Controlling and Monitoring the Workload

Version 2 Release 3
Controlling and Monitoring the Workload

Version 2 Release 3
Before using this information and the product it supports, be sure to read the general information under "Notices" on page ix.

ISO 9001 Certification

This product was developed using an ISO 9001 certified quality system.

Certification has been awarded by the Italian quality system certification group, CSQ (Certification No. CISQ/CSQ 9150.IBM7).

CSQ is a member of the mutually recognized organization of European assessors, ITQS, which assesses and certifies quality systems in the field of information technology enterprises.

Third Edition (December 1999)

This is a major revision of, and obsoletes, SH19-4377-01.

This edition applies to Version 2 Release 3 Modification Level 0 of Tivoli Operations Planning and Control, Program Number 5697-OPC, and to all subsequent releases and modifications until otherwise indicated in new editions or technical newsletters. See the “Summary of Tivoli OPC Version 2 Release 3 Enhancements” on page xv for the changes made to this manual. Technical changes or additions to the text to describe the Tivoli Job Scheduling Console Support are indicated by a vertical line to the left of the change. Make sure you are using the correct edition for the level of the product.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address below.

IBM welcomes your comments. A form for readers' comments appears at the back of this publication. If the form has been removed, address your comments to:

Tivoli OPC Information Development  
Rome Tivoli Laboratory  
IBM Italy S.p.A.  
Via Sciangai, 53  
00144 Rome  
Italy

Fax Number (+39) 06 5966 2077

Internet ID: ROMERCF at VNET.IBM.COM

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

© Copyright International Business Machines Corporation 1991, 1999. All rights reserved.  
Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.
## Contents

**Notices** ................................................................. ix
**Trademarks** .............................................................. ix

**Preface** ................................................................ xi
Who Should Read This Book ........................................... xi
Required Product Knowledge .......................................... xi
How This Book Is Organized ............................................. xi
Tivoli OPC Publications ................................................. xii
  Tivoli OPC Online Books .............................................. xiii
  Online Message Facility .............................................. xiii
Books about Related Products ........................................ xiv

**Summary of Tivoli OPC Version 2 Release 3 Enhancements** .... xv
Job Scheduling Console .................................................... xv
Catalog Management — Data Availability ........................ xv
OS/390 Workload Manager Support ................................ xvi
OS/390 Automatic Restart Manager Support ...................... xvi
Program Interface (PIF) Enhancements ............................ xvi
Enhancements for Non-OS/390 Tracker Agents .................... xvi
Usability Enhancements .................................................. xvi
New and Changed Installation Exits ................................. xvii
New and Changed Initialization Statements ....................... xvii
Version 2 Release 2 Summary ......................................... xviii
Version 2 Release 1 Summary .......................................... xxii

**Chapter 1. What Is Tivoli OPC?** .................................. 1
How Tivoli OPC works ..................................................... 1
  How Does Tivoli OPC Keep Track of Jobs? ....................... 3
  How Does Tivoli OPC Help with Job Preparation? .............. 3
  How Can Tivoli OPC Help You Recover Jobs? .................... 3
  Can Tivoli OPC Help with Online Systems? ...................... 4
  Can You Run Jobs outside Tivoli OPC? .......................... 4
Finding Out about Tivoli OPC ......................................... 5

**Chapter 2. Using the Tivoli OPC Dialogs** ....................... 7
Tivoli OPC ISPF Dialogs ................................................. 7
  Setting Options ....................................................... 8
  Common Dialog Commands and Facilities ...................... 13
Workload Monitor/2 ...................................................... 19
Tivoli OPC Utilities ...................................................... 20

**Chapter 3. Monitoring the Workload** ............................ 21
Using the Ready List Dialog .......................................... 21
  Selecting a Ready List Layout .................................... 22
  Creating Your Own Ready List Layout ......................... 23
  Ready List Layout User Exit ...................................... 24
Using the Ready List .................................................... 27
  Setting the Status of an Operation .............................. 28
  Viewing Operator Instructions .................................. 29
  Preparing Jobs at a Setup Workstation ....................... 30
Figures

1. A Rolling Long-Term Plan ........................................... 2
2. Extending the Current Plan ......................................... 2
3. EQQOPCAP—Tivoli OPC Main Menu ............................... 7
4. EQQXOPTP—Defining Tivoli OPC Parameters and Options ...... 8
5. EQQXDATP—Setting Tivoli OPC Date and Time Format .......... 9
6. EQQXCOLP—Setting Tivoli OPC Color and Highlight Attributes 10
7. EQQXAOIP—Setting Tivoli OPC AD/OI Consistency Check ....... 12
8. EQQXJCLP—Setting JCL Edit Tool Information .................. 13
9. EQQSOPFP—Selecting Operations ................................... 15
10. EQQXSRTL—Sorting a List .......................................... 17
11. ISPOPT3B—PF Key Definitions and Labels ....................... 19
12. EQQXSUBP—Generating JCL for a Batch Job ..................... 20
13. EQQSTOPP—Communicating with Workstations ................. 21
14. EQQRTOPP—Defining Ready List Criteria ....................... 22
15. EQQRLYLL—Ready List Layouts .................................. 23
16. EQQRLYCL—Creating a Ready List Layout ....................... 24
17. EQQRLLRM—Ready List .............................................. 27
18. EQQRJCLE—Editing JCL for an Operation ....................... 30
19. EQQRVAL—List of JCL Preparation Variables to Be Set ....... 31
20. EQQRSRLP—Specifying Ready List Criteria ..................... 32
21. EQQSOPSP—Selecting Application Occurrence and Operation Information .................................................. 36
22. EQQSTOPP—Current Plan and Status Inquiry .................... 37
23. EQQSAOSP—Selecting Application Occurrence Information .... 38
24. EQQSMC1L—Browsing Most Critical Occurrences ............. 39
25. EQQSOPSP—Selecting Application Occurrence and Operation Information .................................................. 40
26. EQQSPG1L—All Dependencies of an Operation (Left Part) .... 40
27. EQQSWSSP—Browsing Summary of Activities at a Workstation 41
28. EQQSGCPP—Browsing General Current Plan Information .... 42
29. EQQMTOPP—Modifying the Current Plan ......................... 45
30. EQQMAADDP—Adding Applications to the Current Plan ....... 47
31. EQQMAADL—Selecting Applications to Add to the CP ........ 48
32. EQQMAOCP—Adding an Application to the Current Plan ...... 48
33. EQQMOPL—Modifying Operations in the Current Plan ....... 52
34. EQQMODP—Modifying an Operation in the Current Plan ....... 52
35. EQQMMADP—Creating a Dependency in the Current Plan .... 53
36. EQQMMDDL—Defining Dependencies in the Current Plan .... 53
37. EQQMAAGL—Adding an Occurrence Group to the CP .......... 54
38. EQQMAOML—Modifying Occurrences Added to the Current Plan 56
39. EQQMOCLL—Modifying Occurrences in the Current Plan ...... 57
40. EQQMOCL—Rerunning an Occurrence in the Current Plan .... 58
41. EQQMOSTL—List Dependency Status Change ................... 59
42. EQQMOPL—Modifying Operations in the Current Plan ....... 62
43. EQQHISTL—Operations History List .............................. 64
44. EQQHIPUP—Specifying Occurrence Arrival ..................... 65
45. EQQMWSPR—Modifying a Workstation in the Current Plan .... 69
46. EQQMWSP—Modifying Workstation Status in the Current Plan 69
47. EQQMWGTR—Modifying Open Time Intervals in the CP ...... 70
48. EQQMEP1L—Handling Operations Ended in Error (Left Part) 73
49. **EQQMERRP**—Specifying Ended In Error List Criteria .......................... 74
50. **EQQELYLL**—Selecting an Error List Layout ........................................... 75
51. **EQQELYCL**—Creating an Error List Layout ........................................... 76
52. **EQQMERTP**—Confirm Restart of an Operation ....................................... 78
53. **EQQMERSL**—Step Restart Selection List ............................................... 79
54. **EQQSJCLB**—Browsing Operation Input/Output Stream ........................... 80
55. How Tivoli OPC validates JCL for step restart ...................................... 82
56. Modified JCL for a Step Restart ................................................................. 83
57. **EQQMCMDL**—Modifying Catalog Management Actions ......................... 85
58. An Example of RECOVER statements ....................................................... 86
59. How Attributes Are Preserved across Intervals ........................................ 92
60. **EQQMSEP**—Specifying Resource Monitor List Criteria .......................... 93
61. **EQQQMLSL**—Special Resource Monitor ............................................... 94
62. **EQQQMIML**—Special Resource Monitor - In Use List ............................ 95
63. **EQQQMWML**—Special Resource Monitor - Waiting Queue .................... 96
64. **EQQQMMOP**—Modifying a Special Resource ........................................... 97
65. **EQQQDIML**—Modifying Intervals for a Special Resource ....................... 98
66. **EQQQDWML**—Modifying Connected Workstations for a Special Resource 99
67. Tivoli OPC Components and Controller–Tracker Links ............................ 103
68. View of Monitors Defined for a Tivoli OPC controller ............................ 104
69. Monitors Defined in Tivoli OPC Instrumentation for Controller–Tracker Relationships ................................................................. 105
70. The Set of Operational Tasks Defined for the Tivoli OPC Controller Component ................................................................. 106

### Tables

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The INCLUDE Statement</td>
<td>xxiii</td>
</tr>
<tr>
<td>2</td>
<td>The INIT Statement</td>
<td>xxiv</td>
</tr>
<tr>
<td>3</td>
<td>Changes to Installation Exits</td>
<td>xxv</td>
</tr>
<tr>
<td>4</td>
<td>Primary Commands for Tivoli OPC Panels</td>
<td>14</td>
</tr>
<tr>
<td>5</td>
<td>Using the MCP DIALOG</td>
<td>44</td>
</tr>
<tr>
<td>6</td>
<td>Fields Available for Display in Ready and Error Lists</td>
<td>121</td>
</tr>
</tbody>
</table>
References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program, or service. Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the user's responsibility.

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

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

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
P.O Box 12195
3039 Cornwallis
Research Triangle Park, NC 27709-2195
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

Trademarks

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

AIX
BookManager
IBM
OPC
OS/390
RACF
Tivoli
TME 10

AIX/6000
GDDM
MVS/ESA
OS/2
OS/400
RISC System/6000
TME

In Denmark, Tivoli is a trademark licensed from Kjøbenhavns Sommer - Tivoli A/S
Microsoft, Windows, Windows NT, and the Windows logo are trademarks or registered trademarks of Microsoft Corporation.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

C-bus is a trademark of Corollary, Inc.

Java and all Java-based trademarks or logos are trademarks of Sun Microsystems, Inc.

PC Direct is a trademark of Ziff Communications Company and is used by IBM Corporation under license.

ActionMedia, LANDesk, MMX, Pentium, and ProShare are trademarks or registered trademarks of Intel Corporation in the United States and other countries.

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

DEC
HP-UX
NT
NuTCRACKER
Solaris
Sun
SunOS
VMS

Digital Equipment Corporation
Hewlett-Packard Corp.
Microsoft Corp.
DataFocus Inc.
Sun Microsystems Inc.
Sun Microsystems Inc.
Sun Microsystems Inc.
Digital Equipment Corporation
Preface

This book shows you how to use the Tivoli OPC ISPF dialogs to run and control planned and ad hoc jobs. Your workload can run on various platforms, but you control it from a central MVS/ESA system that runs the Tivoli OPC controller feature.

Who Should Read This Book

This book is intended for those involved in controlling and monitoring the workload in the production department of a computer installation. It explains how to use Tivoli Operations Planning and Control (Tivoli OPC) for these tasks. Throughout this book, Tivoli Operations Planning and Control is referred to as Tivoli OPC.

Required Product Knowledge

You must know how to use ISPF and TSO.

How This Book Is Organized

The information in this book is organized as follows:

Chapter 1, What Is Tivoli OPC?
Introduces you to Tivoli OPC. This chapter explains what Tivoli OPC is and how it works.

Chapter 2, Using the Tivoli OPC Dialogs
Explains how to use the Tivoli OPC ISPF dialogs. This chapter also introduces Workload Monitor/2 and the Tivoli OPC utilities.

Chapter 3, Monitoring the Workload
Explains how to use the Ready List dialog to process operations that are ready to run and to prepare JCL where this is necessary. It also explains how to use the Query Current Plan dialog to display operations in the plan.

Chapter 4, Updating the Current Plan
Explains how to use the Modify Current Plan dialog to change, add, and delete operations and occurrences. You can also change dependencies and rerun jobs in the current plan and jobs in the history database.

Chapter 5, Handling Operations that End in Error
Explains what to do when an operation fails.

Chapter 6, Monitoring Special Resources
Explains how to use the Special Resource Monitor dialog to change resource allocation and status.

Chapter 7, Managing Tivoli OPC with Tivoli Global Enterprise Manager
Explains how to use the Tivoli Global Enterprise Manager to monitor and manage your Tivoli OPC installation.

Appendix A, MVS Commands Supported by Tivoli OPC
Describes the supported MVS/ESA commands.
Appendix B, Status, Error, and Reason Codes
Explains what the status, error, and reason codes mean.

Appendix C, Fields Displayed in Ready and Error Lists
Describes the fields you can display in the ready list and in the error-handling list.

Tivoli OPC Publications

This book is part of an extensive Tivoli OPC library. These books can help you use Tivoli OPC more effectively:

<table>
<thead>
<tr>
<th>Task</th>
<th>Publication</th>
<th>Order number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Evaluating Tivoli OPC</td>
<td>General Fact Sheet</td>
<td>GH19-4370</td>
</tr>
<tr>
<td>Evaluating Tracker Agents</td>
<td>Tracker Agent Features Fact Sheet</td>
<td>GH19-4371</td>
</tr>
<tr>
<td>Planning Tivoli OPC</td>
<td>Licensed Program Specifications</td>
<td>GH19-4373</td>
</tr>
<tr>
<td>Understanding Tivoli OPC</td>
<td>General Information</td>
<td>GH19-4372</td>
</tr>
<tr>
<td>Learning Tivoli OPC concepts and terminology</td>
<td>Getting Started with Tivoli OPC</td>
<td>SH19-4481</td>
</tr>
<tr>
<td>Using the Java GUI</td>
<td>Tivoli Job Scheduling Console Guide for OPC Users</td>
<td>GC32-0402</td>
</tr>
<tr>
<td>Using the Java GUI</td>
<td>Tivoli Job Scheduling Console Release Notes</td>
<td>GI10-9233</td>
</tr>
<tr>
<td>Interpreting messages and codes</td>
<td>Messages and Codes</td>
<td>SH19-4480</td>
</tr>
<tr>
<td>Installing Tivoli OPC</td>
<td>Installation Guide</td>
<td>SH19-4379</td>
</tr>
<tr>
<td>Customizing and tuning Tivoli OPC</td>
<td>Customization and Tuning</td>
<td>SH19-4380</td>
</tr>
<tr>
<td>Planning and scheduling the workload</td>
<td>Planning and Scheduling the Workload</td>
<td>SH19-4376</td>
</tr>
<tr>
<td>Controlling and monitoring the current plan</td>
<td>Controlling and Monitoring the Workload</td>
<td>SH19-4377</td>
</tr>
<tr>
<td>Using Workload Monitor/2</td>
<td>Workload Monitor/2 User’s Guide</td>
<td>SH19-4482</td>
</tr>
<tr>
<td>Writing application programs</td>
<td>Programming Interfaces</td>
<td>SH19-4378</td>
</tr>
<tr>
<td>Quick reference</td>
<td>Quick Reference</td>
<td>GH19-4374</td>
</tr>
<tr>
<td>Diagnosing failures</td>
<td>Diagnosis Guide and Reference</td>
<td>LY19-6405</td>
</tr>
<tr>
<td>Controlling the AIX, UNIX**, VMS, OS/390 Open Edition workload</td>
<td>Tracker Agents for AIX, UNIX, VMS, OS/390 Open Edition Installation and Operation</td>
<td>SH19-4484</td>
</tr>
<tr>
<td>Controlling the OS/2 and NT workload</td>
<td>Tracker Agents for OS/2 and Windows NT Installation and Operation</td>
<td>SH19-4483</td>
</tr>
<tr>
<td>Controlling the OS/400 workload</td>
<td>Tracker Agent for OS/400 Installation and Operation</td>
<td>SH19-4485</td>
</tr>
</tbody>
</table>

A Master Index, SH19-4375, is published for the Tivoli OPC library.

Maximizing Your OPC Throughput, SG24-2130, contains useful information for tuning the OPC installation.
Tivoli OPC Online Books

All the books in the Tivoli OPC library, except the licensed publications, are available in displayable softcopy form on CD-ROM in the following Softcopy Collection Kit:

- OS/390, SK2T-6700

You can read the softcopy books on CD-ROMs using these IBM licensed programs:

- BookManager READ/2 (program number 5601-454)
- BookManager READ/DOS (program number 5601-453)
- BookManager READ/6000 (program number 5765-086)

All the BookManager programs need a personal computer equipped with a CD-ROM disk drive (capable of reading disks formatted in the ISO 9660 standard) and a matching adapter and cable. For additional hardware and software information, refer to the documentation for the specific BookManager product you are using.

Updates to books between releases are provided in softcopy only.

Online Message Facility

The Online Message Facility (OMF) is an OS/2 program that provides online access to information from BookManager softcopy books. It helps you diagnose problems without interrupting your work. You can retrieve the description of a message by clicking on a message number in a Communications Manager emulator window. Additional information about OMF is available on the Messages and Codes CD-ROM.
# Books about Related Products

These publications are referred to in the text or contain information that you might need:

<table>
<thead>
<tr>
<th>Short Title</th>
<th>Publication</th>
<th>Order Number</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Graphical Data Display Manager Version 3 User's Guide</td>
<td>SC33-0875</td>
</tr>
<tr>
<td>ISPF Reference Summary</td>
<td>ISPF/PDF Version 3 MVS Reference Summary</td>
<td>SC34-4214</td>
</tr>
<tr>
<td></td>
<td>ISPF Version 4 Reference Summary</td>
<td>SC34-4445</td>
</tr>
<tr>
<td>ISPF Planning and Customization</td>
<td>ISPF and ISPF/PDF Planning and Customization</td>
<td>SC34-4257</td>
</tr>
<tr>
<td></td>
<td>ISPF Version 4 Planning and Customization</td>
<td>SC34-4433</td>
</tr>
<tr>
<td>ISPF Guide and Reference</td>
<td>ISPF/PDF Guide and Reference</td>
<td>SC34-4258</td>
</tr>
<tr>
<td></td>
<td>ISPF Version 4 User's Guide</td>
<td>SC34-4484</td>
</tr>
<tr>
<td>ISPF Examples</td>
<td>ISPF Dialog Management Services Examples</td>
<td>SC34-4215</td>
</tr>
<tr>
<td></td>
<td>ISPF Version 4 Examples</td>
<td>SC34-4415</td>
</tr>
<tr>
<td>JCL Reference</td>
<td>MVS JCL Reference</td>
<td>GC28-1654</td>
</tr>
<tr>
<td></td>
<td>MVS SP5 JCL Reference</td>
<td>GC28-1479</td>
</tr>
<tr>
<td></td>
<td>MVS SP5 JCL User's Guide</td>
<td>GC28-1473</td>
</tr>
<tr>
<td>System Commands</td>
<td>MVS/ESA System commands</td>
<td>GC23-1626</td>
</tr>
<tr>
<td></td>
<td>MVS/ESA SP5 System commands</td>
<td>GC23-1442</td>
</tr>
<tr>
<td>JES2 Commands</td>
<td>MVS/ESA JES2 commands</td>
<td>GC23-0084</td>
</tr>
<tr>
<td></td>
<td>MVS/ESA SP5 JES2 commands</td>
<td>GC23-1443</td>
</tr>
<tr>
<td>JES3 Commands</td>
<td>MVS/ESA JES3 commands</td>
<td>GC23-0090</td>
</tr>
<tr>
<td></td>
<td>MVS/ESA SP5 JES3 commands</td>
<td>SC28-1444</td>
</tr>
<tr>
<td>NetView General Information</td>
<td>NetView General Information and Planning</td>
<td>GC31-7098</td>
</tr>
<tr>
<td>RACF Command Reference</td>
<td>Resource Access Control Facility Command Language Reference</td>
<td>SC28-0733</td>
</tr>
<tr>
<td>Maestro Supplemental Documentation Set</td>
<td>Unison Maestro Supplemental Documentation Set</td>
<td>SK3T-3566</td>
</tr>
</tbody>
</table>
Summary of Tivoli OPC Version 2 Release 3 Enhancements

Job Scheduling Console

The new Tivoli Job Scheduling Console (JSC) is a Java-based, client/server application. The key advantages of the JSC are the ability to perform administration and operation tasks in a graphical manner and the ability to access multiple OPC controllers from a single console.

The JSC can:

- Display lists of objects already defined to OPC, from the database and from the current plan, by using flexible filtering criteria
- Work with application descriptions including jobs and their dependencies, time restrictions (input arrival time, deadline, duration), and run cycles
- Work with special resource and workstation definitions
- Modify occurrences, workstation status, and special resource information from the current plan.

The JSC retains the OPC security model. Each data access request is validated by the controller as it is done currently for ISPF users.

The JSC is a real-time interface with OPC and can be used concurrently with the ISPF interface. It is available for various UNIX platforms, Windows NT, and Windows 98. The OPC Connector, which is a backend component supporting the JSC, is available for various UNIX platforms and Windows NT.

