post-Processing Parameters ON and DO Statements

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ON</strong></td>
<td>The ON statement constitutes the beginning of an ON/DO conditional block that is relevant only in the specified situation. The ON block defines a set of circumstances. If those circumstances are met, the specified DO actions are performed. ON criteria have two parts: steps and codes. When defining the ON block, you specify a step or set of steps (program and/or procedure steps), and possible results for those steps (condition codes, abend codes, system codes and so on). If this combination of steps and codes is satisfied, the DO block is performed. A complex set of values, including predefined keyword values such as ANYSTEP, can be specified in the ON block. If more than one ON statement is specified in the block, the A/O (And/Or) field determines the relationship between them. Valid values are A (And) and O (Or).</td>
</tr>
<tr>
<td><strong>DO</strong></td>
<td>DO statements specify actions to be performed if the ON criteria are satisfied. Many different DO statements are available. For example: DO COND adds or deletes a prerequisite condition. DO FORCEJOB forces a job to be scheduled. DO SHOUT sends, or shouts, a message to a specified destination. DO RERUN reruns the job. When you type the name or abbreviation of the DO statement, such as COND, or SHOUT, in the DO field, the appropriate subparameter fields for the statement are displayed. If by mistake you request the wrong DO statement, you can delete it by selecting the DO statement name, such as COND, and pressing Enter.</td>
</tr>
</tbody>
</table>
Post-Processing Parameters Section

Note that this section contains a SHOUT WHEN statement. This statement, which is not examined in this chapter, is different than the DO SHOUT statement.

You had previously decided to request the following post-processing activities:

- If the job ends OK, which is a condition code of less than C0008
  
  — the prerequisite condition of the successful run from the previous month, should be deleted
  
  — a new prerequisite condition indicating that the job ended OK should be created.

- If the job ends NOTOK, that is, it has a condition code of C0008 or higher
  
  — a condition must be created to indicate that the job ended NOTOK.
  
  — an appropriate message must be shouted to your user ID.
  
  — auxiliary job IDJOB4 must be run to perform additional processing.

You can now define the necessary values.

20 Fill in the OUT fields with the following value:

| ID | JOB3-ENDED-OK  | ODAT + |

After filling in the second condition, a new, blank OUT condition line is opened to enable you to specify another condition.

21 In the ON block, specify ANYSTEP in the PGMST (program step) field, and specify the value \textgreater{}C0007 in the CODES field. A value equal to or greater than C0008 is synonymous with a value greater than C0007.

22 In the DO field, type COND and press Enter.

The DO COND statement is opened, as is a blank DO statement for specification of another DO statement.

23 Fill in the DO COND fields with the following values:

| IDJOB3-END-NOTOK | ODAT + |

24 In the blank DO field, type SHOUT and press Enter.

The DO SHOUT statement is opened, as is a blank DO statement for specification of another DO statement, as shown below.

<table>
<thead>
<tr>
<th>DO</th>
<th>SHOUT</th>
<th>TO</th>
<th>URGENCY</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>=</td>
<td>=</td>
<td>=</td>
</tr>
<tr>
<td></td>
<td>DO</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
25 In the DO SHOUT statement, specify your user ID in the TO subparameter field; replace the R (Regular) with U (Urgent) in the URGENCY field; and specify the message **FRIDAY FISCAL RESULTS RUN FAILED** following the equal sign (=).

Note that the only urgency value, other than R and U, is **V**, which signifies a Very Urgent message.

26 In the blank DO field, type **FORCEJOB** and press **Enter**.

The DO FORCEJOB statement is opened, as is a blank DO statement for specification of another DO statement, as shown below.

```
DO FORCEJOB  TABLE  JOB  DATE
  LIBRARY
  DO
```

27 In the DO FORCEJOB statement, specify the current table name (**IDGS1**) in the TABLE field; specify **IDJOB4** in the JOB field; specify **ODAT** in the DATE field; and specify the current scheduling library name in the LIBRARY field.

This section of the job scheduling definition appears as shown below:
You have now finished defining IDJOB3.

28 Exit the job scheduling definition of IDJOB3.

The Job List screen is displayed. This is where you can take a closer look at the use of IN, OUT, and DO COND conditions, and their date qualifiers, to create job dependencies.

**Conditions, Date Qualifiers, and Job Dependencies**

Conditions are generally used to create job dependencies. In the previous chapter you saw that when IDJOB1 ended OK, it set an OUT condition (IDJOB1-ENDED-OK) that satisfied the prerequisite IN condition to IDJOB2. This is necessary because IDJOB2 could not be submitted until IDJOB1 ended OK.

Earlier in this chapter you defined a DO COND condition. Both OUT and DO COND are post-processing statements that add or delete conditions. But the OUT condition is only added or deleted when the job ends OK; the DO COND condition is only added or deleted when the criteria in the ON block are satisfied.

As mentioned in the last chapter, every condition has an associated date. This can be an explicitly defined date, such as 0909, but more commonly, it is a keyword that represents an actual date. For example, the date keyword ODAT stands for the original scheduling date of the job. The meaning of original scheduling date is discussed in Chapter 5, “AutoEdit and JCL,” and is referred to in the discussion of date concepts in the *Control-M for z/OS User Guide*. 

<table>
<thead>
<tr>
<th>OUT</th>
<th>IDJOB3-ENDED-OK</th>
<th>ODAT +</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTO-ARCHIVE</td>
<td>SYSDAT</td>
<td>MAXDAYS</td>
</tr>
<tr>
<td>RETENTION: # OF DAYS TO KEEP</td>
<td># OF GENERATIONS TO KEEP</td>
<td></td>
</tr>
<tr>
<td>SYSOUT OP (C,D,F,N,R)</td>
<td>MAXRREUN</td>
<td>RERUNMEM</td>
</tr>
<tr>
<td>STEP RANGE</td>
<td>FR (PGM.PROC)</td>
<td>.</td>
</tr>
<tr>
<td>ON PGMST ANYSTEP</td>
<td>PROCST</td>
<td>CODES &gt;C0007</td>
</tr>
<tr>
<td>DO COND</td>
<td>IDJOB3-END-NOTOK</td>
<td>ODAT +</td>
</tr>
<tr>
<td>DO SHOUT</td>
<td>TO ID</td>
<td>URGENCY U</td>
</tr>
<tr>
<td>= FRIDAY FISCAL RESULTS RUN FAILED</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DO FORCEJOB</td>
<td>TABLE IDGS1</td>
<td>JOB IDJOB4</td>
</tr>
<tr>
<td>LIBRARY CTM.TEST.SCHEDULE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ON PGMST</td>
<td>PROCST</td>
<td>CODES</td>
</tr>
<tr>
<td>DO</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
A condition is actually identified by the combination of its name and date. Two conditions having the same name but different dates are considered two different conditions. In the last chapter, you wanted the run of IDJOB2 to be dependent on the run of IDJOB1 from the same day. Therefore, not only did the condition names of both jobs have to match, but so did the dates. Use of the keyword ODAT for both the IN and the OUT condition ensured that match. Had they run on different original scheduling dates, there would have been no match.

In this chapter you create another type of dependency—a dependency between different runs of the same job on different dates. IDJOB3 cannot run this month unless it successfully ran last month. The same condition name is used (IDJOB3-ENDED-OK), but with different dates for the IN condition and OUT condition.

- The date of the OUT condition is the date of the current job run; in other words, ODAT.
- The IN condition can only be satisfied by the run from the previous month. Therefore, the date of the OUT condition is PREV, a keyword that resolves to the previous original scheduling date (that is, the last run) of the job.

## Displaying the Scheduling Plan for the Job

Before beginning to define IDJOB4, display the schedule plan for IDJOB3.

Option P (PLN) in the Job List screen is used to request a graphic display of a schedule for job planning.
Enter option P for IDJOB3. A window containing FROM DATE and TO DATE fields opens. This window enables you to specify the time period for which the scheduling plan should be displayed.

The default FROM DATE value is the current working date. The default TO DATE value is one month, or thirty days, following the current working date.
30 Set the FROM DATE to the first day of the current month. Set the TO DATE to the last day of the current month, and press Enter.

The Job Scheduling screen is displayed.

**Figure 25  Job Scheduling Screen**

The Job Scheduling screen displays a calendar in graphic format. The dates within the specified date range on which the job is to be scheduled are marked with an asterisk.

--- **NOTE**

If the specified date range spans multiple months, a different calendar panel is displayed for each month, beginning with the first month. You can move between panels by pressing relevant PF keys. PF keys and their associated commands are explained in the following table.

### Table 8  PF Keys and Screen Navigation Commands for Job Scheduling Plan Screens

<table>
<thead>
<tr>
<th>PF Keys</th>
<th>Associated Commands</th>
</tr>
</thead>
<tbody>
<tr>
<td>PF07/PF19</td>
<td>UP (scroll backward)</td>
</tr>
<tr>
<td>PF08/PF20</td>
<td>DOWN (scroll forward)</td>
</tr>
<tr>
<td>PF10/PF22</td>
<td>PREV (where applicable)</td>
</tr>
<tr>
<td>PF11/PF23</td>
<td>NEXT (where applicable)</td>
</tr>
</tbody>
</table>
Months in the date range in which the job is not scheduled are not displayed.

You can also display the Job Scheduling screen from the Job Scheduling Definition screen. To do so, enter PLAN in COMMAND field. This is especially useful for checking your definition of a complex schedule while you are working on it.

31 Exit the Job Scheduling screen to display the Job List screen.

Defining JOB4

Save the definition of IDJOB3 and create IDJOB4, as explained in the following steps.

32 Exit the Job List screen.

The Exit option window is displayed.

Because you are saving the job scheduling definition in a table that already exists (IDGS1), you should use the SAVE option, not the CREATE option.

33 Specify Y in the SAVE field and leave the CREATE field blank.

The job scheduling definition is saved in the table. Because you bypassed the Table List screen when you entered the job scheduling definition, the entry panel is displayed.

You can now create JOB4 by specifying a new JOB name in the entry panel, as was done when you created IDJOB1 and IDJOB3, or by performing an Insert procedure in the Job List screen, as was done when you created IDJOB2.

Because JOB4 will be very similar to JOB2, you should use the Insert method. Before going to the Job List screen, however, you should first enter the Table List screen and select the table you want to use as a template for the new job.

34 Ensure the TABLE and JOB name fields are blank in the Entry panel, and enter the correct library name.

The Table List screen is displayed.

35 Enter the SELECT option for table IDGS1.

The Job List screen is displayed for table IDGS1. It lists jobs IDJOB1, IDJOB2, and IDJOB3.

36 Enter the INSERT option for IDJOB2.
A new job scheduling definition is displayed. It contains several values found in IDJOB2.

37 Make the following changes in the job scheduling definition.

A Specify IDJOB4 in the MEMNAME field.

B Change the value of the GROUP field to IDGRP2.

C Add the following description in the DESC field:

AUXILIARY JOB WHEN IDJOB3 ENDS NOTOK.

D Delete the ALL value from the DAYS parameter.

E Replace the existing IN condition with IDJOB3-END-NOTOK, but do not change the date value ODAT.

F Change the OUT condition to IDJOB4-ENDED-OK ODAT +.

The job scheduling definition appears as follows:

**Figure 26 Job Scheduling Definition for job IDJOB4**

<table>
<thead>
<tr>
<th>JOB: IDJOB4</th>
<th>LIB CTM.TEST.SCHEDULE</th>
<th>TABLE: IDGS1</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMAND ===</td>
<td>SCROLL===</td>
<td>CRSR</td>
</tr>
<tr>
<td>MEMNAME IDJOB4</td>
<td>MEMLIB CTM.TEST.JCL</td>
<td></td>
</tr>
<tr>
<td>OWNER ID</td>
<td>TASKTYPE JOB</td>
<td>PREVENT-NCT2 Y DFLT N</td>
</tr>
<tr>
<td>APPL GROUP IDGRP2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DESC AUDIT</td>
<td>JOB WHEN IDJOB3 ENDS NOTOK</td>
<td></td>
</tr>
<tr>
<td>OVERLIB</td>
<td>STAT CAL</td>
<td></td>
</tr>
<tr>
<td>SCHENV</td>
<td>SYSTEM ID</td>
<td></td>
</tr>
<tr>
<td>SET VAR</td>
<td>NJE NODE</td>
<td></td>
</tr>
<tr>
<td>CTB STEP AT</td>
<td>NAME</td>
<td>TYPE</td>
</tr>
<tr>
<td>DOCMEM IDJOB4</td>
<td>DOCLIB</td>
<td></td>
</tr>
<tr>
<td>DAYS</td>
<td>DCAL</td>
<td></td>
</tr>
<tr>
<td>WDDAYS</td>
<td>WCAL</td>
<td></td>
</tr>
<tr>
<td>MCODES</td>
<td>RETRO N MAXWAIT 00 D-CAT</td>
<td></td>
</tr>
<tr>
<td>DEFINITION ACTIVE FROM</td>
<td>UNTIL</td>
<td></td>
</tr>
<tr>
<td>IN IDJOB3-END-NOTOK</td>
<td>ODAT</td>
<td></td>
</tr>
<tr>
<td>CONTROL RESOURCE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIPE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FROM TIME</td>
<td>+ DAYS</td>
<td>UNTIL TIME</td>
</tr>
<tr>
<td>DUE OUT TIME</td>
<td>+ DAYS</td>
<td>PRIORITY 00 SAC CONFIRM</td>
</tr>
<tr>
<td>TIME ZONE:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OUT IDJOB4-ENDED-OK</td>
<td>ODAT +</td>
<td></td>
</tr>
<tr>
<td>AUTO-ARCHIVE</td>
<td>SYSDB</td>
<td>MAXDAYS MAXRUNS</td>
</tr>
<tr>
<td>RETENTION: # OF DAYS TO KEEP</td>
<td># OF GENERATIONS TO KEEP</td>
<td></td>
</tr>
</tbody>
</table>
You can save the job using the following procedure:

38 Exit the job scheduling definition.

The Job List screen is displayed.

39 Exit the Job List screen.

The Exit option window is displayed.

40 Specify Y in the SAVE field and leave the CREATE field blank.

This time, because you entered the Table List screen on the way to the job scheduling definition, the Table List screen is displayed.

41 Enter the SELECT option for table IDGS1.

Shown below is that section of the Job List screen for table IDGS1 that lists jobs IDJOB1, IDJOB2, IDJOB4 and IDJOB3.

**Figure 27  Job List Screen**

<table>
<thead>
<tr>
<th>OPT</th>
<th>NAME</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>IDJOB1</td>
<td>THIS JOB MUST BE RUN BEFORE IDJOB2</td>
<td></td>
</tr>
<tr>
<td>IDJOB2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IDJOB4</td>
<td>AUXILIARY JOB WHEN IDJOB3 ENDS NOTOK</td>
<td></td>
</tr>
<tr>
<td>IDJOB3</td>
<td>FISCAL RESULTS ON LAST FRIDAY OF MONTH</td>
<td></td>
</tr>
</tbody>
</table>

You can save the job using the following procedure:

38 Exit the job scheduling definition.

The Job List screen is displayed.

39 Exit the Job List screen.

The Exit option window is displayed.

40 Specify Y in the SAVE field and leave the CREATE field blank.

This time, because you entered the Table List screen on the way to the job scheduling definition, the Table List screen is displayed.

41 Enter the SELECT option for table IDGS1.

Shown below is that section of the Job List screen for table IDGS1 that lists jobs IDJOB1, IDJOB2, IDJOB4 and IDJOB3.

**Figure 27  Job List Screen**

<table>
<thead>
<tr>
<th>JOB LIST</th>
<th>LIB: CTM.TEST.SCHEDULE</th>
<th>TABLE: IDGS1</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMAND</td>
<td></td>
<td>SCROLL&gt;&gt;&gt; CRSR</td>
</tr>
<tr>
<td>OPT</td>
<td>NAME ----- DESCRIPTION -----------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>IDJOB1</td>
<td>THIS JOB MUST BE RUN BEFORE IDJOB2</td>
<td></td>
</tr>
<tr>
<td>IDJOB2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IDJOB4</td>
<td>AUXILIARY JOB WHEN IDJOB3 ENDS NOTOK</td>
<td></td>
</tr>
<tr>
<td>IDJOB3</td>
<td>FISCAL RESULTS ON LAST FRIDAY OF MONTH</td>
<td></td>
</tr>
</tbody>
</table>

**Job Ordering and Job Forcing**

In the previous chapter, you ordered IDJOB1 and IDJOB2 by requesting the O (Order) option for those jobs in the Job List screen. The requested job orders were placed in the active environment, and you saw those job orders listed in the Active Environment screen.
An ordered job is placed in the active environment only if its Basic scheduling criteria are satisfied when the job is ordered, that is, only if the job should otherwise be ordered for that day. This was the case when you ordered IDJOB1 and IDJOB2, because both jobs had a DAYS value of ALL and a Y (Yes) value for every month, which means that the Basic scheduling criteria of the job are satisfied every single day of the year.

In this last exercise, you made IDJOB3 eligible for ordering on the last Friday of each month. If you try to order the job on any other day, it will not be placed in the active environment. There is an option, however, that enables you to force the job into the active environment even when its Basic Scheduling criteria are not satisfied. This is the F (Force) option that is shown at the bottom of the Job List screen.

The following procedure explains how to force the scheduling of IDJOB3. You should be aware that you do not force IDJOB4, because IDJOB4 is a conditional job that is forced by a DO FORCEJOB statement when job IDJOB3 fails with a condition code of C0008 or above.

42 Enter option F (Force) for IDJOB3, as shown in the following Job List screen.

**Figure 28 Job List Screen**

43 If a Confirmation window is displayed, confirm the forcing of the job.

The Message window now displays a confirmation of the job forcing.

44 Exit the Message window.

The Job List screen is again displayed.
You can now display the Active Environment screen and see the results of the forced scheduling of IDJOB3.

45 Enter =3 in the COMMAND field of the Job List screen.

The Active Environment screen is displayed. IDJOB3 is in Wait Schedule status. This means that the Runtime Scheduling criteria of the job are not satisfied.

46 Ensure that filter IDGS is displayed. If it does not appear in the Filter field, enter SHOW IDGS in the COMMAND field.

47 If the Option line is not displayed at the bottom of the screen, enter OPT in the COMMAND field to display the options.

Through a combination of screens, you can determine why the job is in a Wait Schedule status and correct the problems so that the job is submitted. Some of those screens are displayed and explained in the following sections.

**Why Screen**

To see why the job did not run, request the ? (Why) screen. This screen can be accessed from the Active Environment screen by requesting the ? option for the job. The ? option is the first option listed in the option information line at the bottom of the screen.
Enter option ? (Why) for IDJOB3.

The Why screen is displayed for IDJOB3. As the title line of the screen indicates, this screen provides a scheduling analysis for the job.

**Figure 29  Why Screen**

| COMMAND ====> IDJOB3 SCHEDULING ANALYSIS ====>(3.?). |
| OPT DESCRIPTION |
|-----------------|-----------------|-----------------|
| TIME LIMIT FROM 2300 UNTIL 0400 |
| RESOURCE IDTAPE-DRIVE QUANTITY 0001 |
| RESOURCE IDCPOWER QUANTITY 0005 |
| CONDITION IDJOB3-ENDED-OK ODATE 0202 |

---

**NOTE**

Your site-defined start of workday, that is, the time of New Day Processing, can impact how the TIME FROM and UNTIL range determines when the job should be submitted.

For more information, see the discussion of date definition concepts in Chapter 1, “Introduction to Control-M,” and the description of the TIME parameter in the Control-M for z/OS User Guide.
The two missing RESOURCES (IDTAPE-DRIVE and IDCPOWER) and their required quantities are listed. These quantities are not listed if by chance you actually have resources with these names in use and allotted at your site.

Finally, the missing IN CONDITION (IDJOB3-ENDED-OK) is listed, unless you have an actual job that added this condition.

The problems listed above are the runtime scheduling requirements that prevent your job from running.

Runtime scheduling requirements are defined for a purpose, and it is generally not desirable to nullify them or work around them. However, sometimes it does become desirable to manually intervene to enable job submission, and Control-M allows for such intervention. The following sections illustrate how you can correct the problems and enable the job run.

Adding a Missing Condition in the Why Screen

Begin by adding the missing condition. You can do this in the Why screen. Notice the A (ADD CONDITION) option listed at the bottom of the Why screen.

49 Tab to the CONDITION line and request option A.

A confirm window is displayed.

50 Confirm the request.

Figure 30 Confirmation Window for Adding Condition in Why Screen

| COMMAND ==| IDJOB3 SCHEDULING ANALYSIS | SCROLL==| CRSR |
| OPT DESCRIPTION |
| TIME LIMIT | FROM 2300 | UNTIL 0400 |
| RESOURCE | IDTAPE-DRIVE | QUANTITY 001 |
| RESOURCE | IDCPOWER |
| A CONDITION | IDJOB3-ENDED-OK |
| END OF "WH" | CONFIRM Y DDMM 0202 |
The condition disappears from the Why screen, and a message appears at the top of the screen stating that the condition has been added.

**IOA Conditions/Resources Screen**

As already mentioned, two quantitative resources, IDTAPEDRIVE (1 unit) and IDCPOWER (5 units), are missing. You can add these missing conditions in the IOA Conditions/Resources screen.

51 Enter =4 in the COMMAND field of the Why screen.

The IOA Conditions/Resources screen is displayed.

The IOA Conditions/Resources screen displays the list of existing Control and Quantitative resources, which are stored in the Control-M Resources file, and the list of existing conditions stored in the IOA Conditions file.
Filtering the IOA Conditions/Resources Screen

When the list of conditions and resources in the IOA Conditions/Resources screen is very long, say ten or twenty or even fifty pages, it can be very useful to filter the display.

You can filter the display based on prefix and/or based on type of resource or condition. You can now filter the display so that only conditions and resources beginning with your ID are showing.

52 Enter ID in the PREFIX field. Ensure that the COND, CONTROL and RES values are set to Y (Yes) to display conditions, Control resources and Quantitative resources.

Only the condition you previously added in the Why screen should appear, unless there are other conditions and resources that are prefixed by your ID. The display appears as shown below:

<table>
<thead>
<tr>
<th>COMMAND ====&gt;</th>
<th>SCROLL ====&gt; CRSR</th>
</tr>
</thead>
<tbody>
<tr>
<td>PREFIX ====&gt; ID</td>
<td>COND Y CONTROL Y RES Y STAT Y</td>
</tr>
<tr>
<td>OPT TYPE</td>
<td>CONDITION/RESOURCE</td>
</tr>
<tr>
<td>COND</td>
<td>idJOB3-ENDED-OK</td>
</tr>
</tbody>
</table>

Adding a Condition or Resource

Using the ADD command listed at the bottom of the screen, you can now add the missing resources to the Control-M Resources file.

53 Type ADD RESOURCE in the COMMAND field of the IOA Conditions/Resources screen.

The Add Resources window is displayed.

54 Enter IDTAPE-DRIVE in the NAME field and 0001 in the COUNT field, and press Enter.

Figure 32 Add Resources Window in IOA Conditions/Resources Screen

<table>
<thead>
<tr>
<th>COMMAND ====&gt;</th>
<th>SCROLL ====&gt; CRSR</th>
</tr>
</thead>
<tbody>
<tr>
<td>PREFIX</td>
<td>PLEASE FILL IN QUANT RES NAME, COUNT AND PRESS ENTER</td>
</tr>
<tr>
<td>OPT TY</td>
<td></td>
</tr>
<tr>
<td>CO</td>
<td>NAME ====&gt; IDTAPE-DRIVE</td>
</tr>
<tr>
<td>======</td>
<td>&lt;&lt;</td>
</tr>
</tbody>
</table>
The window is closed and the resource is added to the Control-M Resources list in the specified quantity.

55 Repeat the previous step, but specify **IDCPOWER** in the NAME field, and a **0005** in the COUNT field.

The IOA Conditions/Resources screen now contains the required condition and the necessary quantities of the required resources.

Note that the condition you added from the Why screen could have instead been added using the ADD command in the IOA Conditions/Resources screen. In this case, the command would have been ADD CONDITION instead of ADD RESOURCE. A Control resource can be add using the command ADD CONTROL.

Note also, in accordance with the instruction line at the bottom of the screen, that the IOA Conditions/Resources screen permits the deletion of resources and conditions.

### A Second Look at the Why Screen

You can now return to the Why screen and see the results of your adding the missing condition and resources.

56 Enter **=3** in the COMMAND field of the IOA Conditions/Resources screen.

The Why screen, which was your last displayed screen in the Tracking and Control facility, is displayed.

57 Exit from the Why screen and return to it.

Now that the required conditions and resources have been added, the Why screen shows the only remaining problematic runtime criteria: The TIME LIMIT problem.
You can correct this remaining problem in a screen called the Zoom screen, which is described in the following section.

**Zoom Screen**

The Zoom screen is very similar to the Job Scheduling Definition screen. Instead of displaying the permanently defined job scheduling definition, however, the Zoom screen displays the job scheduling definition of the requested job order.

In the Zoom screen, you can change or delete TIME FROM and TIME UNTIL criteria for a job order. You can now display the Zoom screen, which is requested from the Active Environment screen.

58 Exit the Why screen.

The Active Environment screen is displayed. Note the Z (Zoom) option at the bottom of the Active Environment screen. This option displays the Zoom screen.

You will recall from Chapter 2, “Scheduling Definition and Manual Intervention,” that before you could delete a job from the Active Environment screen, you had to place the job in Held status. To make changes through the Zoom screen, you must first hold the job in the Active Environment screen, and then enter the Zoom screen.

59 Enter option H for IDJOB3. Press Enter several times until the status of the job is changed to Held Wait Schedule.

60 Enter option Z for IDJOB3.
The Zoom screen is displayed.

**Figure 33  Zoom Screen**

There are differences between the permanent definition displayed in the Job Scheduling Definition screen, and the job order definition displayed in the Zoom screen. For example, the definition in the Zoom screen does not display Basic scheduling parameters—these parameters are irrelevant once the job order is in the Active Environment. And the Zoom screen displays parameters such as SEARCH COUNTER, which are not found in the job scheduling definition, but which are relevant to the job order processing and have values set by Control-M.

The runtime scheduling parameters you defined for IDJOB3 appear in the Zoom screen.

Of the runtime scheduling criteria you defined, the TIME FROM and TIME UNTIL range remains the only problem. You can resolve this by simply changing the TIME values in the Zoom screen.
61 Change TIME FROM to a value that is an hour earlier than your current system time. Change the UNTIL value to a time that is several hours after the current system time.

At this point, it is important that you note the following:

- Any change to a job order in the Zoom screen changes the value for that job order only. The permanent definition that you defined in the Job Scheduling Definition screen remains unchanged.

- Instead of using the Control-M Why and IOA Conditions/Resources screens to satisfy the other runtime parameters, you could have deleted those parameters from the job order in the Zoom screen.

- The most important thing to remember is that in actual production circumstances, you do not merely delete parameter values that “get in the way,” and you do not intervene in the job control process unless you are fully aware of the potential impact of that intervention.

These exercises are designed to show you the workings of scheduling parameters and to show you how you can intervene if absolutely necessary. But runtime criteria are defined for a reason, and unless you are certain of what you are doing, intervention can do much more harm than good. For example, intervening to cause the submission of a job before its predecessor job has run can potentially do great damage.

Therefore, intervention in job processing should be left only to those who are specifically authorized and who have the knowledge necessary to intervene appropriately.

You can now exit the Zoom screen and see the result.

62 Enter SAVE in the COMMAND field of the Zoom screen. The Active environment screen is displayed.

Freeing a Held Job

Because you held the job before entering the Zoom screen, the job cannot continue processing until you free it. You do this by entering option F (Free) for the job.

63 Enter option F for the job order for IDJOB3. A message at the top of the screen indicates that the freeing of the job is performed. The status of the job changes to Requested Free Wait Schedule.
Periodically refresh the Active Environment screen by pressing **Enter**.

During this process, you see the following actions:

- IDJOB3 changes to Wait Schedule status, and then goes through the regular submission process. The job ends NOTOK (with a condition code of C0008).

- The Shout message you defined is sent to your terminal. The message is similar to the following:

```
CTM- FRIDAY FISCAL RESULTS RUN FAILED  02.02 14:12
CN(INTERNAL)
```

- As a result of the condition code from IDJOB3, job IDJOB4 is forced and is displayed in the Active Environment screen. Job IDJOB4 ends OK.

If desired, you can check the results of the job runs in the Log file. As you did in the last chapter, you can enter option **L** for either or both jobs.

Alternatively, you can enter **=5** in the COMMAND line to request option 5 of the IOA Primary Option menu. This option displays the IOA Log screen. Once in the IOA Log screen, you can define a filter to filter the job display in much the same way that you did in the Active Environment screen.

Exit the Online facility.

This completes the steps in this chapter of the *Control-M for z/OS Getting Started Guide*. 

---

**Example Output:**

```
Filter: IDGS
O Name    Owner    Odate  Jobname  JobID   Typ ----------- Status ------------
IDJOB3    ID       020201                  JOB Requested Free Wait Schedule
IDJOB4    M21     020201 /31867   JOB Ended "OK"
```

---

```
Filter: IDGS
O Name    Owner    Odate  Jobname  JobID   Typ ----------- Status ------------
IDJOB3    ID       020201                  JOB Requested Free Wait Schedule
IDJOB4    M21     020201 /31867   JOB Ended "OK"
```
In this chapter you

- looked closely at the sections in the Job Scheduling Definition screen, and learned about the following parameters that were not discussed in the last chapter
  - W D A Y S
  - M O N T H S
  - (D A T E S)
  - C O N T R O L
  - R E S O U R C E
  - T I M E F R O M / U N T I L
  - P R I O R I T Y
  - O N
  - D O C O N D
  - D O S H O U T
  - D O F O R C E J O B

- used the following options of the Job List screen
  - P (Display the Job Scheduling Plan screen)
  - F (Force a job order)

- used the following options of the Active Environment screen
  - H (Hold)
  - F (Free)
  - ? (Display the Why screen)
  - Z (Display the Zoom screen)

- viewed the scheduling plan of a job in the Job Scheduling Plan screen

- used the Why screen to see why a job didn’t run, and to add a missing condition

- filtered the display of the IOA Conditions/Resources screen, and used the screen to add Quantitative resources

- used the Zoom screen to alter the runtime parameters of a job order

- learned about the difference between ordering and forcing a job

- learned more prerequisite conditions and their associated dates, including the PREV date
Recommended Reading

Before continuing with the next chapter, it is recommended that you read the following in the Control-M for z/OS User Guide.