Catalog Management — Data Availability

The new Catalog Management – Data Availability feature improves OPC performance for job restart and job log retrieval functions. Job runtime information, for example, the sysout datasets, is maintained locally on the tracked system. The controller retrieves this information only when needed for catalog management actions, eliminating the network and processing overhead associated with the transmission of superfluous data. The runtime information at the tracked system is managed by a new component, the OPC Data Store. Using the OPC Data Store, OPC Tracker processes are bypassed and are dedicated to the time-critical job submission and tracking tasks. A new feature is provided to selectively determine how long job runtime information is kept in the Data Store. This new feature is especially useful when a joblog archiving product is used concurrently with OPC.

OS/390 Workload Manager Support

OS/390 Workload Manager, when used in goal mode, provides a new, policy-based management of deadlines for critical jobs. Some CPU-type operations can now be marked as critical in OPC. When such a critical operation is late, according to the specified policy, OPC interfaces with Workload Manager to move the associated job to a higher performance service class. Thus the job receives appropriate additional system resource to reduce or eliminate the delay. Several policies are available to
decide when a job is late, considering characteristics such as duration, deadline time, and latest start time.

OS/390 Automatic Restart Manager Support

OS/390 Automatic Restart Manager increases the availability of OPC components. In the event of program failure, OPC components, for example, the Controller, the OS/390 Tracker and the Server can now be restarted automatically by the Automatic Restart Manager.

Program Interface (PIF) Enhancements

The Program Interface (PIF) has been extended to increase the flexibility of OPC, allowing users to have extended access to OPC data from other application programs. Tivoli OPC Version 2 Release 3 significantly enhances the ability to access current plan data from the PIF by providing:

- Full support for special resources data
- Read access to special resource usage information for operations
- The ability to modify the workstation open intervals
- The ability to modify the successor information for an operation

New resource codes have been added to the Program Interface (PIF):

- CPOPSRU: Current plan operation segment with information for the operation in relation to a special resource
- CPSUC: Current plan successor segment
- CSR: Current plan special resources
- CSRCOM: Current plan special resource common segment
- IVL: Current plan workstation interval segment

Enhancements for Non-OS/390 Tracker Agents

The OPC Tracker Agents for non-OS/390 platforms have been enhanced:

- A new version of the OPC Tracker Agent for OpenVMS is available. This new version runs in the native OpenVMS environment, thus removing the requirement to install the POSIX shell.
- The security features for the UNIX OPC Tracker Agents have been enhanced. Stricter file permissions are now used for temporary work files.
- The installation process of the OPC Tracker Agent for OS/390 UNIX System Services has been simplified.

Usability Enhancements

New features increase the overall usability of the product, thus increasing user productivity:

- OPC can perform variable substitution within inline procedures, thus increasing the flexibility of the job setup feature. It is possible to customize OPC so that
jobs are submitted also when variables are not defined in the OPC variable tables. This means that, when variables are substituted outside OPC, duplicate variable definitions are avoided.

- During Catalog Management actions, OPC can delete datasets with an expiration date.
- A new Modify command (JSUACT) has been provided to start or stop the job submission function. This feature enables automation products, for example, Tivoli NetView to have control over the OPC job submission activity.
- The Mass Update utility has been enhanced with a new sample job. This downloads all the applications belonging to a group in a sequential file for use as input to the Batch Loader utility, thus easing the management of group applications from the batch administration.
- The sample library now contains the DSECT sections for the Program Interface (PIF) data areas. This eases the process of writing PIF applications and the migration of existing PIF applications to new OPC releases.

New and Changed Installation Exits

User exit EQQUX001 has three new parameters:

- **NEWREC** Number of JCL lines in new JCLAREA
- **NEWJCL** New JCLAREA
- **USDREC** Number of JCL lines used in new JCLAREA

User exit EQQUX007 has the new extended status (PEXSTAT) as part of its parameters set.

The Job Submission exit (installation exit 1) now allows changes to the size of JCL being processed. This enhancement gives users more flexibility to customize their operating environment.

The Operation Status Change exit (installation exit 7) has been enhanced to receive extended status information. This means that full status information is available within this exit to allow more detailed processing.

The samples set has two new samples: EQQCMX01 and EQQCMX05.

New and Changed Initialization Statements

Two initialization statements have been added to enhance the JCL variable substitution:

- **VARFAIL** If VARFAIL is specified, JCL variable substitution error is bypassed for the specified types and variables are left unresolved in the submitted JCL.
- **VARPROC** Specifies whether or not the variables must be resolved also in the inline procedures.

Three initialization statements have been added to handle the OPC Data Store options:
FLOPTS  Defines the options for the FL (Fetch Job Log) task. A Controller uses this statement when OPCOPTS DSTTASK (YES) is specified.

DSTOPTS  Specifies options for the OPC Data Store.

DSTUTIL  Specifies options for the Data Store batch utilities and the clean up subtask.

Parameters have been added to, or changed in, the JOBOPTS statement so as to handle the new Data Store options:

JOBLOGRETRIEVAL
  A new value DELAYEDST has been added to this keyword for specifying that the job log is to be retrieved by means of the OPC Data Store.

DSTCLASS  A new parameter to define the reserved held class that is to be used by the OPC Data Store associated with this tracker.

DSTFILTER  A new parameter to specify if the job-completion checker (JCC) requeues to the reserved Data Store classes only the sysouts belonging to these classes.

Parameters have been added to, or changed in, the OPCOPTS statement so as to handle the new catalog management functions:

DSTTASK  Specifies whether or not the OPC Data Store is to be used.

JCCTASK  A new DST value has been added to specify if the JCC function is not needed, but the Data Store is used.

A parameter has been added to the OPCOPTS and the SERVOPTS statements:

ARM  Activates automatic restart (via the Automatic Restart Manager) of a failed OPC component.

A parameter has been added to the OPCOPTS statement for the Workload Manager (WLM) support:

WLM  Defines the WLM options, that is, the generic profile for a critical job. The profile contains the WLM service class and policy.

---

**Version 2 Release 2 Summary**

**Instrumentation for Tivoli Global Enterprise Manager**

Tivoli Global Enterprise Manager (GEM) is the industry's first solution for unifying the management of cross-platform business applications that run businesses and make them competitive. Tivoli GEM helps you to manage strategic applications from a unique business systems perspective, focusing your IT resources on keeping these systems working properly and productively. Tivoli OPC has been enhanced to support the Job Scheduling Business System of the Tivoli GEM Systems Management Business System. From the Tivoli GEM console, which provides a single point of management, a Tivoli OPC user has complete control of all the Tivoli OPC components, regardless of the platform on which they run. In more detail, the Tivoli OPC instrumentation for Tivoli GEM enables you to do the following:
• Show all the Tivoli OPC components, including controllers, stand-by controllers, OS/390 trackers, AS/400 tracker agents, TCP/IP connected tracker agents.

• Show the different links between the above components. This provides, at a glance, a check on the health of the connections. For example, an OS/390 tracker might be running but might have no connection to the controller.

• For each component, manage a set of status parameters (monitors) specific to that component. These monitors might report the status of some vital OPC controller data sets such as database, current plan, and long-term plan)

• Manage this set of monitors graphically. You can:
  – Ask for value of the monitor
  – Be notified when the value of the monitor changes
  – Associate a severity (such as normal, warning, severe, or critical) with each monitor value

• Start or stop Tivoli OPC trackers without logging them on.

• Know at a glance, in a sysplex environment, which is the active controller and which the stand-by.

• Execute commands on Tivoli OPC components, from a single point of control, regardless of the platform and operating system used for that component.

SAP R/3 support
Tivoli OPC has been enhanced to exploit the Extended Agent technology of the Tivoli Workload Scheduler product. This technology enables Tivoli OPC to interface with a number of third party applications that can perform scheduling. By using this technology, you can now start and track a SAP R/3 job from Tivoli OPC. You can also retrieve and display the job log at the Tivoli OPC controller. This function requires the Tivoli OPC Tracker Agent for one of the following platforms:

• AIX
• Digital UNIX
• Sun Solaris
• Windows NT
• HP–UX

TCP/IP communication improvements
The TCP/IP communication component that enables the controller to communicate with the TCP/IP connected tracker agents has been restructured to use the standard TCP/IP C–Socket interface. This change enables Tivoli OPC for the latest OS/390 releases and provides for the use of the standard TCP/IP features, such as the KEEPALIVE option.

Catalog management enhancements
The logic that Tivoli OPC uses when determining which catalog management actions to perform has been extended to manage the following situations:

• Some steps in a job are not executed, but are flushed. The datasets referred to in those steps are ignored by the catalog management function.
• A dataset referred to with disposition NEW in one step is also referred to in other steps. Logic to determine the action to perform in these cases has been added to the Catalog Management function.

**Dataset Delete function (EQQDELDL) improvements**
The Dataset Delete function has been enhanced to determine the correct action when a dataset referred to with disposition NEW in one step is also referred to in other steps. Logic to determine the correct action to perform in these cases has been added to the Dataset Delete function. The Dataset Delete function has also been improved to do the following:

- Delete datasets for which an expiration date was specified.
- Issue diagnostic information when the IDCAMS DELETE command or the DFHSM ARCHDEL command fails to delete a dataset.

**Current plan occurrence limit removal**
The maximum number of occurrences in the current plan has been increased from 32767 to 9999999. This enhancement enables you to manage the current plan more flexibly when you have large workloads.

**Operations in AD limit removal**
You can now define up to 255 operations in each Application Description. This enhancement provides for more flexibility in the definition of the workload.

**AD and OI consistency check**
The consistency between the Application Description and the Operator Instruction OPC databases is now enforced by OPC. For instance, whenever an operation is deleted the associated operator instructions is also deleted.
Some usability enhancements have also been implemented in the Application Description dialogs when defining operator instructions. For instance, you can now also access temporary operator instructions.

**JCL editing from Application Description dialogs**
You can now customize the Tivoli OPC dialogs so that a library management application used in the customer's environment to manage the production jobs can be invoked from the Application Description OPC dialogs, thus increasing user productivity. New row commands have been added to invoke such an application from the Operation List panel while working with an Application Description.

**OPC Control Language tool**
The OPC Control Language (OCL) tool enables you to access and manipulate Tivoli OPC data by using a REXX-like language. Several macro-functions are made available that perform, in a single action, what would require several invocations of the OPC Program Interface functions. The OCL tool acts as an extension to the REXX language processor. Therefore, normal REXX statements can be coded together with OCL statements. This tool runs in a batch TSO session.

**Tracker agents**
New Tracker Agents are provided to control the workload on:

- Digital UNIX
- OS/390 Open Edition
SmartBatch coexistence
Tivoli OPC has been extensively tested to make sure that all the features continue to work correctly when the production workload is under SmartBatch control.

Other enhancements to functions
- EQQZSUBX 16 MB limit removal: because it is no longer necessary to move the JCL buffer below the 16 MB line before submitting it to JES2 or JES3, this processing has been removed from Tivoli OPC.
- To improve the robustness of Tivoli OPC, the STIMERM macro is now invoked, wherever the STIMER macro was previously invoked.
- Tivoli OPC Job-Submit user exit (EQQUX001) has been improved by adding two new parameters: WorkstationID and ErrorCode. When ErrorCode is set, Tivoli OPC will not submit the job.
- Tivoli OPC Operation-Status-Change user exit (EQQUX007) has been improved by adding the procsstep name to the JOBAREA parameter. This enhancement provides for fully automated problem management.
- Debugging aids for performance problems: new statistics are now produced by Tivoli OPC to trace all the activities performed during the job submission process. These statistics are especially useful when you tune your systems to maximize job throughput in Tivoli OPC. You can dynamically activate and deactivate these statistics by means of new MODIFY commands.

New and changed installation exits
User exit EQQUX001 has two new parameters:

- RETCO The error code
- WSNAME The workstation name of submission process

User Exit EQQUX007 has a new field in the JobArea called procedure step name.

Changes to commands
The following modify commands have been added:

- CPQSTA Activates or deactivates CP lock statistic messaging
- EVELEM Sets a new value for the EVELEM keyword of the JTOPTS statement
- EVESTA Activates or deactivates EVENT statistic messaging
- GENSTA Activates or deactivates GS task statistic messaging
- HB Issues a heartbeat message for an OPC controller or for all trackers connected to that controller
- JCLDBG Activates or deactivates the JCL debugging trace
- QUELEN Sets a new value for the QUEUELEN keyword of the JTOPTS statement
- STATIM Sets a new value for the STATIM keyword of the JTOPTS statement
- STATUS Returns status information about the OPC controller and the tracker agents connected to it.
- WSASTA Activates or deactivates WSA task statistic messaging

New and changed initialization statements
The following values have been added to the STATMSG keyword of the JTOPTS statement:
EVELIM    Makes customizable the event number criterion for statistic messaging.

STATIM    Uses an interval time criterion to issue statistics messaging.

WSATASK   Activates new statistics for WSA task.

The following new values have been added to the SUBFAILACTION keyword of the JTOPTS statement:

**XC, XE and XR**
To specify how OPC must handle values returned by the Job Submission Exit (EQQUX001) for the RETCO parameter.

A new keyword has been added to the BATCHOPT statement:

**MAXOCCNUM** Set the maximum number of occurrences in the current plan for the daily planning function.

A new keyword has been added to the JTOPTS statement:

**MAXOCCNUM** Set the maximum number of occurrences in the Current Plan for the dialog, ETT, Automatic Recovery and PIF functions.

**Changes to programming interfaces**
The OPC Programming Interface (PIF) has been extended as follows:

- A new subsegment has been added to the Workstation record called the Workstation Access Method Information (WSAM).
- A new keyword, ADOICHK, has been added to the OPTIONS request to activate the consistency check between Application Description and Operator Instruction records.

**Version 2 Release 1 Summary**

Tivoli OPC Version 2 Release 1 became generally available in March 1997. Major enhancements compared to OPC/ESA Release 3.1 are described in the following sections.

**Tracker agents**
New Tracker Agents are provided to control the workload on:

- Digital OpenVMS
- Pyramid MIPS ABI

**Shared parm library in Sysplex environment**
MVS controllers and trackers can share common controller and tracker initialization statements and started task JCLs, making it easier to install many OPC subsystems inside an MVS/ESA sysplex environment.

**Controller configuration in Sysplex environment**
Tivoli OPC support of MVS/ESA sysplex (base and parallel) has been extended to enable any one of many cloned controllers on different MVS images to switch from standby to active status. An OPC controller is started on each member of the XCF group. The first potential controller that becomes active is the active controller and the others are standby controllers. If the active controller fails, a standby controller running in another MVS/ESA image of the sysplex environment takes over automatically as the active controller.
Single system image
This enhancement allows OPC TSO dialog users and PIF users to be on a different MVS/ESA image from the OPC controller. Dialog users and PIF applications can also be on MVS systems outside the sysplex where the controller runs. Remote communication is based on APPC.

Extended dialog filter
The dialog filter has been extended to allow more specific search arguments and to define the interpretation of wildcard characters.

Reparseing of NOERROR
New operator commands allow the operator to dynamically update the NOERROR table using the NOERROR initialization statements defined by the OPC PARMLIB member, and to read the statements from a member of the EQQPARM DD concatenated libraries. In addition a new initialization statement allows the inclusion of NOERROR statements from members of the EQQPARM DD concatenated libraries.

PIF extension
Program Interface has been greatly extended to support almost all OPC database record types.

Job tracking log
This enhancement provides to user exit 11 job tracking records on which an effective disaster recovery procedure can be based. The customer through exit 11 receives job tracking records, and can send this data to a remote controller that, in case of failure of the active controller, will take over as controller.

GMT clock support improvement
The GMTOFFSET keyword in the OPCOPTS statement lets the user define an offset between the GMT time set in the MVS system and the actual GMT time. The OPC controller uses the GMT clock to validate an OPC Tracker Agent trying to connect; this improvement addresses the need of some users to have the MVS GMT clock independent of the actual GMT time, while keeping the ability to use Tracker Agents.

Batch command interface tool
A batch command interface tool is supplied to perform most of the actions supported by the PIF interface by means of a batch command interface.

New and changed initialization statements
Initialization statements have been added and changed in Tivoli OPC Version 2. The following sections summarize the differences.

The INCLUDE statement
Added in Tivoli OPC Version 2, the INCLUDE statement lets you reduce the size of the parameter library member that contains the OPCOPTS and JTOPTS statements and reduce the associated maintenance activities.

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Short description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NOERROR</td>
<td>Specifies to read NOERROR information from other members of the EQQPARM library.</td>
</tr>
</tbody>
</table>
The INIT statement

Added in OPC/ESA Release 3.1, the INIT statement lets you define run-time options for processing requests from a PIF application. These settings override the values set by the INTOPTS statement in EQQPARM. The statement is defined in a second parameter file that is identified by the EQQYPARM DD statement in the JCL procedure of the PIF application. In Tivoli OPC Version 2 the LUNAME keyword has been added.

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Short description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CWBASE</td>
<td>Specifies the origin for the century window used by the PIF application</td>
</tr>
<tr>
<td>HIGHDATE</td>
<td>Specifies the high date presented to the PIF application in valid-to fields</td>
</tr>
<tr>
<td>LUNAME</td>
<td>Specifies a server or controller LU name for the PIF application</td>
</tr>
<tr>
<td>SUBSYS</td>
<td>Identifies the Tivoli OPC subsystem controller</td>
</tr>
<tr>
<td>TRACE</td>
<td>Specifies the level of trace information to write to the diagnostic file.</td>
</tr>
</tbody>
</table>

Changes to commands

These modify commands have been added:

- **NEWNOERR** Requests that the NOERROR statements be reprocessed.
- **NOERRMEM** (member) Requests that the NOERROR information be read from the specified member.

The MODIFY command has been extended to accept stop and start of the server started tasks:

- `F ssname, P=SERV`
- `S ssname, P=SERV`

Changes to programming interfaces

The Programming Interface is extended as follows:

- UPDATE is supported for calendars, periods, workstations, and all workstations closed.
- BROWSE and UPDATE are supported for ETT and special resources.

The LIST request has been extended to support a new keyword, MATCHTYP, to specify whether generic search arguments (* and %) are to be treated as normal characters.

A new keyword, ADVERS, has been added to the OPTIONS request, to activate the support of AD versioning.

New and changed installation exits

Table 3 on page xxv summarizes the changes to installation exits in Tivoli OPC Version 2.
Table 3. Changes to Installation Exits

<table>
<thead>
<tr>
<th>Exit name</th>
<th>Short description of change</th>
</tr>
</thead>
<tbody>
<tr>
<td>EQQUX001</td>
<td>Tivoli OPC Version 2 now also supports the addressing modes RMODE(24) and AMODE(31).</td>
</tr>
<tr>
<td>EQQUX011</td>
<td>Sample job tracking log write exit.</td>
</tr>
</tbody>
</table>

Messages

Messages have been changed, deleted, and added in Tivoli OPC Version 2. Refer to *Tivoli OPC Messages and Codes* for the complete message text and descriptions. Note that in Version 2 the message text and explanations refer to the product as OPC/ESA.
Chapter 1. What Is Tivoli OPC?

This chapter gives a description of Tivoli OPC and introduces its terminology. Read this chapter if you are new to Tivoli OPC.

One system in your complex is designated as the controlling system: it runs the Tivoli OPC controller. From this system, you can automatically plan, control, and monitor your entire production workload. All the systems in your complex must run the Tivoli OPC tracker. The tracker acts as the communication link between the system that it runs on and the Tivoli OPC controller.

How Tivoli OPC works

If you do not have an automated planning system such as Tivoli OPC, you submit jobs on request, or according to run sheets. If they fail, you correct the error and resubmit them, perhaps after running recovery jobs. The jobs depend on many conditions, such as:

- Hardware, such as tape drives.
- Online systems, such as CICS. Often a system must be shut down before a batch job can run.
- Operating system resources, such as JES initiators, that you need to run a job of the correct class.
- Other jobs. You cannot run the job to print pay slips until the payroll tax deduction program has completed successfully.
- Job parameters that you must change each run.
- The time of day.
- The day of the week or year. Some jobs must be run on Friday. There are sometimes complex rules that specify what you do when the normal day is a holiday.

When you run jobs and started tasks under Tivoli OPC, these dependencies are defined in Tivoli OPC databases by someone in your enterprise who is the Tivoli OPC administrator. The administrator defines your workload to Tivoli OPC like this:

1. Create one or more calendars with the holidays that you take.
2. Define applications, which are sets of jobs and other steps such as job preparation and print processing. Applications can themselves be grouped into application groups.
3. Create a long-term plan (LTP). This lists all occurrences of the applications that will run in a long period of typically a few months and the dependencies among them.
4. Create a current plan (CP). This is a detailed plan, typically for one day, that lists the applications that will run and the operations in each application. An operation can be a computer job, but it can also be any other operation that you want to control with Tivoli OPC, such as printing and job preparation.

The administrator’s tasks are described in Planning and Scheduling the Workload.
You work mostly with the current plan. It is created by a batch job, usually at a fixed time each day. The CP is really a Tivoli OPC dataset, which is continually updated by events on the processors, but you can have a printed plan, which is a report that is produced when the CP is created.

Strictly speaking, the CP is created only once and the daily planning process is called extending the plan. Extension is a better term than creation, because the old CP is also part of the new CP. Look at the long-term plan in Figure 1.

Figure 1. A Rolling Long-Term Plan

The LTP starts officially at the origin start date (1 July 1995 in the example), but completed application occurrences are removed, so the LTP is about the same length (2 months of occurrences) after each monthly extension.

The CP should always stretch for some hours or days into the future. Extend the CP at regular intervals, using the EXTEND option of the DAILY PLANNING menu. You can extend the CP to a fixed date and time, or you can extend it by a period of hours and minutes.

Figure 2 shows a 48-hour CP. The initial CP lasts 48 hours: every morning, the CP is extended by a further 24 hours.

Figure 2. Extending the Current Plan
Input is taken from both the LTP and from the present CP. The planning performed on Tuesday for Tuesday’s work considers the actual situation (both completed and outstanding work) as reflected in the CP.

The extended CP always keeps uncompleted application occurrences, but the CP will usually be about 48 hours long, extended by 24 hours every 24 hours.

How Does Tivoli OPC Keep Track of Jobs?

Tivoli OPC submits jobs to the operating system. Tivoli OPC-supplied code in the operating system (in the case of an MVS tracker, this code is in JES and SMF exits) tells Tivoli OPC when the jobs have finished and also *how* they have finished (Tivoli OPC looks at abend and return codes and can also look for certain error messages in the job log that do not always give rise to nonzero return codes).

Tivoli OPC calculates the latest time that a job can be started before it is in danger of missing its deadline. Tivoli OPC continually adjusts its estimate of how long a job takes, taking the actual run times into account.

When a job or other operation is running late, Tivoli OPC can issue alerts. An alert can be a message to the operator console, but it can also trigger other events.

How Does Tivoli OPC Help with Job Preparation?

Tivoli OPC helps in two ways:

- Run-time variables can often be automatically substituted, even if they vary from run to run. Many such variables are related to the date, and Tivoli OPC can build a string in the format required by a program.
- When jobs need manual job preparation, the administrator specifies a job setup operation as a predecessor for the job operation. Tivoli OPC does not submit the job until you have finished preparing the job statements.

Do not edit and submit the job outside Tivoli OPC. Instead, use the Tivoli OPC Ready List dialog to edit the job: Tivoli OPC submits the job for you when you have prepared the job (and when other dependencies have been met).

How Can Tivoli OPC Help You Recover Jobs?

Tivoli OPC supports automatic recovery by having its own job statements that take effect when a job fails: these job statements look like comments to MVS and JES.

For jobs tracked on an MVS system, Tivoli OPC also notices when the catalog has been updated by a job and is able to undo the catalog updates (step by step, if required) to the point before the job ran, for all datasets allocated with JCL DD statements. This facility is called Catalog Management. For example, when a job creates a dataset, a rerun often fails because the dataset already exists. With Catalog Management active for the job, Tivoli OPC uncatalogs and deletes the dataset before resubmitting the job.
Can Tivoli OPC Help with Online Systems?

An online system, such as a CICS system, is a job or started task, so it can be started like any other operation defined to Tivoli OPC.

Many batch applications cannot start until the online system shuts down. One advantage of defining an online system to Tivoli OPC is that a batch application can be made dependent on the online system and Tivoli OPC can start the batch application automatically (assuming other dependencies are met) when the online system ends.

The Tivoli OPC administrator can specify that Tivoli OPC issues a message when an operation deadline passes before the operation is complete. This is called a **DEADLINE WTO**. If a DEADLINE WTO is specified for the operation representing an online system on an MVS system, Tivoli OPC sends a write-to-operator (WTO) message to the operator console when the online system must shut down. This message can trigger events in NetView, such as the broadcasting of a “Closing in 5 minutes” message to online users and the initiation of the shutdown transaction 5 minutes later.

Can You Run Jobs outside Tivoli OPC?

Jobs fall into four categories:

1. Jobs that are in the current plan and are submitted by Tivoli OPC.
   These are scheduled jobs, or jobs that you have added to the current plan using the MCP dialog.

2. Jobs that are in the current plan, but are not submitted by Tivoli OPC.
   These are usually jobs that are generated and submitted by some other subsystem, such as CICS. Tivoli OPC can track these jobs and take account of the resources that they use. It is also possible for other subsystems to submit held jobs and for Tivoli OPC to release them when all dependencies are met.

3. Jobs that are not submitted by Tivoli OPC, but trigger events in Tivoli OPC.
   These jobs are specified with event-triggered tracking (ETT). Refer to Planning and Scheduling the Workload for more details of ETT.

4. Jobs that are completely ignored by the Tivoli OPC controller.

One disadvantage of having jobs outside Tivoli OPC (categories 3 and 4) is that Tivoli OPC cannot take account of the resources that they use. Tivoli OPC can schedule and control its jobs to avoid contention for resources (such as tapes, datasets, and JES initiators), but if other jobs use these resources, Tivoli OPC may submit its jobs when a resource is unavailable.
Finding Out about Tivoli OPC

If you are new to Tivoli OPC, the number of pages in its library can be daunting, but you do not need to read it all. Start with this book, using the index and the table of contents to find help for the task that you have to do.

Do

- Read the reports that are produced when the current plan is extended.
- Use the dialogs, especially the dialogs mentioned in this book (Ready List, Query Current Plan, and Modify Current Plan), to find out more about Tivoli OPC and your application and workstation definitions.
- Press PF1 for help when you are stuck in the dialogs.
- Suggest improvements to your administrator if jobs do not run smoothly or if you very often have to make changes through the dialogs.
- Tell the administrator about any tasks that you often do and you think could be automated by Tivoli OPC.
- Use Getting Started with Tivoli OPC for a quick introduction to Tivoli OPC.

Do not

- Run Tivoli OPC-controlled jobs (or their associated job preparation tasks) outside Tivoli OPC, except where this is planned.
- Attempt to get work going by changing the status of jobs or by using the EXECUTE command. If a job is not being submitted, there must be a reason: use the Tivoli OPC dialogs to find the reason and resolve the dependency. Changing the status of an operation and using the EXECUTE command are for exceptional occasions.
Chapter 2. Using the Tivoli OPC Dialogs

To perform most operator tasks, you use the Tivoli OPC dialogs, which run under Interactive System Productivity Facility (ISPF).

This chapter shows you how to get help, how to tailor the PF keys and ISPF options, and how to use the Tivoli OPC filter panels to reduce the amount of data displayed in lists.

Tivoli OPC ISPF Dialogs

How you invoke the dialogs on your system depends on your installation standards. The department that installed Tivoli OPC can tell you how to invoke Tivoli OPC at your installation. Usually, you select the Tivoli OPC option from the ISPF main menu.

Tivoli OPC administration is performed on the controlling system in a multisystem complex, so you must use the dialogs on the system in your configuration that is running the Tivoli OPC controller.

You can reach all the Tivoli OPC controller dialogs from the OPERATIONS PLANNING AND CONTROL/ESA menu, (called the Tivoli OPC main menu). See Figure 3.

```
EQQOPCAP  ------------ OPERATIONS PLANNING AND CONTROL/ESA  ---------------
Option ===>
Welcome to OPC/ESA. You are communicating with the OPCC subsystem.
Select one of the following options and press ENTER.
0 OPTIONS - Define OPC/ESA dialog user parameters and options
1 DATABASE - Display or update OPC/ESA data base information
2 LTP - Long Term Plan query and update
3 DAILY PLANNING - Produce daily plans, real and trial
4 WORK STATIONS - Work station communication
5 MCP - Modify the Current Plan
6 QCP - Query the status of work in progress
7 OLD OPERATIONS - Restart old operations from the DB2 repository
9 SERVICE FUNC - Perform OPC/ESA service functions
X EXIT - Exit from the OPC/ESA dialog
```

Figure 3. EQQOPCAP—Tivoli OPC Main Menu

Before you can use any Tivoli OPC dialog, you must have the authority to access it. See your security administrator or system programmer if you do not have the access that you need.
Using the Dialogs

Setting Options

You do not have to set options every time you use the dialogs. The options, and many of the parameters that you enter in the dialogs, are saved when you leave ISPF (though not if the session is not terminated normally) and will be the default next time.

Select option 0 on the main menu to display this panel:

```
EQQXOPTP -------- DEFINING OPC/ESA PARAMETERS AND OPTIONS ----------------
Option ===>
Select one of the following:
0 REINIT       - Re-initialize the application profile values
1 SUBSYSTEM NAME - Set or change name of Subsystem and Server LU
2 DATE          - Specify date/time formats and default calendar
3 COLOR         - Specify panel color and highlight attributes
4 ISPF OPTIONS  - Specify ISPF/PDF options
5 AD/OI CHECKS  - Specify AD/OI consistency checks
6 JCL EDIT      - Specify JCL edit tool
```

Figure 4. EQQXOPTP—Defining Tivoli OPC Parameters and Options

After you specify the options you prefer, they are used throughout the Tivoli OPC dialog. The options are stored in your ISPF profile dataset. When you use the dialog, the options are retrieved from your profile.

REINIT

Use this option to set the Tivoli OPC profile to the default values defined at installation time. This is done automatically the first time that you use Tivoli OPC. If you start to use Tivoli OPC for a new language feature, perform a REINIT.

SUBSYSTEM NAME

Select this option to specify the name of the Tivoli OPC controller subsystem with which the dialog is to communicate. The name must be an alphanumeric string of not more than 4 characters. If the controller is on a different MVS, the SERVER LU NAME must be given. The LU name can be a fully qualified LU name networkid.luname, 3–17 characters. The LU name is sufficient if the Server is on the same net as the dialogs.
DATE

Use this option to specify the format of dates and times in the Tivoli OPC dialog and, if required, to set a local time offset. The following panel is displayed:

```
EQQXDATP ----------- SETTING OPC/ESA DATE AND TIME FORMAT -----------------------
Command ==> 
Enter/change data below:

DATE-FORMAT ===> YY/MM/DD Combine the characters for year ( YY or CCYY ), and month ( MM ) and
day ( DD ), or day number ( DDD ).
You can use separation characters (such as - or /) if space permits.

TIME-FORMAT ===> HH.MM Combine the characters for
hours ( HH ) and minutes ( MM ).
Optionally separated by any character.

LOCAL TIME OFFSET ===> /zerodot__ Specify local time offset in minutes.
The value must be in the range 0 to 1439.
TIME OFFSET SIGN ===> + Specify - if local time is before OPC/ESA.
Specify + if local time is after OPC/ESA .
CALENDAR ID ===> ________________ Default calendar identification
```

Figure 5. EQQXDATP—Setting Tivoli OPC Date and Time Format

DATE-FORMAT

You can specify dates with these formats:

- CCYYMMDD or YY/MM/DD, where CC is the century. CC, YY, MM, and DD can be in any order.

- YY/DDD or CCYY/DDD, where DDD is the day number in the Julian calendar. CC, YY, and DDD can be in any order.

The delimiter character, shown as a slash (/), is optional and can be any character other than C, Y, M, or D. If you specify CCYYMMDD, you cannot use delimiters. The date format can be no more than 8 characters. Example panels in this book use the format YY/MM/DD.

TIME-FORMAT

Similarly, you can specify the time format as HH.MM or MM.HH. The delimiter character, shown as a period (.), is optional and can be any character other than H or M. Example panels in this book use the format HH.MM.

LOCAL TIME OFFSET

If you are in a different time zone from the Tivoli OPC controller, you can specify a local time offset. This means that actual start and end times are adjusted to take your local time into consideration. The local time offset is the number of minutes your local time is ahead of or behind controller subsystem time; that is, the Tivoli OPC controlling processor clock time. The local time offset specified on this panel applies only to the ISPF profile that you are using.

Note: All time values stored in Tivoli OPC controller datasets are specified in controller subsystem time so that you cannot, for example, use the local-time offset option to specify or display workstation open intervals, input arrival times, or run-cycle start times, in your local time.
Reports created by Tivoli OPC batch jobs always have time values expressed in controller subsystem time and not your local time.

**TIME OFFSET SIGN**
This option is used with LOCAL TIME OFFSET to specify whether your local time is ahead of or behind Tivoli OPC controller subsystem time. For example, you specify + if your local time is ahead of controller subsystem time.

**CALENDAR ID**
Specify the default calendar that Tivoli OPC uses for dialog functions such as the Long-Term Plan dialog, the GENDAYS command for checking run cycles, and substituting JCL variables at job setup. For batch functions, Tivoli OPC uses the calendar specified in the BATCHOPT initialization statement. In both cases, Tivoli OPC looks for a calendar called DEFAULT if no other calendar is specified. For some functions (ETT and the EQUSIN subroutine), Tivoli OPC has no access to BATCHOPT or the dialog default, so will always look for the calendar DEFAULT if no calendar is specified.

If no calendar is specified and there is no calendar called DEFAULT, Tivoli OPC treats all days as work days.

**COLOR**
Select this option to display this panel, where you specify color and highlighting attributes for different panel elements:

```
    EQQXCOLP ------ SETTING OPC/ESA COLOR AND HIGHLIGHT ATTRIBUTES --------
    Command ===> 
    Enter/change data below:

    COLOR  HILITE  PANEL ELEMENT CATEGORY
    WHITE_  _______ Panel titles and data items
    BLUE___ _______ Directional lines and explanatory text
    BLUE___ REVERSE Header text
    WHITE_ _______ Option numbers and command text
    BLUE___ _______ Normal status (for instance output text)
    WHITE_ _______ Important status (for instance output data)
    RED____ _______ Command input
    GREEN__ _______ Optional input
    RED____ _______ Required input
    RED____ BLINK__ Error flagged input

Valid color specifications are:
    WHITE, RED, BLUE, GREEN, PINK, YELLOW, and TURQ
Valid highlight specifications are:
    USCORE, REVERSE, BLINK, and blank for no highlighting
```

Figure 6. EQQXCOLP—Setting Tivoli OPC Color and Highlight Attributes

Tivoli OPC panels have different panel elements; for example, the title of the panel and the command input field. For each of these elements, you can specify color and highlighting. If you set a color to blank, the installation default is used.

To test the effect of the color and highlighting attributes, press Enter to redisplay the panel with the specified attributes.
All panel elements in the Tivoli OPC dialog are subsequently displayed with
the attributes specified on this panel. You can also use the COLOR
command at any time.

**ISPF OPTIONS**
Select this option on the DEFINING TIVOLI OPC PARAMETERS AND OPTIONS
panel to change the following:

**TERMINAL**
Specify these terminal characteristics:
- Terminal type
- Number of PF keys
- Input field pad characters: nulls or blanks
- Command delimiter for stacking commands
- Screen format.

**LOG/LIST**
Specify the log dataset and list dataset options, print process
options, and job statement information for the system printer.

**PF KEYS**
Specify the number of PF keys and the operations they will
perform.

**DISPLAY**
Specify whether the command line is to be placed at:
- The top (ASIS) of the panel (as shown in the examples in this
guide)
- The bottom (BOTTOM) of the panel.

**LIST**
Specify the list dataset record format (FBA or VBA), logical record
length, and line length.

**GRAPHIC**
Specify GDDM graphic print attributes.

**ENVIRON**
Use the ENVIRON commands to trace the TPUT, TGET, and
PUTLINE buffers, to produce system ABEND dumps when not
running in ISPF TEST mode, and to gather terminal status
information.

**KEYLIST**
Modify keylist function. You can issue the KEYLIST command
from the command line of a panel. If you issue the KEYLIST
command from a panel displaying a keylist, that keylist will be
marked

*** CURRENTLY ACTIVE KEYLIST ***.

**DIALOG TEST**
Specify dialog test options. You can specify that ending Dialog
Test (option 7 on the ISPF Primary Panel) will restore the
TEST/TRACE values that were in effect when Dialog Test was
called.

**COLOR**
Change the default color specifications for ISPF panels.

**CUAATTR**
Use the CUA Color/Emphasis Change Utility to change the color,
intensity, and highlight values for panel elements on CUA panels.

Refer to *ISPF Guide and Reference* for a complete description of the fields on
the panel displayed when the ISPF OPTIONS alternative is selected.
AD/OI CHECKS

Use this option to specify whether you want AD/OI consistency checks done each time an Application is created/deleted/modified.

Consistency checks consist of looking for a match in the AD data base for the operator instruction used in the Application. All operator instructions for which no match is found are deleted.

These checks are made immediately after the AD dialog action has been completed, either confirming or canceling it.

For example, if you start the creation of a new Application, APPLX, with one operation, 001, select option 7 from the OPERATION DETAILS panel and create an operator instruction, APPLX-001. Finally, press CANCEL instead of PF3 so that the APPLX application is not created; the consistency checks will lead to the deletion of the newly created operator instruction, because an operation with the key APPLX-001 could not be found in the AD data base.

When you select option 0.5, the following panel is displayed:

---

Figure 7. EQQXAOIP—Setting Tivoli OPC AD/OI Consistency Check

CONSISTENCY CHECK

If Y is specified, consistency checks will be done from the Application Description dialog. Y is the default.

CONFIRM PANEL

This field is meaningful only if Consistency Check has been set to Y.

If Y is specified, a confirmation panel is displayed before deleting any operations selected by the consistency checks. Y is the default.

JCL EDIT

Use this option if you have a tool to edit JCL, and you want to be able to call it from the Tivoli OPC AD dialogs.
When you select option 0.6 the following panel is displayed:

```
EQQXJCLP ------ SETTING JCL EDIT TOOL INFORMATION ---------------------
Command ===> 
Enter/change data below:

PANEL NAME ===> EQQAJCLE Specify the name of the panel to be displayed when editing JCL from AD data base.
This panel name is used to provide a way to call a user tool to edit JCL.
Default value is EQQAJCLE which is a dummy panel.
```

Figure 8. EQQXJCLP—Setting JCL Edit Tool Information

**PANEL NAME**

A link between Tivoli OPC AD dialogs and a user tool for editing JCL is established by means of a panel name. When the JCL Edit option is selected from the AD dialogs, then the panel above is displayed.

The panel must exist, or you will get an ISPF 'panel not found' error.

The default value for the panel name is EQQAJCLE, which is the name of a dummy panel.

**Common Dialog Commands and Facilities**

The common commands that you use on Tivoli OPC panels are described in the following text. Special commands, as they relate to a specific panel, are described in the individual task descriptions.

The following sections also provide some hints and tips for customizing, locating, and displaying Tivoli OPC information promptly.

**Concatenating Options**

You can concatenate Tivoli OPC selection options in the standard ISPF manner. For example, entering 4.1.0 on the Tivoli OPC main menu takes you directly into the workstation ready list that you last displayed. Tivoli OPC lets you enter concatenated options both in row input fields and on the command line.

**Note:** You cannot use the concatenated option to pass through a confirmation panel without displaying it. For example, entering 9.5.Y from the main panel simply takes you into option 9.5.

You can also use the ISPF command delimiter to concatenate options in the Tivoli OPC dialog. For example, enter this command string:

```
9.2;y;=9.1;y;=4.1.cpu1
```

to start and stop the job submission function and return to the ready list display for the CPU1 workstation. When you use the ISPF command delimiter, you pass through confirmation panels without displaying them. This can make completing or deleting a long list of applications or occurrences much faster.
From some Tivoli OPC list panels, you can also concatenate row commands. When the row command input field allows more than 1 character to be specified, you can concatenate options to reach faster the data to which you need access. For example, if you are browsing a list of occurrences in the current plan, you can enter s.2 as the row command for an occurrence in the list to display the list of operations in the selected occurrence. Entering s.6 as the row command for any computer operation in the subsequent list displays the job for the selected operation. Concatenating row commands lets you bypass the intervening panels that present the list of available options.

**Quick Return Command**

As with all ISPF applications, the END command returns you to the previously displayed panel. For example, if you select option 4.1.cpu1 from the main menu, the first panel displayed is the ready list. When you enter END or press PF3 from the ready list, the next panel displayed is the main menu. If you choose not to concatenate the options, it takes three panel displays to reach the ready list and three to return.

You can use the ISPF quick return command (=) as a fast path through the dialog. For example, to return to the ready list from wherever you are in the Tivoli OPC dialog, enter =4.1.0 on the command line.

When you are displaying occurrences and operations in the current plan, you can sometimes navigate through an exceptionally long path. In this case, you should consider using the quick return command even when you want to return to the same area of the dialog.

**Primary Commands**

You can use normal ISPF commands freely throughout the dialog. Table 4 shows some of the ISPF commands used on the command line of Tivoli OPC panels.

<table>
<thead>
<tr>
<th>Command</th>
<th>Action</th>
</tr>
</thead>
</table>
| END     | Save all changes made on the panel and return to the previous panel if no errors were found. END often starts the operation described on the panel.  
**Note:** If there is an error in an input field on a panel, you cannot leave that panel by pressing the End PF key or by issuing the END command; either correct the error or issue the CANCEL command. |
| RETURN  | Return to main menu. An END operation is executed for each panel in the sequence leading back to the main menu; all changes on the individual panels are saved. |
| ENTER   | Verify and save all changes. The panel is then redisplayed (if no error was found), unless you press Enter on a menu or selection criteria panel, in which case the next logical panel is displayed. If an error is discovered, a short error message is issued. You can obtain more information (a long error message) by pressing the Help PF key. |
| CANCEL  | Return to the previous panel without making any changes. |
| UP nn   | Scroll upward through the data by the number of lines specified by nn. |
Table 4 (Page 2 of 2). Primary Commands for Tivoli OPC Panels

<table>
<thead>
<tr>
<th>Command</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>DOWN (nn)</td>
<td>Scroll downward through the data by the number of lines specified by (nn).</td>
</tr>
<tr>
<td>RIGHT</td>
<td>Display the right part of the data. This is available only from panels that have the text LEFT PART in the panel title.</td>
</tr>
<tr>
<td>LEFT</td>
<td>Display the left part of the data. This is available only from panels that have the text RIGHT PART in the panel title.</td>
</tr>
<tr>
<td>HELP</td>
<td>Display help information.</td>
</tr>
<tr>
<td>SORT</td>
<td>Sort information in a list.</td>
</tr>
<tr>
<td>LOCATE (lparm)</td>
<td>Scroll to the field specified. If the field is not found, the list is displayed starting with the entry before which the specified field would have occurred. If the list is sorted by application name, then (lparm) is the name of the application; if sorted by job name, (lparm) is a job name.</td>
</tr>
<tr>
<td>GRAPH</td>
<td>Display a network of dependencies.</td>
</tr>
<tr>
<td>GDDM</td>
<td>Execute GDDM functions on a graphically displayed network.</td>
</tr>
<tr>
<td>ATTR</td>
<td>Set graphic attributes.</td>
</tr>
</tbody>
</table>

Specifying List Criteria
Throughout the Tivoli OPC dialog, options are provided to let you list data elements. You can use commands to edit the data and to perform other operations on these lists.