In Chapter 2

Description of the Job Scheduling Plan screen, Why screen, Zoom screen and IOA Conditions/Resources screen

In Chapter 3

Detailed parameter descriptions of the parameters you have used (mentioned above, under “Review”)

--- NOTE ---

The descriptions of the ON and WDAYS parameters are quite detailed. Read those descriptions for a general understanding of parameter usage and possible parameter formats, without trying to memorize all the detail. For now, you can skip the information related to subparameter WCAL in the WDAYS parameter description.
Restarts under Control-M/Restart

This chapter includes the following topics:

Overview ................................................................. 92
Preparations ............................................................... 92
Defining Restart in the Job Scheduling Definition ...................... 93
Editing JCL from the Active Environment ................................ 99
Restarting the Job ...................................................... 100
Review ................................................................. 104
  Recommended Reading ............................................. 104
Overview

In the last chapter, you used a DO FORCEJOB statement in an ON/DO block to force a “remedial” job following a job failure. However, rather than run a remedial job following job failure, it is more likely that you will want to correct the problem and then restart the job that failed.

In this chapter, you will learn to use Control-M/Restart to perform job restarts when they become necessary. Before you do, however, you should be clear about the difference between a job rerun and a job restart.

Job rerun is the re-execution of a scheduled job, starting from the beginning. For example, if a job fails, the entire job can be rerun. At best, rerunning a job can waste processing time on already successfully completed job steps; and unless certain precautions are taken, if successful job steps from the prior run performed updates before the job failed, rerunning the job can create problematic results by repeating those updates.

Job restart is the re-execution of a job beginning at a particular step. In general, the results of successful job steps before the failure are utilized, and re-execution continues from the end of the last successful step.

Control-M/Restart automates restart by identifying the step at which to initiate a job restart, and by performing necessary tasks to ensure that job restart is error-free.

Two separate processes are required for Control-M/Restart to restart under Control-M:

- Defining the restart parameters in the Control-M job scheduling definition appropriately, so that restart can be performed if it becomes necessary.

- Activating the restart process when restart becomes necessary.

In this chapter, you will define restart parameters in the job scheduling definition, and then, following job failure and correction of the problem, see and involve yourself in the process of restart.

Preparations

For this chapter, you will create a new job and its job scheduling definitions. You will use the name IDJOB5, and the same library and table that you used in the preceding chapter.
Defining Restart in the Job Scheduling Definition

To create the JCL for IDJOB5, begin with a copy of the TESTUTIL JCL, such as the JCL you used for IDJOB3. Recall that for job IDJOB3 this JCL had one step, which you defined to end with a condition code of C0008. For IDJOB5, you should make the following changes:

1. Change the step so that it ends with a condition code of C0000 (so the step ends OK).

2. Copy the changed step and its accompanying DD statements four times, so that you have five steps in the job. Name those steps S1, S2, S3, S4, and S5, respectively.

3. Change the third step (S3) so that it ends with a condition code of C0008.

Your job should now have five steps, four of which end okay, but one, the third, that ends with a condition code of C0008.

You can now continue with the first part of these exercises; the creation of the job scheduling definition.

Defining Restart in the Job Scheduling Definition

1 Enter the IOA Online Facility and open a job scheduling definition for job IDJOB5.

2 Ensure that the following values are part of the job scheduling definition:

- In the MEMNAME field, specify IDJOB5. Fill in the appropriate JCL library name in the MEMLIB field.
- In the GROUP field, specify IDGRP3.
- In the DESC field, specify RESTART JOB UNDER CTM/RESTART.
- In the DAYS field, specify ALL, and specify Y in all the MONTHS fields. These are the only Basic Scheduling parameters you should define.
- Do not defined any Runtime Scheduling parameters.
- In the OUT fields, define the condition

  IDJOB5-ENDED-OK   ODAT    +

You are now ready to define the ON and DO statements. These will include your restart parameters.
3  In the ON block, specify ANYSTEP as the program step (PGMST) value. Specify >C0004 as the codes value.

The meaning of the >C0004 value should be clear. Generally, a step is considered to have ended OK if the routine returns a code of C0004 or lower. Therefore, a CODES value of >C0004 instructs Control-M to perform the accompanying DO statements if the job ended NOTOK.

In the preceding chapter, and again in this step, you specified ANYSTEP as the PGMST step values. These ON step values should be examined more closely.

**ON Steps**

The ON step fields identify the possible steps against which Control-M will check for the specified CODES values.

Two types of step values can be specified:

- Program Step (PGMST) value
- Procedure Step (PROCST) value

You can specify either or both types of values, but you must specify at least one value if you use an ON block.

Valid step values can be any of the following:

- Literal value (for example, S3, if this is a step name)
- Keyword value that represents a step

Valid keyword values are

- ANYSTEP, which is valid only as a PGMST value
  
  DO statements are performed if the CODES criteria are satisfied for any program step.

- +EVERY
  
  DO statements are performed only if the CODES criteria are satisfied for all steps, program and/or procedure, depending on the definition.

- Step Range Name, which is valid only for a PGMST value

  You can define a range of steps in the STEP RANGE statement, immediately above the ON statement. You must assign a name to this step range.
You can then specify this step range name, preceded by an asterisk, as the step value in the ON statement. The asterisk prefix in the ON statement instructs Control-M to check the program step range defined in the STEP RANGE field, rather than looking for an actual program step by that name.

DO statements are performed if the CODES criteria are satisfied for any program step in the step range.

4 Define the following DO COND statement as the first DO statement in the ON block:

IDJOB5-END-NOTOK  ODAT  +

You can now to define a Shout statement to the operator. However, because this is only an exercise and you do not want the shout to actually go to the operator, you will instead send the shout to yourself.

5 Define a DO SHOUT statement as the next DO statement in the ON block. The target location, or TO value, defaults to your user ID. Specify the message:

IDJOB5 RUN FAILED. CORRECT PROBLEM AND CONFIRM RESTART.

You are now ready to define your restart parameters. Two statements are generally used in combination, to define a restart:

- DO IFRERUN
- DO RERUN

The DO IFRERUN statement is used to indicate that a restart is desired. It defines parameters to be used for restart, such as FROM step and TO step. This statement tells Control-M that if the job is rerun, hence the name DO IFRERUN, it should be rerun in accordance with the restart values specified in the DO IFRERUN statement.

The DO RERUN parameter merely instructs Control-M to run the job again. If you wanted a rerun without a restart, you would specify only the DO RERUN parameter, and the job would be rerun from the beginning.

The DO IFRERUN statement precedes the DO RERUN statement.

6 Enter IFRERUN in the blank DO statement. The DO IFRERUN statement is opened.

Note that this statement contains FROM, TO, and CONFIRM fields. The default value of the CONFIRM field should be N (No).

7 Set the CONFIRM value to Y (Yes).
Defining Restart in the Job Scheduling Definition

--- NOTE ---
The CONFIRM field of the DO IFRERUN statement is similar in meaning to the CONFIRM runtime scheduling parameter that you used in Chapter 1, “Introduction to Control-M.” However, it applies only to restarted jobs, whereas the CONFIRM runtime scheduling parameter applies to all job runs.

You can now take a look at the FROM and TO fields.

The FROM field indicates the step from which the restart should begin, and the TO field indicates the step to which the restart should continue.

Both the FROM and the TO fields allow specification of a program step, to the left of the period, and/or a procedure step, to the right of the period. You can specify either or both values.

- A TO step value can only be a literal value, such as S3, if this is a step name, or blank. If no TO value is specified, job processing continues to the end.
- The FROM field allows specification of either a literal value or a keyword that represents a step. Some of the valid keyword values for the FROM step are listed in the following table:

Table 9  DO IFRERUN: Selected FROM Keyword values

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$FIRST</td>
<td>First step of the job.</td>
</tr>
<tr>
<td>$ABEND</td>
<td>Step of the job that ended NOTOK due to system abend, user abend, condition code C2000 (PL/1 abend), or JFAIL (job failed on JCL error). $ABEND is a subset of $EXERR, described below.</td>
</tr>
<tr>
<td>$FIRST.$ABEND</td>
<td>First step of the abended procedure.</td>
</tr>
<tr>
<td>$EXERR</td>
<td>Job step that ended with any error, including an abend, or that ended with a condition code that is redefined using the ON and DO statements, as ENDED NOTOK</td>
</tr>
</tbody>
</table>

8 Specify $EXERR in the FROM field.

9 Enter RERUN in the blank DO statement. This statement has no subparameter values.

Before exiting the job scheduling definition, return and define one more OUT condition.

10 In the OUT condition line, add the following condition as the second condition on that line:
If the job fails, and then successfully finishes following restart, this OUT condition deletes the DO COND condition that was added by the job failure. If the job does not fail, there is no IDJOB5-END-NOTOK condition to delete. In this case, no deletion occurs and processing continues.

Your job scheduling definition appears as shown in Figure 34.

**Figure 34  Job Scheduling Definition**

```
JOB: IDJOB5  LIB CTM.TEST.SCHEDULE  TABLE: IDGS1
COMMAND ===>  SCROLL==> CRSR
+-----------------------------------------------------------------------------+
MEMNAME IDJOB5  MEMLIB CTM.TEST.JCL
OWNER ID  TASKTYPE JOB  PREVENT-NCT2  DFLT N
APPL
DESC
OVERLIB  STAT  CAL
SCHENV  SYSTEM ID  NJE  NODE
SET VAR
CTB  STEP  AT  NAME  TYPE
DOCMEM IDJOB5  DOCLIB
+-----------------------------------------------------------------------------+
DAYS  ALL  DCAL
WDAYS  AND/OR
DATES  CONFCAL  SHIFT  RETRO N MAXWAIT 00  D-CAT
MINIMUM  PDS
DEFINITION ACTIVE FROM  UNTIL
+-----------------------------------------------------------------------------+
IN  CONTROL
RESOURCE
PIPE
FROM TIME  +  DAYS  UNTIL TIME  +  DAYS
DUE OUT TIME  +  DAYS  PRIORITY  SAC  CONFIRM Y
TIME ZONE:
+-----------------------------------------------------------------------------+
OUT  IDJOB5-ENDED-OK  ODAT  +  IDJOB5-END-NOTOK  ODAT  -
AUTO-ARCHIVE Y  SYSDB Y  MAXDAYS  MAXRUNS
RETENTION: # OF DAYS TO KEEP  # OF GENERATIONS TO KEEP
SYSOUT OP (C,D,F,N,R)  FROM
MAXRERUN  RERUNMEM  INTERVAL  FROM
STEP RANGE  FR (PGM.PROC)  TO .
ON PGMST ANYSTEP  PROCST  CODES  $0004  A/O
  DO COND  IDJOB5-END-NOTOK  ODAT +
  DO SHOUT TO ID  URGENCY R
  = IDJOB5 RUN FAILED, CORRECT PROBLEM AND RESTART
  DO IFRERUN FROM $EXERR .  TO .  CONFIRM Y
  DO RERUN
  DO
ON PGMST  PROCST  CODES  A/O
  DO SHOUT WHEN  TIME  +  DAYS  TO  URGN
```
11 Exit the Job Scheduling Definition screen to the Job List screen.

12 Exit the Job List screen and save the changes, by entering Y in SAVE field of the Exit Option window. The Table List screen is displayed.

13 Reenter the Job List screen for the table, and order job IDJOB5. The Job Order Messages screen is displayed. and the job is ordered.

14 Exit the Job Order Messages screen and display the Active Environment screen by entering =3 in the COMMAND field. The Active Environment screen is displayed.

15 Ensure that filter IDGS is displayed. If it does not appear in the Filter field, enter SHOW IDGS in the COMMAND field).

16 Refresh the display as often as needed. A message similar to the following message is Shouted to your terminal:

---
CTM- IDJOB5 RUN FAILED. CORRECT PROBLEM AND RESTART 02.02 12:38 CN(INTERNAL)
---

The job goes through a process of submission and execution, and finally ends with the status: **Ended- Not “OK” Due to CC - Rerun Needed:**

The Active Environment screen appears as follows. If the jobs from the exercises in the preceding chapter have not been deleted by site maintenance, they will also appear in the screen.

---
Filter: IDGS --------- Control-M Active Environment ------- UP <D> - (3)
COMMAND ===> SCROLL ===> CRSR
O Name Owner Odate Jobname JobID Typ --------- Status ---------
IDJOB5 ID 020201 M21 /29162 JOB Ended- Not "OK" Due to CC -
Rerun Needed
---

The CC in the status refers to a condition code. You can now review the job log and identify the condition code problem.

17 Call up the log for the job by entering L in the OPTION field. The Control-M Log screen is displayed for the job.
Each event in the life cycle of the job appears as a message issued by Control-M. Note the following messages:

- **SEL216W** identifies a problem: An unexplained condition code of 0008 in step S3, reflecting what you deliberately defined in the job.

- **SEL214I** indicates that rerun is needed, reflecting what you deliberately defined in the job.

- **SEL219I** indicates that the job ended “NOT OK”

### Editing JCL from the Active Environment

Clearly there is no point in restarting a job that will continue to fail because the problem has not been corrected. But now that you know the cause of the failure, you can correct it and then restart the job.

You can correct the JCL of this job by using the JCL option in the Active Environment screen.

18 Exit the Control-M Log screen. The Active Environment screen is displayed.

Note the option J (JCL) at the bottom of the screen. If the list of commands is displayed instead of the list of options, enter the **OPT** command to toggle to the list of options.
19 Enter option J for the job. The JCL is displayed in ISPF edit mode.

20 Change the condition code of C0008 in step S3 to C0000, and exit the JCL. The Active Environment screen is displayed. You can now restart the job.

**Restarting the Job**

21 Enter option R (Rerun) for the job. Option R performs job rerun.

However, as was discussed earlier, when restart instructions are defined in a DO IFRERUN statement, restart is performed when the job is run again.

When a rerun is requested, a window is opened. The window is different for regular reruns and restart reruns. Because you defined a DO IFRERUN statement in the job scheduling definition, you see the Confirm Restart window:

**Figure 35 Confirm Restart Window**

![Confirm Restart Window]

In the top half of the window you see that

- the first line informs you which job (IDJOB5) is to be rerun
- the next line asks for confirmation, and you will shortly specify Y (Yes)
- the next line tells you that the rerun has been defined to include a restart, it defaults to Y, but you can specify N (No) if you prefer a full rerun
The bottom half of the window deals with restart information. In this exercise, you will only look at the first line, which tells you from which step, and to which step, the restart will be performed.

■ The FROM value is S3. This makes sense because steps S1 and S2 ended successfully.

■ The TO step is blank, which means that once restart begins, it will continue till the end of the job. Consider the following:

   — If you do not want the steps after the restart step to run again, you can specify restart step S3 as the TO step.

   — If you defined the JCL so that steps after the failed step do not run, and you want them to run following the restart, you should leave the TO value blank.

At this point, all you need to do is enter Y in the CONFIRM field, and the job will restart. However, you should not take that action at this time.

There might be instances in which you want the job to restart from a different step than the one determined by Control-M/Restart as the logical restart step. It is possible to change the FROM and TO steps in the Confirm Restart window. To facilitate this change, you can display the list of steps in the job.

Notice that ? is a valid value for the With Restart field. Entering ? displays the list of steps.

22 Enter ? in the With Restart field. The Restart Step List window is opened over the Restart Window.
Restarting the Job

The Control-M/Restart Step List window sequentially lists all the steps in the job, assigning each of them a sequence number.

At the bottom of the window are three options that can be specified in the O (Option) field for the appropriate step:

- Option F can be used to specify a From step.
- Option T can be used to specify a To step.
- Option O can be used to indicate that only the specified step should be rerun.

You can specify option F for step S3, but this is not necessary, since that value is already indicated in the Restart Window. So just exit the Restart Step List window.

23 Press PF03/PF15 to exit the Restart Step List window. The Restart Step List window is closed, and the Restart Window is displayed.

24 Enter Y in the Please Confirm field. The Restart window is closed, and the rerun with restart now begins.

Notice the progression of status changes for the job in the Active Environment screen. When the rerun with restart is complete, the job appears as shown below:
Notice that there are two status descriptions for the job—each belonging to one of the runs:

- The current status, Ended “OK” (Restarted), applies to Run2. The job was successfully restarted.

- The original status with the problematic CC now appears as being associated with the prior run.

You can now look at the message log for the restarted job.

25 Call up the log of the job by entering L in the OPTION field. The Control-M Log screen is displayed for the job.

Notice the message SEL208I. This message indicates that the job ended “OK.”

26 Exit the Online facility.

This completes the steps in this chapter of the Control-M for z/OS Getting Started Guide.
Review

In this chapter you

- defined a restart in your job scheduling definition using parameters DO IFRERUN and DO RERUN
- learned valid restart step keyword values and specified that the job should restart from step $EXERR
- entered the JCL of the failed job from the Active Environment screen by entering the J (JCL) option, and corrected the JCL
- confirmed a rerun/restart request (Option C) for the failed job in the Active Environment screen, and in the process you displayed the Confirm Restart Window and Restart Step List window, in which you confirmed the restart
- checked the log of the job following the failed run, and again following the restart

Recommended Reading

Before continuing with the next chapter, it is recommended that you read the following:

- In the Control-M/Restart User Guide
  - all of Chapter 1.
  - in Chapter 2, the online material related to the Scheduling Definition facility and the Tracking and Control facility.
- In the Control-M for z/OS User Guide.
  - In Chapter 2, the description of the Control-M/Restart information related to the Confirm Restart window and Rerun/Restart window, and the Restart Step List window.
  - In Chapter 3, detailed parameter descriptions of parameters DO IFRERUN and DO RERUN.
SMART Table Scheduling

This chapter includes the following topics:

- Overview .......................................................... 106
- Preparations ...................................................... 106
- Creating a SMART Table ....................................... 107
  - SMART Table Entity ......................................... 108
  - Planned Logic of the Job Scheduling Definitions ................. 115
  - Jobs Scheduling Definitions in the SMART Table ................. 115
  - Saving the Table .............................................. 126
  - Points to Remember Concerning SMART Table Scheduling .... 126
  - SMART Table Entity Ordering and Job Ordering .................. 127
  - Advantages of SMART Table Scheduling ......................... 130
- Review .............................................................. 131
- Recommended Reading ......................................... 132
Overview

When defining job scheduling definitions, it is good practice to group together related job scheduling definitions. This is not a requirement—related jobs can belong to different groups, or not belong to any group—but it can be useful. For example, if related jobs belong to the same group, you can use the group name to filter the Active Environment screen so that only the jobs in that group are displayed. But greater functionality than just ease of filtering a display can be provided by assigning a set of jobs to a SMART Table.

The major advantage of defining SMART Tables is that they enable you to define basic scheduling criteria, runtime scheduling criteria and post processing criteria that apply to the jobs in the SMART Table as a whole.

Preparations

In the exercises in the previous chapters of this guide, you created the JCL for five jobs, IDJOB1 through IDJOB5. To simplify preparation for this chapter, you will utilize all five of these JCLs.

In addition, you will utilize one of the job scheduling definitions that you created in Chapter 1, “Introduction to Control-M,” the job scheduling definition for IDJOB1, in table IDGS1. For the remaining jobs, you will create new job scheduling definitions in a new table (IDGS2), as part of the exercises.

Before continuing with the exercises, you should perform the following preparatory steps:

- In the JCL of IDJOB5, change the JCL of step S5 so that it sets a return code of C0008 instead of C0000.
- Ensure the following, which should be the case if you completed all of the steps in the previous chapters of this guide:
  - The JCL for step S1 in IDJOB3 sets a return code of C0000, and not C0008.
  - All conditions from previous exercises in this guide do not appear in, or are deleted from, the IOA Conditions/Resources screen.
  - All job orders from previous exercises in this guide do not appear in, or are deleted from, the Active Environment screen.

You are now ready to look at SMART Table scheduling. For purposes of these exercises, you should assume that the new table contains jobs that must run at end of week and at end of month.
Creating a SMART Table

1. Enter the IOA Online Facility and display the Scheduling Definition Facility entry panel.

In this chapter, you will create a new table called IDGS2; in the previous chapters of this guide you used a table called IDGS1.

Notice the field SMART TABLE. Valid values for this field are Y (SMART Table) or [Enter] (Table that handles jobs individually). In previous chapters of this guide, you ignored this field because the default value [Enter] served your purpose when you worked with table IDGS1. Now, however, you want to define the new table, IDGS2, to be a SMART Table.

A SMART Table contains a mechanism for defining parameters that apply to the table as a whole.

In the previous chapters, whenever you defined a job scheduling definition in table IDGS1, the parameters in the job scheduling definition could only be applied to that job. For example, a TIME FROM value, or a DO SHOUT value, in the job scheduling definition for IDJOB3, applied only to IDJOB3. If you wanted to apply the same parameters to IDJOB4, you had to define the parameters in the job scheduling definition of IDJOB4.

SMART Tables have a special kind of scheduling definition, called a SMART Table Entity, that is used to apply parameters to all jobs in the table. Each SMART Table has one, and only one, SMART Table Entity, as well as its job scheduling definitions. When you first request creation of a SMART Table from the Scheduling Definition entry panel, the SMART Table entity definition screen is displayed.
SMART Table Entity

You can now request creation of SMART Table IDGS2, and take a look at the SMART Table Entity.

2. In the Entry Panel, enter IDGS2 in the TABLE field and Y in the SMART TABLE field. (Note: Once you specify a SMART Table name in the TABLE field, you will not be able to change the name from the SMART Table Entity Scheduling Definition panel.)

The SMART Table Entity scheduling definition is displayed.

**Figure 37  SMART Table Entity Scheduling Definition**

<table>
<thead>
<tr>
<th>TBL</th>
<th>CTM.TEST.SCHEDULE(IDGS2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMAND</td>
<td>CTM.TEST.SCHEDULE(IDGS2)</td>
</tr>
<tr>
<td>SCROLL</td>
<td>CRSR</td>
</tr>
<tr>
<td>TBL</td>
<td>CTM.TEST.SCHEDULE(IDGS2)</td>
</tr>
<tr>
<td>LIBRARY</td>
<td>CTM.TEST.SCHEDULE</td>
</tr>
<tr>
<td>TABLE</td>
<td>IDGS2</td>
</tr>
<tr>
<td>JOB</td>
<td></td>
</tr>
<tr>
<td>SMART TABLE</td>
<td>Y (Y - for new SMART Table)</td>
</tr>
</tbody>
</table>

Although the SMART Table Entity scheduling definition is similar to the job scheduling definition, there are differences. The first thing to note is that the term TBL in the upper left corner identifies the definition as a SMART Table Entity.
The next thing to note is that the first field in the definition, under the COMMAND line, is the TABLE field, followed by the GROUP field on the same line. Note also that there is no MEMLIB field.

In a regular job scheduling definition, the MEMNAME parameter contains the name of the JCL member of the job. However, a SMART Table Entity does not have a JCL member.

By default, the name you specified in the TABLE field automatically gets placed in the DOCMEM field.

3 In the GROUP field, specify the name IDGRP4, which is the name of our group.

4 In the DESC field, specify SMART TABLE ENTITY FOR TABLE IDGS2. The entries in the General parameters section appear as shown below:

```
+----------------------------------------------------------------------------+
| TABLE IDGS2 GROUP IDGRP4 |
| OWNER ID                |
| APPL DESC               |
| SMART TABLE ENTITY FOR TABLE IDGS2 |
| ADJUST CONDITIONS N TBL MAXWAIT 05 |
| SET VAR DOCMEM IDGS2 DOCLIB |
+----------------------------------------------------------------------------+
```

### Schedule RBCs and Other Basic Scheduling Parameters

You can now look at the Basic Scheduling parameters section. The first field to note is the SCHEDULE RBC field. This field only appears in definitions in SMART Tables.

```
+----------------------------------------------------------------------------------------------------------+
| SCHEDULE RBC DAYS LEVEL DCAL AND/OR WDAYS WDATE MTHNCTS MTHNCTU MTHNCTY MTHNCTY MTHNCTY MTHNCTY MTHNCTY MTHNCTY |
| SCHEDULE RBC ACTIVE FROM UNTIL |
+----------------------------------------------------------------------------------------------------------+
```

5 In the SCHEDULE RBC field, specify the name EOW, which stands for END-OF-WEEK. In the LEVEL field, specify either TBL or CTM. (If the Schedule RBC’s level is set to TBL, the rule-based calendar is defined in the SMART Table Entity. If the Schedule RBC’s level is set to CTM, the rule-based calendar is defined in the IOA Calendar Facility.) In the WDAYS field, specify the value 0, which means the last day of the week.
Notice that a new, blank set of Basic Scheduling criteria opens up.

Thus far, in each of your job scheduling definitions, you have been limited to defining only one set of Basic scheduling parameters. This is because a job scheduling definition can have only one occurrence of the Basic Scheduling parameter section.

SMART TABLE Entities, on the other hand, can have multiple occurrences of the Basic scheduling parameters section. Once you have defined a set of Basic Scheduling parameters, a new, blank section of Basic Scheduling parameters opens to enable you to define another set.

You distinguish between the sets of Basic Scheduling parameters in the SMART TABLE Entity by assigning a name to each set in the SCHEDULE RBC field.

In the “Preparations” section above, you determined that the jobs in this table will run at the end of the week and at the end of the month. You have just defined the basic scheduling criteria for the end of the week run. You can now define the basic scheduling criteria for the end of the month run.

6 In the empty SCHEDULE RBC field, specify the name **EOM**, which stands for END-OF-MONTH. In the LEVEL field, specify either **TBL** or **CTM**. In the DAYS field, specify the value **L1**, which means the last day of the month.

In an actual operational environment, these sets of Basic Scheduling criteria would be enough to achieve your goals. However, because you want to be able to perform these exercises on any day of the month, you will define one more set of Basic scheduling criteria for purposes of this guide only.

7 In the blank SCHEDULE RBC field, specify the name **EXERCISES**. In the LEVEL field, specify either **TBL** or **CTM**. In the DAYS field, specify the value **ALL**.
Your entries in the Basic Scheduling parameter section are displayed as shown in Figure 38.

**Figure 38  Basic Scheduling Parameter Section**

<table>
<thead>
<tr>
<th>SCHEDULE RBC</th>
<th>EOW</th>
<th>LEVEL</th>
<th>TBL</th>
<th>DAYS</th>
<th>DCAL</th>
<th>AND/OR</th>
<th>WDCAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>WDAYS</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MONTHS</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DATES</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCHEDULE RBC</td>
<td>ACTIVE FROM</td>
<td>UNTIL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SCHEDULE RBC</th>
<th>EOM</th>
<th>LEVEL</th>
<th>TBL</th>
<th>DAYS</th>
<th>DCAL</th>
<th>AND/OR</th>
<th>WDCAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>WDAYS</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MONTHS</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DATES</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCHEDULE RBC</td>
<td>ACTIVE FROM</td>
<td>UNTIL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SCHEDULE RBC</th>
<th>EXERCISES</th>
<th>LEVEL</th>
<th>TBL</th>
<th>DAYS</th>
<th>DCAL</th>
<th>AND/OR</th>
<th>WDCAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>WDAYS</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MONTHS</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DATES</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCHEDULE RBC</td>
<td>ACTIVE FROM</td>
<td>UNTIL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SCHEDULE RBC</th>
<th>LEVEL</th>
<th>DAYS</th>
<th>DCAL</th>
<th>AND/OR</th>
<th>WDCAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>WDAYS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MONTHS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DATES</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCHEDULE RBC</td>
<td>ACTIVE FROM</td>
<td>UNTIL</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Runtime Scheduling Parameters**

All Runtime Scheduling fields found in the SMART TABLE Entity are also found in the job scheduling definition.

In the “Preparations” section above, you determined that you would continue to use the job scheduling definition of IDJOB1 in table IDGS1.

Assume that job IDJOB1, in table IDGS1, must end OK before any of the jobs in table IDGS2 can run. You will now add the condition necessary to establish this job dependency.
8 In the IN field, specify the condition **IDJOB1-ENDED-OK  ODAT**. Your entry in the Runtime Scheduling parameter section is displayed.

<table>
<thead>
<tr>
<th>IN</th>
<th>IDJOB1-ENDED-OK</th>
<th>ODAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONTROL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FROM TIME</td>
<td>+ DAYS</td>
<td>UNTIL TIME + DAYS</td>
</tr>
</tbody>
</table>

---

### On Table End and Other Post-Processing Parameters

Like the job scheduling definition, the SMART Table Entity contains OUT and SHOUT WHEN statements, and DO blocks. However, instead of regular ON blocks, the SMART Table Entity contains ON TABLE-END blocks.

Unlike the ON block, which applies to the individual job, the ON TABLE-END block applies to the jobs in the SMART Table as a whole. Valid values for the ON TABLE-END field are as follows:

**Table 10  Valid Values For the ON TABLE-END Statement**

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OK</td>
<td>Subsequent DO actions are performed for each job in the SMART Table only if the end status of the SMART Table Entity is OK, that is, all scheduled jobs in the SMART Table ended OK.</td>
</tr>
<tr>
<td>NOTOK</td>
<td>Subsequent DO actions are performed for each job in the SMART Table if the end status of the SMART Table Entity is NOTOK, that is, at least one job in the SMART Table ended NOTOK.</td>
</tr>
</tbody>
</table>

Not all DO statements that are valid in the job scheduling definition are valid in the SMART Table Entity. The DO statements listed below can be specified in the SMART Table Entity. Several of these DO statements have not been described in this guide, but they are all described in the *Control-M for z/OS User Guide*:

- DO COND
- DO OK
- DO MAIL
- DO FORCEJOB
- DO SET
- DO NOTOK
- DO SHOUT
You can now fill in the Post-processing parameters of the SMART Table Entity.

9 In the OUT field, specify the condition IDGS2-ENDED-OK ODAT +.

10 In the ON TABLE-END field, specify the value OK.

11 In the accompanying DO field, specify SHOUT, and enter the following message in the DO SHOUT statement:

   PROCESSING OF TABLE IDGS2 ENDED OK.

12 In the blank ON TABLE-END field, specify the value NOTOK.

13 In the accompanying DO field, specify COND, and fill in the DO COND statement with the following: IDGS2-END-NOTOK ODAT +.

14 In the next DO field, specify SHOUT, and enter the following message in the DO SHOUT statement:

   PROBLEM WITH TABLE PROCESSING. AT LEAST ONE JOB DID NOT END OK.

Your entries in the Post-processing parameter section are displayed as shown in Figure 39.