You use selection criteria to specify the contents of lists in the Tivoli OPC dialog. An example of the panel where you specify list criteria follows:

```
EQQSOPFP ------------------- SELECTING OPERATIONS -------------------
Command ===>
Specify selection criteria below and press ENTER to create an operation list.

JOBNAME ===> P/c5197______
FAST PATH ===> Y Valid only along with jobname
APPLICATION ID ===> ________________
OWNER ID ===> ________________
AUTHORITY GROUP ===> ________
WORK STATION NAME ===> ____
PRIORITY ===> _ Low priority limit
MANUALLY HELD ===> _ Y Yes, N No
STATUS ===> __________ List of status codes: A R * S I C E W U and D
FROM ===> ________ _____
TO ===> ________ _____
GROUP DEFINITION ===> ________________
```

Figure 9. EQQSOPFP—Selecting Operations

You can use blanks, complete names, IDs, or generic search arguments in the input fields on this panel to determine the contents of the list.
When you request a list of operations in the current plan, Tivoli OPC scans sequentially through all operations defined in the plan to determine which operations match the criteria you have specified. If the current plan is large, this search can take a long time. On some selection criteria panels, you can choose the fastpath option to reduce the overhead of the search. When fastpath is used, Tivoli OPC searches through operations included in the current plan job name table, which only has operations on automatic reporting workstations (such as print and computer workstations). When it finds a job name that matches, Tivoli OPC includes all operations with that job name, whether on an automatic workstation or not. So the search in Figure 9 on page 15 will find all automatic workstation operations with a job name beginning with P and all other operations with the same name.

Using Generic Search Arguments
Many input fields in the Tivoli OPC dialog accept generic search arguments. You specify a generic search argument by entering either an asterisk (*) or a percent sign (%) in an input field. You can enter these characters by themselves or in combination with other characters.

Use an asterisk (*) to represent any character string or a null string. The percent sign (%) represents any single character.

If you want to select all applications whose first three letters are PAY, enter these characters in the input field:

```
APPLICATION ID ===> PAY*________
```

If you want to select all applications where P is the first letter and Y is the third letter, you enter:

```
APPLICATION ID ===> P%Y*________
```

The percent sign in this example results in a search for application identifiers where there is any single character between P and Y.

Some section panels contain the:

```
TYPE OF MATCH ===> 
```

field. Using this field you can specify the type of match that should be applied for filters, allotating wild characters (*) and %) to be treated as normal characters. If the field is left blank, then standard generic matching is done.

When double-byte character set (DBCS) mode is specified for kanji, the DBCS asterisk (X'425C') and the DBCS percent sign (X'426C') are accepted as generic search arguments. These search arguments can be used for application and owner IDs.

Sorting List Output
In all list displays, you can enter the SORT command at the command prompt to display a panel where you can specify the order of the list items. The sort order you request remains in effect for that specific list type until changed.
For example, if you request a list of occurrences in the current plan and then enter SORT when the list is presented, the following panel is displayed:

```
EQQXSRTL ------------------------ SORTING A LIST -------------- ROW 1 TO 14 OF 49
Command ===> Scroll ===> CSR

Enter/change data in rows
or enter DELETE in the command field to delete the active sort:
Sort order = 1-9, where 1 is the first sort field.
Direction = A for ascending (default) or D for descending.
```

<table>
<thead>
<tr>
<th>Sort Direction</th>
<th>Title of field</th>
<th>Description of field</th>
</tr>
</thead>
<tbody>
<tr>
<td>op.</td>
<td>Operation no. 1</td>
<td>first critical op.</td>
</tr>
<tr>
<td>time</td>
<td>Actual input arrival time of occ.</td>
<td></td>
</tr>
<tr>
<td>time</td>
<td>Actual completion time of the occ.</td>
<td></td>
</tr>
<tr>
<td>time</td>
<td>Actual start time of first critical op.</td>
<td></td>
</tr>
<tr>
<td>time</td>
<td>Latest start time first critical op.</td>
<td></td>
</tr>
<tr>
<td>time</td>
<td>Deadline time of first critical op.</td>
<td></td>
</tr>
<tr>
<td>time</td>
<td>Input arrival time of first critical op</td>
<td></td>
</tr>
<tr>
<td>time</td>
<td>Planned start time of first critical op</td>
<td></td>
</tr>
<tr>
<td>time</td>
<td>Deadline time of the occurrence</td>
<td></td>
</tr>
<tr>
<td>Adding function</td>
<td>What added occurrence to current plan</td>
<td></td>
</tr>
<tr>
<td>Application</td>
<td>Application ID</td>
<td></td>
</tr>
<tr>
<td>Application text</td>
<td>Verbal description of the application</td>
<td></td>
</tr>
<tr>
<td>Arrived</td>
<td>Actual input arrival date of occ.</td>
<td></td>
</tr>
<tr>
<td>Authgrp</td>
<td>Authority group</td>
<td></td>
</tr>
</tbody>
</table>

Figure 10. EQQXSRTL—Sorting a List

DBCS identifiers in sorted lists appear in binary order.

**Locating Data Strings in List Output**

You can enter LOCATE at the command prompt on any list display panel to find a particular data string in the primary sort field. The command also supports generic search strings. For example, if you enter LOC ABC* to find any item in the list beginning with ABC, the list scrolls to the specified field. If the field is not found, the list is displayed starting with the entry before which the specified field would have occurred.

If application name is the primary sort field, request LOCATE applname; similarly, if sorted by job name, request LOCATE jobname. If you need to regularly issue a locate command against a list of items that is not sorted by the item you want to locate, you can change the sort order by entering the SORT command.

This command is useful when working with lists of special resources or event-triggered-tracking criteria.

**Graphically Displaying Lists**

If you have GDDM installed and have a terminal capable of displaying graphics, you can display lists of applications, occurrences, and operations graphically. Graphic displays contain the same information as edit or selection lists, only the format is different. On a graphically displayed list, you can see dependency connections that might be difficult to visualize from a conventional list. An entire network of applications can be represented.

To see a graphically displayed list, enter GRAPH at the command prompt of a panel on which graph is an option. The clarity of the graph depends on how much data GDDM is trying to present on your display terminal. For example, if you request a
GRAPH of all operations in the current plan, you would probably see an attractive geometric design that resembles a bunch of grapes: there is simply too much data to present graphically on a standard workstation display terminal.

You can use the GDDM command to display the GDDM menu where you can choose the zoom and print functions. With GDDM Version 3, you can print a zoomed area. Before printing, change the colors so that you have a black-and-white image. It is often useful to save the graph in the ADMGDF format, which you can manipulate with other programs. For more information, refer to GDDM User's Guide.

The attributes of a graphically displayed list can be customized. Enter ATTR at the command prompt from the graph panel. You can specify the color, shape, and line type for different items displayed in the graph. Press PF1 (HELP) for possible values for the attributes.

**PF Key Assignment**

The Tivoli OPC dialog maintains separate program function (PF) keys from your normal ISPF key assignments. Enter KEYS at the command prompt to display or change the current assignment.

You can assign a PF key to a Tivoli OPC command that you use regularly; for example, to display the ready list. To ensure that the command will be executed correctly regardless of the Tivoli OPC panel it is entered from, assign the PF key like this:

```
PF5 ===> ;=4.1.cpu1
```

where ; is your ISPF command delimiter.

You can assign PF keys separately for each Tivoli OPC panel. For example, if you regularly use the application description dialog, you can assign PF keys to the OPER and RUN commands. The PF keys that you assign for a particular panel are in effect only for that panel; in other parts of the dialog, your standard Tivoli OPC PF keys are in effect.

It is recommended that you do not alter the key assignments for PF1 (HELP), or PF12 (RETRIEVE). PF12 (RETRIEVE) returns the command you last executed to the command prompt; press PF12 again, and the command before that is returned. A stack of approximately 25 commands is maintained.
In support of the PFSHOW command, the PF KEY DEFINITIONS AND LABELS panel lets you optionally assign labels to the PF key definitions. A label is used for display in place of the corresponding definition when the user issues the PFSHOW command.

```
ISPOPT3B ------- PF KEY DEFINITIONS AND LABELS - PRIMARY KEYS ------------------
COMMAND ===> 
NUMBER OF PF KEYS ===> 24 TERMINAL TYPE ===> 3278
PF13 ===> ;=4.1.oper
PF14 ===> ;=4.1.cpu1
PF15 ===> ;=5.4.0
PF16 ===> ;=5.1
PF17 ===> ;=5.2
PF18 ===> ;=6.1
PF19 ===> ;=6.3
PF20 ===> ;=3.1
PF21 ===> ;=3.2
PF22 ===> ;=2.1
PF23 ===> ;=2.2
PF24 ===> ;=1.4.1
PF13 LABEL ===> rl_oper PF14 LABEL ===> rl_cpu1 PF15 LABEL ===> errors
PF16 LABEL ===> mcp_add PF17 LABEL ===> mcp_mod PF18 LABEL ===> qcp_appl
PF19 LABEL ===> qcp_job PF20 LABEL ===> dpreplan PF21 LABEL ===> dpreplan
PF22 LABEL ===> ltp_look PF23 LABEL ===> ltpbatch PF24 LABEL ===> ad_look
Press ENTER key to display alternate keys. Enter END command to exit.
```

Figure 11. ISPOPT3B—PF Key Definitions and Labels

When you enter the PFSHOW command from any Tivoli OPC panel, the labels displayed at the bottom of Figure 11 are presented on your Tivoli OPC panels. To remove the display from your panels, enter PFSHOW OFF.

**Workload Monitor/2**

Workload Monitor/2 is a feature that runs on a personal workstation that is running IBM Operating System/2 (OS/2) Version 2 or later. It communicates with, and presents data from, active Tivoli OPC systems.

With Workload Monitor/2, you can:
- Filter and display data about operations in the current plan.
- Filter and display workstation data
- Display detailed data about individual operations and workstations
- Display detailed information about the current plan
- Tailor and display graphical views of the workload
- Define alerts for operations that you want to monitor
- Update the status of planned operations.

For more information about Tivoli OPC Workload Monitor/2, refer to *Tivoli OPC Workload Monitor/2 User's Guide*. 

Chapter 2. Using the Tivoli OPC Dialogs
Using the Dialogs

Tivoli OPC Utilities

The LTP and the current plan are created and extended using Tivoli OPC batch programs. Other Tivoli OPC programs and utilities update and report on data in the Tivoli OPC databases. You can submit these programs using the Tivoli OPC dialogs or using normal job submission. For several of the batch processes, particularly extending the plans, it is convenient to define the job in a Tivoli OPC application and use the full facilities of Tivoli OPC to schedule, submit, and track the processing.

When you submit the jobs from the Tivoli OPC dialog, you see the panel in Figure 12.

```
**EQQXSUBP** -------------- **GENERATING JCL FOR A BATCH JOB** --------------

Command ===> 

Enter/change data below and press ENTER to submit/edit the JCL.

JCL to be generated for: REPLAN CURRENT PLAN PERIOD

SYSOUT CLASS ===> C  (Used only if output to system printer)
LOCAL PRINTER NAME ===> ______ (Used only if output on local printer)
DATASET NAME ===> ____________________________________________  
                   (Used only if CLASS and LOCAL PRINTER are both blank). If blank default name used is XMAWS.EID4.DPREP.LIST
SUBMIT/EDIT JOB ===> S  S to submit JOB, E to edit

Job statement :

===> //XMAWSM JOB (B25401,NOBO),’OPC/ESA EID4’,CLASS=B,MSGCLASS=Q,______
===> // MSGLEVEL=(1,1),TIME=5,NOTIFY=XMAWS_______________________
===> ___________________________________________________________________
===> ___________________________________________________________________
```

**Figure 12. EQQXSUBP—Generating JCL for a Batch Job**

When you submit the jobs that update or report on Tivoli OPC data using the dialog:

- The job is submitted using TSO submit; as a result, the authority of the submitting user is assigned to the job.
- The JCL for the job is not saved in the Tivoli OPC JCL repository. If the job needs to be rerun, you must re-create the JCL.
- If you select E for the SUBMIT/EDIT option on the EQQXSUBP panel, an ISPF edit panel containing the JCL is displayed. Enter submit on this panel if you want the job to run. When you enter End or use PF3 from the ISPF edit panel, the job is not submitted. From the edit panel, you can save the job using the CREATE command.
Chapter 3. Monitoring the Workload

All operations in the current plan are associated with a workstation. You can use the Tivoli OPC dialogs to display the operations. For an overall view, you use the Query Current Plan dialog (described later in this chapter in “The QCP Dialog” on page 37). However, if you want to find out which operations are due to start or are already started at a workstation, use the workstation ready list.

Using the Ready List Dialog

The ready list contains operations that have no outstanding predecessors, operations defined to the workstation that are waiting for a particular time or resource, operations that have already started, and operations that have ended in error. The status codes of operations in the ready list can be A, I, R, *, S, or E. The ready list displayed by Tivoli OPC can also include one operation in C status, the last operation that you manually set to complete. This operation is maintained on the ready list to give you an opportunity to reset the operation if you change your mind.

To display the ready list for a workstation, select option 4 from the Tivoli OPC main menu, which will then display the COMMUNICATING WITH WORKSTATIONS menu (Figure 13). The ready list functions are available by selecting option 1 on this panel.

```
EQQRTOPP ---------- --- COMMUNICATING WITH WORK STATIONS -----------------------
Option ===>
Select one of the following:
1 READY LIST   - Using the ready list
2 WAITING LIST - Review submitted jobs that have a waiting status
3 JOB SETUP    - Setup the JCL for jobs
4 WORK STATIONS - Review the status of work stations
9 DEFINE RL    - Define alternative ready list layouts
```

Figure 13. EQQRTOPP—Communicating with Workstations

You can customize the layout of the workstation ready list to suit your needs, either by selecting one of the supplied layouts or by building your own layout.
Selecting a Ready List Layout

When you select the ready list for a workstation for the first time, you must specify the layout to use for the list. Approximately 90 different fields can be displayed in a ready list (see Appendix C, “Fields Displayed in Ready and Error Lists” on page 121), but not all of these can be displayed at once. Information on the fields that are to be displayed, and the order in which they should be displayed, is kept in a named ready list layout. There might be many different layouts in your installation. Each supplied default layout is designed for a specific workstation type.

If you have not specified a layout for the workstation that you have selected, Tivoli OPC displays a list of layouts, so you can then select a layout from this list. Tivoli OPC stores the layout you choose for each workstation in your ISPF profile. When you request the ready list for a workstation, Tivoli OPC retrieves the name of the layout ID you prefer from your ISPF profile. If you want to change the layout of the ready list, enter the name of another layout in the corresponding field in the ready list: Tivoli OPC displays the ready list again using the new layout you specified.

You can get a list of all the ready list layouts by typing an asterisk (*) in the LAYOUT ID field on the SPECIFYING READY LIST CRITERIA panel and then pressing Enter.

```
Command ===> 

Enter/Change data below and press ENTER to create a ready list.

WORK STATION NAME ===> CPU1 (Blank presents a list.)
LAYOUT ID ===> * An id, blank for default, * for a list

Selection criteria:
APPLICATION ID ===> 
OWNER ID ===> 
JOB NAME ===> 
LOWEST PRIORITY ===> Lowest priority to be selected.
OPERATION STATUS ===> List of status codes, A R S I E or blank

Latest input arrival: Select only operations with input
DATE ===> arrival before this date and time.
TIME ===> (Format YY/MM/DD and HH.MM)

Status sort order ===> CES List of status codes, A R S I E or C
(Any three must be selected, or all blank)
```

Figure 14. EQQRSRLP—Specifying Ready List Criteria
Creating Your Own Ready List Layout

Ready list layouts are kept in two ISPF tables: one for your own use, and one that contains your installation-defined layouts. Your own layout overrides the installation layout.

To change a ready list layout, select option 9, the DEFINE RL option on the COMMUNICATING WITH WORK STATIONS panel. Tivoli OPC then displays a list of both your own and any installation-defined layouts (Figure 15). From this list, you can create new layouts or select a layout to delete, copy, modify, or browse. Any modified installation-defined layout is stored as a private copy of the layout in your ISPF profile. You can delete your own layout, but you cannot delete an installation layout.

To make a set of private layouts available for your colleagues:

1. Create a complete set of layouts. If you want to include a supplied (or old) layout, edit and save it so that it becomes part of your private library.

2. Save (by renaming) the old EQQRLDEF member in the common table library.

3. Copy the EQQRLOUT member from your ISPF profile library to the common table library, renaming it to EQQRLDEF.

```
<table>
<thead>
<tr>
<th>Row</th>
<th>Layout</th>
<th>Description</th>
<th>Owner</th>
<th>Last update</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>C1</td>
<td>Layout 1 for CPU work station</td>
<td>LEIFTO2</td>
<td>96/09/15 08.21</td>
</tr>
<tr>
<td>2</td>
<td>C2</td>
<td>Layout 2 for CPU - times</td>
<td>LEIFTO2</td>
<td>96/09/13 13.12</td>
</tr>
<tr>
<td>3</td>
<td>C3</td>
<td>Layout 3 for CPU - options</td>
<td>LEIFTO2</td>
<td>96/09/13 13.19</td>
</tr>
<tr>
<td>4</td>
<td>WTO</td>
<td>Ready list layout for WTO</td>
<td>XRAYNER</td>
<td>97/02/03 12.28</td>
</tr>
<tr>
<td>5</td>
<td>LAYOUT1</td>
<td>My layout</td>
<td>XRAYNER</td>
<td>97/04/20 13.18</td>
</tr>
</tbody>
</table>
```

Figure 15. EQQRLYLL—Ready List Layouts
When you modify or create a layout, Tivoli OPC presents a list of the available ready-list fields (see Appendix C, “Fields Displayed in Ready and Error Lists” on page 121 for a complete list). From this list, shown in Figure 16, you select the fields to be contained in the ready list layout, and the column order. You can also specify whether the field should be highlighted when Tivoli OPC displays it. All the items previously selected for display are placed at the top of the list in column order (1).

Note that not all fields are applicable for all kinds of workstations or operations. For example, the actual duration field, Act dur, will not be applicable for a general automatic workstation as its operations disappear from the Ready List when complete.

On the CREATING A READY LIST LAYOUT panel, you can also specify a text string used by the ISPF select service to invoke a user exit.

![Figure 16. EQQRLYCL—Creating a Ready List Layout](image)

### Ready List Layout User Exit

The user exit is passed the status of an operation in the ready list output. It can set the next logical status plus the error code and duration of the selected operation.

#### Invoking the User Exit from the Tivoli OPC Dialog

The user exit is invoked when you enter the row command R (reset) or N (set next status) in a row of the Ready List panel, EQQRLRLM. It is invoked for only those workstation that have been defined as follows:

<table>
<thead>
<tr>
<th>Type</th>
<th>General</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reporting attribute</td>
<td>Manual start and completion</td>
</tr>
<tr>
<td>Job setup</td>
<td>N</td>
</tr>
</tbody>
</table>
The actual status of the operation must be one of the following:

- Waiting for arrival
- Ready
- Interrupted
- Ready nonreporting

Although defined, the user exit is not invoked for other cases.

**Defining and Setting Up the User Exit**

You can specify a user exit routine in the USER EXIT field of the Creating a Ready List Layout panel, EQQRLYCL. This specification can be in the form of one of the following command strings, depending on the type of user exit routine:

- `CMD(listname)`
- `PANEL(panelname)`
- `PGM(modulename)`

Store CLISTs, panels, and programs that comprise the user exit routine in a selected dataset that is concatenated in the SYSPROC, ISPPLIB, ISPMLIB, and STEPLIB statements of the procedure or CLIST used for Tivoli OPC Dialog setup. (For more information, refer to “Setting up the controller ISPF environment” on page 71 of the *Tivoli OPC Installation Guide*.)

**Communicating with the User Interface Routine**

Communication between the Tivoli OPC dialog and user exit routine is through the ISPF variable defined in shared pool. The Tivoli OPC dialog passes the following variables to the user exit routine:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UROSTAT</td>
<td>Current operation status</td>
</tr>
<tr>
<td>URNSTAT</td>
<td>New operation status</td>
</tr>
<tr>
<td>UROPERR</td>
<td>New operation error code</td>
</tr>
<tr>
<td>UROPDUR</td>
<td>New operation duration</td>
</tr>
<tr>
<td>N3P_COM</td>
<td>ISPF variables related to the operation</td>
</tr>
</tbody>
</table>

The UROSTAT variable contains the status value of the current operation. New operation parameters are passed initialized to blank.

<table>
<thead>
<tr>
<th>N3P_COM Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPGROUP</td>
<td>Authority group</td>
</tr>
<tr>
<td>OPADI</td>
<td>Application ID</td>
</tr>
<tr>
<td>OPIAD</td>
<td>Application input arrival after MCP</td>
</tr>
<tr>
<td>OPIAT</td>
<td>..Date and time</td>
</tr>
<tr>
<td>OPTXT</td>
<td>Textual description</td>
</tr>
<tr>
<td>OPJBN</td>
<td>OS jobname or blank</td>
</tr>
<tr>
<td>OPWSN</td>
<td>Workstation name</td>
</tr>
<tr>
<td>OPNUM</td>
<td>Operation number</td>
</tr>
<tr>
<td>OPJCL</td>
<td>JOB-, SYSOUT-class, or blank</td>
</tr>
<tr>
<td>OPFRM</td>
<td>From number or blank</td>
</tr>
<tr>
<td>OPPSD</td>
<td>Planned start,</td>
</tr>
<tr>
<td>OPPST</td>
<td>..Date and time</td>
</tr>
<tr>
<td>OPPED</td>
<td>Planned end,</td>
</tr>
<tr>
<td>OPPET</td>
<td>..Date and time</td>
</tr>
<tr>
<td>OPOID</td>
<td>Operation input arrival,</td>
</tr>
<tr>
<td>OPOIT</td>
<td>..Date and time</td>
</tr>
<tr>
<td>OPODD</td>
<td>Operation deadline,</td>
</tr>
<tr>
<td>N3P_COM Variable</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>OPODT</td>
<td>Date and time</td>
</tr>
<tr>
<td>OPLOD</td>
<td>Latest out for operation,</td>
</tr>
<tr>
<td>OPLOT</td>
<td>Date and time</td>
</tr>
<tr>
<td>OPASD</td>
<td>Actual start,</td>
</tr>
<tr>
<td>OPAST</td>
<td>Date and time</td>
</tr>
<tr>
<td>OPAAD</td>
<td>Actual arrival,</td>
</tr>
<tr>
<td>OPAAT</td>
<td>Date and time</td>
</tr>
<tr>
<td>OPISD</td>
<td>Intermediate start if</td>
</tr>
<tr>
<td>OPIST</td>
<td>Interrupted, date and time</td>
</tr>
<tr>
<td>OPAED</td>
<td>Actual end,</td>
</tr>
<tr>
<td>OPAET</td>
<td>Date and time</td>
</tr>
<tr>
<td>OPEDU</td>
<td>Estimated duration, hh.mm</td>
</tr>
<tr>
<td>OPADU</td>
<td>Actual duration, hhhh.mm</td>
</tr>
<tr>
<td>OP£PS</td>
<td>Parallel servers required</td>
</tr>
<tr>
<td>OP£R1</td>
<td>WS resources required (r1)</td>
</tr>
<tr>
<td>OP£R2</td>
<td>WS resources required (r2)</td>
</tr>
<tr>
<td>OPCST</td>
<td>Current status</td>
</tr>
<tr>
<td>OPERR</td>
<td>Error code</td>
</tr>
<tr>
<td>OPAEC</td>
<td>Automatic error completion (Y or N)</td>
</tr>
<tr>
<td>OPPRI</td>
<td>Priority</td>
</tr>
<tr>
<td>OPXST</td>
<td>Extended status</td>
</tr>
<tr>
<td>OPESU</td>
<td>Number of successors</td>
</tr>
<tr>
<td>OP£PR</td>
<td>Number of predecessors</td>
</tr>
<tr>
<td>OPESR</td>
<td>Number of special resources</td>
</tr>
<tr>
<td>OPAJR</td>
<td>Automatic hold/release (Y or N)</td>
</tr>
<tr>
<td>OPASUOP</td>
<td>Automatic submit (Y or N)</td>
</tr>
<tr>
<td>OPTJT</td>
<td>Time job (Y or N)</td>
</tr>
<tr>
<td>OPRESTA</td>
<td>Restartable operation</td>
</tr>
<tr>
<td>OPRERUT</td>
<td>Reroutable operation</td>
</tr>
<tr>
<td>OPWRER</td>
<td>Operation was rerouted</td>
</tr>
<tr>
<td>OPAWS</td>
<td>Operation alternate WS</td>
</tr>
<tr>
<td>OPD WTO</td>
<td>Operation deadline WTO</td>
</tr>
<tr>
<td>OPMX LVL</td>
<td>Maximum nesting level</td>
</tr>
<tr>
<td>OPUPDA</td>
<td>Operation userdata field</td>
</tr>
<tr>
<td>OPMHLD</td>
<td>Operation manually held field</td>
</tr>
<tr>
<td>OPNOP</td>
<td>Operation nop field</td>
</tr>
<tr>
<td>OPEX EC</td>
<td>Operation execute field</td>
</tr>
<tr>
<td>OPMCUP</td>
<td>Last MCP update</td>
</tr>
<tr>
<td>OPJES</td>
<td>Job ID</td>
</tr>
<tr>
<td>OPMCJST</td>
<td>Joblog status</td>
</tr>
<tr>
<td>OPEX DES</td>
<td>Execution destination</td>
</tr>
</tbody>
</table>
Using the Ready List

These are some of the tasks you can perform using the ready list:

- Diagnosing delays. See page 36.
- Setting the status of an operation. See page 28.
- Resetting an operation to its previous state. See page 29.
- Interrupting an operation. See page 29.
- Reporting an operation as ended-in-error. See page 29.
- Viewing operator instructions. See page 29.
- Preparing JCL at a setup workstation. See page 30.
- Delaying an operation, and releasing it. See page 33.
- Removing an operation from the current plan, and restoring it. See page 34.
- Running an operation immediately with EXECUTE. See page 35.
- Viewing a list of operations in the history database. See page 63.

Select option 1 on the COMMUNICATING WITH WORKSTATIONS panel (Figure 13 on page 21) to see the ready list. Figure 17 shows a typical ready list for a computer workstation.

```
Figure 17. EQQRLRLM—Ready List

Enter the SORT command to display the list-field headings and their meanings. The ready list by default sorts the operations by job name within status. However, you can change the sort fields presented on the ready list. See Appendix B, “Status, Error, and Reason Codes” on page 115 for a complete list of status codes and extended status codes. If you sort operations by the actual start times, the jobs that have been running the longest will appear at the top of your display.

You cannot override sort criteria that you have specified on a filter panel with the SORT command.
```
Setting the Status of an Operation

When you work with a ready list, you are normally changing the status of a Tivoli OPC operation in the cases where the workstation does not report the status change automatically. To do this, you can let Tivoli OPC assign the next logical status, or you can set the status explicitly.

Letting Tivoli OPC assign the next status: Letting Tivoli OPC assign the next status is the easier way: enter the row command $N$ beside the operation whose status is to be changed. Tivoli OPC then changes the status to the next logical status for that type of workstation:

- For automatic reporting workstations, the order is: A, R, S, C.
- For manual start and completion workstations, the order is: A, R, S, C.
- For completion only workstations, the order is: A, R, C.
- For nonreporting workstations, the order is: A, C.

Sometimes you see operations on a ready list in $*$ status. This means the operation is ready, but one or more predecessors are defined on a nonreporting workstation.

Setting the status explicitly: To set the status explicitly, enter the row command $N-x$, where $x$ is the status you want set for the operation. You might want to do this to immediately set the status of an operation to complete, or to report an operation as interrupted or as ended-in-error. Sometimes Tivoli OPC requests more information when you set the status; for example, it might request the duration of the operation, or the error code. When Tivoli OPC requests more information, it displays a panel where you should enter the information.

Notice that:

- Setting the operation status to $S$ (started) for operations on computer workstations does not cause the job or started task that is associated with the operation to be submitted. First establish why the operation is not started. When you have done this, you can remove the cause of the delay so that Tivoli OPC can run the work normally. See “Diagnosing Delays” on page 36.
- Setting a WTO operation defined to a general workstation to started does not cause a WTO message to be issued.
- Setting the status of a started operation to $E$ (error), or to any other status, does not cancel the job.
- Setting the status of an operation to $C$ (completed) allows successor operations to start (other dependencies permitting), even though the job might still be executing.

See Chapter 4, “Updating the Current Plan” on page 43.
Resetting an Operation to Its Previous State
Tivoli OPC retains the most recent operation that you set to X status on the ready list. You can reset the operation to its previous state with the R row command, but only if the status of successor operations has not changed.

Interrupting an Operation
The status of an active operation can be set to I (interrupted) automatically if it is active on a workstation that has the splittable attribute. A computer workstation never has this attribute. You might want to interrupt job setup, for example, if you are half-way through editing a job when you have to break for a meeting.

When you interrupt the operation, the resources that the operation uses are freed. While the operation is in interrupted status, its duration is not incremented. There is no limit to the number of times that an operation can be started and interrupted.

If the status of an operation is not S (started), use the MCP dialog to set the status to interrupted. See the MODIFYING OPERATIONS IN THE CURRENT PLAN panel in Figure 42 on page 62 for details.

Reporting an Operation As Ended-in-Error
Tivoli OPC automatically reports jobs and started tasks that fail with E (ended-in-error) status. You can manually report operations as ended-in-error on any type of workstation. In these circumstances, you must decide what the error status signifies. The advantage of reporting the operation as ended-in-error is that a problem is highlighted. Tivoli OPC puts the operation in the ended-in-error list, which you should review regularly so that error-recovery action can be taken. Refer to Planning and Scheduling the Workload for information on Tivoli OPC automatic recovery.

To manually report an operation as ended-in-error, enter the N-E row command beside the operation. Tivoli OPC then prompts you for an error code. It is good practice to reserve some error codes for specific error situations.

Setting the status of a started operation to E (error), or to any other status, does not cancel the job.

Viewing Operator Instructions
Some operations might require specific instructions on how they are to be handled. These instructions are known as operator instructions. You can tell whether instructions exist for an operation by looking at the code in the OI ready-list field:

N There are no operator instructions.
Y There are operator instructions.
+ Some operator instructions have been changed recently. The default is 30 days, but the definition of “recent” depends on the setting of the NEWOILIMIT keyword of the JTOPTS parameter at your installation. See your Tivoli OPC administrator.

To browse operator instructions, enter the O command beside the operation. Tivoli OPC then displays the instructions using the ISPF/PDF browse function.
Preparing Jobs at a Setup Workstation

A setup workstation is a general workstation used for preparing jobs. In Tivoli OPC, the operation of preparing jobs is immediately followed by the operation that runs the job on the computer workstation. If it is not waiting for other conditions to be met, the job can be started as soon as job setup is complete.

When you set the next logical status for an operation at a job setup workstation (with the N row command), Tivoli OPC reports the status of the setup operation as $ (started). The action that Tivoli OPC takes depends on whether it finds promptable variables in the job that have not been resolved.

**Note:** When you edit the job using the Ready List or MCP dialogs, you edit the latest job from the JS file, which is where Tivoli OPC places modified jobs. The original job is always left unaltered in the partitioned dataset allocated to the ddname EQQJBLIB (JBLIB). To force Tivoli OPC to read a fresh copy of the job from JBLIB, delete all the lines and end the edit.

Preparing Jobs without Unresolved Promptable Variables

Type the N row command beside the job to edit it and start the setup operation. Tivoli OPC automatically resolves any nonpromptable job variables that are defined to be resolved at setup time and then invokes the ISPF editor.

You can edit the job as required on the EDITING JCL FOR AN OPERATION panel (Figure 18).

---

**Figure 18. EQQRJCLE—Editing JCL for an Operation**

---
The job in Figure 18 on page 30 has several errors:

1. The error message 2 refers to the line following (3). CDATE is a dynamic-format variable whose format must be specified in a SETFORM directive before being used.

2. When variables are not found, other variables, such as those on the job card, are not substituted either, even though Tivoli OPC has found them. You do not see any substituted variables unless Tivoli OPC has found them all.

3. The variable on 1 is wrong. You cannot have variables in the Tivoli OPC RECOVER directive. Tivoli OPC has not pointed out this error because it does not scan recovery statements unless the job fails.

When you have finished editing the job, exit by entering one of the following commands:

- END, which causes Tivoli OPC to save the modified job in the JCL repository, set the status of the setup operation to C (complete), and start the successor operation (the job itself) unless there are other dependencies.
- CANCEL, which causes Tivoli OPC to exit without saving the job in the JCL repository. The status of the setup operation is still R (ready).
- TSAV E, which saves your edited job in the JCL repository for later editing and changes the setup operation to status I (interrupted). Use this command if you want to keep your changes and continue editing later.

Preparing Jobs with Unresolved Promptable Variables

Tivoli OPC:

1. Scans a job for variables that are not specified as substitute-at-submit variables.

2. Displays all promptable variables on the LIST OF JCL PREPARATION VARIABLES TO BE SET panel (Figure 19) so that you can enter values.

```
Application id : SCRIPT7 An AIX/6/zerodot/zerodot/zerodot transfer script
Operation : CPU7 /zerodot15
Jobname : SCRIPT7
EDIT JCL ===> Y Edit the tailored JCL: Y, or N

Row Variable Variable Variable
cmd name description value
' PROMPT1 Line 1 of data greetings ________________________________
' PROMPT2 Line 2 of data from AIX/6000 _______________________

*********************************** BOTTOM OF DATA ****************************
```

Figure 19. EQQRLVAL—List of JCL Preparation Variables to Be Set
3. Ends when you enter one of these commands:
   - CANCEL. Tivoli OPC skips variable substitution and does not save the job.
   - END, which causes Tivoli OPC to:
     a. Validate all promptable variable values
     b. Substitute promptable variables in the job
     c. Substitute all nonpromptable variables
     d. Store the job in the JCL repository.

If errors are found, a message is displayed on the variable list panel.

If you specify that you want to edit the job on the LIST OF JCL PREPARATION VARIABLES TO BE SET panel, Tivoli OPC invokes the ISPF edit function when you enter the END command and displays the EDITING JCL FOR AN OPERATION panel (Figure 20).

```
Converted JCL below and press END to complete, CANCEL to reject and reset, or TSAVE to save changes and interrupt the operation.

Application : SCRIPT7 An AIX/6000 transfer script
Operation : CPU7 015
Jobname : SCRIPT7 JCL last updated by: XRAYNER

# Create a data file
echo 'greetings ' > $jclfile
echo 'from AIX/6000' >> $jclfile

# Create a file to send
echo 'open SYSTEM' > $ftppfile
echo 'site file=jes' >> $ftppfile
echo 'site lrecl=80' >> $ftppfile
echo "put $jclfile" >> $ftppfile

# Invoke FTP to send the file
ftp < $ftppfile
rm $jclfile

# Invoke command and save return code
$command
src=$?
if [ "$src" -eq 0 ]
  then status=C
  else status=E
fi
```

Figure 20. EQQRJCLE—Editing JCL for an Operation

After you have modified the job, the END command saves the job in the JCL repository, and you exit from ISPF edit. If you enter the CANCEL command, you do not cancel the variable values set because they are already stored in the JCL repository.

If the setup operation is for several processor operations (all with the same job name), Tivoli OPC displays a list of the operations that you can choose from.

**Note:** The standard ISPF edit SAVE command has no effect when you are editing a job within Tivoli OPC.
Other Ways of Editing a Job

There are several ways to do job setup using the dialog:

- Select the JOB SETUP option from the COMMUNICATING WITH WORK STATIONS panel. You see a selection list that presents a list of operations eligible for setup. Type J next to one of the operations. This puts you into ISPF edit on the job for that operation. Change the job and press PF3. The changed job is stored in the JCL repository, and the operation is automatically set to C status (complete).

  **Note:** You can change any job from the JOB SETUP option, even if the operation does not have a job setup operation associated with it. The changed job is stored in the JCL repository as usual. Because there is no setup operation associated with this activity that can be marked as complete, this editing procedure has no effect on the Tivoli OPC schedule.

- Use the MCP dialog to edit the job for any operation in the current plan. See “Modifying an Operation” on page 62.

- Set up the job for an individual occurrence in the Long-Term-Plan dialog. The edited job for the future occurrence is stored in the JCL repository.

Delaying an Operation, and Releasing It

Sometimes you must delay the start of an operation because of a situation beyond your control. For example, the application programmer is manually editing some production files to incorporate an urgent program fix. In such situations, when the operations concerned are already in the current plan and waiting only for a certain time or for predecessors to be complete, you must do something to stop the operation from being started when the scheduling criteria are satisfied. You can:

- Manually HOLD the operation by using the MCP dialog, or the ready list if the operation predecessors are already complete.

- Modify the job to include a deliberate error; for example, a comma at the end of the job card for an MVS job. The job is submitted when all the scheduling criteria are met but does not actually execute until the syntax error is corrected.

- Modify the occurrence to include an extra operation on a general workstation, which becomes a predecessor for the operation you need to delay.

The manual HOLD command, MH, can be issued for an operation on a computer workstation with automatic reporting or on any workstation with no reporting, if the current status of the operation is A, R, *, W, C, or E. Tivoli OPC does not start any operation that has been manually placed in HOLD by a dialog user, even though the status of the operation will change when the operation start criteria make the operation eligible to be started. All operations that have been manually placed on HOLD are identified by the extended status code H.

When you no longer want the operation held, you can issue the RELEASE command, MR, and the operation extended status code changes to reflect the current situation. If all start criteria for this operation are met, the operation can start immediately.

If you need to HOLD or RELEASE an operation that is not on the ready list, you can use the MCP dialog. See “Modifying an Operation” on page 62 for more details. These commands can also be entered from the ended-in-error list.
You cannot invoke the next logical status row command against an operation that has been manually held, but you can set a specific status. This does not alter the HOLD indication.

The MH command gives the operation the HOLD property. To remove this property from the operation, use the MR command. Neither the MH command nor the MR command changes the status of an operation directly. A HOLD operation is not automatically scheduled for processing.

Removing an Operation from the Current Plan, and Restoring It

If you need to remove (NOP) an operation that is already in the current plan, enter the NP command beside the row in the ready list. NP can be issued for any operation on the ready list that has status A, R, *, W, or (for computer workstations with automatic reporting only) C.

Tivoli OPC processes status changes for NOP operations until they reach a status of A, R, or * (ready). Tivoli OPC ignores time dependency, use of special resources, and other constraints. When a NOP operation reaches the ready status A, R, or *, Tivoli OPC immediately sets the operation to status C; the operation is not submitted, and successor operations are eligible to start. NOP operations are identified by the N extended status code.

Attention: Make sure that successor operations in a dependency chain are removed (NOP) before their predecessor operation. This prevents the successor operations from starting when you remove (NOP) their predecessor.

If you want to restore the operation, use the UN command. The UN command does not affect the status of the operation. It merely removes its NOP characteristics. The operation can be treated as any normal operation on the ready list.

The NP command gives to the operation the property NOP. This property can be removed from the operation by the UN command only. Neither the NP command nor the UN command changes the status of an operation directly. A NOP operation is automatically changed to completed status when it reaches ready status, regardless of whether other constraints are met.

Note: The EXECUTE command starts an operation even if it is NOP. See “Running an Operation Immediately with EXECUTE” on page 35.

Use the MCP dialog for an operation that is not on the ready list. See “Modifying an Operation” on page 62 for more details.
Running an Operation Immediately with EXECUTE

The EXECUTE command overrides normal scheduling rules except dependencies. You can use the EXECUTE command when:

- An operation is waiting for a resource that is not actually required
- You want only one job to be submitted through Tivoli OPC
- Automatic job submission is not active for all operations
- A planned shutdown of a workstation is in progress and an operation is not submitted by Tivoli OPC because the job cannot finish in time.

The EXECUTE command, EX, can be issued for an operation on a computer workstation with automatic reporting if the status of the operation is A, *, or R. The EXECUTE command causes Tivoli OPC to start the operation without regard to normal scheduling criteria. An operation that you EXECUTE will be started even when:

- Job submission is not active.
- Job options for the operation do not specify automatic submit.
- Time dependency for the operation is not satisfied.
- Required resources are not available.
- The operation has H (manual hold) extended status. It remains in held status.
- The operation has N (NOP) extended status. It remains in NOP status.

Note: If the workstation the operation is defined to is not active and there is no active alternate workstation connected, or the operation is not reroutable, the EXECUTE command is rejected and the operation status is not changed.

You can also request EXECUTE from the MCP dialog. See “Modifying an Operation” on page 62 for more details.
Diagnosing Delays

Sometimes you need to run a specific operation in the plan immediately or you need to know why a certain job is not started. Here are some reasons why a computer workstation operation may not be started:

- The workstation is not open.
- Predecessors are not complete.
- No parallel server is available.
- The workstation is offline or has failed and no rerouting is in effect.
- The workstation is active but not connected.
- Not enough workstation resources are available.
- Not all the required special resources are available.
- The operation is waiting for a specific time of day.
- The operation has been manually held.
- The automatic-job-submission option is set to NO for the operation.
- There was an error during job submission.

Many of these reasons are indicated by a unique extended status code. For example, operations that are waiting until a particular time of day will have T extended status code. If you are not familiar with the code or if there is no code, request additional information about the operation by entering row command I. See Appendix B, “Status, Error, and Reason Codes” on page 115 for a full list of error codes.

The SELECTING APPLICATION OCCURRENCE AND OPERATION INFORMATION panel in Figure 21 shows an operation that is ready but cannot be submitted because job submission is deactivated (1).

![Figure 21. EQQSOPSP—Selecting Application Occurrence and Operation Information](image_url)

The ALL DEPS option can be particularly helpful. You can use this function to find out what outstanding predecessors remain before an operation will start and to see the impact of its being late or failing.
The Query Current Plan Dialog

The Query Current Plan (QCP) dialog provides answers to your production status queries. You can request detailed or summary information on individual applications, operations, or workstations, and summary information concerning all the operations. The QCP dialog looks at the current plan, which is continuously updated as the operations are processed. You can use the QCP dialog to:

- Determine why an operation has not been started.
- Provide status information.
- Display a list of operations that have ended-in-error.
- Decide if intervention is required to speed up the processing of specific applications. You can display the applications that are most critical and those that have missed, or are close to missing, the defined deadline.
- Check information before making modifications to the current plan.
- Display a list of all dependencies for an operation. This function is of particular benefit to quickly identify which outstanding predecessors are not completed. Tivoli OPC displays up to 999 levels of dependencies.
- Determine the impact of an operation that has ended in error.