**Figure 39  Post-Processing Parameter Section**

```
+-----------------------------------------------------------------------------+
| OUT  IDGS2-ENDED-OK       ODAT +                                             |
| ON TABLE-END OK                                                         |
|  DO SHOUT  TO  TSO-M21        URGENCY R                          |
|    = PROCESSING OF TABLE IDGS2 ENDED OK                                  |
|  DO                                                         |
| ON TABLE-END NOTOK                                                      |
|  DO COND  IDGS2-END-NOTOK       ODAT +                                |
|  DO SHOUT  TO  TSO-M21        URGENCY R                          |
|    = PROBLEM WITH TABLE PROCESSING. AT LEAST ONE JOB DID NOT END OK      |
|  DO                                                         |
| ON TABLE-END                                                        |
|  DO SHOUT WHEN TIME + DAYS TO URGN                                  |
|                                                                 |
+-----------------------------------------------------------------------------+
```
Logic of the SMART Table Entity Scheduling Definition

The following points about the logic of your SMART Table Entity should be noted:

■ The SMART Table Entity is used to help determine and control the processing of the jobs in the table.

  — You defined the following three sets of Basic Scheduling Criteria:

  — EOW criteria result in jobs being scheduled on the last day of the week.

  — EOM criteria result in jobs being scheduled on the last day of the month.

  — EXERCISES criteria result in jobs being scheduled every day, for purposes of the exercises in this guide.

You will soon see how these sets of scheduling criteria apply to the job scheduling definitions in this table.

■ You defined an IN condition: IDJOB1-ENDED-OK. This condition is set when job IDJOB1 in table IDGS1 ends successfully. This means that successful completion of IDJOB1 is a prerequisite to the submission of the jobs in table IDGS2 that are scheduled according to Basic scheduling criteria in the SMART Table Entity.

■ You defined an OUT condition that is set only when all scheduled jobs in table IDGS2 end OK. This condition can be used as an IN condition to some other job that requires successful completion of the jobs in table IDGS2 as a prerequisite.

■ You defined two sets of ON TABLE-END/DO parameters:

  — The first set applies if all the scheduled jobs in the table end OK. It shouts a message to that effect,

  — The second set applies if any of the scheduled jobs in the table ends NOTOK. It shouts a message to that effect; and it sets an appropriate condition that can, for example, be used to run an auxiliary job.

You are about to exit the SMART Table Entity and create the job scheduling definitions in table IDGS2. However, before doing so, outline the logic you would like to see in the job scheduling definitions.
Planned Logic of the Job Scheduling Definitions

As mentioned above, you want successful completion of job IDJOB1 to be a prerequisite to the running of the jobs in table IDGS2. In addition to this, the following logic should apply:

- Table IDGS2 will contain jobs IDJOB2, IDJOB3, IDJOB4, and IDJOB5.
- IDJOB2 is a prerequisite to IDJOB3, and IDJOB4 is a prerequisite to IDJOB5. However, IDJOB3 is NOT a prerequisite to IDJOB4.
- IDJOB2 should be run both at end of week and end of month, and during the exercise.
- IDJOB3 should be run following IDJOB2, at end of week or month, and during the exercise, provided that the day is a weekday, that is, on Monday, Tuesday, Wednesday, Thursday, or Friday.
- IDJOB4 should be run both at end of week and at end of month, but should also run on several other days of the month: the 8th, the 14th, and, so the exercise will work, the day that you perform this exercise.
- IDJOB5 should run following IDJOB4, but only at end of week and end of month, and during the exercise. It should not run the other days specified.

Jobs Scheduling Definitions in the SMART Table

You should now exit the SMART Table Entity and create the job scheduling definitions in table IDGS2.

Exit the SMART Table Entity. Upon exiting the SMART Table Entity the first time, a blank Job Scheduling Definition screen is displayed.
As with all job scheduling definitions, JOB: is displayed in the upper left corner.
Notice, however, that the GROUP field already has a value, which is IDGRP4. The name specified in the GROUP field of the SMART Table Entity is not automatically assigned as the GROUP name of the jobs in the SMART Table. Only jobs whose GROUP name is blank are assigned the GROUP name from the SMART Table Entity.

Notice that job scheduling definitions in SMART Tables contain two Basic Scheduling parameter fields not found in tables with disabled SMART Table attributes: SCHEDULE RBC and RELATIONSHIP.

The SCHEDULE RBC field enables you to assign to the job scheduling definition any desired sets of basic scheduling criteria that you defined in the SMART Table Entity.

16 Specify IDJOB2 in the MEMNAME field and the name of the JCL library you are using in the MEMLIB field.

17 Specify JOB2 in the DESC field. Your entries in the General parameters section are displayed as follows:

Notice that job scheduling definitions in SMART Tables contain two Basic Scheduling parameter fields not found in tables with disabled SMART Table attributes: SCHEDULE RBC and RELATIONSHIP.
18 Specify **EOW** in the SCHEDULE RBC field. A new SCHEDULE RBC field opens up.

19 Specify **EOM** in the new SCHEDULE RBC field, and then specify **EXERCISES** in the third SCHEDULE RBC field that opens up.

This job is now eligible for scheduling if either the EOW or EOM sets of basic scheduling criteria in the SMART Table Entity are satisfied. The EXERCISES criteria must also be satisfied to enable you to complete these instructions.

Your entries in the Basic Scheduling parameters section are displayed as follows:

```
SCHEDULE RBC  EOW
SCHEDULE RBC  EOM
SCHEDULE RBC  EXERCISES
SCHEDULE RBC
RELATIONSHIP (AND/OR)  O
DAYS
                  DCAL AND/OR
WDAYS
DATES
CONF CAL   SHIFT   RETRO N MAXWAIT 05  D-CAT
MINIMUM   POS
DEFINITION ACTIVE FROM    UNTIL
```

It is unnecessary to specify an IN condition for this job. Although IDJOB1 is a prerequisite to this job, you already defined IDJOB1-ENDED-OK as an IN condition for the entire table.

Therefore, the Runtime Scheduling parameter section should be empty, as shown in **Figure 40**.

**Figure 40  Runtime Scheduling Parameter Section**

```
IN
CONTROL
RESOURCE
PIPE
FROM TIME   + DAYS UNTIL TIME   + DAYS
```

The Post-processing section requires an OUT condition for this job because it is a prerequisite to IDJOB3. But you must decide whether to define Post-processing parameters in case the job ends NOTOK, that is, you need to decide whether the SMART Table Entity post-processing parameters are sufficient, or whether IDJOB2 requires its own additional post-processing definitions. For purposes of this exercise, assume that the SMART Table Entity Post-processing actions are sufficient.
Now you can define the Post-processing parameters.

20 In the OUT field, specify the condition **IDJOB2-ENDED-OK ODAT +**. Your entries in the Post-processing parameters section are displayed as follows:

```
OUT   IDJOB2-ENDED-OK   ODAT +
AUTO-ARCHIVE Y   SYSDB Y   MAXDAYS   MAXRUNS
RETENTION:   # OF DAYS TO KEEP   # OF GENERATIONS TO KEEP
SYSOUT OP   (C,D,F,N,R) FROM
MAXRERUN   RERUNMEM INTERVAL FROM
STEP RANGE   FR (PGM.PROC)   . TO   .
ON PGMSR   PROCST CODES   A/D
DO
SHOUT WHEN   TIME +   DAYS TO   URGN
MS
```

21 Save this job scheduling definition and exit the job scheduling definition. The Job List screen is displayed.

```
JOB LIST   LIB: CTM.TEST.SCHEDULE   TABLE: IDGS2
COMMAND ===>   SCROLL==> CRSR
OPT  NAME ---   TYP ---   DESCRIPTION -----   GROUP: IDGRP4
---   ---   ---   ---
IDGS2   T   SMART TABLE ENTITY IN TABLE IDGS2
IDJOB2   J   PRELIMINARY CALCULATIONS
```

Both the SMART Table Entity and the job scheduling definition that you created are listed in the Job List screen. Note the TYP field. This field identifies the type of entity listed. Valid values are: T, which identifies SMART Table Entity, and J, which identifies Job Scheduling Definition.

You can now create the remaining job scheduling definitions, beginning with IDJOB3.

22 Perform an Insert (Option I) next to IDJOB2. A job scheduling definition that contains many of the same values as found in IDJOB2 is displayed.

23 Specify **IDJOB3** in the MEMNAME field (the MEMLIB field is already filled in).

24 Specify **JOB3** in the DESC field. The Basic Scheduling parameters found in IDJOB2 appear in IDJOB3.
Recall that you want this job to be scheduled at end of week and end of month (just like IDJOB2), but only if those days do not fall on a Saturday or Sunday. Therefore, you will keep these Basic Scheduling parameter values, and specify several additional values as well.

Previously, you learned that the SCHEDULE RBC field is followed by a RELATIONSHIP field and a set of Basic Scheduling Criteria:

- The Basic Scheduling criteria allow you to define scheduling criteria apart from the criteria found in the Schedule RBCs.
- The RELATIONSHIP field determines the relationship between these criteria and the Schedule RBCs. Valid values are A (And) or O (Or):
  
  - The A value is more restrictive. In addition to the Schedule RBC criteria being satisfied, the basic scheduling criteria of the job, itself, must also be satisfied.
  
  - The O value is less restrictive. Even if the Schedule RBC criteria of the job are not satisfied, the job can be scheduled if its basic scheduling criteria are satisfied.

You should schedule job IDJOB3 only if the EOM, EOW (or EXERCISES) criteria are satisfied and the day is weekday (Monday through Friday).

25 Specify A (And) in the RELATIONSHIP field.

26 Specify 1,2,3,4,5 in the WDAYS field.

---

**NOTE**

You want this job to be scheduled on the day you perform this exercise. Adjust the WDAYS value so that it includes the day on which you are working.
Your entries in the Basic Scheduling parameters section are displayed as follows, except as adjusted to ensure that the WWDAYS field contains your current working day:

```
SCHEDULE RBC EOW
SCHEDULE RBC EOM
SCHEDULE RBC EXERCISES
SCHEDULE RBC
RELATIONSHIP (AND/OR) A
DAYS

WWDAYS 1,2,3,4,5
DATES
CONFCAL SHIFT RETRO N MAXWAIT 05 D-CAT
MINIMUM PDS
DEFINITION ACTIVE FROM UNTIL
```

There were no Runtime scheduling parameters specified in IDJOB2, so none appear in IDJOB3. But you must add the relevant condition to make this job dependent on IDJOB2.

27 Add the IN condition: **IDJOB2-ENDED-OK ODAT**. Your entry in the Runtime Scheduling parameters section is displayed as follows:

```
IN IDJOB2-ENDED-OK ODAT
CONTROL RESOURCE PIPE
FROM TIME + DAYS UNTIL TIME + DAYS
```

The Post-processing parameters contain the same OUT condition as IDJOB2. Normally, you would change the string JOB2 to JOB3 in the OUT condition to make the OUT condition correct. But for this exercise you will do something different.

In the exercises you have previously performed in this guide, you defined OUT conditions indicating that the job ended OK, even when that job is not a prerequisite to other jobs. This is good practice, but not a requirement. To demonstrate this, since no other job is dependent on this job, you should delete the condition.

28 Delete the OUT condition. The Post-processing parameter section will be empty, and the entire job scheduling definition is displayed as follows:
Exit the job scheduling definition. You can now create the job scheduling definition for IDJOB4.

Perform an INSERT (I) next to IDJOB3. A job scheduling definition similar to that belonging to IDJOB3 is displayed.

Specify IDJOB4 in the MEMNAME field.

Specify JOB4 in the DESC field. The Basic Scheduling parameters found in IDJOB3 appear in IDJOB4.
You can now make the necessary changes.

33 Specify O (Or) in the RELATIONSHIP field.

34 Delete the values in the WDOMS field.

35 Specify 08,14 and the current working day of the month in the DAYS field.

By defining the current working day in the DAYS field, you can see how the Or Relationship works.

36 Delete the EXERCISES value from the SCHEDULE RBC field.

37 If the day on which you perform these exercises satisfies the EOW or EOM criteria, delete the schedule RBCs of the satisfied criteria from the job scheduling definition.

The Basic scheduling criteria should only be satisfied by the DAYS parameter, not by any schedule RBCs in the job scheduling definition. If the Or relationship works as expected, the job will still be scheduled.

The IN condition from IDJOB3 now appears in IDJOB4. It should be deleted.

38 Delete the following IN condition: IDJOB2-ENDED-OK ODAT.

There was no OUT condition in IDJOB3, so none appears in IDJOB4. You should add the appropriate condition.

39 Add the following OUT condition: IDJOB4-ENDED-OK ODAT +. The job scheduling definition for IDJOB4 is displayed as shown in Figure 41.
### Figure 41  Job Scheduling Definition for IDJOB4

<table>
<thead>
<tr>
<th>JOB: IDJOB4</th>
<th>LIB CTM.TEST.SCHEDULE</th>
<th>TABLE: IDGS2</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMAND ===&gt;</td>
<td>SCROLL====&gt; CRSR</td>
<td></td>
</tr>
<tr>
<td>+-----------</td>
<td></td>
<td>+</td>
</tr>
<tr>
<td>MEMNAME IDJOB4</td>
<td>MEMLIB CTM.TEST.JCL</td>
<td></td>
</tr>
<tr>
<td>OWNER ID</td>
<td>TASKTYPE JOB</td>
<td>PREVENT-NCT2</td>
</tr>
<tr>
<td>APPL GROUP IDGRP4</td>
<td>DESC JOB4</td>
<td>OVERLIB STAT CAL</td>
</tr>
<tr>
<td>SCHEDENV SYSTEM ID</td>
<td>NJE NODE</td>
<td>SET VAR</td>
</tr>
<tr>
<td>CTB STEP AT NAME TYPE</td>
<td>DOCMEM IDJOB4</td>
<td>DDCLIB</td>
</tr>
</tbody>
</table>

| SCHEDULE RBC EOW | SCHEDULE RBC EOM |
| SCHEDULE RBC RELATIONSHIP (AND/OR) O |
| DAYS 02,08,14 DCL |
| WDAYS AND/OR WCAT |
| DATES CONFCAL SHIFT RETRO N MAXWAIT 05 D-CAT |
| MINIMUM PDS DEFINITION ACTIVE FROM UNTIL |

| IN CONTROL |
| RESOURCE PIPE |
| FROM TIME + DAYS UNTIL TIME + DAYS DUE OUT TIME + DAYS PRIORITY SAC CONFIRM |
| TIME ZONE: |

| OUT IDJOB4-ENDED-OK ODAT + AUTO-ARCHIVE Y SYSDB Y MAXDAYS MAXRUNS |
| RETENTION: # OF DAYS TO KEEP # OF GENERATIONS TO KEEP |
| SYSOUT OP (C,D,F,N,R) FROM MAXXRENUM RERUNMEM INTERVAL FROM |
| STEP RANGE FR (PGM.PROC) . TO . ON PGMST PROCST CODES A/O DO |
| SHOUT WHEN TIME + DAYS TO URGN |
| MS |

---------- >>>>>>>>>>>>>>>>>>> END OF SCHEDULING PARAMETERS <<<<<<<<<<<<<<<< -------

COMMANDS: EDIT, DOC, PLAN, JOBSTAT 15.01.32

40 Exit the job scheduling definition.

You can now create the job scheduling definition for IDJOB5.

41 Perform an INSERT (I) next to IDJOB4. A job scheduling definition similar to that belonging to IDJOB4 is displayed.

42 Specify IDJOB5 in the MEMNAME field.
Specify **JOB5** in the DESC field. The Basic Scheduling parameters found in IDJOB4 appear in IDJOB5.

You can now make the necessary changes.

Replace all deleted schedule RBCs so that the job scheduling definition contains schedule RBCs EOW, EOM, and EXERCISES.

Delete the values specified for the **DAYS** parameter.

The job should only be scheduled according to the specified schedule RBCs. You should make this job dependent on successful completion of IDJOB4.

Add the following IN condition: **IDJOB4-ENDED-OK ODAT**.

In the OUT condition, change the string **JOB4** to **JOB5**. The job scheduling definition is displayed as follows:
Exit the job scheduling definition. The Job List screen is displayed as shown in Figure 42.
Saving the Table

49 Exit the Job List screen. The Exit Option window is displayed.

50 Specify Y in the SAVE field to save table IDGS2. The table is saved, and the entry panel is displayed. (The first time, you must create the Smart table before saving it, by specifying Y in CREATE field.)

Points to Remember Concerning SMART Table Scheduling

Before ordering the SMART Table Entity and its jobs, note the following points about SMART Table scheduling:

- Only the SMART Table Entity can be ordered. Individual jobs in the SMART Table cannot be ordered, although they can be forced.

- When a SMART Table Entity is ordered, its schedule RBCs are checked to see if the SMART Table Entity is eligible for scheduling. If none of the sets of scheduling criteria for the SMART Table Entity is satisfied, neither the SMART Table Entity, nor any of its jobs, are ordered. In this case, even if a the basic scheduling criteria of the job itself are satisfied, and an OR relationship is defined, the job is not scheduled.

- If at least one set of scheduling criteria is satisfied, each job is checked to see if it can be scheduled according to the combination of its schedule RBCs, basic scheduling criteria, and the defined AND/OR relationship.
SMART Table Entity Ordering and Job Ordering

You can now order the SMART Table Entity and all of the jobs in table IDGS2. Remember that these jobs are dependent on successful completion of IDJOB1 in table IDGS1, but you will not immediately order that job.

51 Reenter table IDGS2.

52 Enter the ORDER option by the SMART Table Entity only. The SMART Table Entity and all its jobs are ordered.

When a SMART Table Entity is ordered, all the jobs in the SMART Table that should be scheduled that day are automatically ordered as well. Because all the jobs have the Schedule RBC EXERCISES, they were all ordered.

53 Enter the Active Environment screen and filter the display using the IDGS filter. The screen is displayed as follows:

<table>
<thead>
<tr>
<th>Filter: IDGS</th>
<th>--------</th>
<th>Control-M</th>
<th>Active Environment</th>
<th>------</th>
<th>UP</th>
<th>&lt;D&gt;</th>
<th>(3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Owner</td>
<td>Odate</td>
<td>Jobname</td>
<td>JobID</td>
<td>Typ</td>
<td>Status</td>
<td></td>
</tr>
<tr>
<td>IDGS2</td>
<td>ID</td>
<td>020201</td>
<td>TBL</td>
<td>Wait Schedule</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IDJOB2</td>
<td>ID</td>
<td>020201</td>
<td>JOB</td>
<td>Wait Schedule</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IDJOB3</td>
<td>ID</td>
<td>020201</td>
<td>JOB</td>
<td>Wait Schedule</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IDJOB4</td>
<td>ID</td>
<td>020201</td>
<td>JOB</td>
<td>Wait Schedule</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IDJOB5</td>
<td>ID</td>
<td>020201</td>
<td>JOB</td>
<td>Wait Schedule</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>========= &gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;</td>
<td>Bottom of Jobs List</td>
<td>&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Notice that the SMART Table Entity, and all its jobs, have a status of Wait Schedule.

54 Request the Why screen, Option ?, for the SMART Table Entity.

<table>
<thead>
<tr>
<th>COMMAND ===&gt;</th>
<th>SCROLL ===&gt; CRSR</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPT DESCRIPTION</td>
<td></td>
</tr>
<tr>
<td>CONDITION</td>
<td>JOB1-ENDED-OK</td>
</tr>
<tr>
<td>========= &gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;</td>
<td>END OF &quot;WHY&quot; LIST</td>
</tr>
</tbody>
</table>

As predicted, the job is waiting for condition IDJOB1-ENDED-OK. You should now order IDJOB1.

55 Display the Job List screen for table IDGS1 and job order IDJOB1.

56 Return to Active Environment screen. Job IDJOB1 is scheduled and awaiting confirmation.

57 Confirm IDJOB1 and then watch the impact on the SMART Table Entity and jobs in table IDGS2.
Job IDJOB1 changes to a status of Wait Submission, then is submitted, then executed, and finally, it ends OK. At that point, the status of the SMART Table Entity changes to Active, and the status of jobs IDJOB2 and IDJOB4 changes to Wait Submission.

<table>
<thead>
<tr>
<th>Filter: IDGS</th>
<th>Control-M</th>
<th>Active</th>
<th>Environment</th>
<th>UP</th>
<th>&lt;D&gt; - (3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMAND ====&gt;</td>
<td>SCROLL ====&gt; CRSR</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O Name</td>
<td>Owner</td>
<td>Odate</td>
<td>Jobname</td>
<td>JobID</td>
<td>Typ</td>
</tr>
<tr>
<td>IDGS2</td>
<td>ID</td>
<td>020201</td>
<td>TBL</td>
<td>Active</td>
<td></td>
</tr>
<tr>
<td>IDJOB2</td>
<td>ID</td>
<td>020201</td>
<td>JOB</td>
<td>Wait Submission</td>
<td></td>
</tr>
<tr>
<td>IDJOB3</td>
<td>ID</td>
<td>020201</td>
<td>JOB</td>
<td>Wait Schedule</td>
<td></td>
</tr>
<tr>
<td>IDJOB4</td>
<td>ID</td>
<td>020201</td>
<td>JOB</td>
<td>Wait Submission</td>
<td></td>
</tr>
<tr>
<td>IDJOB5</td>
<td>ID</td>
<td>020201</td>
<td>JOB</td>
<td>Wait Schedule</td>
<td></td>
</tr>
<tr>
<td>IDJOB1</td>
<td>ID</td>
<td>020201</td>
<td>COURSE7A/14348</td>
<td>JOB</td>
<td>Ended &quot;OK&quot;</td>
</tr>
</tbody>
</table>

From that point on, the processing continues. Once job IDJOB2 ends OK, job IDJOB3 is submitted; once IDJOB4 ends OK, job IDJOB5 is submitted.

Job IDJOB3 ends OK, but IDJOB5 does not, due to the condition code C0008. Because at least one of the jobs in the SMART Tables — IDJOB5 — ended NOTOK, the SMART Table Entity ends NOTOK. At the end of processing, the status of the jobs appears as follows:

<table>
<thead>
<tr>
<th>Filter: IDGS</th>
<th>Control-M</th>
<th>Active</th>
<th>Environment</th>
<th>UP</th>
<th>&lt;D&gt; - (3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMAND ====&gt;</td>
<td>SCROLL ====&gt; CRSR</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O Name</td>
<td>Owner</td>
<td>Odate</td>
<td>Jobname</td>
<td>JobID</td>
<td>Typ</td>
</tr>
<tr>
<td>IDGS2</td>
<td>ID</td>
<td>020201</td>
<td>TBL</td>
<td>Ended- Not &quot;OK&quot;</td>
<td></td>
</tr>
<tr>
<td>IDJOB2</td>
<td>ID</td>
<td>020201 M21 /14373</td>
<td>JOB</td>
<td>Ended &quot;OK&quot;</td>
<td></td>
</tr>
<tr>
<td>IDJOB3</td>
<td>ID</td>
<td>020201 M21 /14375</td>
<td>JOB</td>
<td>Ended &quot;OK&quot;</td>
<td></td>
</tr>
<tr>
<td>IDJOB4</td>
<td>ID</td>
<td>020201 M21 /14374</td>
<td>JOB</td>
<td>Ended &quot;OK&quot;</td>
<td></td>
</tr>
<tr>
<td>IDJOB5</td>
<td>ID</td>
<td>020201 M21 /14376</td>
<td>JOB</td>
<td>Ended- Not &quot;OK&quot; Due to CC</td>
<td></td>
</tr>
<tr>
<td>IDJOB1</td>
<td>ID</td>
<td>020201 COURSE7A/14372</td>
<td>JOB</td>
<td>Ended &quot;OK&quot;</td>
<td></td>
</tr>
</tbody>
</table>

You can go back and correct the JCL for IDJOB5, and then rerun the job.

58 Edit the JCL for IDJOB5, which you can do using the JCL option in the current screen, and change the output condition code in step S5 from C0008 to C0000. Then exit back to the Active Environment screen

In the previous chapter, you learned about reruns in context of restarts, but it is also possible to rerun a failed job from the beginning. Do this for IDJOB5.

59 Enter option R (Rerun) for job IDJOB5. The Rerun/Restart window is displayed. In this case, the With Restart value is N (No restart).
60 Enter Y (Yes) in the Confirm field of the window. The job is rerun.

The second run of the job ends OK, and so does the SMART Table Entity. The Active Environment screen is displayed as follows:

Go into the IOA Conditions/Resources screen and see the results of these job runs.

61 Go to the IOA Conditions/Resources screen, Screen 4. Specify your ID as a prefix to filter the display. Remember to ensure that the current date appears in the DATE UNTIL field under the Scroll amount.
Notice the following:

- The conditions follow the order in which they were issued:
  
  - There is an ENDED-OK condition for jobs IDJOB1, IDJOB2, and IDJOB4. (IDJOB3 did not have an OUT condition).
  
  - There is an END-NOTOK condition for SMART Table IDGS2 because of the CC problem in IDJOB5. There is no such condition for IDJOB5 because you did not define it.
  
  - This is followed by the ENDED-OK for IDJOB5, after the rerun, and finally, the ENDED-OK status is displayed for the SMART Table IDGS2.

Advantages of SMART Table Scheduling

SMART Table scheduling is not always desired. But in those situations in which SMART Table scheduling is desirable, use of a SMART Table simplifies the process and provides several advantages. Some of these are discussed below:

- Schedule RBCs simplify basic scheduling.

  Once a set of basic scheduling parameters is defined in the SMART Table Entity, all jobs that require that set of scheduling parameters can utilize them simply by specifying the schedule RBC name in the appropriate field. The more complex and detailed the scheduling parameters, the greater the advantage.

- Multiple sets of scheduling criteria can be specified for each job.

  Regular job scheduling definitions provide for one set of scheduling criteria. However, in a SMART Table, the SMART Table Entity can include multiple sets of scheduling criteria, and any number of these can be specified in any job scheduling definition in the SMART Table. This enables the job to utilize multiple sets of basic scheduling criteria.

- Easier definition of prerequisite conditions and required resources is enabled for each job.

  Defining an IN condition in a SMART Table Entity makes all jobs in that table dependent on that condition without defining the IN condition in all the individual job scheduling definitions. The more jobs in the SMART Table that have no predecessor job other than the predecessor of the SMART Table Entity, the greater this advantage.

- Easier Post-processing definition is enabled for each job.
If you want to define Post-processing that should be performed only if any of the jobs ended NOTOK, it is much easier to define the conditional Post-processing once in the SMART Table Entity, rather than repeatedly in each job scheduling definition.

If you want to define Post-processing that should be defined if all the jobs ended OK, this is also easier to do in SMART Tables if there are jobs not belonging to the same hierarchy of dependence.

- Easier triggering of jobs in a different table is facilitated.

If the submission of a job in another table depends on the outcome of all the jobs in the current table, this is easier to define in a SMART Table.

62 Exit the Online facility.

This completes the steps in this chapter of the *Control-M for z/OS Getting Started Guide*.

**Review**

In this chapter you

- created a SMART Table

- defined a SMART Table Entity with multiple sets of scheduling criteria and with a dependency on successful completion of a job from another table

- used Schedule RBCs to appropriately (according to job) assign sets of scheduling criteria to each job

- defined additional criteria for two of the jobs in the SMART Table—using an AND relationship with the additional criteria in one of the jobs, and using an OR relationship in the other job

- defined two separate lines of dependency in the jobs in the SMART Table—IDJOB3 dependent on IDJOB2, and IDJOB5 dependent on IDJOB4—but no dependency between IDJOB3 and IDJOB4

- ordered the SMART Table Entity with the result that all jobs in the table that should have been scheduled that day were also ordered

- observed that the SMART Table Entity ended NOTOK when one of the jobs in the SMART Table ended NOTOK
- learned the advantages of using a SMART Table over a table that handles jobs individually

## Recommended Reading

Before continuing with the next chapter, you should read the following in the *Control-M for z/OS User Guide*:

In Chapter 2

- information in the Scheduling Definition facility topics relating to SMART Tables and SMART Table scheduling
- status changes of the Active Environment screen related to SMART Table scheduling and job scheduling definitions in SMART Tables
- reasons in the Why screen related to SMART Table jobs and SMART Table Entities
- the SMART Table Entity in the Zoom screen

In Chapter 3

- summary information related to SMART Table scheduling
- detailed parameter descriptions of the following SMART Table scheduling parameters
  - ADJUST CONDITIONS
  - ON TABLE-END
  - RELATIONSHIP
  - SCHEDULE RBCS
- information that relates to SMART Table scheduling in the following parameters
  - DO statements
  - MEMNAME
  - ON
This chapter includes the following topics:

Overview ................................................................. 134
Preparations ............................................................... 135
Using AutoEdit System Variables in the JCL .......................... 135
  System Date, Working Date and ODATE ......................... 137
  Date System Variables ........................................... 138
  Non-Date System Variables ...................................... 139
Introduction to AutoEdit Control Statements and Functions ....... 141
AutoEdit Function Details ............................................ 142
  %%$CALCDTE ....................................................... 142
  %%$JULIAN .......................................................... 143
Other Available Functions ........................................... 144
Supplying Variable Values through the Job Scheduling Definition .. 145
Checking AutoEdit Syntax and Results ................................ 147
  Activating Utility CTMAESIM ................................... 148
  AutoEdit Variables in Other Job Scheduling Definition Statements 151
Supplying Variable Values through an External Member ............ 152
  Pointing to External Members .................................... 154
Additional AutoEdit Control Statements ................................ 155
Complex AutoEdit Terms ............................................. 157
Operators ................................................................. 158
Local Variables and Global Variables ................................ 158
Defining Global Variables ............................................ 160
  Global Variable Syntax .......................................... 161
  Global Variable Access by Another Job .......................... 162
  Global Variable Update by Another Job ......................... 164
JCL Setup Operation Flow ........................................... 165
Review ................................................................. 167
  Recommended Reading ............................................. 168
Overview

In the production environment, JCL must often be manually modified prior to submission of a job, as in the following cases:

- Changing a parameter or a date card
- Supplying tape numbers in JCL procedures
- Eliminating steps under different run conditions, for example, end of month processing versus normal daily run.

Manual modification of the JCL is inconvenient at best, and it can be error-prone and lead to serious problems. The JCL and AutoEdit facility offers an automated alternative to manual JCL update.

The JCL and AutoEdit facility enables AutoEdit terms, variables, functions, and control statements to be specified in the JCL, in place of values that change from job submission to job submission. AutoEdit terms are prefixed by a pair of percent symbols, %%, which distinguishes them from other terms. For example, the term %%%ODAY is recognized as an AutoEdit term.

At time of job submission, AutoEdit terms in the JCL are resolved to, or replaced by, their actual values. The inclusion of AutoEdit terms in the job stream can eliminate the need to change JCL once it is defined.

Certain AutoEdit terms can also be used within job scheduling definitions, and can include system and user-defined variables, functions, operators, and control statements.

Control-M provides an AutoEdit simulation facility that enables you to check the results of AutoEdit inclusion before you run a job in the production environment.

You can also define Global AutoEdit variables in a database that allows the variables to be accessed and updated by different jobs.

In this chapter, you will

- see the results of using AutoEdit terms in the JCL of the job, and examine a number of AutoEdit variables
- use one or two AutoEdit functions
- use AutoEdit terms in the job scheduling definition
- look at the online utility facility and use the AutoEdit simulation facility to check the results of our AutoEdit usage
Preparations

For this chapter, do the following:

- Create a new table, not a SMART Table, called IDGS3.

- In table IDGS3, create four very basic job scheduling definitions, for IDJOB6, IDJOB7, IDJOB8 and IDJOB9, as follows:
  
  - In the MEMNAME field, specify, respectively: IDJOB6, IDJOB7, IDJOB8, or IDJOB9.
  - In the GROUP field, specify: IDGRP5.
  - In the DESCRIPTION field, specify: THIS IS IDJOB6 (or, respectively, IDJOB7, IDJOB8, or IDJOB9).
  - In the DAYS field, specify: ALL.
  - In the MONTHS field, specify: Y.
  - Do not specify any IN or OUT conditions, nor any Runtime Scheduling parameters.
  - Do not specify any Post-processing parameters.

- In your JCL table, create separate JCLs that do nothing—for example, IEFBR14—for jobs IDJOB6, and IDJOB7, IDJOB8, and IDJOB9. Use the JCL from IDJOB1 as the basis for this JCL.

You will make changes to the JCL and job scheduling definitions, as required, during the performance of these exercises.

Using AutoEdit System Variables in the JCL

Suppose that before a job submission you had to update the JCL with the current date, or the jobid of a previously run job. As mentioned in the introduction, such updates are inconvenient at best, and can be a source of errors, as well.

You can use AutoEdit terms to eliminate the need for such updates. In your exercise, you will use several AutoEdit variables in trivial comment lines, but these comment lines will nonetheless demonstrate the usefulness of AutoEdit terms.

1 Edit the JCL of IDJOB6, add the following comment lines, and then exit:

```c
/* Today's working date is: %%ODATE
/* Today's system date is:  %%DATE
/* Today's working day is: %%ODAY
/* The current job order id is: %%ORDERID
```
2 Enter the Scheduling Definition facility and order IDJOB6 (in table IDGS3). The Message screen is displayed indicating that the job ran.

3 Go to the Active Environment screen and request the Sysout display for the job (enter option V for the job to display the Job Order Execution History screen, and then enter option S to display the Sysout Viewing screen). The Sysout for job IDJOB6 is displayed.

4 Page down, or enter the command N (NEXT) in the COMMAND field (Command NEXT is discussed in Chapter 6, “Navigating The Active Environment.”).

The Sysout contains the comment lines you inserted in the JCL, as shown in Figure 43 on page 136). Notice that the AutoEdit variables in the JCL resolved to their appropriate values.

Figure 43   Sysout for Job IDJOB6

It you were to run this job again tomorrow, then the variables would resolve accordingly. It should be apparent, that use of AutoEdit system variables such as %ODATE can eliminate the need to manually update the JCL before a job is run.

Before continuing with the exercise, take a closer look at certain date concepts and AutoEdit date variables.
System Date, Working Date and ODATE

Control-M distinguishes between system date, current working date, and ODATE (original date).

System Date

System date corresponds to the actual calendar date. At any hour on the 9th of September in the year 2000, the system date is the 090900. At the end of the day, when midnight is reached, the system date changes to the 10th of September (100900).

(Of course, your system clock can be set incorrectly. For example, if the system clock is an hour slow, then your system date will not advance at midnight, but rather at 1:00 a.m. Assume that your system clock is set correctly, and that the system time and date correspond to the true time and date.)

Current Working Date

Working date is a Control-M concept that is generally useful in data centers. Basically it allows you to define your day to begin and end at some other time than midnight.

The advantage of using a working day in the production data center is that end of day jobs performed after midnight can have the same working date as jobs performed before midnight; they all have the same working day.

Each site defines its own working day for Control-M. Actually, the site defines the start time of the new working day, which also marks the end of the old working day.

For example, if the new working day at the site begins at 5:00 a.m.:

- A job that runs at 11:00 p.m. on September 9th, 2000 has the following dates (yyymmdd format):
  - System date: 000909
  - Current working date: 000909

- A job that runs at 3:00 a.m. on September 9, 2000 has the following dates (yyymmdd format):
  - System date: 000910
  - Current working date: 000909
**ODATE**

You have seen the term ODAT several times when using date references in conditions in IN and OUT and DO COND statements. ODAT is a four-letter abbreviation of ODATE; the term is abbreviated in conditions because the date field of conditions only allows entry of four characters.

ODATE is a keyword that means Original scheduling date, and it refers to (and resolves to) the working date on which the job was originally scheduled.

The important points to note are that:

- ODATE refers to the working date, not the system date.
- The ODATE may in some cases be different than the current working date. For example, if a job was delayed so that it did not run on its ODATE, but instead ran on the next working day, then its current working date would be one day later than its ODATE.

The ODATE allows Control-M (and you) distinguish between different orders of the same job, and determine which job order of the job applies to which date. The same applies to conditions.

**Date System Variables**

In the preceding steps, you specified a number of variables in the JCL, including, %%ODATE, %%DATE, and %%ODAY. These variables are predefined system date variables, with the following characteristics:

- Predefined system date variables defined to Control-M with recognized meanings.
- Predefined system date variables have values that are not user-supplied, but are resolved from values that are system-supplied.
- Predefined system date variables resolve according to date information.

The AutoEdit facility recognizes many system date variables. These are listed and defined in the JCL and AutoEdit facility chapter of the *Control-M for z/OS User Guide*. However, rather than considering them individually, it is easier to think of system date variables in terms of groupings. Once you have considered the groupings, you can probably anticipate the correct variable name, as illustrated below:

- Date variables can be used for system date, original scheduling date, and current working date.
For example, you saw %%DATE resolve to the system date and %%ODATE resolve to the original scheduling date. You also could have specified %%RDATE, which resolves to the current working date.

While there is no special prefix to indicate system dates, original scheduling date variables have a prefix of %%O, and current working date variables have a prefix of %%R.

- Different date units can be requested.

For example, you saw %%ODAY provide the day unit (1 - 13) of the original scheduling date.

- Some of the more common system date units are %%DAY, %%MONTH, %%YEAR, %%WDAY, which indicates weekday, and %%WEEK, which indicates the week number in the year, 1 through 53.

- The corresponding variables, such as %%OWDAY or %%RWDAY, are available for original scheduling dates and current working dates.

- Other date units of the original scheduling date include %%OMONTH, %%OYEAR, %%OWDAY, for weekday, and %%OWEEK, for week of the year, 1 through 53.

- Julian day variables can also be requested. These are all suffixed JULDAY.

  %%JULDAY, %%OJULDAY, and %%RJULDAY are examples of valid Julian date variables.

- Years in the variables described above are 2-character years. If a 4-character year is desired, a $ should follow the %%.

  For example, %%$DATE, %%$ODATE, and %%$RDATE each return the appropriated date in yyyymmd format, or a different 4-character year format, depending upon the standards in place at your site.

**Non-Date System Variables**

In addition to the date system variables discussed above, many system variables that do not relate to dates are available. %%ORDERID, which you specified in the JCL of IDJOB6, and which indicates the ID assigned by Control-M to the job order, is a non-date system variable. Available AutoEdit non-date system variables are listed in the following table:
Thus far, you have used AutoEdit predefined system variables. These included date system variables, such as \%\%ODATE, and non-date system variables, such as \%\%ORDERID.
Introduction to AutoEdit Control Statements and Functions

5 Edit the JCL of IDJOB6, delete the previously added comment lines, and add the following comment lines in their place. Do NOT exit the JCL.

```/* */ XXXSET %A=%%CALCDATE %%$DATE -1
/* */ THE VALUE OF A IS: %A```

You can now examine the components of the preceding `%%SET` statement.

`%% SET`

where `%%SET` is an AutoEdit control statement.

Control statements are used to define the AutoEdit environment, and to control AutoEdit processing, in the JCL. Control statements can appear anywhere in the JCL member to be submitted. When a control statement is detected in a JCL line, for example, in a JCL remark statement, the line containing the control statement is submitted as part of the job. If the control statement appears in a non-JCL line, for example, in a line beginning without a `//` symbol, the control statement is resolved and the resolved value can be applied to subsequent JCL lines, but the control statement is not submitted as part of the job.

Control statement `%%SET` sets values of user-defined variables. The format of the statement is:

```
%%SET %varname=expression
```

where:

- `varname` is the user-defined variable
- `expression` can consist of
  - a value, such as 5
  - a variable, such as `%%$DATE`
  - a combination of values, variables, operators, functions, and so on, such as
    `%%GENERATION_NUMBER %%PLUS 1`
%%A

%%A is a user-defined AutoEdit variable.

Unlike system variables whose names and meanings are predefined and recognized by Control-M, user-defined variables are defined by the user. You could just as easily have called this variable %%FRED or, if you wanted a more meaningful name, %%BACKDATE.

%%$CALCDTE

%%$CALCDTE is an AutoEdit function. Like system variables, AutoEdit functions are predefined and have set meanings. In the JCL, AutoEdit functions can only be specified in a %SET control statement.

Function %%$CALCDTE performs date manipulation by adding or subtracting a specified number of days from a specified date.

AutoEdit Function Details

%%$CALCDTE

In the preceding step, the principal AutoEdit term is %%$CALCDTE. It is true that the function needed to be placed in a %SET control statement, and it is also true that the returned result needed to be represented by a user-defined AutoEdit variable. But the purpose of this %SET control statement is to perform the date calculation function. The control statement and user-defined variable are incidental to the function.

The format of function %%$CALCDTE is

%%$CALCDTE date ± quantity

where

- date must be, or must resolve to, a date in format yyyyymmdd.
- quantity is a number, or numeric AutoEdit expression, of days to add or subtract from the date

In this step of your exercise, %%$CALCDTE subtracts one day from the ODATE.
NOTE
A related function, %%CALCDATE, makes the same calculation for dates having a 2-character year, that is, dates resolving to format yymmd. This function has a 366 day limit on the number of days that can be added or subtracted. The %%$CALCDATE function is supported for reasons of backward compatibility—it is recommended that %%$CALCDE be used whenever possible.

%%$JULIAN

You can examine one more function before exiting the JCL.

6 Add the following lines below the %%$CALCDE function in the JCL:

```c
/* SET =%%JULIAN %A
// THE VALUE OF B IS: %B
```

Function %%$JULIAN converts a Gregorian date with a 4-character year, to a Julian date with a 4-character year.

The format of function %%$JULIAN is:

```c
%%$JULIAN date
```

where `date` must be, or must resolve to, a date in format yyyymmdd.

In your exercise, you know that %A, which is defined in the preceding %%$CALCDE function, resolves to your ODATE -1. Therefore, %B should resolve to the Julian version of this date.

7 Exit the JCL. You can now run the job.

8 Return to the Job List screen and order IDJOB6.

9 Go to the Active Environment screen, request the Job Order Execution History screen for the job, then request the Sysout Viewing screen for the job execution, and check the Sysout of the job. Page down to the middle section of the Sysout.
Observe the following:

- In the first comment statement, ODATE resolves to the current original scheduling date.
- In the second and third comment lines, %A resolved to the previous day.
- In the fourth comment line, %B resolved to the Julian value of %A.

Other Available Functions

Shown in the following table is a list of some of the other AutoEdit functions available, besides %CALCDTE and %JULIAN. They are described in detail in the Control-M for z/OS User Guide.

NOTE

All date-related functions with a $ symbol, that is, those beginning %$, require and return 4-character years, in format yyyy.

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%GREG</td>
<td>Converts a Julian date to a Gregorian date.</td>
</tr>
<tr>
<td>%LEAP</td>
<td>Determines whether a specified Gregorian date falls in a leap year.</td>
</tr>
<tr>
<td>WCALC</td>
<td>Performs a shift from the specified date to a working date in the specified calendar, according to indicated instructions.</td>
</tr>
<tr>
<td>WEEK#</td>
<td>Calculates in which of weeks 1 through 53 a specified date falls. The function uses the site-defined start of the week—Sunday or Monday— as the first day of each week, and assumes that January 1st falls in the first week.</td>
</tr>
<tr>
<td>WEEKDAY</td>
<td>Calculates on which day of the week a specified date falls. The resolved value is an integer from 1 through 6, or 0, where 1 corresponds to the first day of the week—Sunday or Monday, depending on the site-standard—and 0 corresponds to the last day of the week, either Saturday or Sunday.</td>
</tr>
</tbody>
</table>
Table 12: Other AutoEdit Functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$%$YEARWK#</td>
<td>Calculates in which of weeks 1 through 53 a specified date falls, and returns the year and the week number according to ISO8601 standards. The function uses the site-defined start of the week—Sunday or Monday—as the first day of each week, and assumes that the first week is the week that includes January 4th.</td>
</tr>
<tr>
<td>$%$SUBSTR</td>
<td>Extracts a substring from a string.</td>
</tr>
</tbody>
</table>

Supplying Variable Values through the Job Scheduling Definition

As mentioned earlier, AutoEdit variables that resolve to desired values can be placed in the JCL. This is useful at sites where changes to the JCL are discouraged or not allowed.

If the variables are system variables, the values, by definition, are automatically supplied by the system. But if the variables are user-defined variables, the user must somehow provide the values for those variables. One method of providing those values is through the job scheduling definition.

For example, suppose the JCL must be provided the ID of a tape that is being mounted and used. A user-defined AutoEdit variable representing the tape ID can be placed in the JCL. Then, before the job is run, the value for that tape ID can be provided in the job scheduling definition.

You can now add a user-defined variable for the tape ID to the JCL, and set its value in the job scheduling definition. To keep this exercise at a basic level, you will again place the variable in a comment line in a trivial JCL.

10 Edit the JCL of IDJOB7, add the following comment line, and then exit the JCL:

```/* TAPE ID PROVIDED: $\%$TAPEID */```

You can now edit the job scheduling definition for IDJOB7.

11 Enter the job scheduling definition for IDJOB7.

Notice that the General Parameters section contains a field called SET VAR.

MEMNAME IDJOB7 MEMLIB CTM.TEST.JCLLIB
The SET VAR statement is used for specifying a value for a user-defined AutoEdit variable. Assume that the ID of the tape is 046453.

12 Specify `%%TAPEID=046453` in the SET VAR statement. It appears as follows:

```
OWNER   ID          TASKTYPE JOB    PREVENT-NCT2   DFLT  N
APPL    IDGRP5
DESC     THIS IS IDJOB7
OVERLIB  STAT CAL
SCHENV   SYSTEM ID
SET VAR
CTB STEP AT NAME TYPE
DOCMEM  IDJOB7 DOCLIB
```
13 Exit the job scheduling definition, and then exit the Job List screen. Save the changes in the Exit Option window of the Job List screen.

Before running the job, take a look at the AutoEdit Simulation facility.

Checking AutoEdit Syntax and Results

When Control-M detects an AutoEdit syntax error in a JCL member during submission, the submission is canceled by Control-M. Therefore, it is good practice to check the syntax of AutoEdit statements while the member is being prepared.

Furthermore, when the syntax is correct, you may want to verify that the AutoEdit statements return the desired results. For example, you may want to check that you specified the correct AutoEdit date variables for a job that performs end-of-year processing.

The AutoEdit Simulation facility, utility CTMAESIM tests AutoEdit syntax and JCL setup. This utility simulates the actions of the Control-M submission mechanism, which performs AutoEdit processing and JCL setup, and produces a printed report of the process.

Utility CTMAESIM can operate in either JCL Library mode or Scheduling Library mode:

- In JCL Library mode, the utility checks the AutoEdit statements in the JCL of the job, but not in the job scheduling definition.

This mode becomes operational when you specify a JCL member in the utility panel; in this case, the table that contains the job scheduling definition is unknown to the utility, and the job scheduling definition therefore remains unchecked.
- In Scheduling Library mode, the utility not only checks the AutoEdit statements in the JCL of the job, it also checks the impact that SET VAR statements in the job scheduling definition have on the JCL.

This mode becomes operational when you specify a job scheduling definition in the utility panel. Because the job scheduling definition points to the JCL member, both the job scheduling definition and the JCL can be, and are, checked.

### Activating Utility CTMAESIM

14 Display the Online Utilities menu by selection Option 6 of the IOA PRIMARY OPTION MENU.

**NOTE**
If Option 6 from the IOA Primary Option Menu does not bring you to the Utilities menu, ask your INCONTROL administrator to assist you in accessing the Utilities menu from TSO, or advise you how to access utility CTMAESIM directly.

The IOA Online Utilities menu is displayed. The particular options displayed in the menu depend on the INCONTROL products available at your site, as well as any limitations your INCONTROL administrator may have placed on the display. The IOA Online Utilities menu, and available Control-M utilities, are described in detail in the *INCONTROL for z/OS Utilities Guide*.

```
--- ON-LINE UTILITIES ---
OPTION  ===>
I1  PREREQ CONDITION - Add/Check/Delete a Prerequisite Condition
M1  JOB ORDER ISSUE  - Issue a Job Order
M2  AUTOEDIT SIMUL    - Perform an AutoEdit Simulation
M3  SIMUL/TAPE PULL   - Prepare Simulation/Tape Pull List Job
M4  PARAM PROMPTING   - Parameter Prompting Facilities
M5  QUICK SCHEDULE    - Quick Schedule Definition
M6  USER INTERFACE    - End-User Job Order Interface
R1  CONTROL-R SIMUL   - Control-R Simulation
R2  DATASET CLEANUP   - Control-R Dataset Cleanup
R3  JOB DATASET LIST  - Prepare a Job Dataset List
R4  STANDALONE        - Control-R Standalone
U1  DOCU/TEXT         - Invoke DOCU/TEXT
X   EXIT              - Exit This Menu

USERID - M21
TIME   - 18:06
TERMINAL - 3278
```

15 Enter **M2** in the OPTION field to access utility CTMAESIM. The AutoEdit Simulation panel is displayed.
Fill in the SCHEDULING LIBRARY MODE parameters with the name of the scheduling library, table, and job scheduling definition, and enter YES in the ENTER YES TO CONTINUE field. Do not change the default values in the GLOBAL LIBRARY, WDATE and ODATE fields; and type LIST in the FUNCTION field.

The screen should look similar to the following before you press Enter:
The simulation is performed. The output of the simulation is displayed as follows. This is a JCL and AutoEdit scan.
Notice that in the comment line you added to the JCL, the user-defined AutoEdit variable is resolved to, or replaced by, the value provided by the SET VAR statement in the job scheduling definition:

/** TAPE ID PROVIDED: 046453

You can now see that if you run this job, the user-defined variable will resolve correctly.

AutoEdit Variables in Other Job Scheduling Definition Statements

In addition to specifying a value for a user-defined AutoEdit variable in a SET VAR statement in the job scheduling definition, a value can also be provided in a DO SET statement. However, because a DO SET statement, like any DO statement, is a conditional Post-processing parameter, it is only activated on a subsequent run of the job, such as a rerun or restart, and then, only provided that the particular ON criteria are satisfied.
AutoEdit terms can also be specified in the following statements in the job scheduling definition:

- **SYSOUT and DO SYSOUT**

  File names for SYSOUT and DO SYSOUT handling can be specified using AutoEdit variables whenever SYSOUT Option F, the copy to file or sysout archiving option, is specified, as shown in this example:

  ```
  SYSOUT OP F PRM GPL.%%JOBNAME.D%%ODATE.%%JOBID.T%%TIME
  ```

- **SHOUT, DO SHOUT, and DO MAIL**

  System AutoEdit variables can be used in shouted messages, such as

  ```
  MSG JCL ERROR IN JOB %%JOBID %%STEP
  ```

- **MEMLIB and OVERLIB**

  AutoEdit variables can be used in the MEMLIB and OVERLIB fields to specify the appropriate library, as shown in this example:

  ```
  OVERLIB CTM.LIB.JCL.OVER%%ODATE
  ```

---

**Supplying Variable Values through an External Member**

In the previous steps, you provided the value for a user-defined AutoEdit variable by updating the job scheduling definition. But you can alternatively provide such values through special members dedicated to this purpose. This enables you to provide the values without updating the JCL, and without updating the job scheduling definition.

You can provide these values through an AutoEdit member identified by the AutoEdit control statements `%%LIBSYM` and `%%MEMSYM`.

17 Create a member IDTAPE in a work library. For purposes of this exercise, you will call the library CTM.TEST.AUTOEDIT.

18 In member IDTAPE, place the following lines:

```
%%BRANCH01_TAPE=5554444
%%BRANCH02_TAPE=3336666
%%BRANCH03_TAPE=7654321
```
19 Edit the JCL for IDJOB6 as shown in the following instructions:

A Delete the comment lines, that is, those containing AutoEdit terms, that you added earlier.

B Add the following statements to the JCL, then exit the JCL:

```plaintext
//** %LIBSYM CTM.TEST.AUTOEDIT %MEMSYM IDTAPE
//** TAPE ID PROVIDED: %BRANCH01_TAPE
//** TAPE ID PROVIDED: %BRANCH03_TAPE
```

20 Enter the AutoEdit Simulation utility and run the simulation for IDJOB6. The results of the simulation look similar to those shown in Figure 45. Observe the resolved values for the two //** TAPE ID PROVIDED: statements.

**Figure 45  AutoEdit Simulation Output**

```
Top of Data

SUB131I SUBMITTER STARTED
14.50.17 AES175I JOB SUBMISSION SIMULATION STARTED
SCHEDLIB CTM.TEST.SCHEDULE
TABLE  IDGS3
JOB IDJOB6
WDATE  020201
ODATE  020201
LIST
14.50.17 AES177I START SUBMISSION SIMULATION OF MEMBER IDJOB6 LIBRARY CTM.T

//M21 JOB APERLMAN,CLASS=A,
// MSGCLASS=X,NOTIFY=ID,
// MSGLEVEL=(1,1)
// *NET ID=AESUSER
//----- SUBMITTED BY CONTROL-M (FROM MEMLIB) ODATE=010202
//----- SCHEDULE CTM.TEST.SCHEDULE(IDGS3)
//----- JCL CTM.TEST.JCL(IDJOB6)
//----- CONTROL-M JOB IDENTIFICATION: ORDER ID=ZZZZZ RUN NO.=00001
//** %LIBSYM CTM.TEST.AUTOEDIT %MEMSYM IDTAPE
//** TAPE ID PROVIDED: 5554444
//** TAPE ID PROVIDED: 7654321
//S1 EXEC PGM=IEFBR14
SUB133IM21GRP3 IDJOB6 ZZZZZM21 SUBMITTED FROM LIBRARY (P)

-------------
14.50.18 AES178I END SUBMISSION SIMULATION OF MEMBER IDJOB6 LIBRARY CTM.T

14.50.18 AES176I JOB SUBMISSION SIMULATION ENDED
SUB159ISHUT DOWN UPON REQUEST OF MAIN TASK
```

Bottom of Data
In the previous steps, values for user-defined variables were provided in an external member pointed to by a %%LIBSYM / %%MEMSYM control statement. Actually, there are several ways to point to external members, as described in the following table:

<table>
<thead>
<tr>
<th>AutoEdit Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%%LIBSYM</td>
<td>Includes the contents of the specified member at the location of the %%LIBSYM / %%MEMSYM control statement in the JCL. The member is read by Control-M before submission. The included member can only contain a list of AutoEdit variables and their values in the format: %%term=value This control statement is useful for providing lists of values to be used during job processing, when both the JCL and the job scheduling definition should not be updated.</td>
</tr>
<tr>
<td>%%INCLIB</td>
<td>Includes the contents of the specified member at the location of the %%INCLIB / %%INCMEM statement in the JCL. The member is read by Control-M before submission. This statement is useful for inserting the following types of information into the JCL: JCL statements and/or parameters to be passed to the JCL (for example, SYSIN). AutoEdit control statements, including other %%INCLIB and %%INCMEM statements. Example of usage: If a long segment of JCL must be repeated at different points in the JCL, the segment can alternatively be placed in its own member, and an %%INCLIB / %%INCMEM control statement that points to the member can be specified at the required insertion points.</td>
</tr>
</tbody>
</table>
Additional AutoEdit Control Statements

In addition to the %%SET statement that you used at the beginning of this chapter, and control statements that point to external members, several other AutoEdit control statements are available. These are outlined in the following table:

<table>
<thead>
<tr>
<th>AutoEdit Term</th>
<th>Description</th>
</tr>
</thead>
</table>
| %%GLOBAL     | Includes the contents of the specified member at the location of the %%GLOBAL control statement in the JCL. Like %%MEMSYM, the included member can only contain a list of AutoEdit variables and their values in the format: %term=value Unlike the %%LIBSYM / %%MEMSYM control statement, %%GLOBAL identifies only the member name; it does not identify the library. The specified member is always taken from the library pointed to by DD statement DAGLOBAL.
### Table 14  Additional AutoEdit Control Statements

<table>
<thead>
<tr>
<th>AutoEdit Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%%LABEL lblnam</td>
<td>%%LABEL creates a label of name lblnam at the location of the statement.</td>
</tr>
<tr>
<td>%%GO lblnam</td>
<td>%%GO lblnam can then be used to pass control to the location identified by the %%LABEL lblnam statement.</td>
</tr>
</tbody>
</table>
| %%IF         | Provide the AutoEdit facility with Boolean “IF” logic capability. These statements, in conjunction with control statements %%GOTO and %%LABEL, permit branching based on submission time criteria. Job steps, DD statements, and so on are easily excluded or included. Format of the statements is:  

```plaintext
%%IF conditional-expression

statements

[%%ELSE]

statements

%%ENDIF
```
| %%RANGE      | Limits the handling of AutoEdit functions and variables to a specified column range. Contents of all columns outside the range remain unchanged. Format of the statement is:  

```plaintext
%%RANGE fromcol tocol
```
| %%RESOLVE    | By default, Control-M must resolve all AutoEdit terms in the JCL or the job is not submitted. This default can be overridden by specifying an appropriate %%RESOLVE statement in the JCL. For details, see the description of the %%RESOLVE statement in the *Control-M for z/OS User Guide*. |
Complex AutoEdit Terms

Multiple AutoEdit variables, and constants, can be joined together into a complex term.

The basic rule of AutoEdit resolution is that when a term contains multiple variables, those variables are resolved from right to left.

Multiple variables can be joined together in the following ways:

- Two variables can be joined to form a single complex variable by linking them together, as follows:

  `%%BACKUP_TAPE%%ODAY%`

  On the third day of the month, the variable partially resolves to `%%BACKUP_TAPE03`.

  If the value of `%%BACKUP_TAPE03` is known to Control-M as `EE1022`, this statement would fully resolve to `EE1022`.

- Two variables can be concatenated into two distinct but joined variables by placing a period between them.

  `%%ODAY.%%OMONTH`

  On the 4th of December, `%%ODAY.%%OMONTH` resolves to `0412`.

- Two variables can be concatenated into two distinct variables joined by a period, by placing two periods between them.

  `%%ODAY..%%OMONTH`

  On the 4th of December, `%%ODAY..%%OMONTH` resolves to `04.12`.

- A constant can be appended to a variable by prefixing the constant with the concatenation symbol `%%`.

  `CTM%%MODE%%.01.JCL`

  If `%%MODE=TEST`, then `CTM%%MODE%%.01.JCL` resolves to `CTM.TEST01.JCL`. 
Operators

AutoEdit operators are used to add or subtract values from AutoEdit variables in the JCL. These operators can only be specified in a %SET statement. Valid AutoEdit operators are shown in the following table:

**Table 15 Valid AutoEdit Operators**

<table>
<thead>
<tr>
<th>Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%%PLUS</td>
<td>Adds a value to an AutoEdit variable.</td>
</tr>
<tr>
<td>%%MINUS</td>
<td>Subtracts a value from an AutoEdit variable.</td>
</tr>
</tbody>
</table>

AutoEdit operators are generally used as follows:

```
%%SET variable=operand operator operand
```

where

- **operand** is an expression that resolves to a numeric value.
- **operator** is %%PLUS or %%MINUS.

Only one operator can be specified in each %SET statement. Increase the number of generations (%GENERATION_NUMBER) by one, as follows:

```
// %SET %GENERATION_NUMBER=%GENERATION_NUMBER %%PLUS 1
```

If the value of %GENERATION_NUMBER was initially 1, it is now set to 2.

### Local Variables and Global Variables

User-defined AutoEdit variables are classified as either Local variables or Global variables.

Thus far, all the user-defined variables that you have used in this chapter have been Local variables. A characteristic of Local variables is that values given to them by a job do not carry beyond that job—those values cannot be accessed or changed by a different job.

By contrast, Global variables are stored in the IOA Global Variable database and can be accessed and updated by other jobs.
Shortly, you will create Global variables and demonstrate their global nature. Before doing that, however, you should observe the local nature of the Local variables.

21 Enter the job scheduling definition for IDJOB6 and add the following SET VAR statements, and then exit the job scheduling definition:

```
SET VAR %%LOC1=111
SET VAR %%BRANCH01_TAPE=222222
```

22 Enter the job scheduling definition for IDJOB7 and delete the SET VAR definition. The job scheduling definition now contains no SET VAR definition.

23 Edit the JCL for job IDJOB7, and do the following:

   A  Delete the comment line:

   ```
   //* TAPE ID PROVIDED: %%TAPEID
   ```

   B  Add, in its place, the following comment lines:

   ```
   //* %%LIBSYM CTM.TEST.AUTOEDIT %%MEMSYM IDTAPE
   //* BRANCH01_TAPE HAS THE VALUE: %%BRANCH01_TAPE
   //* LET'S SEE IF THIS RESOLVES: %%LOC1
   ```

   C  Exit the JCL.

24 Order the job scheduling definition of IDJOB6.

25 After job IDJOB6 has successfully ended, order the job scheduling definition of job IDJOB7.

26 Check the results of the job orders in the Active Environment screen. Job IDJOB6 ended “OK”, but job IDJOB7 was not submitted.

27 Request the log for IDJOB7, by using Option L.

   The log indicates that IDJOB7 was not submitted because variable %%LOC1 could not be resolved. Since IDJOB6, which successfully executed, defined a value for %%LOC1, it is clear that IDJOB7 has no access to this value. This is because %%LOC1 is a local variable.

28 Edit the JCL for job IDJOB7, delete the following comment line, and then exit the JCL:

   ```
   //* LET'S SEE IF THIS RESOLVES: %%LOC1
   ```
Rerun job IDJOB7 through Option R in the Active Environment screen. The second run of IDJOB7 ended “OK”.