You can reach the QCP dialog from anywhere in the Tivoli OPC dialogs by entering 6 at the command prompt. This takes you to the QCP menu, displayed in Figure 22. However, you can invoke QCP functions from many places in the Tivoli OPC dialog. For example, if you enter row command I from the ready list, the QCP panel SELECTING APPLICATION OCCURRENCE AND OPERATION INFORMATION is displayed. This can save time because you do not need to leave an area of the dialog to get information.

```
EQQSTOPP -------------- CURRENT PLAN AND STATUS INQUIRY ------------------------
Option ===>
Select one of the following:
1 APPLICATIONS    - Query application occurrences
2 MOST CRITICAL    - Query most critical uncompleted application occurrences
3 OPERATIONS       - Query operations (jobs)
4 ENDED IN ERROR   - Query operations ended in error
5 WORK STATIONS    - Query work station activities
6 GENERAL          - Query general information about current plan
```

Figure 22. EQQSTOPP—Current Plan and Status Inquiry

The options available from this menu are described in the following sections.
Query Current Plan Dialog

Querying Application Occurrences
When you select option 1 (APPLICATIONS), Tivoli OPC displays a filtering panel where you can specify the selection criteria that determines the applications displayed. For example, you can list only those applications that are not complete, or applications added by the ETT function. If the selection fields are left blank, all application occurrences in the plan in status \( W, S, C, E, \) or \( U \) will be displayed. Deleted applications are displayed only when you specifically request applications in \( D \) status.

If you request additional information by using the \( S \) row command, you see the panel in Figure 23:

\[
\begin{align*}
\text{EQQSAOSP} & \quad \text{SELECTING APPLICATION OCCURRENCE INFORMATION} \\
\text{Option} & \quad \text{--- Selecting Application Occurrence Information} \\
\text{Select one of the following:} & \\
1 \text{ APPLICATION} & \quad \text{- Detailed information} \\
2 \text{ OPERATION LIST} & \quad \text{- Operations of the application occurrence} \\
3 \text{ EXTERNAL DEPS} & \quad \text{- External dependencies of the occurrence} \\
\text{Application} & \quad : \text{EID401} \quad \text{Extend EID4 by 24hrs.} \\
\text{Owner} & \quad : \text{EID} \quad \text{External Interface Dev.} \\
\text{Status} & \quad : \text{Started} \\
\text{Priority} & \quad : 5 \\
\text{Variable table} & \quad : \\
\text{Input arrival time:} & \\
\text{Planned} & \quad : 95/05/26 07.00 \\
\text{Actual} & \quad : 95/05/26 07.01 \\
\text{Group Definition} & \quad : \text{DAILYOPC}
\end{align*}
\]

Figure 23. EQQSAOSP—Selecting Application Occurrence Information

From this panel you can request detailed information about the occurrence, the operations defined in the occurrence, or the external dependencies established with the occurrence.

From the CURRENT PLAN AND STATUS INQUIRY panel, select option 2 \( \text{MOST CRITICAL} \) to display a list of uncompleted occurrences sorted by the latest start time.

The latest start time is the latest time that the operation can start in order to meet the deadline. This calculation considers the operation deadline, estimated duration, resource requirements, and successor processing. When Tivoli OPC creates the current plan, it calculates the latest start time for all the operations in a chain of dependencies, starting at the last one.
The occurrences that missed or will miss the defined deadline time are at the top of the list. You see the panel in Figure 24:

```
Row L Application S P First critical oper
cmd id text ws no. stat latest start
'''' JOB8 test var dep E 7 CPU1 029 E 08 23.55
'''' APP6 test variable dep W 7 SETP 005 A 08 23.56
'''' APP1 application 1 W 7 SETP 030 A 09 00.05
'''' APP1 application 1 W 7 SETP 030 A 09 00.06
'''' APP1 application 1 W 7 SETP 030 A 09 00.25
'''' CP current plan W 7 CPU1 050 A 09 13.00
'''' PAYDAILY daily payroll jobs W 5 WTO1 005 A 09 15.51
'''' PAYBACKP backup payroll database W 5 CPU1 015 W 10 05.54
******************************************************************************
```

**Figure 24. EQQSMC1L—Browsing Most Critical Occurrences**

The list displayed by Tivoli OPC can be scrolled left and right. You can switch between them by entering the left/right scroll commands.

If the workstation name contains **** and the displayed job name contains ******** when you scroll right, the critical path in the occurrence is complete but uncompleted operations still exist. A new critical path will be calculated for this occurrence if it is still uncompleted when the next daily plan extend or replan is run.

The shift supervisor and Tivoli OPC administrator often use this list to measure how well the production work is progressing relative to the agreed service levels.
Querying Operation Information

When you are viewing a list of operations, you can request detailed information about a particular operation by typing S beside the operation. This takes you to the SELECTING APPLICATION OCCURRENCE INFORMATION panel. You can reach the panel from the MCP, QCP, or workstation communication dialogs — from anywhere that a list of operations can be displayed. You see the panel in Figure 25:

```
Query Current Plan Dialog

Querying Operation Information

When you are viewing a list of operations, you can request detailed information about a particular operation by typing S beside the operation. This takes you to the SELECTING APPLICATION OCCURRENCE INFORMATION panel. You can reach the panel from the MCP, QCP, or workstation communication dialogs — from anywhere that a list of operations can be displayed. You see the panel in Figure 25:

```

Figure 25. EQQSOPSP—Selecting Application Occurrence and Operation Information

The ALL DEPS option shows the dependencies of an operation and helps you determine the impact of late or failed processing. After you select a dependency type, you see the panel in Figure 26:

```
Figure 25. EQQSOPSP—Selecting Application Occurrence and Operation Information

The ALL DEPS option shows the dependencies of an operation and helps you determine the impact of late or failed processing. After you select a dependency type, you see the panel in Figure 26:

```

In the list displayed in Figure 26 the immediate predecessor is defined at level 1 (1 in the Lev column) and the subsequent predecessors at level 2. You can display up to 999 levels of dependencies.
Checking the Status of a Workstation

Select option 5 WORK STATIONS to see a list of the workstations that includes the current status and reporting attribute. When you request additional summary information with the row command S, the statuses of operations at the workstation are displayed.

```
EQQSWSSP ----- BROWSING SUMMARY OF ACTIVITIES AT A WORK STATION -------------
Command ==>  

Work station     : CPU1 Computer Automatic
Type              : Computer JOB ability
Reporting form    : Automatic reporting
Work Station status : Active

Current Plan created : 95/05/29 07.04
End of planning period: 95/06/01 23.25

<table>
<thead>
<tr>
<th>Operations</th>
<th>Number</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Completed, C</td>
<td>2</td>
<td>0.40</td>
</tr>
<tr>
<td>Interrupted, I</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Started, S</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Ready, R * and A</td>
<td>10</td>
<td>1.14</td>
</tr>
<tr>
<td>Waiting, W</td>
<td>62</td>
<td>1.24</td>
</tr>
</tbody>
</table>
```

Figure 27. EQQSWSSP—Browsing Summary of Activities at a Workstation

The estimated duration is the total planned duration (HH.MM) of the operations in that status. For status W, note that the estimated duration is the time that the waiting operations are expected to run once they are started—it does not include the waiting time.

From the BROWSING WORKSTATION ACTIVITY panel, you can browse the system information for the destination of a computer workstation by entering the row command I. However, a non-local workstation can be browsed only if it is communicates through XCF, NCF, or the submit/release dataset.
Query Current Plan Dialog

Checking the Status of the Current Plan

Option 6 GENERAL provides details about the current plan, such as:

- Creation date and time
- End date and time
- When the last current plan backup was taken
- When the first event was written to the job-tracking-event log since the latest backup of the current plan
- If a new current plan is being produced
- If a new current plan has been produced and is being brought into production
- The ddname of the following resources:
  - Current plan
  - Job-tracking-event log
  - JCL repository.

You see the panel in Figure 28.

```
EQQSGCPP --------- BROWSING GENERAL CURRENT PLAN INFORMATION ---------------
Command ===>
Current plan created : 18/04/95 10.47
Planning period end : 19/04/95 23.00

Backup information:
Last CP backup : 18/04/95 10.48
First logged event after backup : 18/04/95 10.49 Time stamp: 0095108F 08460341

Daily planning status:
Under production : No
NCP ready : No

In use ddname of:
Current plan : EQQCPIDS
Job-tracking log : EQQJT03
JCL repository : EQQJSIDS
```

Figure 28. EQQSGCPP—Browsing General Current Plan Information
Chapter 4. Updating the Current Plan

This chapter describes the Modify Current Plan (MCP) dialog, which is the most important tool for operators controlling the daily Tivoli OPC processing.

Tivoli OPC schedules work according to the current plan, which is created from information in the Tivoli OPC databases. However, unplanned situations that require one-time changes to schedules constantly arise.

Situations that might require changes in plans are:

- Hardware failures
- Urgent hardware maintenance
- Last-minute changes to business schedules
- Business system reruns
- Unplanned jobs that must run immediately
- A change in work priorities
- Jobs that fail
- Data that arrives late
- Operations that run longer than planned

If you are new to Tivoli OPC, read this chapter as a guide to the MCP dialog. If you are already familiar with Tivoli OPC, use Table 5 on page 44 to find the information you need quickly.

The history function lets you rerun completed operations that are no longer in the current plan. “Rerunning Operations in the History Database” on page 63 describes how to use the history function.

Two options in the MCP dialog are large enough to need separate chapters:

- Chapter 5, “Handling Operations that End in Error” on page 73 describes the error list, which you use to handle failed operations.
- Chapter 6, “Monitoring Special Resources” on page 87 describes the Special Resource Monitor, which you use to change the status of resources and the allocation of resources.

You can handle most situations using the Modify Current Plan (MCP) dialog, but you can also perform some from the Ready List, which is described in Chapter 3, “Monitoring the Workload” on page 21.

Note: You can edit jobs with the MCP dialog (for example, using option 6), but you **must** perform job setup operations from the Ready List.
### Table 5. Using the MCP DIALOG

<table>
<thead>
<tr>
<th>If you need to ...</th>
<th>Use fast path</th>
<th>See page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diagnose delays</td>
<td>4.1 or 6.3</td>
<td>36 or 40</td>
</tr>
<tr>
<td>View the job log</td>
<td>5.2 or 5.4</td>
<td>58 or 77</td>
</tr>
<tr>
<td>Run work on request</td>
<td>5.1</td>
<td>46</td>
</tr>
<tr>
<td>Restart an application occurrence from the beginning</td>
<td>5.2</td>
<td>57</td>
</tr>
<tr>
<td>Rerun an occurrence from a specific operation</td>
<td>5.2</td>
<td>58</td>
</tr>
<tr>
<td>Delete an application occurrence</td>
<td>5.2</td>
<td>60</td>
</tr>
<tr>
<td>Change external dependencies to an occurrence</td>
<td>5.2</td>
<td>61</td>
</tr>
<tr>
<td>Rerun an operation in the history database</td>
<td>4.1, 5.3, 5.4</td>
<td>63</td>
</tr>
<tr>
<td>Change dependencies to an operation</td>
<td>5.3 or 5.2</td>
<td>61 or 51</td>
</tr>
<tr>
<td>Change the details of an operation</td>
<td>5.3 or 5.2</td>
<td>61 or 62</td>
</tr>
<tr>
<td>Set the status of an operation</td>
<td>4.1 or 5.2</td>
<td>28 or 61</td>
</tr>
<tr>
<td>Reset an operation to its previous state</td>
<td>4.1</td>
<td>29</td>
</tr>
<tr>
<td>Interrupt an operation</td>
<td>4.1</td>
<td>29</td>
</tr>
<tr>
<td>Report an operation as ended-in-error</td>
<td>4.1</td>
<td>29</td>
</tr>
<tr>
<td>Prepare jobs at a setup workstation</td>
<td>4.1</td>
<td>30</td>
</tr>
<tr>
<td>Delay an operation, and release it</td>
<td>4.1 or 5.3</td>
<td>33 or 62</td>
</tr>
<tr>
<td>Add and delete operations</td>
<td>5.2 or 5.3</td>
<td>62 or 62</td>
</tr>
<tr>
<td>Remove and restore operations with NOP and UNNOP</td>
<td>4.1 or 5.3</td>
<td>34 or 62</td>
</tr>
<tr>
<td>Handle operations that have ended in error</td>
<td>5.4</td>
<td>73</td>
</tr>
<tr>
<td>Complete an ended-in-error operation</td>
<td>5.4</td>
<td>76</td>
</tr>
<tr>
<td>Modify a job that has failed</td>
<td>5.4</td>
<td>77</td>
</tr>
<tr>
<td>Restart an ended-in-error operation</td>
<td>5.4</td>
<td>77</td>
</tr>
<tr>
<td>Get rerun or recovery instructions (view operator instructions)</td>
<td>5.4 or 4.1</td>
<td>76 or 29</td>
</tr>
<tr>
<td>Specify automatic restart for operations that fail</td>
<td>5.4</td>
<td>85</td>
</tr>
<tr>
<td>Recover when automatic restart of an operation fails</td>
<td>5.4</td>
<td>85</td>
</tr>
<tr>
<td>Control and monitor catalog management</td>
<td>5.4</td>
<td>84</td>
</tr>
<tr>
<td>Inform Tivoli OPC of unplanned changes in resources</td>
<td>5.5</td>
<td>66</td>
</tr>
<tr>
<td>Change workstation availability</td>
<td>5.5</td>
<td>67</td>
</tr>
<tr>
<td>Redirect work to an alternate workstation</td>
<td>5.5</td>
<td>68</td>
</tr>
<tr>
<td>Keep plans up-to-date</td>
<td>3.1 or 3.2</td>
<td>67</td>
</tr>
<tr>
<td>Maintain special resources</td>
<td>5.7</td>
<td>87</td>
</tr>
<tr>
<td>Complete an application occurrence</td>
<td>5.2</td>
<td>60</td>
</tr>
<tr>
<td>Handle operations with the OSEQ error code</td>
<td>5.4</td>
<td>77</td>
</tr>
<tr>
<td>Select an ended-in-error list layout</td>
<td>5.4</td>
<td>74</td>
</tr>
<tr>
<td>Create your own ended-in-error list layout</td>
<td>5.9</td>
<td>75</td>
</tr>
<tr>
<td>Run an operation immediately with EXECUTE</td>
<td>4.1 or 5.3</td>
<td>35 or 62</td>
</tr>
</tbody>
</table>
Using Fast Paths
Table 5 on page 44 shows the fast path to the correct panel in the MCP dialog. For example, if the fast path is 5.1, enter \texttt{=5.1} on the command line to reach the panel, which is the same as selecting option 5 (MCP dialog) on the Tivoli OPC main menu and selecting option 1 (ADD) on the MODIFYING THE CURRENT PLAN menu (Figure 29).

You can avoid the filter panel, if there is one, by adding .; for example, \texttt{5.3.0} instead of \texttt{5.3}. If you do this, Tivoli OPC uses the previous filter criteria.

Accessing the MCP Dialog
To enter the dialog, select the MCP option from the Tivoli OPC main menu, and the panel in Figure 29 is displayed.

![EQQMTOPP Dialog](image)

Figure 29. EQQMTOPP—Modifying the Current Plan

Specifying Selection Criteria
When you select an option from the Tivoli OPC main menu, you might see a panel where you can specify selection criteria to reduce the number of entries in the list. The selection criteria are saved between sessions, so you can bypass the panel (for example, by specifying \texttt{2.0} instead of \texttt{2} for the LIST panel) if you do not need to change the criteria.
Running Work on Request

You can add work to the plan on request using the MCP dialog. Before you can add an application occurrence to the plan, however, a description of the application must exist in the application description database. Refer to *Tivoli OPC Planning and Scheduling the Workload* for details.

If your installation frequently adds work to the plan that does not have an application description defined for it, consider using this method:

1. Create dummy model applications that match the work most commonly added. The simplest model application consists of one computer workstation operation. The job names in these model applications should be dummy names indicating that they are only models.

   **Note:** Do not specify run cycles for these applications; if you do, Tivoli OPC automatically adds occurrences for these applications to the LTP and current plan.

2. When you must add work to the plan, select the model application that best corresponds to the work you want to add using the method described in “Adding Occurrences to the Current Plan.” Change the operation details to match those of the work that you want to add. For example, you can change the job name, workstation name, and input arrival times for the operations. You can also add or delete operations, external dependencies, and internal dependencies.

Adding Occurrences to the Current Plan

To add an application occurrence to the current plan, you can:

- Select option 1 from the MODIFYING THE CURRENT PLAN panel. (Figure 29 on page 45).
- Enter =5.1 from the command prompt in any other part of the Tivoli OPC dialog.
- Use the CREATE command from the list panel (option =5.2).
Selecting Occurrences
You specify selection criteria on the ADDING APPLICATIONS TO THE CURRENT PLAN panel:

```
EQQMADDP ---------- ADDING APPLICATIONS TO THE CURRENT PLAN ------------------------
Command ==> 

Specify the information below and press ENTER to add the occurrence, or specify selection criteria to create a list of applications.

APPLICATION ID ===> PAYEMDMP

Input arrival:
DATE ===> 95/04/10 Date in format YY/MM/DD
TIME ===> _____ Time in format HH.MM

Deadline:
DATE ===> _____ Date in format YY/MM/DD
TIME ===> _____ Time in format HH.MM

PRIORITY ===> _ 1-9
ERROR CODE ===> ____ If rerun of occurrence

AUTOMATIC DEP ===> Y Automatic dependency add, Y P S or N
RESOLVE REQUIRED ===> N Auto deps must be resolved, Y or N
GROUP DEFINITION ===> ________________ Group definition filter
```

Figure 30. EQQMADDP—Adding Applications to the Current Plan

These fields can have a significant effect on the added occurrences:
- ERROR CODE; see “Specifying an Error Code” on page 49
- RESOLVE REQUIRED and AUTOMATIC DEP; see “Including Dependencies Defined in the Database” on page 50
- GROUP DEFINITION; see “Grouping Occurrences” on page 54.

Values specified on this panel are carried forward to the ADDING AN APPLICATION TO THE CURRENT PLAN panel (Figure 32 on page 48) for each application that you select. **Input arrival date** always contains the current date. These fields contain the values from your last MCP add:

- APPLICATION ID
- AUTOMATIC DEP
- RESOLVE REQUIRED

You might find it faster to specify only the application ID, group definition, and dependency indicators. If you do not specify input arrival or deadline times, Tivoli OPC extracts them from the run cycles defined to the application in the application description database (if any run cycles exist). These fields will be filled in when you get to the ADDING AN APPLICATION TO THE CURRENT PLAN panel (see “Adding Occurrences” on page 48).
When you need to add multiple occurrences, or if you do not remember the name of the application, leave the APPLICATION ID field blank, or specify a generic application ID such as PAY* to generate a list of applications.

Figure 31. EQQMAADL—Selecting Applications to Add to the CP

Select the single occurrences that you want to add from this list with the row command A. If you want to add part or all of the application group that an application is a member of, select option G. Adding application groups to the current plan is described in detail in “Adding an Application Group to the Current Plan” on page 54.

Adding Occurrences
When you select an individual application you want to add to the current plan, Tivoli OPC displays this panel:

Figure 32. EQQMAOCP—Adding an Application to the Current Plan

Any values specified on the ADDING APPLICATIONS TO THE CURRENT PLAN panel (Figure 30 on page 47) are carried forward to this panel. The priority is extracted
Modify Current Plan (MCP) Dialog

from the application description. The panel indicates the number of operations in the occurrence and the number of external predecessors.

Tivoli OPC does not accept any command, except CANCEL, until the input arrival and deadline dates and times are specified. If you want the input arrival time to be the current time, you can either specify the necessary time or press Enter. Tivoli OPC uses the current time as default for the input arrival time. If you press Enter again, Tivoli OPC uses the current date as default for the deadline date.

You can use the ISPF command delimiter to set up a chain command to do this quickly. For example, if you type ;;; and press Enter, the panel is redisplayed with the cursor at the deadline time field. All other date and time fields are set by Tivoli OPC.

Note: If the added application has a run cycle defined, the input arrival and deadline time is taken from the first run cycle description. If you try to add an application with the same date and time as another occurrence of the same application already in the plan (even if deleted or completed), Tivoli OPC rejects it. If you mean to do this, change the time by one minute until the added occurrence is unique. But be careful if you want Tivoli OPC to resolve external dependencies: external dependencies can depend on the input arrival time of the added occurrence. For example, if application B depends on application A, what happens if you add an extra occurrence of A and B? When you add A, you must give its occurrence a different input arrival time to the regular occurrence. When you add B, you give its occurrence the same or later input arrival time, and it should become dependent on the added occurrence of A, because that is the closest occurrence with an equal or earlier input arrival time. See “Including Dependencies Defined in the Database” on page 50 for a description of how dependencies are resolved when there are several candidates. But you can have problems if the operations in A and B have explicit input arrival times—these are not affected by the occurrence input arrival time that you specify on the ADDING AN APPLICATION TO THE CURRENT PLAN panel, and you should alter these dependencies manually. See “Changing External Dependencies to an Occurrence” on page 61 for details.

You can use the DEP and OPER commands to alter dependencies and operation details for this occurrence.

When you have specified all the occurrence information, add it to the current plan by issuing the END command.

Specifying an Error Code
ERROR CODE is an optional field containing a user-defined value up to 4 characters. This value is included in the daily planning reports for the occurrence when it completes. You can use this to identify particular categories of added processing.

For example, you could have an installation standard that all report reprints are added to the current plan with an error code of REPT. Similarly, occurrences added as part of a business system rerun could have an error code RER.

The specification of an error code does not affect the submission or tracking of operations, but it can be a useful tool for measuring the unplanned workload on your system.
You can use the error code to identify the name or the department of the user who requested the processing. This can be helpful when you are converting regularly added occurrences to an automated method by using ETT or PIF.

**Including Dependencies Defined in the Database**

When you add application occurrences to the current plan, decide if the occurrence or group should be added with the dependencies defined in the application database. With the AUTOMATIC DEP field, you can:

- Add both predecessor and successor dependencies.
- Add only predecessor dependencies.
- Add only successor dependencies.
- Ignore any dependencies specified in the application description.

When you request that dependencies be considered for individual occurrences, you can also specify if the dependencies must be resolved. If you specify RESOLVE REQUIRED – Y, Tivoli OPC issues a message if any dependency defined in the application cannot be resolved. If this happens, Tivoli OPC does not accept the application occurrence into the current plan until you enter DEP and delete the unresolved dependency.

If predecessors are added to an application in the AD database, this does not take effect in the current plan until that application is scheduled by daily planning or manually added to the current plan.

You can also delete external dependencies or specify additional dependencies when you are adding an occurrence—see “Changing and Adding Dependencies” on page 51.

Successor dependencies are added automatically only if the dependency is defined in the applications database at the time the successor occurrence was added to the current plan. When an occurrence is added to the plan, either manually using the MCP dialog or automatically scheduled by daily planning, Tivoli OPC creates a potential predecessor record for predecessors that are not in the current plan. This can happen if the predecessor (job A) is run on demand. When its successor (job B) is added to the plan (at daily planning), Tivoli OPC notices that its predecessor (A) is missing. When you add the on-demand application (A), and specify that Tivoli OPC is to resolve successor dependencies, Tivoli OPC looks for predecessors that match, and the added occurrence A becomes a predecessor to B, unless B has started or has already completed.

If there are several occurrences of B in the plan, Tivoli OPC must choose which occurrence of B to make the successor. The best successor is the first not-started operation that has the correct application name, the correct operation number, and an occurrence input arrival later than or equal to that of the occurrence being added.

For each missing predecessor, Tivoli OPC can keep track of up to 1000 successors. If this limit is reached, no more successors are added and a warning message is issued.

If you add an occurrence to the current plan from the MCP dialog, and later run a daily planning EXTEND or REPLAN job, daily planning never makes the manually added occurrence a predecessor or successor of an occurrence that it adds from the long-term plan, even if the dependency is specified in the database. This is
because daily planning does not search the current plan to resolve dependencies—it takes the dependencies straight from the long-term plan. For example:

1. You add an occurrence of PAYDAILY using the MCP dialog. PAYDAILY is specified in the database as a predecessor of PAYBACKP.
2. You extend the current plan, which adds an occurrence of PAYBACKP.
3. Even though the application description for PAYBACKP states that it has an external predecessor PAYDAILY, PAYBACKP will not have this dependency, because occurrence dependencies are not added when the current plan is extended unless they were present in the LTP.

When you specify resolution of predecessor dependencies, Tivoli OPC looks for occurrences that are predecessors to the added operations, but this is different:

- The predecessor operation can be completed (or started).
- If the added operation has an explicit input arrival time, Tivoli OPC takes the predecessor occurrence with the input arrival time earlier and closest to, or the same as, the input arrival time of the added operation.
- If the added operation has no explicit input arrival time, Tivoli OPC takes the predecessor occurrence with the input arrival time earlier and closest to, or the same as, the input arrival time of the added occurrence, as specified on the ADDING AN APPLICATION TO THE CURRENT PLAN panel in Figure 32 on page 48.

**Note:** When resolving a dependency, Tivoli OPC uses the input arrival time of the predecessor *occurrence*, not the input arrival time of the operation specified in the dependency.

**Changing and Adding Dependencies**

When you add an occurrence, you can make the operations dependent on operations in the plan, even though these dependencies are not defined in the database.
Modify Current Plan (MCP) Dialog

On the ADDING AN APPLICATION TO THE CURRENT PLAN panel, shown in Figure 32 on page 48, enter the OPER command. Tivoli OPC displays this panel:

```
EQQMOPL -------- MODIFYING OPERATIONS IN THE CURRENT PLAN - ROW 1 TO 1 OF 1
Command ===> Scroll ===> PAGE

Enter the GRAPH command above to view operations graphically or change data in the rows, and/or enter any of the following row commands:
I(nn) - Insert, R(nn),RR(nn) - Repeat, D(nn),DD - Delete
J - Edit JCL, O - Browse operator instructions, S - Modify operation details
A - Catalog Management actions, L - Browse joblog

Application : PAYQUERY AD-HOC payroll query
Owner : SAMPLE payroll application
Input arrival : 95/07/28 14.11
Status : Being added

Row Operation Jobname PS Dura- Opt Ext Res Status
cmd ws no. text tion S T S/P S R1 R2 New Cu
```

Figure 33. EQQMOPL—Modifying Operations in the Current Plan

Select the operation. Tivoli OPC displays this panel:

```
EQQMODP -------- MODIFYING AN OPERATION IN THE CURRENT PLAN ------------
Option ===> 

Select one of the following:
1 DEPENDENCIES - Delete and add (internal and external)
2 SPEC RESOURCES - Special resources
3 AUTOMATIC OPTIONS - Job, WTO, and print options
4 TIME - Time specifications
5 JCL - Edit JCL for MVS job
6 GENERAL - General information

Application : PAYQUERY AD-HOC payroll query
Input arrival : 95/07/28 14.11
Operation : CPU1 050 run as required
Duration : 0.05
Jobname : PAYQUERY
External predecessors : No
External successors : No
Special resources : Yes

```

Figure 34. EQQMODP—Modifying an Operation in the Current Plan
Select option 1 (DEPENDENCIES). You see the panel in Figure 35.

![EQQMAADP panel](image)

**Figure 35. EQQMAADP—Creating a Dependency in the Current Plan**

You can specify the details of the predecessor or successor on the CREATING A DEPENDENCY IN THE CURRENT PLAN panel, but it is easier to use a generic search character in the ID (P* in Figure 35) and press Enter.

You see the panel in Figure 36.

![EQQMMDL panel](image)

**Figure 36. EQQMMDL—Defining Dependencies in the Current Plan**

This is especially useful when you have many operations with the same name. Use these row commands to specify the dependencies:

- **P** Makes the operation in the row a predecessor of the operation that you are adding.
Modify Current Plan (MCP) Dialog

**S**  Makes the operation in the row a successor of the operation that you are adding.

**blank**  Removes the dependency between the operation in the row and the one that you are adding.

**Note:** Changes that result in a conflicting status (such as specifying that an operation is a predecessor of a completed operation) are rejected.

**Grouping Occurrences**
When you add occurrences to the current plan, you can specify that the occurrence should become a member of an existing occurrence group or that a new occurrence group should be created. Related occurrences can be handled as a single entity when they are defined in an occurrence group. An occurrence group, and therefore all member occurrences, can be completed or deleted from the current plan using a single dialog request.

The daily planning process automatically creates occurrence groups in the current plan if such a group exists in the long-term plan for the corresponding period.

When an occurrence is defined as a group member, it cannot be individually completed or deleted. Group members must be removed from the group if individual complete or delete is required.

**Adding an Application Group to the Current Plan**
If you enter the G row command on the SELECTING APPLICATIONS TO ADD TO THE CP panel beside an application that belongs to a group, you can add all or part of that group to the current plan.

---

**Figure 37. EQQMAAGL—Adding an Occurrence Group to the CP**
Excluding Some Applications
When Tivoli OPC displays the ADDING AN OCCURRENCE GROUP TO THE CP panel, you see a list of all applications that are members of the group you have selected. From the list, you can exclude specific applications defined in the group. On this panel, you enter the input arrival and deadline time that will be used for all occurrences in the group. Note that only the applications that are valid for the input arrival date and time that you specify are contained in the list. If an application that is normally part of the group is not valid for the specified input arrival date, it is not listed as part of the group.

Specifying Dependency Resolution
You can specify whether dependencies should be resolved and if all occurrences in the group should be added using the same priority or JCL variable table. When you request automatic dependency resolution for an application group, OPC/ESA will first attempt to establish any dependencies to other applications within the group that is being added.

The applications in an application group are handled as a single entity, but when the group is added to the current plan, the occurrences are generated one at a time. If Tivoli OPC is abnormally terminated during such a transaction, the occurrences already added to the current plan will not be backed out, and the other occurrences will not be automatically added when Tivoli OPC is restarted. The order that Tivoli OPC creates the occurrences depends both on the application name and the structure of the dependencies between the applications. The order does not affect how the dependencies are created.

When an occurrence group is added, Tivoli OPC resolves dependencies only within the group if you specify option G on the AUTOMATIC DEP field in Figure 37 on page 54. For all other options, dependencies will be resolved as normal for each added occurrence, with the exception that dependencies are always resolved within the group if possible.

Before adding any occurrence, Tivoli OPC checks for dependency loops in the occurrence group. If it detects a loop, one of the dependencies forming the loop will not be resolved within the group. However, if there is another occurrence, external to the group, that matches the dependency criteria, the dependency will be resolved to this external occurrence if option Y or P is specified in the AUTOMATIC DEP field. A dialog warning message is issued.

If an application defines a dependency that cannot be satisfied within the group, OPC/ESA will attempt to establish the dependency to another occurrence, outside the group, in the current plan. You can see the dependencies established for occurrences created by adding the application group by entering the GRAPH command on the MODIFYING OCCURRENCES ADDED TO THE CURRENT PLAN panel (Figure 38 on page 56).

When you have entered the required data and deleted any rows that you do not want to include in the group, enter END or press PF3. The applications required in the group are added to the plan. Initially, these occurrences are held by Tivoli OPC to let you modify the operation data or modify dependencies for any occurrences in the group. If you do not want to proceed with the group add, enter CANCEL on the ADDING AN OCCURRENCE GROUP TO THE CP panel (Figure 37 on page 54).
When the occurrences are added to the plan, Tivoli OPC displays this panel, which lists only those occurrences added as a result of adding the group:

```
EOQAMOPL ---- MODIFYING OCCURRENCES ADDED TO THE CURRENT PLAN
Command ===> Scroll ===> PAGE

Enter the DELETE command to delete all occurrences and exit, or
Enter the GRAPH command to display occurrence list graphically or
enter any of the row commands below:
B - Browse, D - Delete, M - Modify, RG - Remove from group,
C - Complete, W - Set to Waiting, R - Rerun

Row Application Input arrival S P G
cmd id text date time
'PAYM1' MONTHLY PAYROLL JOBS 95/03/14 12.00 W 5 Y
'PAYM2' MONTHLY PAYROLL TRANSFER 95/03/14 12.00 W 5 Y
```

Figure 38. EOQAMOPL—Modifying Occurrences Added to the Current Plan

From the MODIFYING OCCURRENCES ADDED TO THE CURRENT PLAN panel, you have access to the complete range of MCP services. At this point, the occurrences are still in a hold state; if you enter CANCEL from this panel, the occurrences remain held. The occurrences are already added to the plan, but you can mark them deleted with the DELETE command. If you do this, you cannot add these occurrences later with exactly the same input arrival date and time (because the occurrences, even though marked deleted, are still in the plan). If you need to add them again later, change the arrival time slightly.

After making any modifications, enter END or press PF3 to release the occurrences. If your conversation with the Tivoli OPC subsystem is cancelled for any reason while the occurrences are still in a hold state, they remain held.

Note: While you are using the panel in Figure 38, nobody else can access the listed occurrences. Release the lock as soon as you can.
Restarting an Occurrence from the Beginning

The LIST option of the MCP dialog takes you to the MODIFYING OCCURRENCES IN THE CURRENT PLAN panel:

```
Command ===> Scroll ===> PAGE
```

Enter the CREATE command to add a new occurrence or enter the GRAPH command to display occurrence list graphically or enter any of the row commands below:

- B - Browse, D - Delete, M - Modify, RG - Remove from group, DG - Delete Group,
- C - Complete, W - Set to Waiting, R - Rerun, CG - Complete Group

```
Row Application Input arrival S P G Add
  CP current plan 95/06/08 11.00 C 7 N
  PAYBACKP backup payroll database 95/06/08 12.00 W 5 N
  PAYDAILY daily payroll jobs 95/06/08 12.00 W 5 N
  PAYW weekly payroll jobs 95/06/09 14.00 W 5 Y D
```

Figure 39. EQQMOCLL—Modifying Occurrences in the Current Plan

To restart an application occurrence from the beginning, enter W in the row command field for the occurrence. After you confirm the request, Tivoli OPC resets the status of all operations in the occurrence to W (waiting). All operations (whose predecessors have completed) are then automatically resubmitted if all other criteria for submission are fulfilled. In this process, all external successors that have a status of ready are reset to W (waiting) status.

If an external successor has started, however, the request to restart is rejected, and an error message is displayed. If this happens, you can delete the external successor dependency and then reset the occurrence to waiting status. See “Changing Dependencies to an Operation” on page 61 for details on how to delete dependencies.

If you want to reset the status of successors as well, you need RERUN instead of SET TO WAITING. See “Rerunning an Occurrence in the Current Plan from a Specific Operation” on page 58.

Notes:

1. Row commands are usually executed immediately. Thus, if you cancel the rerun against which you have issued a NOP command, the operation remains removed (no-oped), because the NOP command has already been processed.

2. If at some point an operation was set to HOLD or NOP, the HOLD or NOP status will stay in effect when the occurrence is set to waiting. To see which operations have been set to HOLD or NOP, request a list of the operations using option 5.3.

3. You may need to HOLD or NOP operations that will be reset by the command.

4. When an operation is rerun, Tivoli OPC uses the job that was last submitted for the operation.
Rerunning an Occurrence in the Current Plan from a Specific Operation

If an occurrence is still in the current plan, you can rerun it from a particular operation by selecting the occurrence with the row command R on the MODIFYING OCCURRENCES IN THE CURRENT PLAN panel (Figure 39 on page 57). The RERUNNING AN OCCURRENCE IN THE CURRENT PLAN panel (Figure 40) is displayed, where you can specify which operation in the occurrence you want to restart from.

**Note:** If an operation ends in error, always use the error list to rerun it.

![EQQMROCL —— Rerunning an Occurrence in the Current Plan](image)

If there are external dependencies defined to the operation, they are included on the LIST DEPENDENCY STATUS CHANGE panel (Figure 41 on page 59) that is displayed automatically. From the RERUNNING AN OCCURRENCE IN THE CURRENT PLAN panel, you can modify the jobs of the dependent operations before Tivoli OPC resets them. The modified job is saved in the JCL repository when you exit and confirm the occurrence rerun.

You can use RERUN to reset dependent occurrences that have already completed. This can save you the trouble of adding the occurrences again if you need to rerun all or part of a business system and occurrences have not been planned out of the current plan. You select the point from which you must start the rerun and all successor dependencies are automatically set to waiting status.

**Note:** The real status of the operations is still unchanged. Whether any further row command will be allowed depends on the real status, rather than on the status currently displayed. The effective rerun is executed only after confirmation on panel EQQMOSTL.

In cases where an operation changes from C (complete) status to either R (ready) or W (waiting) status, you must decide if catalog-management actions should be performed to clean up the datasets for a complete rerun. If catalog-management actions are required, you can initiate the action from this panel. When you request catalog-management actions and job log information is available, you can choose...
to restart the operation from a particular step, and Tivoli OPC will automatically tailor and verify the job for restart.

If the catalog-management indicator is set to D and dataset information exists, you must either initiate the catalog-management action or discard the action. If you do not initiate or discard the action, the operation is not resubmitted by Tivoli OPC. Instead, it remains on the ready list with an extended status of C, indicating that catalog-management actions are outstanding.

If the operation you select as a restart point is not ready to start, Tivoli OPC sets all predecessor operations to C (complete) status. This can be useful if you need to complete a chain of dependent occurrences that are not defined as a group. This situation is illustrated in Figure 41.

The LIST DEPENDENCY STATUS CHANGE panel shows which operations will have their status changed to C (Complete). Check this list carefully to be sure that you really want to bypass these operations.

```
Figure 41. EQQMOSTL—List Dependency Status Change

Notes:

1. Row commands are usually executed immediately. Thus, if you cancel the rerun against which you have issued a NOP command, the operation remains removed (no-oped), because the NOP command has already been processed.

2. If at some point an operation was set to HOLD or NOP, the HOLD or NOP status will stay in effect when the occurrence is set to rerun. To see which operations have been set to HOLD or NOP, request a list of the operations using option 5.3.

3. You may need to HOLD or NOP operations that will be reset by this command.

4. When an operation is rerun, Tivoli OPC uses the job that was last submitted for the operation.
```
Modifying Current Plan (MCP) Dialog

Completing an Application Occurrence

When you want to complete the entire application occurrence, enter C in the row command field for the occurrence on the MODIFYING OCCURRENCES IN THE PLAN panel. Tivoli OPC presents a confirmation panel that includes a list of the dependent operations to the occurrence for which you have requested completion. The dependencies listed are also considered complete if you enter Y. From this panel, you can request that Tivoli OPC HOLD or NOP a dependent operation. After your confirmation, Tivoli OPC sets all operations to C (complete) status.

**Note:** If you set the status of an application occurrence to complete, Tivoli OPC can submit its successor operations.

Deleting an Application Occurrence

You can delete an application occurrence from the current plan by entering D in the row command field for the occurrence in the MODIFYING OCCURRENCES IN THE CURRENT PLAN panel. Tivoli OPC then displays a confirmation panel showing all external dependencies for the occurrence. After your confirmation, all operations in the occurrence will be deleted.

Then, for each operation in the deleted occurrence, Tivoli OPC automatically connects all its external predecessors to its external successors. By doing this, Tivoli OPC maintains the consistency of the network of operations.

To prevent unwanted dependent operations from starting after the deletion, first change the external dependencies, as described under “Changing External Dependencies to an Occurrence” on page 61. Alternatively, you can HOLD or NOP a dependent operation from the list displayed.

**Note:** Tivoli OPC retains the deleted occurrences for reporting purposes and to ensure that the LTP is updated if the occurrence exists in the LTP. So you cannot add another occurrence of the same application to the current plan with the same input arrival time as the deleted occurrence. You can display deleted occurrences using the Query Current Plan dialog.

Modifying an Application Occurrence

Requests to modify application occurrences in the plan might include requests to change:

- Input arrival and deadline times
- Priority
- Operation information
- JCL variable table used by the occurrence operations
- External and internal dependencies.

To change an occurrence, enter M in the row command field for the occurrence in the MODIFYING OCCURRENCES IN THE CURRENT PLAN panel (Figure 39 on page 57). You can change occurrence times and priority on the resulting MODIFYING AN OCCURRENCE IN THE CURRENT PLAN panel.

You can also change operations (see “Changing the Details of an Operation” on page 61 for detailed information) and display the list of external dependencies for verification and deletion. See “Changing External Dependencies to an Occurrence” on page 61 for detailed information.
Changing External Dependencies to an Occurrence
You can display all external dependencies to an occurrence by entering the DEP primary command on the MODIFYING AN OCCURRENCE IN THE CURRENT PLAN panel. This takes you to the RESOLVING EXTERNAL DEPENDENCIES IN THE CP panel, where you can verify or delete the dependencies. You cannot add external dependencies from this panel. You can add external dependencies only by modifying individual operations.

Changing Dependencies to an Operation
To change dependencies to an operation, get the list of operations by entering OPER in the command line of the MODIFYING AN OCCURRENCE IN THE CURRENT PLAN panel. In the list of operations, enter S in the row command field to get details of the operation. Select option 1 (DEPENDENCIES) on the MODIFYING AN OPERATION IN THE CURRENT PLAN, as described in “Changing and Adding Dependencies” on page 51. You can then delete dependencies and create new ones. You can handle both internal and external dependencies.

Changing the Details of an Operation
To change the details of an operation, display the list of operations by entering OPER in the command line of the MODIFYING AN OCCURRENCE IN THE PLAN panel. In the list that follows (on the MODIFYING OPERATIONS IN THE PLAN panel), you can modify items such as:

- Workstation name
- Job name
- Operation text
- Use of workstation resources
- Submit options
- Time options
- Operation status

To make other changes, enter S in the row command field to display more details of the operation. This takes you to the MODIFYING AN OPERATION IN THE PLAN panel, where you can select several alternatives. For example, you can edit the job with the ISPF/PDF editor by entering J in the row command field of an operation.

Notes:
1. Saved changes to the job take effect even if you leave the MCP dialog with CANCEL.
2. Changes that would force a status change of a started or completed dependent operation are rejected.
3. PF3 is the confirmation of a modify request. By entering PF3, the resulting 'operation values' comprise the changes specified together with the existing values that have not been changed. If a modify request is entered and PF3 is issued without any change, the operation values are set to the existing values (for example, if the submit option was set to Y then the operation is submitted). If a modify request is enter by mistake, then use CANCEL to discard the request.
Modifying Current Plan (MCP) Dialog

Adding and Deleting Operations
To add or delete operations, request a list of operations by entering OPER in the command line of the MODIFYING AN OCCURRENCE IN THE PLAN panel. In the list, you can enter these row commands:

- **I** Inserts a blank row for a new operation
- **R** Copies existing operations to be added as new
- **D** Deletes operations.

When you generate a new row with the insert or repeat row command, the operation number is initially set to 000. This shows that the operation is not yet defined, and that you must assign a unique number to the operation. By default, the new operation is set up as a successor to the operation preceding it in the list.

When an operation is repeated with the repeat row command, all information except the operation number is copied to the new operation.

When an operation is deleted, Tivoli OPC automatically connects all its predecessors with its successors to maintain the consistency of the network of operations. You cannot delete an operation whose deletion would split the operation network of the application occurrence into several operation networks. If you need to delete an operation that would do this, you must first change the dependencies in the application so that the network will not be split.

Modifying an Operation
Modifications to single operations are more easily handled by selecting option 3 (OPERATIONS) from the MODIFYING THE CURRENT PLAN panel (fast path 5.3).