The job order line appears as follows in the Active Environment screen:

```
IDJOB7   ID  020201 M21    708316 JOB Ended "OK" (Run 2)
          Prior Run: Not Submitted
```

Request SYSOUT of job IDJOB7, using Option V in the Active Environment screen followed by Option S in the Job Order Execution History screen, and scroll down to find the value for BRANCH01_TAPE.

```
%%BRANCH01_TAPE resolved to the original value from %%MEMSYM member, 5554444, not the value set by the SET VAR statement in IDJOB6:
```

```
//* BRANCH01_TAPE HAS THE VALUE: 5554444
```

Clearly, the SET VAR statement in IDJOB6 did not impact the value in the external member, which indicates that %%BRANCH01_TAPE is a local variable.

Now that you’ve seen the limitations of Local variables, take a look at Global Variables.

### Defining Global Variables

As mentioned above, a Global variable is a user-defined variable that is placed in the IOA Global Variable database.

%%SET statements in the JCL, and SET VAR or DO SET statements in the job scheduling definition, enable Control-M jobs and SMART Table Entities to define Global variables and place them in the IOA Global Variable database. These variables can then be used and accessed by other jobs. These jobs can use %%SET, SET VAR and/or DO SET statements to change or update the variable values in the database.

You can now define some Global variables in the job scheduling definition and JCL of job IDJOB8.

Enter the job scheduling definition for IDJOB8 and add the following SET VAR statements. When you define a SET VAR statement and press Enter, a new, blank SET VAR statement is opened to allow definition of the next SET VAR statement.

```
SET VAR   %%\VAR1=AAA
SET VAR   %%..\VAR2=BBB
SET VAR   %%..\IDJOB9\VAR1=CCC
```
Exit the job scheduling definition.

Add the following statements to the JCL of job IDJOB8, and then exit the JCL:

```plaintext
/** VAR1 FROM JOB JOB8 EQUALS %\VAR1
 /** VAR2 FROM GROUP GRP3 EQUALS %..\VAR2
 /** VAR1 FROM JOB JOB9 EQUALS %..\IDJOB9\VAR1
```

Order the job scheduling definition of IDJOB8.

Check the sysout of IDJOB8. The following comments appear in the sysout:

```plaintext
/** VAR1 FROM JOB JOB8 EQUALS AAA
 /** VAR2 FROM GROUP GRP3 EQUALS BBB
 /** VAR1 FROM JOB JOB9 EQUALS CCC
```

From this sysout, you can see that the AutoEdit variables resolved as they should. However, because they appeared in the job scheduling definition and the JCL of the same job, you have not yet demonstrated their global nature.

You will demonstrate the global nature of these variables shortly, but first, take a look at Global variable syntax.

**Global Variable Syntax**

Note the following points about Global variable assignment and syntax:

- Global variables are identified, and distinguished from Local variables, by a backslash immediately following the % sign. For example, whereas %VAR1 is a Local variable, %\VAR1 is a Global variable.

- A Global variable is assigned an owner at time of creation. This owner can be the component that creates the variable, such as the job, or it can be any other component in the database, such as the job, group, application, or even Control-M. The IOA Global Variable Database has a hierarchical structure to reflect this component hierarchy.

- Backslashes are used to describe the hierarchical structure of the IOA Global Variable Database, much like they are used to describe the directory and subdirectory structure in Unix and DOS.

The full path of the IOA Global Variable database is indicated as follows:

```
%\product\application\group\job\variablename
```
_paths can be specified using the same rules and shortcuts that are available with directories and subdirectories, instead of the full path:

— A job or SMART Table Entity can assign a Global variable to itself by specifying a backslash immediately following the % symbol.

— Paired dots followed by a backslash (..\) indicate movement to the next level up.

Based on the above, you can see that the variables created in IDJOB8 SET VAR statements have the following owners:

**Table 16  IDJOB8 SET VAR Statement Owners**

<table>
<thead>
<tr>
<th>Variable</th>
<th>Owner</th>
</tr>
</thead>
<tbody>
<tr>
<td>%%\VAR1=AAA</td>
<td>Global variable %%\VAR1 is owned by job IDJOB8</td>
</tr>
<tr>
<td>%%..\VAR2=BBB</td>
<td>Global variable %%\VAR2 is owned by group IDGRP3. (The paired dots moved the variable up the hierarchy to the group to which IDJOB8 belongs.)</td>
</tr>
<tr>
<td>%%..\IDJOB9\VAR1=CCC</td>
<td>Global variable %%\VAR1 is owned by job IDJOB9. (The paired dots move the variable up the hierarchy to group IDGRP3. The \IDJOB9 moves the variable down the hierarchy from IDGRP3 to job IDJOB9.)</td>
</tr>
</tbody>
</table>

Two variables with the same name but different paths are different variables. This is comparable to the fact that two Unix or DOS files with the same name but different paths are different files.

In our example, %%\VAR1=AAA and %%..\IDJOB9\VAR1=CCC result in different variables. As indicated in the above table

— the Global variable %%\VAR1 that equals AAA belongs to IDJOB8
— the Global variable %%\VAR1 that equals CCC belongs to IDJOB9

If the particular path has no Group and/or no Application, for example, the job does not belong to a group or application, Control-M utilizes the keyword values NO_APPL and NO_GROUP in the path, as needed.

**Global Variable Access by Another Job**

You can now take a look at the global nature of these variables.

Enter the job scheduling definition for IDJOB9 and do the following:
A  Enter **ALL** in the DAYS field.

B  Enter **ANYSTEP** in the ON field.

C  Enter **C0000** in the CODES field.

D  Add a DO SHOUT statement whose destination is your user ID, and which contains the following message text:

   **FIRST SHOUT: %\%\VAR1**

E  Add a second DO SHOUT statement whose destination is your user ID, and which contains the following message text:

   **SECOND SHOUT: %\%..\VAR2**

F  Add a third DO SHOUT statement whose destination is your userID, and which contains the following message text:

   **THIRD SHOUT: %\%..\IDJOB8\VAR1**

G  Exit the job scheduling definition.

Before running the job, determine the anticipated results.

**Table 17  Anticipated Results of Job**

<table>
<thead>
<tr>
<th>Shout</th>
<th>Anticipated Resolution and Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIRST SHOUT:</td>
<td>CCC</td>
</tr>
<tr>
<td></td>
<td>The Shout in IDJOB9 specifies Global variable %%\VAR1, which refers to a Global variable %%\VAR1 that is owned by itself (IDJOB9).</td>
</tr>
<tr>
<td></td>
<td>Job IDJOB8 assigned the value CCC to the Global variable %%\VAR1 that is owned by IDJOB9.</td>
</tr>
<tr>
<td>SECOND SHOUT:</td>
<td>BBB</td>
</tr>
<tr>
<td></td>
<td>The Shout in IDJOB9 specifies Global variable %%..\VAR2, which refers to a Global variable %%\VAR2 that is owned by IDGRP3.</td>
</tr>
<tr>
<td></td>
<td>Job IDJOB8 assigned the value BBB to the Global variable %%\VAR2 that is owned by group IDGRP3.</td>
</tr>
</tbody>
</table>
Global Variable Update by Another Job

Table 17  Anticipated Results of Job

<table>
<thead>
<tr>
<th>Shout</th>
<th>Anticipated Resolution and Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>THIRD SHOUT:</td>
<td>AAA</td>
</tr>
<tr>
<td></td>
<td>The Shout in IDJOB9 specifies Global variable %%.\IDJOB8\VAR1, which refers to a Global variable %\VAR1 that is owned by IDJOB8.</td>
</tr>
<tr>
<td></td>
<td>Job IDJOB8 assigned the value AA A to the Global variable %\VAR1 that is owned by itself (IDJOB8).</td>
</tr>
</tbody>
</table>

37 Order the job scheduling definition for job IDJOB9.

38 Check the Sysout of the job. The Sysout contains the following:

<table>
<thead>
<tr>
<th>FIRST SHOUT:</th>
<th>CCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>SECOND SHOUT:</td>
<td>BBB</td>
</tr>
<tr>
<td>THIRD SHOUT:</td>
<td>AAA</td>
</tr>
</tbody>
</table>

These values clearly indicate that these variables have been globally accessed. You can demonstrate that these variables can be globally updated.

Global Variable Update by Another Job

39 Enter the job scheduling definition for IDJOB9 and do the following:

A Add the following SET VAR statements:

- SET VAR %\VAR1=XXX
- SET VAR %..\VAR2=YYY
- SET VAR %..\IDJOB8\VAR1=ZZZ

B Delete the DO SHOUT statements

C Delete the ON STEP / CODES values.

D Exit the job scheduling definition.

The job scheduling definition for job IDJOB9 will now update the values for the Global variables previously defined by IDJOB8.

40 Enter the job scheduling definition for job IDJOB8 and delete the SET VAR statements. Then exit the job scheduling definition.
41 Order job IDJOB9.

42 Once job IDJOB9 has ended, order job IDJOB8.

You should anticipate the following results of these successive job runs.

- The SET VAR statements in job IDJOB9 have updated the Global variables with the new values.
- Because the SET VAR statements were removed from job IDJOB8, this job should not have changed the values that were newly set by IDJOB9.
- Therefore, you can expect that the JCL of job IDJOB8 used the new values set by IDJOB9.

43 Check the sysout of job IDJOB8. The following comments should appear in the sysout:

```plaintext
/* VAR1 FROM JOB JOB8 EQUALS ZZZ
/* VAR2 FROM GROUP GRP3 EQUALS YYY
/* VAR1 FROM JOB JOB9 EQUALS XXX
```

The above values clearly indicate that Global variables can be globally updated.

---

**JCL Setup Operation Flow**

You have seen that values for variables can come from several sources, for example, external members and SET VAR statements. Therefore, it is important to know the order in which these sources are checked, and the process involved in AutoEdit resolution in JCL.

- All JCL setup operations are performed during job submission. At this time, Control-M processes the JCL of the job line by line.
- Control-M scans each line for AutoEdit terms, which are identified by the `%%` symbol prefix, and unless otherwise instructed, tries to resolve them. Control-M resolves all AutoEdit terms in one line before it moves to the next line.
- All changes made during JCL processing, such as variable resolution, are retained only until Control-M has finished submission of the job.
- Control-M resolves system variables by taking the values from the system.
- Control-M resolves Global variables by taking the values from the IOA Global Variable database.
Values for Local user-defined variables can be taken from any of several possible sources, as described below. When Control-M detects a Local user-defined variable in the JCL line being processed, it checks these possible sources in a specific order until a value is found for the variable. Control-M creates a user-defined variable environment in which it places each user-defined variable and its value.

The potential sources for Local user-defined variable values are listed below in the order in which they are generally checked:

- system variable values

- % %SET control statements

These statements can be specified in JCL lines, including JCL comment lines. They assign values to variables.

- SET VAR and DO SET statements

These statements can be specified in the job scheduling definition. They can be used to define new variables, or to assign new values to existing variables.

SET VAR statements can affect all job submissions.

DO SET statements can override values specified by a SET VAR or previous DO SET statement. However, since DO SET statements are Postprocessing parameters, they only affect subsequent runs of a job, specifically, rerun and restarted jobs.

- Local variables and values defined in members specified in % %LIBSYM / % %MEMSYM control statements.

These members define local variables and specify their values. Members are searched in the order they appear in the JCL.

- Local variables and values defined in members specified in % %GLOBAL control statements.

These members define local variables and specify their values. Members are searched in the order they appear in the JCL.

The order in which Control-M checks potential sources for possible AutoEdit variable resolution is important because once Control-M has resolved a variable, it generally stops checking other sources. Potential values from other sources are ignored, and resolved values are not overridden except by % %SET statements in subsequent JCL lines.
Because JCL is processed sequentially one line at a time, the line being processed can only be affected by external members and %%SET control statements that have previously been processed. If a line contains an undefined variable that is only defined in a subsequent line, the variable cannot be resolved.

By default, if Control-M cannot resolve a variable, it stops submission of the job. This default, however, can be overridden by specifying the %%RESOLVE control statement with a value of NO or OFF.

44 Exit the Online facility.

This completes the steps in this chapter of the Control-M for z/OS Getting Started Guide.

**Review**

In this chapter you

- looked at and used AutoEdit system variables (date and nondate variables) in the JCL of a job
- learned about the difference between System date, Current Working date, and ODATE
- used the AutoEdit %%SET control statement in your JCL, and learned about other AutoEdit control statements
- used AutoEdit functions %%$CALCDTE and %%$JULIAN in your JCL, and learned about other AutoEdit functions
- used the SET VAR statement in the job scheduling definition to supplied values to user-defined variables in the JCL of your job
- checked AutoEdit syntax and results using online utility CTMAESIM
- supplied values to AutoEdit variables through an external member that was pointed to by the %%LIBSYM and %%MEMSYM statement, and learned other ways to point to external members
- learned about complex AutoEdit terms and AutoEdit resolution
- learned about AutoEdit operators
- learned the difference between Local and Global variables
Recommended Reading

- learned the syntax of, and defined and updated Global AutoEdit variables
- learned JCL Setup Operation flow

**Recommended Reading**

Before continuing with the next chapter, you should read the following in the *Control-M for z/OS User Guide*:

- in Chapter 1, the topic dealing with AutoEdit
- in Chapter 2, the description of utility CTMAESIM
- in Chapter 3, description of the SET VAR and DO SET statements
- in Chapter 5, the entire chapter
Navigating The Active Environment

This chapter includes the following topics:

- Overview ................................................................. 170
- Preparations ............................................................... 170
- A Closer Look At Filters .............................................. 171
  - Displaying the List of Available Filters ..................... 176
- Changing Active Environment Display Types ................ 178
- Adding a Note to a Job ............................................... 179
- Displaying Table Names for Jobs ................................. 182
- Displaying Job Dependencies ..................................... 183
- Displaying Execution Information from Job Runs .......... 185
  - Job Order Execution History Screen ......................... 185
  - Sysout Viewing Screen .......................................... 187
- Forcing an OK Status for a Job ................................... 189
- Displaying Statistics for a Job .................................... 191
- Displaying Jobs Belonging to a Specific SMART Table ... 192
- Review ...................................................................... 193
  - Recommended Reading ............................................ 193
Overview

You are now going to look at and use some additional options, screens and capabilities of the Active Environment.

In this chapter, you will

- look more closely at the fields in the Show Screen Filter window, define a new filter, and display a list of available filters
- change between display types of the Active Environment screen
- add a note to a job in the Zoom screen and display the content of the note in the Active Environment screen
- display the scheduling library and table names of jobs in the Active Environment screen
- display the list of dependent jobs, both predecessors and successors, in the Job Dependency Network screen
- display the execution history of job orders in the Job Order Execution History screen, and display job Sysout information in the Sysout Viewing screen
- force a status of OK for a job in the Active Environment screen
- display statistics for a job in the Statistics screen
- display the Table to which a job belongs in the Active Environment screen

Preparations

In this chapter, you will be using the following, and their respective JCLs:

- IDJOB1 from table IDGS1
- Each job, from IDJOB2 through IDJOB5, and the table IDGS2 SMART Table Entity
- IDJOB6 from table IDGS3

In preparation, you will

- ensure that job IDJOB1 in table IDGS1 does not require manual confirmation
A Closer Look At Filters

In anticipation of taking a closer look at filters, you can run the jobs you prepared, and then look at the results in the Active Environment screen.

1 Order the following jobs, in the following order:

- Job IDJOB1 in table IDGS1
- SMART Table Entity GRPSCHD in table IDGS2
- Job IDJOB6 in table IDGS3

2 Enter the Active Environment screen and activate the filter IDGS. The following screen is displayed:

- ensure a continuous job dependency from IDJOB1 through IDJOB6, by
  - making the following changes in the appropriate job scheduling definitions in table IDGS2
    - adding, in the job scheduling definition of IDJOB3, OUT condition IDJOB3-ENDED-OK ODAT +.
    - adding, in the job scheduling definition of IDJOB4, IN condition IDJOB3-ENDED-OK ODAT
  - making the following changes in the appropriate job scheduling definitions in table IDGS3
    - adding, in the job scheduling definition of IDJOB6, IN condition IDJOB5-ENDED-OK ODAT
    - adding, in the job scheduling definition of IDJOB6, OUT condition IDJOB6-ENDED-OK ODAT +

- ensure the following situations in the appropriate JCL files:
  - in the JCL of job IDJOB3, the step returns a condition code of C0008
  - in the JCL of job IDJOB5, all steps return a condition code of C0000

- ensure that batch utility CTMJSA has been run in the current working day
  If CTMJSA is not run as part of New Day processing, run it yourself or request that it be run by your administrator
These results are not surprising. Job IDJOB3 did not end OK due to condition code C0008. And because of the defined job dependencies, the remaining jobs have a status of Wait Schedule.

You can now take a closer look at filters in the Active Environment screen. In Chapter 1, “Introduction to Control-M,” you created filter IDGS, and in subsequent chapters, you activated this filter by entering the SHOW IDGS command.

To display the filter criteria of this same filter for editing, you should add the keyword EDIT at the end of the command. You can now edit filter IDGS.

\[3\] Enter the command **SHOW IDGS EDIT**. The Show Screen Filter window displays the filtering criteria for filter IDGS.
Figure 46  Show Screen Filter Window

You have already seen that you choose whether to save a new filter by specifying Y, or N, in the Save field, and pressing Enter. You can use this same field to choose whether to save changes to an existing filter.

**NOTE**

Specifying N (No) does not cancel changes made to a filter. It only means that they will not be permanently saved. They will, however, remain in memory. This applies even if you are editing a new filter, that is, specifying N and exiting the filter leaves the new filter in memory.

To cancel changes to a filter, close the window by pressing PF04/PF16 to enter the RESET command. The changes are canceled regardless of the value specified in the Save field.

The purpose in filtering the display was to ensure that the screen only displays those jobs that you used in your exercises. You took two steps to accomplish this purpose:

1. You ensured that all jobs in the exercises belonged to a Group prefixed by IDGRP.
2. You then filtered the display based on this Group name prefix of IDGRP.

You could have accomplished the purpose using the following filter criteria:

- Because you prefixed each member name with IDGS, you could have filtered using that Memname prefix.
If you ran no other jobs, you could have filtered on Owner ID.

Clearly, filtering on Odate would not have accomplished your purpose. However, Odate is useful for filtering when you only want to see jobs scheduled for a specific Odate.

The middle portion of the window is divided into three columns. These columns are all status related. For example, you can include (or exclude) jobs that have a Wait Schedule, Wait Confirmation, and/or Ended “OK” status. To learn the details of the relationship between these status columns, and their header topics, In Process, Ended, State, refer to the description in the Control-M for z/OS User Guide.

If you filter on such values as a common member name or group prefix, you are likely to display related or connected jobs. By contrast, if you filter only on a status such as Wait Schedule, you are likely to see completely unrelated jobs in the display. This, however, is still very valuable. An operator, for example, may need to check which jobs still have a Wait Schedule or a Wait Confirmation status.

Multiple filtering criteria can be specified. All specified criteria must be satisfied. You can, for example, specify a filter on your Owner name and on a Wait Schedule status. This way, you can see only your jobs that are Waiting Scheduling.

The filter window for filter IDGRP is currently displayed. You can, of course, close it without making changes. And, as mentioned above, you can make changes and either save them or keep them in memory.

But you can also use this filter as the basis for another filter by making the desired changes and specifying a new name in the Filter field.

You can define a new filter that shows only jobs with a group name prefix of IDGRP that have a Wait Schedule status. Name this filter IDWS, as described in the following steps.

4 In the Filter field, change the name from IDGS to IDWS.

5 Specify Y in the Save column.

6 Change the description to read: GS-WAITSCHED.

7 Leave the Group name value, IDGRP, unchanged.

8 In the status sections leave the In Process value (Y) and the Wait Sched value (Y) unchanged. The Active Environment window will not show the jobs that are in Wait Schedule status if the jobs are waiting for a condition. So we must leave the Wait Cond unchanged as (Y). Similarly, if the jobs are waiting for anything else, we must leave the appropriate values (Y). Therefore the Wait time, Wait quant, and Wait contrl values must remain (Y). Also, leave all values in the State column unchanged.
9 In the status sections, make the following changes:

A In the In Process column, change all values (except those of In Process, Wait Sched, Wait time, Wait Cond, Wait quant, and Wait contrl) to **N**.

B Change the Ended column header value to **N**.

The window is displayed as follows. This illustration shows the changes from the previous filter indicated in bold.

<table>
<thead>
<tr>
<th>In Process</th>
<th>Ended</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>N</td>
<td>Y</td>
</tr>
</tbody>
</table>

The window is displayed as follows. This illustration shows the changes from the previous filter indicated in bold.

10 Press **Enter** to exit the window. The Active Environment screen is displayed with the newly defined filter.
Displaying the List of Available Filters

Control-M is supplied with several predefined filters, including a filter called DEFAULT. In addition to any filters you define, the INCONTROL administrator can redefine filters and place them in the General profile.

Site-defined defaults determine whether the last filter used, or the DEFAULT filter, will be activated upon reentry to the Active Environment screen.

If you are unsure of a filter name, you can display the list of available filters in the Display Filters window. The display includes all globally available filters as well as filters that you have defined.

You can then select a filter from the Display Filters window for activation or editing.

11 Enter the command SHOW ? in the COMMAND field. The Display Filters window is opened, displaying the list of available filters.

**Figure 47 Active Environment Screen Display Filters Window**

<table>
<thead>
<tr>
<th>Filter: IDWS</th>
<th>* ------ Control-M Active Environment ------ UP &lt;D&gt; - (3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMAND</td>
<td>SCROLL ==&gt; CRSR</td>
</tr>
<tr>
<td>O Name</td>
<td>Owner Odate Jobname JobID Typ Status</td>
</tr>
<tr>
<td>IDJOB4</td>
<td>ID 020201 JOB Wait Schedule</td>
</tr>
<tr>
<td>IDJOB5</td>
<td>ID 020201 JOB Wait Schedule</td>
</tr>
<tr>
<td>IDJOB6</td>
<td>ID 020201 JOB Wait Schedule</td>
</tr>
</tbody>
</table>

---
| DISPLAY FILTERS |                      |
| CMD ==> SCROLL ==> CRSR |
| O NAME | DESCRIPTION |
| CONFIRM | WAIT CONFIRM. JOBS |
| DEL | ONLY DELETED JOBS |
| END | ALL ENDED JOBS |
| ENDNOTOK | ENDED NOT-OK JOBS |
| ENDOK | ENDED OK JOBS |
| EXEC | EXECUTING JOBS |
| LATE | LATE JOBS |
| WAIT | JOBS ON WAIT QUEUE |
| ECSALL | ALL JOBS IN AJF |
| IDGS | GS-EXERCISES |
| IDWS | GS-WAITSCHED |
| OPTIONS S SELECT E EDIT |

The Display Filters window displays the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>Name of the filter as it appears in the General or User profile.</td>
</tr>
<tr>
<td>DESCRIPTION</td>
<td>Description of the filter (if provided).</td>
</tr>
</tbody>
</table>
You can select a filter from the list for activation or editing by specifying the appropriate option in the O (option) field to the left of the filter name: Valid options are

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>S (SELECT)</td>
<td>Activate the filter. The display of jobs in the Active Environment screen is filtered according to the filter criteria.</td>
</tr>
<tr>
<td>E (EDIT)</td>
<td>Display the filter’s filtering criteria in the Show Screen Filter window to enable editing of the filter.</td>
</tr>
</tbody>
</table>

12 Enter S in the OPTION field by filter IDGS to activate filter IDGS. The window is closed, but the Active Environment screen display has not changed, because you are at the bottom of the display.

13 Use the PFKeys to scroll to the top of the screen. The Active Environment screen display conforms to the criteria of filter IDGS.

To deactivate all filters, that is, to display the full Active Environment screen without any filter showing, you can enter the command SHOW nonfilter EDIT, where nonfilter is NOT the name of an existing filter. Delete the name from the Filter field, and press Enter.
Changing Active Environment Display Types

Various predefined formats are available for the Active Environment screen. To change display type, enter the DISPLAY, abbreviated DI, followed by a space and any one-letter display type abbreviation, as follows:

- D – Default Display Type
- N – Network Fields Display Type
- A – All Fields Display Type

Enter DI A in the COMMAND field. The All Fields display type of the Active Environment screen is displayed.

Figure 48 Display Type A (All Fields)

![Table showing various fields of a job]

Notice that this display type includes many fields not displayed in the Default display type. Because of the larger number of lines displayed for each job, each screen display shows far fewer jobs than the default display type.

Enter DI D in the COMMAND field. The Default display type is redisplayed.

You have not displayed the Net display type, DI N. This display type is generally useful only in the Job Dependency Network screen, which is discussed, below, under the topic “Displaying Job Dependencies.”

Next, you will see how to add a note to one of your jobs.
Adding a Note to a Job

The NOTE command that appears in the list of commands at the bottom of the Active Environment screen is NOT used to add notes. Rather, it is used to display notes that have already been added to a job order.

Notes are added to job orders in the Zoom screen. Remember that before making any changes in the Zoom screen, you must first place the job in Held status.

16 Place job IDJOB6 in Held status and display the Zoom screen for the job by entering Option H, and pressing Enter until the job has a status of Held Wait Schedule. Then enter option Z).

17 Enter command NOTE in the COMMAND field of the Zoom screen. A Note line is opened inside the Zoom screen between two lines of equal signs, as follows:

When you enter text in one line, a new blank line is opened to enable you to enter more text.

18 In the NOTE line, enter the text: THIS NOTE COULD BE IMPORTANT. BUT IT’S NOT.
Adding a Note to a Job

The NOTE section is displayed as follows:

```
===========================================================================
NOTE THIS NOTE COULD BE IMPORTANT. BUT IT'S NOT.
NOTE
===========================================================================
```

Save this new note by typing **SAVE** in the command line and pressing **Enter**.

19 Exit the Zoom screen. The Active Environment screen is displayed.

A Note flag, showing the word ***NOTE*** in the STATUS field for IDJOB6, indicates that the job has an appended note.
Adding a Note to a Job

Chapter 6 Navigating The Active Environment

Observe the Note command in the list of commands at the bottom of the screen.

20 Enter **NOTE** in the COMMAND field. The text of the note is displayed in the Status field.

21 Enter **NOTE** in the COMMAND field again. The text of the note is hidden.
This is because the NOTE command acts as a toggle between displaying and hiding the text of each note appended to a job order.

22 Free job IDJOB6, using Option F.

Displaying Table Names for Jobs

Notice the TABLE command at the bottom of the screen. This command toggles between displaying and hiding the name of the scheduling library and table to which each job order belongs.

23 Enter the TABLE command. The name of the scheduling library and table are displayed for each job in the Status field.

<table>
<thead>
<tr>
<th>Filter: IDGS</th>
<th>Control-M</th>
<th>Active Environment</th>
<th>UP</th>
<th>SCROLL</th>
<th>CRSR</th>
</tr>
</thead>
<tbody>
<tr>
<td>O Name</td>
<td>Owner</td>
<td>Odate</td>
<td>Jobname</td>
<td>JobID</td>
<td>Typ</td>
</tr>
<tr>
<td>IDJOB1</td>
<td>ID</td>
<td>020201</td>
<td>M21</td>
<td>/08915</td>
<td>JOB Ended</td>
</tr>
<tr>
<td>IDJOB2</td>
<td>ID</td>
<td>020201</td>
<td>M21</td>
<td>/08916</td>
<td>JOB Ended</td>
</tr>
<tr>
<td>IDJOB3</td>
<td>ID</td>
<td>020201</td>
<td>M21</td>
<td>/08917</td>
<td>JOB Ended</td>
</tr>
<tr>
<td>IDJOB4</td>
<td>ID</td>
<td>020201</td>
<td></td>
<td></td>
<td>JOB Wait Schedule SCHED-LIB=CTM.TEST.SCHEDULE (IDGS2)</td>
</tr>
<tr>
<td>IDJOB5</td>
<td>ID</td>
<td>020201</td>
<td></td>
<td></td>
<td>JOB Wait Schedule SCHED-LIB=CTM.TEST.SCHEDULE (IDGS2)</td>
</tr>
<tr>
<td>IDJOB6</td>
<td>ID</td>
<td>020201</td>
<td></td>
<td></td>
<td>JOB Wait Schedule SCHED-LIB=CTM.TEST.SCHEDULE (IDGS2)</td>
</tr>
</tbody>
</table>

Commands: Opt Display Show History RBal REFresh Auto Jobstat SHPF Note Table
Opt command toggles between Commands and Options display 21.29.12

24 Enter the TABLE command again. The names of the scheduling libraries and tables are hidden.
Displaying Job Dependencies

You have not displayed the Net display type using the DI N command. This display type is generally useful only in the Job Dependency Network screen, which you will soon display. Beforehand, however, it is advisable to refresh the Active Environment display using the REFRESH command, which appears in the list of commands at the bottom of the screen.

25 Enter REFRESH in the COMMAND field. A message indicates that the display has been refreshed.

Toggle to the list of options at the bottom of the Active Environment screen, so you can see exactly which option you will use to display the Job Dependency Network screen.

26 Enter OPT in the COMMAND field. The list of available options is displayed.

One of these options is N (Net). It displays the list of dependencies for a job; that is, the list of predecessor and successor job of the selected job. Specify this option for job IDJOB5.

27 Enter N in the OPTION field next to IDJOB5. The Job Dependency Network screen is displayed (with display type N).

<table>
<thead>
<tr>
<th>Filter: IDGS</th>
<th>------ CONTROL-M NETWORK OF M21JOB5 ------ UP &lt;N&gt; -(3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMAND ====&gt; SCROLL ====&gt; CRSR</td>
<td></td>
</tr>
<tr>
<td>O Level ----- Name ----- DueIN/Out Elaps Late Prio Res ------ Status -----</td>
<td></td>
</tr>
<tr>
<td>-4 IDJOB1    1158 1159 0001 Ended &quot;OK&quot;</td>
<td></td>
</tr>
<tr>
<td>-3 IDJOB2    1158 1159 0001 Ended &quot;OK&quot;</td>
<td></td>
</tr>
<tr>
<td>-2 IDJOB3    1156 1159 0003 Ended- Not &quot;OK&quot; Due</td>
<td></td>
</tr>
<tr>
<td>to CC</td>
<td></td>
</tr>
<tr>
<td>-1 IDJOB4    1158 1159 0001 Wait Schedule</td>
<td></td>
</tr>
<tr>
<td>--&gt; IDJOB5   1158 1159 0001 Wait Schedule</td>
<td></td>
</tr>
<tr>
<td>+1 IDJOB6    1156 1159 0003 Wait Schedule</td>
<td></td>
</tr>
<tr>
<td>*** Note ***</td>
<td></td>
</tr>
<tr>
<td>=========== &gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt; Bottom of Jobs List &lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt; ===========</td>
<td></td>
</tr>
</tbody>
</table>

The Job Dependency Network screen is a special case of the Active Environment screen. It indicates predecessor and successor jobs, and the levels of those jobs, relative to the selected job.

Note the following about the jobs listed in the screen:

- An arrow appears beside the job for which the N option was requested. This job is the starting point for looking at predecessor and successor jobs.

- Immediately below IDJOB5 is its the only successor job, IDJOB6, with a level number of +1. The plus sign indicates that IDJOB6 is a successor job; the integer 1 indicates that IDJOB6 is only one level removed from IDJOB5, that is, it is an immediate successor.

- Above IDJOB5 is the list of predecessor jobs, each displaying a negative level number. The minus signs in the level numbers indicate that these are predecessor jobs of IDJOB5. The integer in the level number indicates the number of levels the job is removed from IDJOB5. For example, IDJOB4, level -1, is the immediate predecessor; IDJOB2, level -3, is three levels away from IDJOB5.

- Jobs appear in the screen in job flow order, from earliest predecessor to latest successor.

Job dependencies are determined according the prerequisite IN and OUT conditions of the job. DO COND conditions are ignored because they are conditional rather than constant.

The screen also displays the following information about the jobs, much of it also found in the Zoom screen:

**Table 18  Job Dependency Fields (part 1 of 2)**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DueIN</td>
<td>Due in time. Time by which the job must be submitted.</td>
</tr>
<tr>
<td>DueOut</td>
<td>Due out time. Time by which the job must finish executing.</td>
</tr>
<tr>
<td>Elaps</td>
<td>Elapse time. Expected time (in minutes) for the job to execute.</td>
</tr>
<tr>
<td>Late</td>
<td>Indication that a job is late. Possible values:</td>
</tr>
<tr>
<td></td>
<td>• X – Actual execution has not completed within the expected execution time. Also indicates that SHOUT WHEN EXECTIME was issued.</td>
</tr>
<tr>
<td></td>
<td>• I – Job was not submitted in time. Also indicates that SHOUT WHEN LATESUB was issued.</td>
</tr>
<tr>
<td></td>
<td>• O – Job is late. Also indicates that SHOUT WHEN LATE was issued.</td>
</tr>
</tbody>
</table>
To refresh the display with the latest information, enter the REFRESH command as you did before displaying the screen. The time of the last refresh is displayed on the top line of the Job Dependency Network screen.

You can change display types in the Job Dependency Network screen, but there is little point in doing so because it is this display type that shows the dependency levels.

Press PF03/PF15 to exit from the Job Network Dependency screen back to the Active Environment screen.

### Displaying Execution Information from Job Runs

### Job Order Execution History Screen

You can now display the execution history for job IDJOB1. As you saw in Chapter 1, “Introduction to Control-M,” you can do this by specifying Option V (View Sysout) for the job.

Enter option V for job IDJOB1. The Job Order Execution History screen for IDJOB1 is displayed.

**NOTE**

This facility requires that Control-M/Restart be operational at your site. If Control-M/Restart is not installed, skip to “Forcing an OK Status for a Job” below.
The following information about the job is displayed at the top of the screen:

### Table 19  Job Information

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEMNAME</td>
<td>Name of the member containing the JCL of the job.</td>
</tr>
<tr>
<td>OWNER</td>
<td>User ID of the owner of the job.</td>
</tr>
<tr>
<td>ORDERID</td>
<td>Job order ID.</td>
</tr>
<tr>
<td>ODATE</td>
<td>Original scheduling date of the job.</td>
</tr>
</tbody>
</table>

The following information is provided for each execution of the job:

### Table 20  Job Execution Information

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JOBNAME</td>
<td>Job name.</td>
</tr>
<tr>
<td>JOBID</td>
<td>JES job number.</td>
</tr>
<tr>
<td>DATE</td>
<td>Execution date of the job.</td>
</tr>
<tr>
<td>START</td>
<td>Start time of the job execution, in format hh:mm.</td>
</tr>
<tr>
<td>ELAPSED</td>
<td>Total elapsed time of the job execution, in format mmmm.nn, where mmmm is minutes, and nn is hundredths of minutes.</td>
</tr>
<tr>
<td>PAGES</td>
<td>Number of pages in the sysout.</td>
</tr>
<tr>
<td>MAX RC</td>
<td>Highest return code of the job execution.</td>
</tr>
<tr>
<td>STATUS</td>
<td>Status assigned to the job by Control-M, based on execution results.</td>
</tr>
</tbody>
</table>
Notice the option S (Select) at the bottom of the screen. You can specify this option to display the sysout of the selected job run. Alternatively, you can enter VIEWALL in the COMMAND field to display the sysout of all runs of the job.

Enter option S for the job run of IDJOB1. The Sysout Viewing screen is displayed.

Figure 49 illustration is obviously more than one screen length. It shows the full sysout.

--- CONTROL-M/CONTROL-R SYSOUT VIEWING ------ PAGE 1 OF 3

COMMAND ===> SCROLL===> CRSR
MEMNAME IDJOB1 OWNER ID JOBNAME M21 ODATE 020201

18.05.20 JOB25002 ---- FRIDAY, 02 FEB 2001 ----
18.05.20 JOB25002 IRR010I USERID STCUSER IS ASSIGNED TO THIS JOB.
18.05.20 JOB25002 ICH70001I STCUSER LAST ACCESS AT 18:03:22 ON FRIDAY, FEBRUARY 18.05.20 JOB25002 $HASP373 M21 STARTED - INIT 3 - CLASS A - SYS OS35
18.05.20 JOB25002 IEF403I M21 - STARTED - TIME=18.05.20
18.05.20 JOB25002 - CPU (Total) Elapse
18.05.20 JOB25002 - Jobname Stepname ProcStep RC I/O hh:mm:ss.th hh:mm:
18.05.20 JOB25002 - S1 00 0 00.02
18.05.20 JOB25002 - IEF404I M21 - ENDED - TIME=18.05.20
18.05.20 JOB25002 - $HASP395 M21 END

--- JES2 JOB STATISTICS ------
- 02 FEB 2001 JOB EXECUTION DATE
- 0 Cards Read
- 44 SYSOUT PRINT RECORDS
- 0 SYSOUT PUNCH RECORDS
- 3 SYSOUT SPOOL KBYTES
- 0.00 MINUTES EXECUTION TIME

**** END OF DATA SET - ADDED BY CONTROL M **** DO NOT USE IT IN YOUR PROGRAM!!
1 //M21 JOB APERLMAN,CLASS=A, // MSGCLASS=X,NOTIFY=ID, // MSGLEVEL=(1,1) //** SUBMITTED BY CONTROL-M (FROM MEMLIB) ODATE=010202 //** SCHEDULE CTM.TEST.SCHEDULE(IDGS1) //** JCL CTM.TEST.JCL(IDJOB1) //** CONTROL-M JOB IDENTIFICATION: ORDER ID=001JW RUN NO.=00001
2 //S1 EXEC PGM=IEFBR14
You can scroll down, up, right, and left through the sysout.

The sysout is divided into sections that you can navigate using the N (NEXT) and P (PREV) commands. The following line appears at the end of each section:

```
**** END OF DATA SET - ADDED BY CONTROL M **** DO NOT USE IT IN YOUR PROGRAM !!
```

31 Enter N in the COMMAND field. The next section of the sysout is displayed.

```
------------------- CONTROL-M/CONTROL-R SYSOUT VIEWING ------ PAGE 2 OF 3
COMMAND ===> Scroll===> CRsr
MEMNAME IDJOB1 OWNER ID JOBNAME M21 ODATE 020201
-------1-------2-------3-------4-------5-------6-------7-------8
1 //M21 JOB APERLMAN,CLASS=A,
   // MSGCLASS=X,NOTIFY=ID,
   // MSGLEVEL=(1,1)
   //*** SUBMITTED BY CONTROL-M (FROM MEMLIB) ODATE=010202
   //*** SCHEDULE CTM.TEST.SCHEDULE(IDGS1)
   //*** JCL CTM.TEST.JCL(IDJOB1)
   //*** CONTROL-M JOB IDENTIFICATION: ORDER ID=001JW RUN NO.=00001
2 //S1 EXEC PGM=IEFBR14
```

32 Enter N in the COMMAND field again. The next section of the sysout is displayed.
Forcing an OK Status for a Job

In some of the exercises in previous chapters, you set up errors in the JCL to achieve certain results, and then edited the JCL to correct those mistakes before rerunning or restarting the job.

There may be situations, however, in which you determine that even though a particular job step had an error, the job should still end OK, that is, where the error is not serious enough to require an Ended Not OK status for the job.

In such a situation, you can use the O (Force OK) option in the Active Environment screen to change a job status to Ended OK, as explained below.

34 Enter Option O for IDJOB3.
Forcing an OK Status for a Job

Free IDJOB3. The status of IDJOB3 changes to Ended “OK” Forced OK.

This, in turn, results in the IDJOB3-ENDED-OK condition to be added to the IOA Conditions file, which in turn allows the remaining jobs (IDJOB4, IDJOB5, and IDJOB6), and the SMART Table Entity to successively end OK.

In the following screen illustration, the status of IDJOB3 has already changed, and IDJOB4 has been changed to Wait Submission.

At the end of the process, the screen is displayed as follows:
Displaying Statistics for a Job

If your site uses utility CTMJSA to collect job statistics, you can display job runtime statistics for a job by entering the S (Statistics) line option.

36 Enter option S for IDJOB1. The Statistics screen is displayed.

For each computer with statistics on the job, an Average Statistics line is displayed followed by individual job or SMART Table Entity statistics for each execution:

- The Average Statistics Line contains the SYSID and SMF ID of the computer for which statistics are calculated, as well as the average ELAPSED, CPU and SRB time for the job on that computer.

- The Individual Execution and/or SMART Table Entity Statistics Line contains the JOBID, Start time, End time, Elapsed time, CPU time, and SRB time for the execution or SMART Table Entity.

37 Exit the Statistics screen to the Active Environment screen.
Displaying Jobs Belonging to a Specific SMART Table

You have already seen that you can filter the Active Environment display on Table name.

But through specification of the T (Table) line option, you can restrict the display to those jobs belonging to the same table, independent of filters. This option applies only to a SMART Table Entity and jobs in a SMART Table.

38 Enter **SHOW nonsense EDIT**. The filter window is displayed with the default edits set, and the nonsense name in the Filter field.

39 Blank out the Filter name and press **Enter**. The Active Environment screen is displayed with the default filter edits, that is, the equivalent of no filter.

40 Scroll to the top of the screen. The full Active Environment is displayed.

41 Find the listing of one of the jobs in the SMART Table (IDJOB2 through IDJOB5), and enter Option T for the job. The Table screen is displayed.

**Figure 50 Table Screen**

<table>
<thead>
<tr>
<th>Filter:</th>
<th>CONTROL-M</th>
<th>TABLE - TBLSCHD</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>O Name</th>
<th>Owner</th>
<th>Odate</th>
<th>Jobname</th>
<th>JobID</th>
<th>Typ</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>TBLSCHD</td>
<td>ID</td>
<td>020201</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IDJOB2</td>
<td>ID</td>
<td>020201</td>
<td>M21</td>
<td>/26694</td>
<td>JOB</td>
<td>Ended</td>
</tr>
<tr>
<td>IDJOB3</td>
<td>ID</td>
<td>020201</td>
<td>M21</td>
<td>/26695</td>
<td>JOB</td>
<td>Ended</td>
</tr>
<tr>
<td>IDJOB4</td>
<td>ID</td>
<td>020201</td>
<td>M21</td>
<td>/26696</td>
<td>JOB</td>
<td>Ended</td>
</tr>
<tr>
<td>IDJOB5</td>
<td>ID</td>
<td>020201</td>
<td>M21</td>
<td>/26697</td>
<td>JOB</td>
<td>Ended</td>
</tr>
</tbody>
</table>

The Table screen is a variation of the Active Environment screen. It contains the SMART Table Entity, and all jobs belonging to the SMART Table of the selected job. The name of the SMART Table appears in the top line of the screen. The display can be further filtered if desired.
42 Exit the screen. The Active Environment screen is displayed.

43 Exit the Online facility.

This completes the steps in this chapter of the Control-M for z/OS Getting Started Guide.

Review

In this chapter you

- looked more closely at the fields in the Show Screen Filter window, defined a new filter, and displayed a list of available filters
- changed between display types of the Active Environment screen
- added a note to a job in the Zoom screen and displayed the contents of the note in the Active Environment screen
- displayed the scheduling library and table names of jobs in the Active Environment screen
- displayed the list of dependent jobs (predecessor and successor jobs) in the Job Dependency Network screen
- displayed the execution history of job orders in the Job Order Execution History screen, and display job sysout in the Sysout Viewing screen
- forced a status of OK for a job in the Active Environment screen
- displayed Statistics for a job in the Statistics screen
- displayed the SMART Table to which a job belongs in the Active Environment screen

Recommended Reading

Before continuing with the next chapter, you should read the following in the Control-M for z/OS User Guide:

In Chapter 2
- information related to filters and filtering the display in the Active Environment screen
- information about display types of the Active Environment screen
- information about adding a note in the Zoom screen and displaying the note in the Active Environment screen
- commands and options of the Active Environment screen
- the Job Dependency Network screen
- the Job Order Execution History screen, and the Sysout Viewing screen
- status information in the Active Environment screen
- the Statistics screen
- any other information about the Active Environment screen that you have not yet read
Job Ordering and New Day Processing

This chapter includes the following topics:

Overview ................................................................. 196
Preparations ................................................................ 196
Job Ordering Through Utility CTMJOB ......................... 196
New Day Processing ..................................................... 199
User Daily Jobs ........................................................... 200
  Defining the JCL of a User Daily Job ......................... 200
  Defining the Date Control Record for a User Daily Job ... 201
  Defining the Job Scheduling Definition of a User Daily Job .... 203
  Additional Points About User Daily Jobs ................. 203
Job Ordering Through Online Utility CTMJOBQ. .............. 205
Other Methods of Job Ordering .................................... 207
Review ................................................................. 208
  Recommended Reading ........................................... 208
Overview

In the previous chapters of this guide, you used the O (Order) or F (Force) option in the Job List screen to order individual jobs. These same options are available in the Table List screen to order or force all the jobs in a particular table.

There are, however, other methods of job ordering. In this chapter of the guide, you will look at job ordering through the following methods:

- Batch utility CTMJOB
- New Day Processing and User Daily jobs
- Online utility CTMJOBRQ

You will also see a brief description of several other methods of job ordering.

--- NOTE ---

Only relevant DD statements are illustrated in the sample JCLs for the exercises in this chapter. Other DD statements, such as DALOG and DACNDF, are not shown.

Preparations

No special preparations are required for this chapter of the guide.

Job Ordering Through Utility CTMJOB

The CTMJOB job ordering utility can be invoked from the job step or by calling the program from a TSO environment and/or application program.

Assume that you want to order jobs IDJOB1, IDJOB2, and IDJOB3 in table IDGS1.

There are quite a few ways to do this using CTMJOB. Several of these ways are illustrated in the following pages.

--- NOTE ---

Specify the appropriate library in place of CTM.TEST.SCHEDULE in each of the alternatives.
Define the JCL for CTMJOB as follows:

```plaintext
//SCHDLIB DD DSN=CTM.TEST.SCHEDULE,DISP=SHR
//DAJOB DD *
ORDER DDNAME=SCHDLIB MEM=IDGS1 JOB=IDJOB1 ODATE=ODATE
ORDER DDNAME=SCHDLIB MEM=IDGS1 JOB=IDJOB2 ODATE=ODATE
ORDER DDNAME=SCHDLIB MEM=IDGS1 JOB=IDJOB3 ODATE=ODATE
   FORCE
```

Notice a few things about this JCL.

- The desired job scheduling definitions are specified in ORDER control statements included in a DAJOB DD statement.
- Each ORDER control statement identifies a library, table and job scheduling definition. To avoid repetition of the scheduling library name in each ORDER statement, the library name was identified in a preceding DD statement.
- The last ORDER statement requested a FORCE. Therefore, job IDJOB3 will be forced regardless of its basic scheduling criteria.

The CTMJOB utility generally uses a DAJOB DD statement to identify the tables and/or jobs to schedule, although an exception will be shown later in this chapter. But the ORDER statements do not have to be included in line in the DAJOB statement. They can, instead, be placed in a parameter member pointed to by the DAJOB statement.

The following steps show how to create an alternative definition for CTMJOB that places the order statements in a PARM member (IDPRM1).

--- NOTE ---

Be sure to specify an appropriate PARM library in place of CTM.TEST.PARM.

---

Define the JCL for CTMJOB as follows:

```plaintext
//SCHDLIB DD DSN=CTM.TEST.SCHEDULE,DISP=SHR
//DAJOB DD DISP=SHR,DSN=CTM.TEST.PARM(IDPRM1)
```

Specify the following contents in the IDPRM1 member in the PARM library:
The above definitions for CTMJOB achieve the same results. The advantage of the second method, using the PARM member, is that you do not have to change the JCL if you want to change the list of jobs to be ordered. You merely need to change the contents of the PARM member.

If the table being scheduled is a SMART Table, you can specify scheduling RBCs to include or ignore in the ORDER statements.

4 Define the following JCL for CTMJOB:

```
//DAJOB     DD    *
ORDER  DDNAME=CTM.TEST.SCHEDULE MEM=IDGS2 ODATE=ODATE
SELECT RBC EXERCISES
IGNORE RBC EOW
IGNORE RBC EOM
//
```

Now that you have seen several instances of using ORDER statements, take a closer look at the ORDER statement syntax.

The syntax for the ORDER statement is as follows:

```
ORDER(DSN=schedlib|DDNAME=dd|DD=dd)|{MEMBER=table|MEM=table}|.JOB=jobnm|
[.ODATE=date|DATE=date]|[.ODATEOPT={VALUE|VAL|RUN}][.FORCE]
[SELECT RBC rbcname1]
[IGNORE RBC rbcname2]
```

Observe the following points about ORDER statements:

- Immediately following the ORDER statement, you must specify either the name of scheduling library (in a DSN parameter), or the name of a DD statement that identifies a scheduling library (in a DD or DD name parameter).
- You must then specify a table name.
- Jobname is optional. If not specified, all jobs in the table are ordered.
- A date parameter, such as ODATE or DATE, is optional if a date control record has been defined.
- The FORCE parameter is optional. It forces a job even if its basic scheduling criteria are not satisfied.
■ IGNORE and SELECT RBC statements are relevant for 'SMART Table scheduling only.

For more information on the parameters in the ORDER statement, see the description of the format of ORDER statements in the Control-M chapter of the INCONTROL for z/OS Utilities Guide.

Thus far, all instances of DD statement DAJOB included, or pointed to, a PARM member that includes ORDER statements. However, DAJOB statements can be defined without ORDER statements.

5 Define the following JCL for CTMJOB:

```
//DAJOB      DD  DISP=SHR,DSN=CTM.TEST.SCHEDULE(IDGS1)
//           DD  DISP=SHR,DSN=CTM.TEST.SCHEDULE(IDGS3)
//ORDERJOB EXEC   PGM=CTMJOB,
//       PARM='ORDER DSN=CTM.TEST.SCHEDULE TABLE=IDGS1 JOB=SORT ODATE=020201'
```

Using this method, the schedule library and table is specified directly in DAJOB DD statement. This method provides no advantage over in-line specification of ORDER statements. It has the disadvantage that you cannot request specific jobs, only whole tables.

It is also possible to use CTMJOB without specifying a DAJOB DD statement. This is done by specifying a PARM parameter in an EXEC statement in the JCL. For example:

```
//ORDERJOB EXEC   PGM=CTMJOB,  
//       PARM='ORDER DSN=CTM.TEST.SCHEDULE TABLE=IDGS1 JOB=SORT ODATE=020201'
```

However, this method is of very limited usefulness because it can only be used to order a single table or job.

If you were to submit any of the defined JCLs for CTMJOB, the specified job scheduling definitions would be ordered. However, instead of submitting the JCLs, for this exercise you should move on to the topic of User Daily jobs.

**New Day Processing**

As mentioned in Chapter 5, “AutoEdit and JCL,” each site defines a time that represents the end of the old working day and the start of the new working day. At the start of the new working day, New Day processing is performed. New Day processing is the process by which daily maintenance is performed and jobs to be scheduled on the new day are placed in the Active Jobs file.
New Day processing is generally performed by a combination of the New Day procedure and User Daily jobs. The topic of New Day Processing is described in detail in the Control-M chapter of the *INCONTROL for z/OS Administrator Guide*.

You can now take a look at User Daily jobs.

**User Daily Jobs**

A User Daily job is a job scheduling definition and JCL job that executes the User Daily procedure CTMDAILY. This procedure, in turn, calls the CTMILU program.

The CTMILU program can be seen as an enhanced CTMJOB, in that it executes CTMJOB, but it also executes other programs that provide additional functionality. You will look at one of these other programs shortly, but first, you should create a User Daily job, as described in the following topic.

**Defining the JCL of a User Daily Job**

6 Based on the above JCL, define a User Daily in member IDUDJ1, but do not exit the JCL:

```plaintext
//JCLLIB ORDER=your.proclib.PROCLIB
//INCLUDE MEMBER=IOASET
//GTSTR5 EXEC CTMDAILY,
//DATEREC=DATERECU <= CHANGE
//DAJOB DD *
ORDER DDNAME=SCHDLIB MEM=IDGS1 JOB=IDJOB1
ORDER DDNAME=SCHDLIB MEM=IDGS1 JOB=IDJOB2
ORDER DDNAME=SCHDLIB MEM=IDGS1 JOB=IDJOB3
```

The DAJOB DD statement looks familiar because these parameters are ultimately passed to CTMJOB. Therefore, any syntax that is valid for passing parameters to program CTMJOB, as demonstrated in previous steps in this chapter, can be used in the User Daily job.

But there is a difference. Note that the order statements do not contain date parameters as they did in the previous steps. Instead the date information is provided by a record called the Date Control record. You can now identify the Date Control record to the job, and then create this Date Control record, using the DATEREC parameter in the procedure that points to the member containing the Date Control record.
Defining the Date Control Record for a User Daily Job

Now note the statement DATEREC=DATERCU. You should examine this item for a moment.

7 In the JCL, change this statement:

// DATEREC=DATERCU

...to this statement:

// DATEREC=IDDCRU

...and then exit the JCL.

Defining the Date Control Record for a User Daily Job

In your PARM library is a member called DATERECU. This is the model Date Control record provided with Control-M. The sample Date Control record is displayed similar to the following:

<table>
<thead>
<tr>
<th>000001</th>
<th>301000</th>
<th>301000</th>
<th>301000</th>
<th>301000</th>
<th>301000</th>
</tr>
</thead>
<tbody>
<tr>
<td>000001</td>
<td>301000</td>
<td>301000</td>
<td>301000</td>
<td>301000</td>
<td>301000</td>
</tr>
</tbody>
</table>

The Date Control record contains an example date that is repeated several times in specific columns. The integrity of these columns must be maintained. The date must appear in the Date Control record of a User Daily job in the following columns:

- 01 - 06
- 18 - 23
- 25 - 30
- 43 - 48
- 50 - 55
- 67 - 72

Each User Daily requires its own Date Control record. You will use this model record to create a Date Control record for the User Daily that you have just created. As you can see from the step you just performed, you are going to call the Date Control record for this User Daily IDDCRU.

8 In the PARM library, copy member DATERECU and call it IDDCRU.

9 Edit member IDDCRU and, without changing the column positions, replace the old dates with the current working date. Then exit the member.
Once you have created a Date Control record for a User Daily, Control-M automatically updates the date information in the record. This is one of the great advantages of using User Dailies to order jobs. You do not have to update date information in a DAJOB DD statement each day, because Control-M provides the updated information through the Date Control record.

This appearance of the date six times (instead of once) in the Date Control record of a User Daily enables Control-M to manage the process of job ordering. At each stage in the job ordering process, the current original scheduling date is placed in one of these fields.

**Enhanced Daily Checkpointing**

An optional, second, Date Control record can be defined for a User Daily to enable Enhanced Daily Checkpointing. If an interruption such as a system crash occurs during job ordering, Enhanced Daily Checkpointing facilitates automatic rerun of the job ordering process by enabling Control-M to identify the last job ordered before the interruption. Control-M can then continue the ordering process from that point.

This second Date Control record has a completely different format than the regular Date Control record. In fact, the Enhanced Daily Checkpointing Date Control record contains far more than dates.

The following table shows the format of the second Date Control record, which is required only if Enhanced Daily Checkpointing is used.

**Table 21 Format of the Second Date Control Record**

<table>
<thead>
<tr>
<th>Column</th>
<th>Constant or Value Added</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>01 - 04</td>
<td>JOB= Constant.</td>
<td></td>
</tr>
<tr>
<td>05 - 12</td>
<td>blank</td>
<td>In this area, Control-M stores the MEMNAME value of the last ordered job.</td>
</tr>
<tr>
<td>13 - 23</td>
<td>,SERIAL_NO= Constant (note the comma before the “S”).</td>
<td></td>
</tr>
<tr>
<td>24 - 28</td>
<td>blank</td>
<td>In this area, Control-M stores its internal sequence number of the last ordered job.</td>
</tr>
<tr>
<td>29 - 37</td>
<td>,ORDERID= Constant (note the comma before the “O”).</td>
<td></td>
</tr>
<tr>
<td>38 - 42</td>
<td>blank</td>
<td>In this area, Control-M stores the order ID of the last ordered job.</td>
</tr>
<tr>
<td>43 - 49</td>
<td>,GROUP= Constant (note the comma before the “G”).</td>
<td></td>
</tr>
<tr>
<td>50 - 69</td>
<td>blank</td>
<td>In this area, Control-M stores the group name of the last ordered job.</td>
</tr>
</tbody>
</table>
When creating this record, you must

- specify the indicated constants, such as JOB=, in the appropriate columns
- leave blank the columns indicated as blank

These columns are filled in by Control-M during User Daily processing.

Defining the Job Scheduling Definition of a User Daily Job

You have now defined the JCL of User Daily job IDUDJ1, and created its Date Control record called IDDCRU. One task remains, however; you must define the job scheduling definition for User Daily IDUDJ1.

10 In your scheduling library, create a new table called IDUDT1, and in it create a job scheduling definition called IDUDJ1. Define the job scheduling definition as follows:

A Ensure that the MEMNAME and MEMLIB fields point to the JCL of IDUDJ1.

B Define the Basic Scheduling parameters so that this job scheduling definition gets ordered every day, for example, by using a DAYS value of ALL.

C Exit the job scheduling definition.

Additional Points About User Daily Jobs

As you saw in previous chapters you can order jobs in the online facility. However, using this method, you must first enter the online facility. Then you must order the jobs, either individually or a table at a time. This method is not bad for an occasional job order, but it certainly is not suitable for ordering a large number or jobs, especially if they are from different tables.

In this chapter, you saw that you can order a large number of jobs using a single run of CTMJOB. And you can keep the list of jobs you specified to CTMJOB for reuse. This eliminates the disadvantages of using online job ordering on regular daily basis.

For daily processing, employing User Daily jobs provides an additional advantage—the Date Control record. The Date Control record has many advantages, including the following facts:

- It can be used in conjunction with the RETRO parameter, described in the parameters chapter of the Control-M for z/OS User Guide, to enable the site to catch up with missed work days—for example, if the system is down for several days.
■ It enables Enhanced Daily Checkpointing, which is useful in case of system crash.

However, even the User Daily job must be ordered. There are several ways this can be done:

■ You can order the User Daily job using the Online facility. But this requires going into the Control-M Online facility—still a manual intervention.

■ You can order the User Daily job using CTMJOB.

■ You can use any of the other job ordering methods available, some of which will be described below. But this, too, involves manual intervention.

■ Finally, you can order User Dailies using the New Day procedure. The New Day procedure is defined and maintained by the Control-M administrator, and is run once each day as part of New Day processing. When User Dailies are defined to the New Day procedure, they are automatically ordered each day that their basic scheduling criteria are satisfied. Most sites use a combination of User Daily jobs and the New Day procedure to completely automate daily job scheduling.

Because the New Day procedure is defined by the Control-M administrator, and is discussed in detail in the INCONTROL for z/OS Administrator Guide, it is not discussed in this guide.

Before moving on to a discussion of alternative methods of job ordering, you should note the following points about User Daily jobs:

■ Control-M provides two sample User Daily job scheduling definitions in member MAINDAY in the SCHEDULE library:
  — DAILYSYS is a sample User Daily for scheduling system jobs
  — DAILYPRD is a sample User Daily for scheduling production jobs.

■ You can define as many User Daily jobs as you want, with each ordering only those jobs that you want it to order. This leaves you great flexibility in organizing your User Daily jobs by whatever system is useful. For example, you can organize User Dailies according to table, application or group, department, basic and runtime criteria of the jobs being ordered, or any other useful criteria or combination of criteria.

■ You can define the scheduling criteria of the User Daily job in any way you wish. For example, if a certain set of jobs is to be processed at the end of the work day, there is no need for them to sit in the Active Jobs file all day. Instead, you can ensure that they are ordered only in the evening or at night, by appropriately defining the TIME FROM criteria of the User Daily that orders those jobs.

■ If you want, you can define a User Daily to run several maintenance procedures that you would like run in succession.
Although the New Day procedure is ordered only once each day, at start of New Day, you can order User Daily jobs whenever you wish, and as often as you wish. Of course, you must ensure that this does not produce unwanted results. If you do not want multiple orders of the same job, you should not run the User Daily more than once.

You can locate your User Daily jobs as you wish. For example, if all the User Daily jobs are placed in a single table, then by ordering that table you order all User Daily jobs that it contains.

Job Ordering Through Online Utility CTMJOBRRQ

As previously discussed, the occasional job scheduling request can be made using the Order or Force option in the Scheduling Definition facility. At some sites, however, access to Scheduling Definition facility may be restricted for security reasons, for example, to prevent the changing or deletion of job scheduling definitions.

As an alternative to ordering or forcing a job in the Scheduling Definition facility, you can order or force a job using Online Utility M1.

Enter the Online Utilities menu and request utility CTMJOBRRQ, which is Option M1. The Job Request Utility panel is displayed.
In this panel you can

■ specify the name of the library, table and job scheduling definition

To request all jobs in a table, specify an asterisk in the job name field.

■ specify the scheduling run date

The basic scheduling criteria of the jobs are checked against this date to see if the job should be ordered as requested.

■ indicate if you want the job forced if it should not be scheduled on that particular run date

■ specify the name of the calendar library

This tells Control-M where to look if a calendar name is specified in the job scheduling definition. Calendars are discussed in Chapter 8, “Additional Features.”

■ specify a group name

--- NOTE ---

The GROUP field is generally useful in tables whose jobs do not inherit the GROUP from the table, and only if an * is specified in the JOB NAME field. It limits the jobs ordered to those belonging to the specified group.
In general, it is probably quicker to use batch utility CTMJOB, especially if you are ordering several jobs. Perhaps the only advantage of online utility CTMJOBRQ is that you do not need to know the format of the order statements—you merely fill in the parameter fields presented in the panel. But CTMJOBRQ is definitely not a powerful utility.

Other Methods of Job Ordering

Thus far, you have seen the following methods of job ordering:

- Order and Force options in the Job and Table List screens
- Batch utility CTMJOB
- User Daily jobs
- Online utility CTMJOBRQ

The following job ordering methods are also available:

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>End User Job Order interface</td>
<td>Enables job ordering through online utility (or CLIST) CTMJBINT M6. Described in the Control-M for z/OS User Guide.</td>
</tr>
<tr>
<td>Utility CTMBLT</td>
<td>Described in the Control-M chapter of the INCONTROL for z/OS Utilities Guide.</td>
</tr>
<tr>
<td>CLIST CTMCJOBS</td>
<td>Enables job ordering directly from the TSO environment. Described in the Control-M for z/OS User Guide.</td>
</tr>
<tr>
<td>Quick submit command CTMQSB</td>
<td>Enables job ordering through Control-M submit command CTMQSB instead of the ISPF submit command. Described in the Control-M for z/OS User Guide.</td>
</tr>
<tr>
<td>Job ordering from special environments</td>
<td>Facilitates job ordering from other environments, such as CICS and ROSCO, through CTMAJO. Described in the Control-M for z/OS User Guide.</td>
</tr>
</tbody>
</table>

12 Exit the Online facility.

This completes the steps in this chapter of the Control-M for z/OS Getting Started Guide.
Review

In this chapter you

■ learned to order jobs using batch utility CTMJOB, and learned several different methods and formats that can be used to let the utility know which jobs to order

■ learned to define the JCL and job scheduling definition for a User Daily job, and how to define its required Date Control record, and a second, optional Date Control record for Enhanced Daily Checkpointing

■ took a look at the CTMJOBRQ online utility panel that can be used to order a job

■ briefly identified several other methods of job ordering

Recommended Reading

Before continuing with the next chapter, it is recommended that you read the following:

■ In the Control-M for z/OS User Guide:
  — In Chapter 2, the description of online utility CTMJOBRQ (M1).
  — In Chapter 3, the description of the RETRO parameter.
  — In Chapter 6, the topic “Job Ordering Methods.”

■ In Chapter 3 of the Control-M for z/OS Administrator Guide, the topic “Job Ordering using New Day Processing.”

■ In Chapter 3 of the Control-M for z/OS Utilities Guide, the description of utility CTMJOB.
Additional Features

This chapter includes the following topics:

Overview ................................................................. 210
Preparations ............................................................. 210
Defining Calendars .................................................... 211
  Structure of the IOA Calendar Definition Facility .... 216
Responding to External Events through CMEM ........ 217
  Event Types Handled by CMEM - Available ON Statements 223
Creating On Spool Jobs ............................................. 224
  Defining On Spool Jobs ......................................... 225
  Bringing the Job On Spool .................................... 229
  Additional Points About On Spool Jobs .................. 230
Batch Utility IOACND ............................................... 231
Miscellaneous Facilities ............................................ 234
  History Jobs File .................................................. 234
  Journaling and Restoration Capability .................... 235
  Accumulating Statistics: Statistics Facility ............ 235
  Automatic Tape Adjustment ................................. 236
  Simulating Job Execution and Forecasting Resource Usage 236
  Control-M/Restart Dataset Cleanup Utility CTRCCLN (R2) 237
  Reporting Facility .............................................. 237
Exit the Online Session ............................................ 238
Review ................................................................. 238
  Recommended Reading ........................................... 239
Overview

In this, the final chapter of the Getting Started guide, you will examine several topics not covered in the other chapters.

What you learn in this chapter will enable you to

■ create a calendar in the IOA Calendar facility
■ take a look at, and create a rule in, the CMEM (Control-M Event Manager) facility
■ bring a job On Spool in CMEM
■ use batch utility IOACND to clean up conditions and resources

You will also learn about the availability of several miscellaneous facilities.

Preparations

To prepare for the exercises in this chapter

■ create a new table called IDGS4
■ create a new job scheduling definition in table IDGS4, called IDJOB10, with these characteristics:
  — In the MEMNAME field, specify IDJOB10
  — In the GROUP field, specify IDGRP5
  — In the DESCRIPTION field, specify THIS IS JOB10
  — In the OUT field, specify IDJOB10-ENDED-OK ODAT +
■ create a trivial JCL for IDJOB10
■ Create a JCL called IDEXT1

The JCL should contain two steps, S1 and S2, each ending with condition code C0000. There should be no job scheduling definition for this JCL.
Defining Calendars

A calendar definition is a collection of parameters, in calendar form, that is used to indicate on what dates of the year scheduling should occur.

Calendars can be very useful for providing basic scheduling criteria, especially when

- scheduling dates do not follow a pattern, or follow a pattern that is not easily specified using regular basic scheduling criteria

  An example might be a WORKDAYS calendar that schedules jobs on all weekdays, Monday through Friday, that are not holidays. Creating this schedule using basic scheduling parameters would be problematic. Defining the weekdays is simple enough, but excluding the holidays will be a problem.

- identical detailed and complex scheduling criteria are used for many jobs

  Instead of detailing the scheduling criteria in the job scheduling definition of each job, the scheduling criteria can be specified once in a calendar, and then only the calendar name need be specified in each the job scheduling definition. An example might be a HOLIDAYS calendar that consists of a number of disparate dates. Specifying this once in a calendar is easier and less error-prone than specifying the same set of dates in many job scheduling definitions.

You can now define a calendar for use in your job scheduling definition.

1. Enter =8 in the COMMAND field. The IOA Calendar Definition Facility entry panel is displayed.

   In the LIBRARY field is the name of the default calendar library at your site. For purposes of illustration, you will call this library IOA.TEST.CAL.
2 In the CALENDAR field, enter the name IDCAL1 and press Enter.
Defining Calendars

Chapter 8 Additional Features

The Calendar Definition screen is displayed for the current year, as indicated by the label under the COMMAND field.

Figure 52 IOA Calendar Definition Screen

<table>
<thead>
<tr>
<th>COMMAND ==&gt;</th>
<th>IOA CALENDAR - IDCAL1</th>
<th>SCROLL==&gt; CRSR</th>
</tr>
</thead>
<tbody>
<tr>
<td>YEAR 2001</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

3 Tab to the field next to the year label, and specify GETTING STARTED. The following screen is displayed:

Notice the layout of the screen:

- It is sectioned according to months. Only the first six months are displayed. You can page down to display the next six months.

- Each section contains three lines:
  - A number line contains up to 31 digits indicating the dates in the month. A plus sign indicates the 10th, 20th and 30th.
  - Directly below the number line is a blank line used for entering a Y value for each date on which scheduling should occur.
  - Directly above the number line is a line containing Ss. Each S indicates either Saturday or Sunday, depending on the defaults defined at your site.
Normally, you would try to create a logical, useful calendar. However, to keep the definition as simple as possible, you will specify Y only for the date on which you are performing these exercises.

4 Specify Y in the field that represents the current working date. The Calendar definition should have only one entry:

```
COMMAND ===> GETTING STARTED
YEAR 2001

01
02
03
04
05
06

TYPE Y IN ALL THE EXECUTION DAYS
```

5 Exit the Calendar Definition screen. The Year List screen is displayed, listing only the definition you just created.
Exit the Year List screen. The Exit Option window is displayed. Type Y in the CREATE field and press Enter to save this newly created calendar. You can now use this calendar in your job scheduling definition.

6 From the IOA Calendar Facility – Entry Panel, type =2 in the COMMAND field and press Enter to enter the Scheduling Definition facility.

7 Edit the job scheduling definition of IDJOB10. In the DCAL field, type IDCAL1 and press Enter, and exit the job scheduling definition.

8 Order IDJOB10. The job is scheduled, and executed, because the basic scheduling criteria defined in calendar IDCAL1 were satisfied.

Although you specified only a calendar name in this exercise, calendars can be used in conjunction with other basic scheduling parameters.

The relationship between DAYS values and the DCAL calendar depends on the format of values specified for DAYS parameter. The same holds true of the relationship between WDDAYS values and the WCAL calendar. For details, refer to the description of parameters DAYS and WDDAYS in the parameters chapter of the Control-M for z/OS User Guide.
Structure of the IOA Calendar Definition Facility

Before continuing with the exercises, you should note some of the similarities between the structure of the Scheduling Definition facility and the structure of the Calendar Definition facility.

- Both job scheduling definitions and calendar definitions are stored in members in a library. Therefore, both facilities have corresponding screens at each relevant level.

  - The gateway to both definition facilities is an entry panel that allows specification of a Library, member, and entity, either a calendar year or a job scheduling definition name.

  - The screen at the next level lists the members in the library

    - The Table List screen lists the members in the Scheduling Definition facility.

    - The Calendar List screen lists the members in the Calendar facility.

  - The screen at the next level lists the entities in the member.

    - The Job List screen lists the entities in the Scheduling Definition facility member.

    - The Year List screen lists the entities in the Calendar facility member.

  - The screen at the next level is the definition screen itself.

    - The Job Scheduling Definition screen provides Scheduling Definition facility information.

    - The Calendar Definition screen provides Calendar Definition facility information.

- The description you provide in the definition screen appears in the Job List screen or the Year List screen.

- If you create or modify a definition, an Exit Option window is displayed upon exiting the Job List screen or Year List screen to enable you to save or cancel your work.
Responding to External Events through CMEM

Until now, almost everything you have done in this guide has been connected to the scheduling of jobs under the Control-M monitor. You learned how to define jobs, how to order the jobs, how to check up on, and intervene in, the processing, how to perform restart, and so on.

However, Control-M provides a facility, the Control-M Event Manager (CMEM) facility, that enables Control-M to perform specified actions in response to external events. External events are events in the system that occur outside direct operation of Control-M, such as submission of a job not under the control of the Control-M monitor.

The CMEM facility is comprised of a monitor and a subsystem. The facility employs sets of user-defined rules that specify events to monitor and actions to perform if a specified event occurs.

You can now enter the CMEM facility.

9 Enter Option C in the IOA Primary Option menu, or =C from any other location. The CMEM entry panel is displayed.

Figure 54   CMEM Rule Definition Facility – Entry Panel

CMEM has a library, member, and rule structure much like the library, member, and job scheduling definition structure of the Scheduling Definition facility.
This is reflected in the similarity between the Scheduling Definition entry panel and the CMEM entry panel, which has entry fields for LIBRARY, TABLE, and RULE.

Other screens of the CMEM facility also correspond to those of the Scheduling Definition facility:

- Table List Screen
- Rule List screen, which corresponds to the Job List screen
- Rule Definition screen, which corresponds to the Job Scheduling Definition screen

--- NOTE ---
Many commands and options in the corresponding screens are also the same.

---
You can now create a CMEM table and rule. Use a test CMEM library if one exists, or use the default CMEM rule library. For the purposes of this guide, you will call this library CMEM.TEST.RULES.

10 In the CMEM entry panel, enter IDCM1 in the TABLE field, and rule IDRUL1 in the RULE field. Use either the default or a test CMEM rule library. For the purposes of this guide, this library is called CMEM.TEST.RULES.

--- NOTE ---
If Control-O is installed, your site should not be running CMEM.

---

<table>
<thead>
<tr>
<th>SPECIFY LIBRARY, TABLE NAME, RULE NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIBRARY  ===&gt; CMEM.TEST.RULES</td>
</tr>
<tr>
<td>TABLE    ===&gt; IDCM1</td>
</tr>
<tr>
<td>(Blank for table selection list)</td>
</tr>
<tr>
<td>RULE     ===&gt; IDRUL1</td>
</tr>
<tr>
<td>(Blank for rule selection list)</td>
</tr>
</tbody>
</table>

The Rule Definition is displayed. It immediately displays the following message at the top of the screen:

IOAE4CE AT LEAST ONE "ON" STATEMENT MUST BE FILLED IN
The screen is displayed as follows:

**Figure 55   CMEM Rule Definition Screen**

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>SCROLL</th>
<th>CRSR</th>
</tr>
</thead>
<tbody>
<tr>
<td>+--------+--------+-------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ON</td>
<td>JArrival JEnd Dsnevent Step</td>
<td></td>
</tr>
<tr>
<td>OWNER ID</td>
<td>GROUP</td>
<td>MODE</td>
</tr>
<tr>
<td>THRESHOLD</td>
<td>DESCRIPTION</td>
<td></td>
</tr>
</tbody>
</table>

---

The CMEM rule definition is much simpler than the job scheduling definition. Basically, it enables specification of the following types of parameters:

- ON parameters specify the situation or event that should trigger the rule.
- DO parameters specify the actions the rule should perform.
- General parameters specify general information about the rule. In the screen, the general information appears immediately under the ON statement.

The CMEM rule lacks Basic and Runtime scheduling criteria. CMEM tables are usually ordered, or loaded to memory, when CMEM is started. They can also be refreshed or loaded by an operator command, or manually, by using the FORCE option in the CMEM Table List screen.

A CMEM rule in memory is triggered, that is, all its DO statements are performed, by the occurrence of the events specified in the ON statements of the rule.

You can now define your CMEM rule. Of course, to define a rule, you need to know its purpose. The rule you are about to define should accomplish the following:

Job IDEXT1 is a job that is NOT submitted under the Control-M monitor. However, step S2 of job IDEXT1 performs certain processing that is required before certain other jobs can be submitted by Control-M. Therefore, the purpose of the rule is to inform Control-M when step S2 in job IDEXT1 ends OK.

11 In the ON statement, enter the value **STEP**. The ON STEP statement is displayed.
12 Fill in the ON STEP parameters as follows:

A Following the = symbol in the ON STEP statement, specify **IDEXT1**, which is (the job name).

B In the PROCSTEP field, specify **S2**.

--- **NOTE**

In this screen, the PGMSTEP field comes after the PROCSTEP field, which is the reverse of the positioning of these fields in the Job Scheduling Definition screen.

C In the STEPRC field, specify **OK**.

13 Fill in the General parameters as follows:

A In the MODE field, specify **TEST**.

B In the DESCRIPTION field, specify **ON JOB IDEXT1 STEP S2:OK ADD COND**.

This section is displayed as follows:

14 In the DO statement, enter **COND**. The DO COND statement is displayed.

This statement is similar to the DO COND statement in a job scheduling definition.

15 Specify **IDS2-IDEXT1-OK** in the condition field, **ODAT** in the date field, and **+** in the action field.

The statement is displayed as follows:
The entire rule definition is displayed as follows:

```

DO COND     = IDS2-IDEXT1-OK      ODAT +
DO

RULES OF LIBRARY: CMEM.TEST.RULES       TABLE: IDCMI
COMMAND ===>                                  SCROLL=== > CRSR
+----------------------------------------------------------+
ON STEP     = IDEXT1   JTYPE   SMFID      SYSTEM        PROCSER          PGMSTEP S2  STEPRC OK
OWNER ID    GROUP                         MODE TEST  RUNTSEC
DESCRIPTION ON JOB IDEXT1 STEP S2:OK  ADD COND
THRESHOLD
----- >> >>>>>>>>>>>>>>> END OF RULE DEFINITION PARAMETERS <<<<<<<<<<<<<<<<<<< -----

16  Exit the CMEM definition. The Rule List screen is displayed.

Figure 56  CMEM Rule List Screen

```

In addition to displaying the rule name and description, the Rule List screen also displays the rule type, which in this case is Z. The rule type is determined by the type of ON statement defined. The letter Z is used to indicate that the rule is an ON STEP rule. There are other rule type codes that will be identified later in this chapter.
17 Exit the Rule List screen. The Exit Option window is displayed.

**Figure 57** CMEM Exit Option Window

<table>
<thead>
<tr>
<th>RULES OF LIBRARY: CMEM.TEST.RULES</th>
<th>TABLE: IDCM1</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMAND</td>
<td>====&gt; CRSR</td>
</tr>
<tr>
<td>OPT R</td>
<td>--------------</td>
</tr>
<tr>
<td>PLEASE SELECT EXIT OPTION</td>
<td>&lt;&lt; =========</td>
</tr>
<tr>
<td>SAVE CREATE LIBRARY CMEM.TEST.RULES</td>
<td></td>
</tr>
<tr>
<td>TABLE IDCM1</td>
<td></td>
</tr>
</tbody>
</table>

18 Enter \textbf{Y} in the CREATE field. The CMEM entry panel is displayed again.

You can now load to memory the table that contains the rule you just defined.

19 Ensure the TABLE and RULE name fields are blank and press Enter. The Table List screen is displayed.

20 Enter Option \textbf{F} (Force) for the table IDCM1. If a Confirmation window is displayed, confirm the request.

The Control-M CMEM Order Messages screen displays a message indicating that the table is about to be ordered by the CMEM monitor.

**Figure 58** Control-M CMEM Order Messages Screen

<table>
<thead>
<tr>
<th>COMMAND ====&gt; CONTROL-M CMEM ORDER MESSAGES ----(C.ORDER) --- SCROLL====&gt; CRSR</th>
</tr>
</thead>
<tbody>
<tr>
<td>02.49.53 M0F15C1 TABLE IDCM1 ODATE 020201 WILL BE ORDERED BY CMEM MON.</td>
</tr>
<tr>
<td>&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt;&gt; END OF MESSAGE LIST &lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt;&lt; ====&gt; CR</td>
</tr>
</tbody>
</table>

21 Exit the Control-M CMEM Order Messages screen. The CMEM Table List screen is displayed.

22 Enter \textbf{=4} to check the IOA Conditions/Resources screen for condition IDS2-IDEXT1-OK.

The condition should not appear because the rule that creates it has not yet been triggered.

23 Submit the JCL of job IDEXT1. Once Step S2 has ended OK, check again for condition IDS2-IDEXT1-OK in the IOA Conditions/Resources screen.
The condition does not appear, due to the rule being in TEST mode, as specified in 13 on page 220. However, if the rule had been defined in PROD mode, the condition would appear, indicating that the CMEM rule detected the successful completion of step S2 and added the specified condition to the IOA Conditions file. Control-M jobs that have this condition as an IN condition can now be submitted, assuming all other scheduling and runtime criteria are satisfied.

**Event Types Handled by CMEM - Available ON Statements**

You have already seen the ON STEP event. Shown below is a table with the complete list of ON statements available to CMEM.

<table>
<thead>
<tr>
<th>Event</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSNEVENT</td>
<td>Dataset disposition, whether cataloged, deleted, or kept, during step termination or dynamic decollation, or the occurrence of a NOT CATLGD 2 event, which usually occurs when a dataset name is created in a job step but not cataloged because its name already exists in the MVS catalog. Specified in an ON DSNEVENT statement in the rule.</td>
</tr>
<tr>
<td>JOBARRIV</td>
<td>Arrival of a job on the JES spool from any source. Examples: Jobs submitted by a TSO user or by CICS, or jobs received over an NJE network. Specified in an ON JOBARRIV statement in the rule.</td>
</tr>
<tr>
<td>JOBEND</td>
<td>Completion of a job regardless of its source. Specified in an ON JOBEND statement in the rule.</td>
</tr>
<tr>
<td>STEP</td>
<td>Termination of a job step. Specified in an ON STEP statement in the rule.</td>
</tr>
</tbody>
</table>

The following list indicates the Rule type code that appears in the Rule List screen, depending on the type of ON statements specified in the rule:

- R – Job arrival
- X – Job end
- D – Dataset
- Z – Step
CMEM Actions - Available DO Statements

You have already used a DO COND statement in your rule. The following table provides the complete list of DO statements available to CMEM:

Table 24  CMEM Action Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DO statement</td>
<td>Action to be performed when the rule is triggered. Subparameters may be displayed. Valid DO statements are:</td>
</tr>
<tr>
<td></td>
<td>■ DO COND–Add or delete a prerequisite condition.</td>
</tr>
<tr>
<td></td>
<td>■ DO FORCEJOB–Force a job order under Control-M.</td>
</tr>
<tr>
<td></td>
<td>■ DO STOPJOB–Stop execution of the remaining steps of the job that triggered the rule.</td>
</tr>
</tbody>
</table>

The following actions can be defined if Control-O is installed:

■ DO RULE–Invoke a Control-O rule from within the current rule.
■ DO SHOUT–Issue a message to a specified destination using the Shout facility.

Creating On Spool Jobs

An On Spool job is an independently submitted job or started task, such as a job submitted by a TSO user, that is brought under the control of the Control-M monitor using a CMEM rule.

Such a CMEM rule must

■ be an ON JOBARRIV rule that is triggered by the arrival of the job on the JES spool

■ have a DO FORCEJOB statement that forces a job scheduling definition that “matches” the arriving job, so that Control-M can use the instructions in the job scheduling definition to control the job

The issue of matching the job scheduling definition to the arriving job is discussed in more detail later.
Control-M then controls the entire life cycle of the job, except submission, according to the instructions in the forced job scheduling definition. Because the job has already been submitted, Control-M does not submit the job. However, if the job is held, Control-M releases the job when the runtime scheduling criteria are met.

Defining On Spool Jobs

The following components are necessary to create On Spool jobs:

- job to be brought On Spool
- CMEM rule
- job scheduling definition

The following tables clarify the connections between the components by identifying the values you will specify and the reasons for those values. This understanding should be gained before you create the necessary components.

**Table 25  Job To Be Brought On Spool - Values for Exercise**

<table>
<thead>
<tr>
<th>Item</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Member</td>
<td>IDEXT2</td>
<td>Name of the JCL member of the External job.</td>
</tr>
<tr>
<td>Library</td>
<td>“CTM.TEST.JCL”</td>
<td>The test (or personal) JCL library you have been using for these exercises. As in the previous exercises, call it CTM.TEST.JCL</td>
</tr>
</tbody>
</table>

**Table 26  CMEM Rule - Values for Exercise  (part 1 of 2)**

<table>
<thead>
<tr>
<th>Item</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule</td>
<td>IDEXP2</td>
<td>Name of rule that will bring job IDEXT2 On Spool.</td>
</tr>
<tr>
<td>Member</td>
<td>IDCM1</td>
<td>Rule Table that contains the rule, which is shown in the same table as in previous exercise.</td>
</tr>
<tr>
<td>Library</td>
<td>“CMEM.TEST. RULE”</td>
<td>CMEM Rule library used in previous CMEM exercise.</td>
</tr>
<tr>
<td>ON</td>
<td>ON JOBARRV IDEXT2</td>
<td>On statement in rule that detects the arrival of the external job to be brought on spool.</td>
</tr>
<tr>
<td>DO</td>
<td>FORCEJOB</td>
<td>Statement that forces the job scheduling definition that matches the arrived job IDEXT2. TABLE, JOB, and LIBRARY are the relevant subparameters of this statement, and are described below:</td>
</tr>
</tbody>
</table>
You can now create the components.

### Creating the JCL of the Job

Using the values in the preceding tables, for the JCL job, create the JCL of the job that will be placed on the JES Spool. The job can be a trivial job, but ensure that the job card has the following characteristics:

- The job should be submitted with TYPRUN=HOLD to delay its execution and permit Control-M to determine when to run the job.
The MSGCLASS sysout of the job should be one of the following to enable Control-M to read the sysout for the job and perform postprocessing according to the job scheduling definition:

– For JES3 users, it must be equal to the Control-M SYSOUT held class.
– For JES2 users, it can be any held SYSOUT class.

Creating the CMEM Rule

Using the values in the preceding tables, for the CMEM rule, create CMEM rule IDEXP2. Ensure that the rule has the following characteristics:

- The ON JOBARRIV statement should have the job name value of IDEXT2 after the = symbol. The rest of the fields can be blank.
- For the General parameters, the OWNER field should contain your user ID, the MODE field should contain the value TEST, and the DESCRIPTION field should say CREATE ON SPOOL JOB.

This section of the rule is displayed as follows:

```
ON JOBARRIV = IDEXT2
OWNER ID ID
DESCRIPTION CREATE ON SPOOL JOB
```

- The DO section should contain a DO FORCEJOB statement with the following subparameters:
  - In the TABLE field, IDGS4.
  - In the JOB field, IDEXT2.
  - In the DATE field, ODAT
  - In the LIBRARY field, the scheduling library that contains table IDGS4, which in this exercise is CTM.TEST.SCHEDULE.

The DO section is displayed as follows:

```
DO FORCEJOB = TABLE IDGS4 JOB IDEXT2 DATE ODAT
LIBRARY CTM.TEST.SCHEDULE
DO
```

The entire rule definition is displayed as follows:
26 Exit the Rule Definition screen, and the Rule List screen. In the Exit Option window, save the rule.

Creating the Job Scheduling Definition

27 Using the values in the preceding tables, for the job scheduling definition, create the job scheduling definition IDEXT2 in table IDGS4. Ensure that the definition has the following characteristics:

- The MEMNAME field contains IDEXT2.
- The MEMLIB field points to the JCL library of the arrived job, which in this exercise is CTM.TEST.JCL.

You can now define the following parameters in this job scheduling definition:

28 In the GROUP field, specify IDGRP6.

29 In the DESC field, ON SPOOL EXERCISE.

30 Fill in the OUT fields with the following values:

   IDEXT2-ENDED-OK  ODAT +

The job scheduling definition is displayed as follows:
31 Exit and save the job scheduling definition.

Bringing the Job On Spool

You can now bring the job On Spool.

32 Force the table that contains rule IDEXT2 to memory.

33 Submit the JCL of job IDEXT2, for example, from TSO.
The CMEM rule detects the arrival of job IDEXT2 on the JES spool, and forces the job scheduling definition of job IDEXT2. Control-M then recognizes this as an On Spool job and tracks and controls the job using the instructions in the job scheduling definition.

34 Check the job log. The job log indicates the appropriate Control-M handling of the job, depending on the results of the execution (assuming that the rule is in PROD mode).

35 Check the IOA Conditions/Resources screen.

The condition IDEXT2-ENDED-OK is displayed in the screen. This indicates that the job was brought under the control of the Control-M monitor.

Additional Points About On Spool Jobs

The following points relate to component definition:

- The job name specified in the ON JOBARRIV statement in this rule must match the name of the job to be monitored. It can be a full job name, or it can be a mask if a group of jobs is to be monitored.

- The job scheduling definition must be forced by the first DO FORCEJOB statement in the CMEM rule.

- The MEMNAME value in the job scheduling definition must match the name of the external job. A mask can be specified in the MEMNAME field if the same job scheduling definition is used for more than one job.

- Appropriate runtime scheduling criteria for the job must be defined in the job scheduling definition. This enables Control-M to control when the job should be run, that is, its execution.

- Desired post-processing actions must be defined in the job scheduling definition.

On Spool jobs are handled as follows:

- When the job arrival event occurs, Control-M forces the requested table or job.
  
  - If the MEMNAME value in the requested table or job does not match the name of the arriving job, the table or job is forced and processed regularly by Control-M, which presumes that a job is submitted when its runtime scheduling criteria are met, and so on.

  - If the MEMNAME value in the requested table or job matches the name of the arriving job, the job becomes an On Spool job and Control-M
- replaces the MEMNAME mask, if a mask was specified in MEMNAME, with
  the name of the arriving job

- assigns the job ID of the job that triggered the event to the forced job

- forces the job

For details and exceptions see the discussion of On Spool job scheduling
definition considerations in the Control-M for z/OS User Guide.

The forced job appears in the Active Environment screen with
WAIT SCHEDULE ON SPOOL as its status.

- Control-M starts processing the forced job when all runtime scheduling criteria
defined in the job scheduling definition are satisfied. If there are no runtime
  scheduling criteria in the job scheduling definition, Control-M starts processing the
  job immediately.

- Control-M looks for the job in the spool and releases it, if required.

  — If the external job is waiting for execution in HELD state, that is, if the job arrives
    on spool with the TYPRUN parameter set to HOLD, Control-M releases it for
    execution.

  — Otherwise, Control-M verifies that the job is still in the spool, and is waiting for
    execution, executing, or ended, before performing postprocessing.

- Control-M waits for the job to finish execution, reads its SYSOUT, analyzes the
  execution results, and performs all the postprocessing actions defined in the job
  scheduling definition.

By default, Control-M can only handle On Spool jobs that originate on the same NJE
node on which Control-M is running.

**Batch Utility IOACND**

Utility IOACND is a utility that is generally included in New Day processing but can
also be run throughout the day on an as-needed basis. The utility adds, verifies the
existence of, deletes, or modifies prerequisite conditions and resources in the IOA
Conditions file and/or the Control-M Resources file. It can be activated as a started
task (STC), from TSO, or from within a user program.

A copy of the utility is pre-supplied with Control-M. You need only provide the
desired control statements, as discussed below:
In the pre-supplied copy of the utility, add the following control statements under the EXEC statement:

```
ADD COND IDGS-MONITOR-UP %%MONTH.%%DAY
DELETE COND IDJOB10-ENDED-OK %%MONTH.%%DAY
CHANGE RESOURCE IDTAPE 3
ADD CONTROL IDDATA1 E
```

This job must be submitted through Control-M to resolve the AutoEdit variables.

Submit the utility.

Enter the IOA Conditions/Resources screen.

- Condition IDGS-MONITOR UP appears with the current ODATE.
- Condition IDJOB10-ENDED-OK with the current ODATE no longer appears.
- Three units of quantitative resource IDTAPE appear in the screen.
- Control resource IDDATA1 has been assigned Exclusive control.

Take a closer look at the control statements you can use in this utility. All control statements must have the following format:

```
action entity_type entity_name qualifiers
```

where

- `action` can be one of the following:
  - ADD Valid for any entity type.
  - DELETE Valid for any entity type.
  - CHECK Valid for COND only. Verifies that the condition exists.
  - CHANGE Valid for RESOURCE only. Changes the resource quantity.

- `entity_type` can be one of the following:
  - COND For prerequisite conditions
  - RESOURCE For quantitative resources
  - CONTROL For control resources

- `entity_name` is the name of the condition or resource.

- `qualifiers` depend on the entity type, as indicated in the following table:
Table 28 Qualifiers and Entity Types

<table>
<thead>
<tr>
<th>Entity Type</th>
<th>Qualifier</th>
<th>Description and Valid Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>COND</td>
<td>cond_date</td>
<td>Condition date. Valid values are:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>literal – mmdd or ddmm site format</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DATE – Current Gregorian date</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WDATE – IOA working date</td>
</tr>
<tr>
<td></td>
<td></td>
<td>STAT – static date.</td>
</tr>
<tr>
<td>RESOURCE</td>
<td>resc_quantity</td>
<td>Resource quantity. In either of two formats:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>n (integer, such as 2 or 3). Valid for ADD and DELETE actions.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>+/- n (such as -2 or +3). Valid for CHANGE actions. It specifies the amount and direction of the change.</td>
</tr>
<tr>
<td>CONTROL</td>
<td>control_type</td>
<td>Type of Control. Valid values are:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>E – Exclusive</td>
</tr>
<tr>
<td></td>
<td></td>
<td>S – Shared</td>
</tr>
</tbody>
</table>

Note the following points about utility IOACND:

- It can be activated as a started task (STC), from TSO, or from within a user program.

- The addition, deletion, modification or verification is recorded three ways:
  - in the job sysout
  - as a message on the operator console
  - as an event in the IOA Log file.

- Parameters can be received by the utility in two ways:
  - using DD statement DACNDIN (or SYSIN)
  - using PARM.

The parameters required depend on whether prerequisite conditions, control resources or quantitative resources are being processed.
If it is necessary to add, delete, verify, or modify prerequisite conditions or resources during a job step, CTMCND can be called as a procedure from within the application program.

IOA Online Utility I1, which is described in the IOA Utilities chapter of the Control-M for z/OS User Guide, can be used for adding, deleting or checking a prerequisite condition in the IOA Conditions file.

### Miscellaneous Facilities

The following facilities may prove of special interest to you. There are no exercises to perform regarding these facilities, and you can find their descriptions in the appropriate guide. They are listed below to ensure you know they are available.

Some of the following descriptions are taken from the introductory chapter of the Control-M for z/OS User Guide.

#### History Jobs File

During New Day processing, jobs that have ended OK or whose retention period has expired, according to job scheduling definition parameters, are deleted from the Active Jobs file.

If Control-M/Restart is installed, these jobs can be placed in the History Jobs file during New Day processing. This is an optional feature that can be activated by the INCONTROL administrator. Activation of this feature is described under parameter HIST in the Control-M chapter of the INCONTROL for z/OS Administrator Guide.

Jobs in the History Jobs file can be restored, by request, to the Active Jobs file, for subsequent restart.

Jobs remain in the History Jobs file until they are deleted according to criteria defined in the job scheduling definition.

The contents of the History Jobs file can be viewed from the History Environment screen, as described in the online facilities chapter of the Control-M for z/OS User Guide.
Journaling and Restoration Capability

The Control-M Journal file collects data about changes in the Control-M Active Jobs file, the IOA Conditions file, and the Control-M Resources file that occur during the Control-M working day.

The Journal file is initialized each day during New Day processing. From that point on, for the rest of the working day, the Control-M monitor records in the Journal file all job processing activities that impact the Control-M Active Jobs file, and all prerequisite condition additions to and deletions from the IOA Conditions file and the Control-M Resources file.

If the Control-M Active Jobs file, and optionally, the IOA Conditions file and the Control-M Resources file, need to be restored, for example, following a system crash, utility CTMRSTR can be run to restore the files. The utility uses data from the Journal file to restore the files to the status they had at any specific time after the last run of the New Day procedure.

The Control-M Journal file is initialized each day during New Day processing. Therefore, the time at which the New Day procedure initialized the Journal file is the earliest time to which the Control-M Active Jobs file, the Control-M Resources file, or the IOA Conditions file can be restored.

Journaling and Restoration is an optional feature that can be activated by the INCONTROL administrator. It is described in the Control-M chapter of the INCONTROL for z/OS Administrator Guide, and activation of this feature is described under parameter JRNL in the chapter on installing Control-M in the INCONTROL for z/OS Installation Guide.

Accumulating Statistics: Statistics Facility

As part of the postprocessing for each job, Control-M determines the elapsed run time of the job. All accumulated information regarding job execution, including the elapsed run time, is written to the IOA Log file.

Periodically, statistics utility CTMJSA, which was mentioned in Chapter 6, “Navigating The Active Environment,” and which is described in detail in the Control-M chapter of the INCONTROL for z/OS Utilities Guide, can be used to scan and analyze the IOA Log file. This utility gathers information about start time, elapsed run time, CPU utilization time, and so on, for each job. The utility places this information in the Statistics file, where averages of these values can be maintained for each job.
Statistics facility averages can be used for several purposes, such as

- determining if the execution time of a job falls outside a statistically normal range of time, which would indicate an execution delay or problem
- determining when a shout message should be issued based on the elapsed time of a job
- simulating job executions and forecast the impact of changes to the system.
- determining if a job can complete execution before the Control-M planned shutdown time (QUIESCE command)
- calculating DUE-IN time for use by the Deadline Scheduling facility

For more information, see the discussion on Automatic Job Flow Adjustment in the introductory chapter of the *Control-M for z/OS User Guide*.

**Automatic Tape Adjustment**

The Automatic Tape Adjustment facility collects and analyzes statistics regarding tape drive usage, and automatically allocates the appropriate number of tape drives at job order time. This facility, which can be implemented by your INCONTROL administrator, overrides any tape drive Quantitative resource value specified in the job scheduling definition. For more information, see the discussion about tape device usage statistics in the description of the Statistics screen and the description of the RESOURCE parameter in the *Control-M for z/OS User Guide*.

**Simulating Job Execution and Forecasting Resource Usage**

Using statistics accumulated by the Statistics facility, the Simulation and Forecasting facility simulates the actions of the Control-M monitor under the conditions specified in simulation parameters.

The Simulation and Forecasting facility enables you to forecast anticipated job load for a specified time in the future, and to forecast the effects that possible changes to the system might have.

The Simulation and Forecasting facility can improve the efficiency of your site. It can help with resource and configuration decisions, and it can help with the planning of workload scheduling to achieve maximum utilization of resources.

The Simulation and Forecasting facility is described in the chapter on the Simulation and Forecasting facility in the *Control-M for z/OS User Guide*. 
The facility utilizes the Simulation and Forecasting procedure that can be activated by online utility CTMCSIM, which is run by Option M3 in the IOA Utility menu. This utility, which can also activate the Tape Pull List procedure, is described in the IOA Utilities chapter of the Control-M for z/OS User Guide.

Control-M/Restart Dataset Cleanup Utility CTRCCLN (R2)

Utility CTRCCLN, the Control-M/Restart Dataset Cleanup Utility, is used to manually request dataset cleanup. The utility places a CONTROLR step in the job stream and submits the job. The CONTROLR step performs the dataset adjustment, including step adjustment, if necessary, and then stops. No further job steps are executed. The utility adds a step that prevents the execution of the other steps in the job.

This utility can be requested by entering option R2 in the IOA Utility menu. The utility is described in detail in the CONTROLR step chapter of the Control-M/Restart User Guide.

Reporting Facility

Control-M supports a comprehensive reporting facility, which can produce the following types of reports:

Table 29  Report Types

<table>
<thead>
<tr>
<th>Reports</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Keystroke Language Reports</td>
<td>These are reports generated with the Keystroke Language (KSL). KSL is a general purpose reporting language, based on the Online facility, capable of producing numerous reports from the database.</td>
</tr>
<tr>
<td>Special Purpose Reports</td>
<td>These reports include the Job Flow reports that are generally used to track the dependencies between jobs, and the Job Plan reports that are used to anticipate which jobs are scheduled each day.</td>
</tr>
</tbody>
</table>

Sample reports are provided in the IOA SAMPLE library. The Reporting facility is described in the Keystroke Language (KSL) chapter of the Control-M for z/OS User Guide. Special purpose reports are described in the INCONTROL for z/OS Utilities Guide.
Exit the Online Session

39 Exit the Online facility.

This completes the steps in this chapter of the Control-M for z/OS Getting Started Guide.

Review

In this chapter you

- created a calendar in the IOA Calendar facility and used it to schedule a job
- created a CMEM table and rule, and loaded the rule to memory
- used a CMEM rule to bring a job On Spool in CMEM
- used batch utility IOACND to cleanup conditions and resources
- learned about the following facilities and capabilities
  - History Jobs File
  - Journaling and Restoration Capability
  - Accumulating Statistics: Statistics Facility
  - Automatic Tape Adjustment
  - Simulating Job Execution and Forecasting Resource Usage and utility CTMCSIM
  - Control-M/Restart Dataset cleanup utility CTRCCLN
  - Reporting Facility
Recommended Reading

It is recommended that you read the following:

■ In the *Control-M for z/OS User Guide*:

  – In Chapter 2, the description of:

    ■ IOA Calendar facility
    ■ CMEM facility
    ■ Control-M History Jobs file
    ■ Control-M Statistics file (and screen)
    ■ Online utility CTMCSIM (M3)

  – In Chapter 4, the description of the CMEM facility, especially relating to On Spool Jobs

  – In Chapter 7, Control-M Simulation and Forecast Facility

■ In the *Control-M/Restart User Guide*:

  – In Chapter 2, the description of online utility CTRCCLN (R2)

■ In the *Control-M for z/OS Utilities Guide*,

  – In Chapter 2, the description of utility IOACND
  – In Chapter 3, the description of utilities CTMJSA
Sample JCLs

Shown below are sample JCLs used for the exercises in this book, according to chapter and job. The samples indicate how the JCL appears at the time of preparation. They do not indicate changes made during the course of the exercises. It may be necessary to customize some of these JCLs, particularly when names, such as a site-defined library name, must be specified.

Chapter 1

JOB1

```
EDIT       CTMP.JCL(M21JOB1) - 01.05  Columns 00001 00072
****** ************************************************************* Top of Data *****************************************************
000001 //M21  JOB APERLMAN,CLASS=A,
000002 //     MSGCLASS=X,NOTIFY=M21
000003 //S1   EXEC  PGM=IEFBR14
****** ************************************************************* Bottom of Data *****************************************************
```

JOB2

```
EDIT       CTMP.JCL(M21JOB2) - 01.00  Columns 00001 00072
****** ************************************************************* Top of Data *****************************************************
000001 //M21  JOB APERLMAN,CLASS=A,
000002 //     MSGCLASS=X,NOTIFY=M21
000003 //S1   EXEC  PGM=IEFBR14
****** ************************************************************* Bottom of Data *****************************************************
```
## Chapter 2

### JOB3

<table>
<thead>
<tr>
<th>EDIT</th>
<th>CTMP.JCL(M21JOB3) - 01.03</th>
<th>Columns 00001 00072</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>*************************** Top of Data ***********************************</td>
<td></td>
</tr>
<tr>
<td>000100</td>
<td>//M21 JOB ,A.PERLMAN',CLASS=A,MSGCLASS=X,NOTIFY=M21</td>
<td></td>
</tr>
<tr>
<td>001400</td>
<td>//S1 EXEC PGM=IOATEST,PARM='TERM=C0008'</td>
<td></td>
</tr>
<tr>
<td>001500</td>
<td>//STEPLIB DD DSN=IOAP.LOAD,DISP=SHR</td>
<td></td>
</tr>
<tr>
<td>001600</td>
<td>// DD DISP=SHR,DSN=SCM.DEV.I600.LOAD</td>
<td></td>
</tr>
<tr>
<td>001700</td>
<td>//DAPARM DD DISP=SHR,DSN=IOAP.PARM</td>
<td></td>
</tr>
<tr>
<td>001800</td>
<td>// DD DISP=SHR,DSN=SCM.DEV.I600.IOA.IOAENV</td>
<td></td>
</tr>
<tr>
<td>001900</td>
<td>//DLOG DD DISP=SHR,DSN=IOAP.LOG</td>
<td></td>
</tr>
<tr>
<td>002000</td>
<td>//SYSPRINT DD SYSPUT=*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>*************************** Bottom of Data ***********************************</td>
<td></td>
</tr>
</tbody>
</table>

### JOB4

<table>
<thead>
<tr>
<th>EDIT</th>
<th>CTMP.JCL(M21JOB4) - 01.00</th>
<th>Columns 00001 00072</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>*************************** Top of Data ***********************************</td>
<td></td>
</tr>
<tr>
<td>000001</td>
<td>//M21 JOB APERLMAN,CLASS=A,</td>
<td></td>
</tr>
<tr>
<td>000002</td>
<td>// MSGCLASS=X,NOTIFY=M21</td>
<td></td>
</tr>
<tr>
<td>000003</td>
<td>//S1 EXEC PGM=IEFBR14</td>
<td></td>
</tr>
<tr>
<td></td>
<td>*************************** Bottom of Data ***********************************</td>
<td></td>
</tr>
</tbody>
</table>
Chapter 3

JOB5

EDIT CTMP.JCL(M21JOB5) - 01.09

****** **************************** Top of Data ****************************

000001 //M21 JOB ,'A.PERLMAN',CLASS=A,MSGCLASS=X,NOTIFY=M21
000002 //S1 EXEC PGM=IOATEST,PARM='TERM=C0000'
000003 //STEPLIB DD DSN=IOAP.LOAD,DISP=SHR
000004 // DD DISP=SHR,DSN=SCM.DEV.1600.LOAD
000005 //DAPARM DD DISP=SHR,DSN=IOAP.PARM
000006 // DD DISP=SHR,DSN=SCM.DEV.1600.IOA.IOAENV
000007 //DALOG DD DISP=SHR,DSN=IOAP.LOG
000008 //SYSPRINT DD SYSOUT=*  
000009 //S2 EXEC PGM=IOATEST,PARM='TERM=C0000'
000010 //STEPLIB DD DSN=IOAP.LOAD,DISP=SHR
000011 // DD DISP=SHR,DSN=SCM.DEV.1600.LOAD
000012 //DAPARM DD DISP=SHR,DSN=IOAP.PARM
000013 // DD DISP=SHR,DSN=SCM.DEV.1600.IOA.IOAENV
000014 //DALOG DD DISP=SHR,DSN=IOAP.LOG
000015 //SYSPRINT DD SYSOUT=*  
000016 //S3 EXEC PGM=IOATEST,PARM='TERM=C0000'
000017 //STEPLIB DD DSN=IOAP.LOAD,DISP=SHR
000018 // DD DISP=SHR,DSN=SCM.DEV.1600.LOAD
000019 //DAPARM DD DISP=SHR,DSN=IOAP.PARM
000020 // DD DISP=SHR,DSN=SCM.DEV.1600.IOA.IOAENV
000021 //DALOG DD DISP=SHR,DSN=IOAP.LOG
000022 //SYSPRINT DD SYSOUT=*  
000023 //S4 EXEC PGM=IOATEST,PARM='TERM=C0000'
000024 //STEPLIB DD DSN=IOAP.LOAD,DISP=SHR
000025 // DD DISP=SHR,DSN=SCM.DEV.1600.LOAD
000026 //DAPARM DD DISP=SHR,DSN=IOAP.PARM
000027 // DD DISP=SHR,DSN=SCM.DEV.1600.IOA.IOAENV
000028 //DALOG DD DISP=SHR,DSN=IOAP.LOG
000029 //SYSPRINT DD SYSOUT=*  
000030 //S5 EXEC PGM=IOATEST,PARM='TERM=C0000'
000031 //STEPLIB DD DSN=IOAP.LOAD,DISP=SHR
000032 // DD DISP=SHR,DSN=SCM.DEV.1600.LOAD
000033 //DAPARM DD DISP=SHR,DSN=IOAP.PARM
000034 // DD DISP=SHR,DSN=SCM.DEV.1600.IOA.IOAENV
000035 //DALOG DD DISP=SHR,DSN=IOAP.LOG
000036 //SYSPRINT DD SYSOUT=*  

****** **************************** Bottom of Data ****************************
Chapter 4

JOB1

EDIT       CTMP.JCL(M21JOB1) - 01.05                  Columns 00001 00072
******   *********************************************** Top of Data ****************************
000001 //M21  JOB APERLMAN,CLASS=A,
000002 //   MSGCLASS=X,NOTIFY=M21
000003 //S1   EXEC PGM=IEFBR14
******   *********************************************** Bottom of Data ****************************

JOB2

EDIT       CTMP.JCL(M21JOB2) - 01.00                  Columns 00001 00072
******   *********************************************** Top of Data ****************************
000001 //M21  JOB APERLMAN,CLASS=A,
000002 //   MSGCLASS=X,NOTIFY=M21
000003 //S1   EXEC PGM=IEFBR14
******   *********************************************** Bottom of Data ****************************

JOB3

EDIT       CTMP.JCL(M21JOB3) - 01.03                  Columns 00001 00072
******   *********************************************** Top of Data ****************************
000100 //M21   JOB ,A.PERLMAN",CLASS=A,MSGCLASS=X,NOTIFY=M21
001400 //S1   EXEC PGM=IOAEST,PARM="TERM=C0000"
001500 //STEPLIB DD   DSN=IOAP.LOAD,DISP=SHR
001600 //     DD   DISP=SHR,DSN=SCM.DEV.I600 LOAD
001700 //DAPARM DD   DISP=SHR,DSN=IOAP.PARM
001800 //     DD   DISP=SHR,DSN=SCM.DEV.I600.IOA.IOAENV
001900 //DALOG   DD   DISP=SHR,DSN=IOAP.LOG
002000 //SYSPRINT DD   SYSOUT=* 
******   *********************************************** Bottom of Data ****************************
EDIT       CTMP.JCL(M21JOB4) - 01.00                  Columns 00001 00072
****** ************************************* Top of Data ************************************************
000001 //M21  JOB  APERLMAN,CLASS=A.
000002 //    MSGCLASS=X,NOTIFY=M21
000003 //S1  EXEC  PGM=IEFBR14
****** ************************************* Bottom of Data ************************************************
EDIT       CTMP.JCL(M21JOB5) - 01.09                  Columns 00001 00072

000001 //M21 JOB ,'A.PERLMAN',CLASS=A,MSGCLASS=X,NOTIFY=M21
000002 //S1 EXEC PGM=IOATEST,PARM='TERM=C0000'
000003 //STEPLIB DD DSN=IOAP LOAD,DISP=SHR
000004 // DD DISP=SHR,DSN=SCM.DEV.I600.LOAD
000005 //DAPARM DD DISP=SHR,DSN=IOAP.PARM
000006 // DD DISP=SHR,DSN=SCM.DEV.I600.IOA.IOAENV
000007 //DALOG DD DISP=SHR,DSN=IOAP.LOG
000008 //SYSPRINT DD SYSOUT=*  
000009 //S2 EXEC PGM=IOATEST,PARM='TERM=CO0000'
000010 //STEPLIB DD DSN=IOAP LOAD,DISP=SHR
000011 // DD DISP=SHR,DSN=SCM.DEV.I600.LOAD
000012 //DAPARM DD DISP=SHR,DSN=IOAP.PARM
000013 // DD DISP=SHR,DSN=SCM.DEV.I600.IOA.IOAENV
000014 //DALOG DD DISP=SHR,DSN=IOAP.LOG
000015 //SYSPRINT DD SYSOUT=*  
000016 //S3 EXEC PGM=IOATEST,PARM='TERM=CO0008'
000017 //STEPLIB DD DSN=IOAP LOAD,DISP=SHR
000018 // DD DISP=SHR,DSN=SCM.DEV.I600.LOAD
000019 //DAPARM DD DISP=SHR,DSN=IOAP.PARM
000020 // DD DISP=SHR,DSN=SCM.DEV.I600.IOA.IOAENV
000021 //DALOG DD DISP=SHR,DSN=IOAP.LOG
000022 //SYSPRINT DD SYSOUT=*  
000023 //S4 EXEC PGM=IOATEST,PARM='TERM=CO0000'
000024 //STEPLIB DD DSN=IOAP LOAD,DISP=SHR
000025 // DD DISP=SHR,DSN=SCM.DEV.I600.LOAD
000026 //DAPARM DD DISP=SHR,DSN=IOAP.PARM
000027 // DD DISP=SHR,DSN=SCM.DEV.I600.IOA.IOAENV
000028 //DALOG DD DISP=SHR,DSN=IOAP.LOG
000029 //SYSPRINT DD SYSOUT=*  
000030 //S5 EXEC PGM=IOATEST,PARM='TERM=CO0000'
000031 //STEPLIB DD DSN=IOAP LOAD,DISP=SHR
000032 // DD DISP=SHR,DSN=SCM.DEV.I600.LOAD
000033 //DAPARM DD DISP=SHR,DSN=IOAP.PARM
000034 // DD DISP=SHR,DSN=SCM.DEV.I600.IOA.IOAENV
000035 //DALOG DD DISP=SHR,DSN=IOAP.LOG
000036 //SYSPRINT DD SYSOUT=*  

****** ****************** Top of Data ******************  

****** **************************** Bottom of Data ****************************
Chapter 5

JOB6

EDIT       CTMP.JCL(M21JOB6) - 01.05                  Columns 00001 00072
****** **************************************** Top of Data *******************************
000001 //M21  JOB APERLMAN,CLASS=A,
000002 //    MSGCLASS=X,NOTIFY=M21
000003 //S1  EXEC  PGM=IEFBR14
****** **************************************** Bottom of Data *******************************

JOB7

EDIT       CTMP.JCL(M21JOB7) - 01.05                  Columns 00001 00072
****** **************************************** Top of Data *******************************
000001 //M21  JOB APERLMAN,CLASS=A,
000002 //    MSGCLASS=X,NOTIFY=M21
000003 //S1  EXEC  PGM=IEFBR14
****** **************************************** Bottom of Data *******************************

JOB8

EDIT       CTMP.JCL(M21JOB8) - 01.05                  Columns 00001 00072
****** **************************************** Top of Data *******************************
000001 //M21  JOB APERLMAN,CLASS=A,
000002 //    MSGCLASS=X,NOTIFY=M21
000003 //S1  EXEC  PGM=IEFBR14
****** **************************************** Bottom of Data *******************************

JOB9

EDIT       CTMP.JCL(M21JOB9) - 01.05                  Columns 00001 00072
****** **************************************** Top of Data *******************************
000001 //M21  JOB APERLMAN,CLASS=A,
000002 //    MSGCLASS=X,NOTIFY=M21
000003 //S1  EXEC  PGM=IEFBR14
****** **************************************** Bottom of Data *******************************
### Chapter 6

**JOB3**

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>000100</td>
<td>//M21 JOB ,'A.PERLMAN',CLASS=A,MSGCLASS=X,NOTIFY=M21</td>
</tr>
<tr>
<td>001400</td>
<td>//S1 EXEC PGM=IOATEST,PARM='TERM=C0008'</td>
</tr>
<tr>
<td>001500</td>
<td>//STEPLIB DD DSN=IOAP.LOAD,DISP=SHR</td>
</tr>
<tr>
<td>001600</td>
<td>// DD DISP=SHR,DSN=SCM.DEV.I600_LOAD</td>
</tr>
<tr>
<td>001700</td>
<td>//DAPARM DD DISP=SHR,DSN=IOAP.PARM</td>
</tr>
<tr>
<td>001800</td>
<td>// DD DISP=SHR,DSN=SCM.DEV.I600.IOA.IOAENV</td>
</tr>
<tr>
<td>001900</td>
<td>//DALOG DD DISP=SHR,DSN=IOAP.LOG</td>
</tr>
<tr>
<td>002000</td>
<td>//SYSPRINT DD SYSOUT=*</td>
</tr>
</tbody>
</table>

******** ************************************** Bottom of Data ****************************
No special preparations required.
Chapter 8

JOB10

EDIT       CTMP.JCL(M21JOB10) - 01.00                  Columns 00001 00072
****** **************************** Top of Data ****************************
000001 //M21 JOB APERLMAN,CLASS=A,
000002 // MSGCLASS=X,NOTIFY=M21
000003 //S1 EXEC PGM=IEFBR14
****** **************************** Bottom of Data ****************************

EXT1

EDIT       CTMP.JCL(M21EXT1) - 01.09                  Columns 00001 00072
****** **************************** Top of Data ****************************
000001 //M21 JOB ,A.PERLMAN,CLASS=A,MSGCLASS=X,NOTIFY=M21
000002 //S1 EXEC PGM=IOATEST,PARM='TERM=C0000'
000003 //STEPLIB DD DSN=IOAP.LOAD,DISP=SHR
000004 // DD DISP=SHR,DSN=SCM.DEV.I600.LOAD
000005 //DAPARM DD DISP=SHR,DSN=IOAP.PARM
000006 // DD DISP=SHR,DSN=SCM.DEV.I600.IOA.IOAENV
000007 //DIALOG DD DISP=SHR,DSN=IOAP.LOG
000008 //SYSPRINT DD SYSOUT=* 
000009 //S2 EXEC PGM=IOATEST,PARM='TERM=C0000'
000010 //STEPLIB DD DSN=IOAP.LOAD,DISP=SHR
000011 // DD DISP=SHR,DSN=SCM.DEV.I600.LOAD
000012 //DAPARM DD DISP=SHR,DSN=IOAP.PARM
000013 // DD DISP=SHR,DSN=SCM.DEV.I600.IOA.IOAENV
000014 //DIALOG DD DISP=SHR,DSN=IOAP.LOG
000015 //SYSPRINT DD SYSOUT=* 
****** **************************** Bottom of Data ****************************
## Index

### Symbols

- `% % SET` control statement 141
- `%%CALCDE` AutoEdit function 142
- `%%CALCDE` function 142
- `%%JULIAN` AutoEdit function 143
- `%%JULIAN` function 143
- `% % A` AutoEdit variable 142

### A

- **Active Environment screen** 36
  - Changing display types 178
  - Checking the job log 41
  - Holding and deleting job orders 44

### ADD command

- **AutoEdit**
  - Checking syntax and results 147
  - Complex terms 157
  - Control statements 155
  - Control statements and functions 141
  - Functions 142, 144
    - `%%CALCDE` 142
    - `%%JULIAN` 143
  - Operators 158
  - Using system variables in the JCL 135

### Autemtatic Tape Adjustment facility

- **Basis scheduling parameters** 60
- **BMC Software, contacting** 2

### C

- **Calendars, defining** 211
- **CMEM rule, creating** 227

### Commands

- **ADD** 83
- **DOWN** 74
- **NOTE** 179

### Conditions

- **UP** 74

### Control statements

- **Adding** 83
- **Control statements** 141
- **AutoEdit** 155

### CONTROL-M

- **Active Jobs file** 235
- **Journal file** 235
- **Main components** 24
- **Reporting facility** 237
- **Restoring files** 235

### CONTROL-M/Restart Dataset Cleanup Utility CTRCCLN (R2)

- **237**

### Conventions Used in This Guide

- **Creating**
  - **CMEM Rule** 227
  - **Job JCL** 226
  - **Job scheduling definition** 228
  - **On Spool jobs** 224

### CTMAESIM utility

- **147**

### CTMJOB utility

- **196**

### CTMJOB utility

- **205**

### CTMRSTR utility

- **235**

### CTRCCLN (R2) Dataset Cleanup utility

- **237**

### customer support

- **3**

### Date qualifiers

- **71**

### Date system variables

- **138**

### Defining

- **Calendars** 211
- **Date Control Record for a User Daily job** 201
- **Global variables** 160
- **JCL of a User Daily job** 201
- **Job scheduling definition of a User Daily job** 203
- **JOBS** 75
- **On Spool jobs** 225
- **Restart in the job scheduling definition** 93

### Displaying

- **Execution information from job runs** 185
- **Job dependencies** 183
- **Jobs belonging to a specific SMART Table** 192
- **List of Available Filters** 176
Scheduling plan for the Job 72
Statistics for a job 191
Table names for jobs 182
DO statement, CMEM rule 224

E
Enhanced Daily Checkpointing 202
Exiting the Online facility 50
External members, pointing to 154

F
Facilities, miscellaneous 234
Filtering
   Active Environment display 37
   IOA Conditions/Resources screen 83
Filters 171
   Displaying list of available 176
Forecasting resource usage 236
Functions 144
   % %$CALCDTE 142
   % %$JULIAN 143

G
General parameters 59
Global variables
   Access by another job 162
   Defining 160
   Syntax 161
   Update by another job 164

H
History Jobs File 234

I
IOA Calendar Definition facility 216
IOA Conditions/Resources screen 82
IOACND batch utility 231

J
JCL
   Editing from the Active Environment 99
   Setup operation flow 165
Job List Screen, formats 50
Job Order Execution History screen 185
Job Ordering methods 207
Job scheduling definition facility, returning to 46

Job scheduling definitions
   Defining restart in 93
   Facility 26
   In the SMART Table 115
   Planned logic of 115
   Saving 46
   Supplying variable values 145
JOB4, defining 75
Jobs
   Checking the Sysout for 42
   Confirming an order 39
   Creating JCL 226
   Creating scheduling definition 228
   Dependencies 71
   Displaying
      Belonging to a specific group 192
      Dependencies 183
      Execution information from runs 185
      Scheduling plan 72
      Statistics for 191
      Table names 182
   Forcing 77
   Forcing an OK status for 189
   Freeing a held job 87
   Identifying scheduling problems 80
   On Spool 230
      Bringing 229
      Creating 224
      Defining 225
   Ordering 77
      Through online utility CTMJOB 196
      Through utility CTMJOB 196
   Restarting 100
   Simulating execution 236
   User Daily 200, 203
      Defining Date Control Record 201
      Defining JCL 200
      Defining job scheduling definition 203
   Journal file, CONTROL-M 235

N
New Day processing 199
NOTE command 179
Note, adding to a job 179

O
ODATE 137
On Spool jobs 230
ON Statements 223
ON TABLE-END parameter 112
Online facility
   Exiting 50
   Primary options 26
P
Parameters
   Basis scheduling 60
   General 59
   ON TABLE-END 112
   Post-processing 112
   Runtime scheduling section 64
   Post-processing
      Parameters 112
      Parameters section 67
   product support 3

R
Reporting facility, CONTROL_M 237
Resource usage, forecasting 236
Resource, adding 83
   Responding to external events through CMEM 217
   Runtime Scheduling Parameters 111

S
Saving the Table 126
Schedule RBCs and Other Basic Scheduling Parameters 109
Screens
   Active Environment 36
      Changing display types 178
      Checking the job log 41
      Holding and deleting job orders 44
   IOA Conditions/Resources 82
      Filtering 83
   Job List, formats 50
   Job Order Execution History 185
   Sysout Viewing 187
   Table List, selecting a table from 48
   Why 79, 84
   Zoom 85
SMART Table
   Entity 108
   Scheduling
      Advantages 130
      Creating 107
      Points to remember 126
SMART Table Entity
   Logic of the SMART Table Entity scheduling definition 114
Statements, ON 223
Statistics, accumulating 235
   support, customer 3
   Sysout Viewing screen 187
   System date 137

T
Table List screen
   Selecting a table from 48
   technical support 3
   Toggling Between Display of Commands and Options 36

U
User Daily jobs 200, 203
Utilities
   CTMAESIM 147
   CTMJOB 196
   CTMJOBRQ 205
   CTMRSTR 235
   CTRCCLN (R2) 237
   IOACND 231
Utility CTMAESIM 147

V
Variables
   % %A 142
   Date 138
   Global
      Access by another job 162
      Defining 160
      Syntax 161
      Update by another job 164
   Local and global 158
   Non-Date system 139
   Supplying values through an external member 152
   Supplying values through the job scheduling definition 145

W
Why screen 79, 84
   Adding a missing condition in 81
   Working date 137

Z
Zoom screen 85