When you have specified the required list criteria, you see the MODIFYING OPERATIONS IN THE CURRENT PLAN panel:

<table>
<thead>
<tr>
<th>Row</th>
<th>Application id</th>
<th>Operation</th>
<th>Jobname</th>
<th>Input</th>
<th>Arrival</th>
<th>Dura-</th>
<th>Opt</th>
<th>Depen</th>
<th>S</th>
<th>Oper</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CP</td>
<td>CPU</td>
<td>050</td>
<td>XRAYNERC</td>
<td>95/04/28</td>
<td>12:00</td>
<td>3.00</td>
<td>Y</td>
<td>Y</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>MAWS</td>
<td>CPU</td>
<td>010</td>
<td>MAWS</td>
<td>95/04/27</td>
<td>16:28</td>
<td>0.01</td>
<td>Y</td>
<td>N</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>PAYBACKP</td>
<td>CPU</td>
<td>015</td>
<td>PAYBACKP</td>
<td>95/04/28</td>
<td>12:00</td>
<td>0.05</td>
<td>Y</td>
<td>N</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>PAYBACKP</td>
<td>WTO1</td>
<td>030</td>
<td>PAYBACKP</td>
<td>95/04/28</td>
<td>12:00</td>
<td>0.01</td>
<td>Y</td>
<td>N</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>PAYDAILY</td>
<td>WTO1</td>
<td>005</td>
<td>PAYDAILY</td>
<td>95/04/28</td>
<td>12:00</td>
<td>0.01</td>
<td>Y</td>
<td>Y</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>PAYDAILY</td>
<td>SETP</td>
<td>010</td>
<td>PAYDAILY</td>
<td>95/04/28</td>
<td>12:00</td>
<td>0.03</td>
<td>Y</td>
<td>N</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>PAYDAILY</td>
<td>CPU1</td>
<td>020</td>
<td>PAYDAILY</td>
<td>95/04/26</td>
<td>12:00</td>
<td>0.05</td>
<td>Y</td>
<td>N</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>PAYDAILY</td>
<td>CPU2</td>
<td>020</td>
<td>PAYDAILY</td>
<td>95/04/27</td>
<td>12:00</td>
<td>0.05</td>
<td>Y</td>
<td>N</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>PAYDAILY</td>
<td>CPU1</td>
<td>020</td>
<td>PAYDAILY</td>
<td>95/04/28</td>
<td>12:00</td>
<td>0.05</td>
<td>Y</td>
<td>N</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>PAYW</td>
<td>CPU1</td>
<td>030</td>
<td>PAYWSLIP</td>
<td>95/04/27</td>
<td>12:00</td>
<td>0.05</td>
<td>Y</td>
<td>N</td>
<td>1</td>
</tr>
<tr>
<td>11</td>
<td>PAYW</td>
<td>PAY1</td>
<td>095</td>
<td>PAYWSLIP</td>
<td>95/04/27</td>
<td>12:00</td>
<td>1.00</td>
<td>Y</td>
<td>N</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 42. EQQMOPRL—Modifying Operations in the Current Plan
This panel lists the operations in the current plan that fulfill the list criteria you specified. Using row commands, you can perform the full range of MCP services on these operations.

The HIST primary command on this panel lets you perform actions on operations that are no longer in the current plan.

---

**Rerunning Operations in the History Database**

You might want to rerun an operation that has completed and is not part of the current plan. To do this, you use the history function.

The history function stores operation data in the history database, a DB2 database that contains operations from the most recently completed occurrences from a previous current plan. The HIST command lets you select operations from the history database for further processing.

**Note:** To rerun an old operation, Tivoli OPC must be able to retrieve the JES JOBLOG, either from the JES spool, an archive database (via user exit EQQUX010), or from the Tivoli OPC JS VSAM database. If the JOBLOG has previously been retrieved into the JS file, and then deleted (either via a PIF cleanup utility or automatically by Tivoli OPC because another occurrence of the same application has been scheduled and completed), then it is not possible to restart the JOBSTEP operation using Tivoli OPC.

---

**Updating the History Database**

If you do not use the history function, the daily plan process creates a current plan that includes new occurrences as well as the noncompleted occurrences from the old current plan. However, completed occurrences from the old plan do not appear in the new one. To save information about completed occurrences, you must use the history function.

When the history function is active, the daily plan process stores information about completed occurrences in the history database when it extends the current plan. Each time a new current plan is created, the history database is updated with completed operations that are defined on computer workstations. The history update stores information about the most recently completed occurrence of an application unless the RETAINOPER keyword is specified in the BATCHOPT statement. If the RETAINOPER keyword is specified, the value of the keyword determines how many calendar days the operation is kept in the database before being removed or replaced by a newer occurrence of the operation. If RETAINOPER is not specified, the operation is kept in the history database indefinitely or until it is replaced by a more recent occurrence.

Before you can use the history function, a DB2 database must be installed with OPC/ESA table definitions and the OPERHISTORY and DB2SYSTEM keywords must have been defined on the BATCHOPT and OPCOPTS statements for your subsystem. This is described in *Installation Guide* and *Customization and Tuning*. 
Processing History Operations

You can display a list of the history operations by selecting option 7 (Old Operations) on the main OPC/ESA menu, or by entering the HIST command on any of these panels:

- MODIFYING OPERATIONS IN THE CURRENT PLAN (5.3)
- HANDLING OPERATIONS ENDED IN ERROR (5.4)
- READY LIST (4.1).

After issuing the HIST command and specifying list criteria, this panel is displayed:

```
Figure 43. EQQHISTL—Operations History List
```
On the OPERATIONS HISTORY LIST panel, you see the row commands that can be issued against the operations. Two row commands, B and J, can be issued before you select an operation.

B lets you browse the operation details. If you issue the command before selecting the operation, the details you see will be from the history database. If you have selected the operation and changed input arrival time before browsing, the details will show the new input arrival time.

J displays the job for editing. If it exists in the JS dataset, the job you edit will be the one used when the occurrence was most recently run. If this copy does not exist, the job will be taken from the job library dataset.

**Note:** If any Tivoli OPC or system variables are used in the JCL, the job submitted by the RESTART processing may be very different from the one originally submitted.

### Selecting a History Operation

When you select an operation, you add it to the current plan and make it available for processing with additional row commands.

You select a history operation with the S row command. Selecting an operation displays this panel, where you can change the input arrival or deadline time:

![Figure 44. EQQHIPUP—Specifying Occurrence Arrival](image)

After you have selected an operation and specified data on the SPECIFYING OCCURRENCE INPUT ARRIVAL panel, you return to the previous panel. The input arrival time has been updated to reflect the time you specified, and the operation has been added to the current plan with the status AH (arriving, manually held).
Modify Current Plan (MCP) Dialog

You should note that no dependency information is stored in the history database. Operations that are added will not have any predecessors or successors even though these might be defined in the application description. Use the M command to add dependencies after you have selected an operation.

Select an operation before performing any of these row commands:

- **DS**  The DS command deletes the operation from the current plan.
- **JR**  Perform a job level restart and if catalog management is active, browse or modify the datasets for the operation. The job is released when you leave the OPERATIONS HISTORY LIST with PF3.
- **L**  Browse the job log. The command retrieves and displays the job log if it exists for the selected operation.
- **M**  This command is the same as the M command on the MODIFYING OPERATIONS IN THE CURRENT PLAN. It lets you change operation details. If you want to run the history operations that you add to the current plan in any certain order, you must add the dependencies here.
- **SR**  Retrieves the job log for the operation. and lets you select a step to begin from when rerunning. The job is released when you leave the OPERATIONS HISTORY LIST with PF3.

Informing Tivoli OPC of Unplanned Changes in Resources

Sometimes a vital resource suddenly becomes unavailable because:

- A hardware error occurs on a tape or disk drive
- A processor is down
- A link is down
- Important staff are not available
- Resources must suddenly be reallocated.

When a vital resource becomes unavailable, you usually change the workstation definitions to prevent Tivoli OPC from starting operations in the wrong order or producing inaccurate schedules. If the resource is a special resource, refer to Chapter 6, “Monitoring Special Resources” on page 87 for details of the Resource Monitor dialog. But to change workstation parallel servers and the workstation fixed (R1 and R2) resources, use the MCP dialog and select the WORK STATIONS option to display a list of all workstations. Enter the S row command beside the workstation you want to select. You can then change the reporting attributes and the open interval information for the workstation. You can also change the alternate workstation for a workstation in this dialog.

Any changes to open intervals that you make here remain in effect until they elapse or until you change them again in the dialog. These changes always override the information calculated from the workstation descriptions when you extend the current plan.

**Note:** If you assign no parallel servers to an interval and specify that Tivoli OPC should control on servers for the workstation, Tivoli OPC will not submit any jobs for that workstation during the interval.
Keeping Plans Up-to-Date

When you have changed the current plan, or when the duration of some work has varied significantly from its schedule, the planned times in the current schedule might be inaccurate. To make Tivoli OPC recalculate these times, taking the current information into account, replan the current plan. You can do this in two ways:

1. Enter the Replan dialog from the Daily Planning dialog (select the REPLAN option on the PRODUCING TIVOLI OPC DAILY PLANS panel). Tivoli OPC will then submit the replan job.

2. Define the replan job as an application in Tivoli OPC so you can schedule the replan as you would any other application, and Tivoli OPC will submit it automatically.

The REPLAN job removes all completed occurrences from the current plan dialogs.

Changing Workstation Availability

Before Tivoli OPC can start an operation, the workstation that it runs on must be available (open and active). If control on servers has been defined for the workstation, servers must be available, too. Select option 5.5 from the Tivoli OPC main menu to specify the times that a workstation is open for work.

Active and Inactive Computer Workstations

To run work, a computer workstation must be active and open. A workstation is active if the Tivoli OPC controller can communicate with it. An active computer workstation can be open or closed. A closed computer workstation is not eligible to have work scheduled on it even if the controller can communicate with the workstation. Workstation status can be changed dynamically (either manually, using the Tivoli OPC dialogs, or automatically, in response to changes in your systems).

An inactive workstation can have one of the following statuses:

**FAILED** The operating system has detected a failure on the system that the workstation is defined on. The workstation is automatically set to status ACTIVE when the system is available again. You can manually set a computer workstation to status FAILED, using the Tivoli OPC dialogs. The WSSTAT command or the EQQUSIN subroutine can be used to report FAILED status for a workstation.

**OFFLINE** Communication is lost between the Tivoli OPC controller and Tivoli OPC tracker on the system that the workstation represents. This might be because the tracker is not yet started or because it ended abnormally. Tivoli OPC does not take any reroute or restart actions until the time specified for the OFFDELAY initialization statement elapses. The status is automatically set to ACTIVE status when the tracker is started. The WSSTAT command or the EQQUSIN subroutine can be used to report OFFLINE status for a workstation.

**UNKNOWN** Tivoli OPC has detected that the workstation is inactive, but no other diagnostic information is available. You can manually reset the workstation to ACTIVE, using the Tivoli OPC dialogs, the WSSTAT command, or the EQQUSIN subroutine, or Tivoli OPC can...
Modify Current Plan (MCP) Dialog

do this automatically. A workstation normally connected via XCF or NCF will be reported in UNKNOWN status, if the XCF or NCF task is not started for the Tivoli OPC controller. Workstations that specify a user-defined destination ID are set to UNKNOWN status when they are first added to the current plan.

In addition to the workstation status, text that describes the status value can be displayed by Tivoli OPC in the current plan dialog. The possible status descriptions are:

**Waiting for manual intervention**
The Tivoli OPC controller has received an automatic online notification for a workstation previously in offline or failed status. The AVAIL action parameter value specified by the WSFAILURE or WSOFFLINE initialization statement specifies manual activation. If the workstation should be receiving work from the controller, vary the workstation status to ACTIVE from the MCP dialog.

**Waiting for connection**
The status of the workstation has been set to ACTIVE, and the Tivoli OPC controller is waiting for the corresponding tracker to communicate. No operations will be started on the workstation until the tracker and the controller synchronize the correct submit position. This condition can identify an error in your Tivoli OPC configuration. Perhaps the tracker events are not being communicated to the controller. Check that the tracker has either EWSEQNO(n) specified in the EWTROPTS or that an event reader has been started by specifying ERDRTASK(n) in the OPCOPTS for the tracker.

**Offline actions pending**
An offline notification has been received. The time Tivoli OPC has been instructed to wait before performing offline actions has not yet been reached.

**Status was set manually**
The status of the workstation has been set by a dialog user.

**Status set by WSSTAT**
The workstation status was set as the result of a workstation availability event generated by the WSSTAT TSO command or by the EQUSIN subroutine.

**Status set by EQQUX009**
The workstation status was set to the current value from the return code issued by the operation-initiation exit, EQQUX009.

**Redirecting Work to Alternate Workstations**
Tivoli OPC supports the redirecting of work from one workstation to another. If a workstation becomes inactive, you can specify—for each of its open intervals—an alternate workstation where work will be redirected. At any time, you can manually redirect the work to the workstation by using the Tivoli OPC Modify Current Plan dialog (option 5.5 from the Tivoli OPC main menu).
When you select a workstation to be modified, the current status of the workstation is presented (see Figure 45). In most cases, the workstation is Active and no further explanation is required. If the workstation status has been modified manually or if an exception has occurred, status text that describes the current state is also displayed.

If you make changes on the MODIFYING A WORKSTATION IN THE CURRENT PLAN panel, the changes take effect when you press PF3 (End)—you must press PF12 (Cancel) if you do not want to make the typed modifications.

If you need to manually change the status of a workstation, enter V from the MODIFYING A WORK STATION IN THE CURRENT PLAN panel, and the following panel is displayed.

```
EQQMWSVP ---- MODIFYING WORK STATION STATUS IN THE CURRENT PLAN -------------
Command ===>

Enter data below, and issue END command to change the status:
Work station : CPU1 Computer Automatic
Current Status : Active
NEW STATUS ===> _ A active, F failed
O offline
Fail/failed options :
STARTED OPERATIONS ===> _ R restart, L leave, E set to error
REROUTE OPERATIONS ===> _ Y to reroute, N to leave operations
ALTERNATE WS NAME ===> ___ Alternate Work station for reroute
```

Figure 46. EQQMWSVP—Modifying Workstation Status in the Current Plan

The values you enter on the MODIFYING WORKSTATION STATUS IN THE CURRENT PLAN panel determine the effect on the workload. If you modify the workstation status to failed or offline, you must specify what should happen to operations that are already started on the workstation and whether Tivoli OPC should reroute operations to an alternate workstation.
Notes:
1. Be careful if you specify that operations are to be restarted on an alternate workstation. They may still be running on the original workstation. Tivoli OPC does not know this. You must decide if they can be safely started on the alternate workstation. If you set them to error, you can rerun them after taking any necessary recovery actions—this is a safer option.

2. When you issue END from this panel, the status change takes effect immediately, even if you subsequently issue CANCEL on panel EQQMWSRP.

3. When defining your operations, you can specify that a particular operation should not be redirected under any circumstances.

Figure 47 shows how you can adjust the workstation fixed resources and parallel servers for a workstation, and specify an alternate workstation.

Figure 47. EQQMWSOL—Modifying Open Time Intervals in the CP

You cannot use this panel to make the workstation unavailable: use the panel in Figure 46 on page 69 instead. In practice, setting the number of parallel servers to zero means that operations cannot run on the workstation.

If you alter the interval details, the altered interval is never replaced by daily planning extend and replan functions. Consider this sequence of events:
1. You modify the number the number of parallel servers between 18.00 and 23.59 on 95/03/10 to 25.
2. You change the database to close CPU1 at 20.00 on 95/03/10.
3. You extend the plan at 15.00.

The result is that the workstation stays open past 20.00, because there is a modified interval that lasts until 23.59. Tivoli OPC considers that it is modified even if you change the number back to the original value.
Browsing System Information

From the MODIFYING WORKSTATIONS IN THE CURRENT PLAN panel, you can browse the system information for the destination of a computer workstation by entering the row command I. However, a non-local workstation can be browsed only if it communicates through XCF, NCF, or the submit/release dataset.
Modify Current Plan (MCP) Dialog
Chapter 5. Handling Operations that End in Error

This chapter explains how to handle failed operations. You use the ended-in-error list, which is part of the Modify Current Plan (MCP) dialog, which was introduced in Chapter 4, “Updating the Current Plan” on page 43.

When an operation that is under the control of Tivoli OPC does not complete successfully, it is reported as having ended in error. Tivoli OPC automatically reports failures of jobs or started tasks, and you can manually report failures of manual operations via the Tivoli OPC dialog. Tivoli OPC keeps a list of all operations that have ended in error. You can display this list using either the MCP dialog, which lets you take some action on the operation, or the QCP dialog, where you can only browse the operation.

When you are using the HANDLING OPERATIONS ENDED IN ERROR panel (the ended-in-error list), you have access to a wide range of options, which include:

- Restarting the operation
- Completing or deleting the operation or the occurrence
- Requesting automatic recovery for the operation
- Complete or delete an occurrence group
- Initiating, discarding, or displaying catalog management actions.

Figure 48 shows an example of an ended-in-error list, showing the PAYDAILY job having ended with an abend.

```
EQQMEP1L ------ HANDLING OPERATIONS ENDED IN ERROR (left part ROW 1 TO 1 OF 1
Command ===> Scroll ===> PAGE

Scroll right or enter the EXTEND command to get extended row command
information, enter the HIST command to select operation history list, or
enter any of the row commands below:
I,O,J,R,R,C,M,MR, or, REL,ARC,W,CM,MOD,DEL,CAT,RG,DS,or CG
LAYOUT ID ===> OPCESA__ Change to switch layout id

Cmd Ended time Application ws no. Jobname Errc
'''' 95/01/01 9.30 PAYDAILY CPU1 20 PAYDAILY SCC

******************************************************************************
```

Figure 48. EQQMEP1L—Handling Operations Ended in Error (Left Part)

Sometimes you may need to restart an operation even though it has completed successfully. See “Restarting an Occurrence from the Beginning” on page 57 for information on restarting operations that have not failed.
Displaying the Ended-in-Error List for Action

To display the ended-in-error list for action, select option 4 (ERROR HANDLING) from the MCP dialog main menu. The SPECIFYING ENDED IN ERROR LIST CRITERIA panel is displayed. Specify selection criteria to display only the list items you want to examine.

You can customize the layout of the ended-in-error list to suit your needs, either by selecting one of the installation-defined layouts or by building your own layout. This is described in the following sections.

After you select an error list layout and display your error list, the list is updated with the latest information when you press Enter or any of the PF keys.

Selecting an Ended-in-Error List Layout

When you select the ended-in-error list, you can also define a layout to use for the list. About 90 fields can be displayed in an ended-in-error list (see Appendix C, “Fields Displayed in Ready and Error Lists” on page 121). Not all of these can be presented at once in the same ended-in-error list. But you can specify two panels for your layout and switch between these panels using the LEFT and RIGHT commands. Information on the fields that are to be displayed, and the order in which they should be displayed, is kept in a named ended-in-error list layout. There might be many different layouts in your installation.

If you have not specified a layout, Tivoli OPC displays a list of existing layouts. You can then select a layout from this list. If you want to change the layout of the ended-in-error list, enter the name of another layout in the corresponding field in the ended-in-error list, and Tivoli OPC will display the ended-in-error list using the layout you specified. You can get a list of all the ended-in-error list layouts by entering an asterisk (*) in the LAYOUT ID field on the SPECIFYING ENDED-IN-ERROR LIST CRITERIA panel:

![EQQMERRP SPECIFYING ENDED IN ERROR LIST CRITERIA](image)

Figure 49. EQQMERRP—Specifying Ended In Error List Criteria
Creating Your Own Ended-in-Error List Layout

Ended-in-error list layouts are kept in two ISPF tables: one for your own use and one that contains the installation-defined layouts. Your own version of a layout overrides the installation layout for your user ID. To change an ended-in-error list layout, select the DEFINE EL option on the MODIFYING THE CURRENT PLAN panel. Tivoli OPC then displays a list of both your own layouts and any installation-defined layouts. From this list, you can create new layouts or select an existing layout to delete, copy, modify, or browse. Modified installation-defined layouts are stored in your own library. You can delete your own layout versions, but you cannot delete an installation-defined layout.

```
EQQELYLL ------------------------ ERROR LIST LAYOUTS ------------------ ROW 1 TO 2 OF
Command ===> Scroll ===> PAG

Enter the CREATE command above to create a new layout or
enter any of the row commands below:
B - Browse, C - Copy, D - Delete, M - Modify

<table>
<thead>
<tr>
<th>Row</th>
<th>Layout</th>
<th>Description</th>
<th>Owner</th>
<th>Last update</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>OPCESA</td>
<td>OPC/ESA standard layout</td>
<td>CONYL2</td>
<td>92/02/11 04.2</td>
</tr>
<tr>
<td>2</td>
<td>CMENH</td>
<td>cm enhancements</td>
<td>XRAYNER</td>
<td>94/05/02 19.3</td>
</tr>
</tbody>
</table>

Figure 50. EQQELYLL—Selecting an Error List Layout

When you modify or create a layout, Tivoli OPC presents a list of the available ended-in-error list fields. See Appendix C, “Fields Displayed in Ready and Error Lists” on page 121. From this list, you select and order the fields to be contained in the ended-in-error list layout. You can also specify whether the field should be highlighted when Tivoli OPC displays it. All the items you select for display are placed as near the top of the selection list as possible.
Error List

Figure 51. EQQELYCL—Creating an Error List Layout

To make a set of private layouts available for your colleagues:

1. Create a complete set of layouts. If you want to include a supplied (or old) layout, edit and save it so that it becomes part of your private library.

2. Save (by renaming) the old EQQELDEF member in the common table library.

3. Copy the EQQELOUT member from your ISPF profile library to the common table library, renaming it to EQQELDEF.

Getting Rerun or Recovery Instructions

When an operation fails, the person responsible for rerunning the operation often needs rerun instructions. These instructions can reside either in the job or in the operator instruction database.

To display the instructions for the failing operations, enter the O row command beside those operations on the HANDLING OPERATIONS ENDED IN ERROR panel. To see the job, enter J in the row command field.

Completing an Ended-in-Error Operation

Sometimes you might consider that an operation has ended successfully, even though it is listed as ended-in-error. To set its status to C (complete), enter the C row command beside that operation in the HANDLING OPERATIONS ENDED IN ERROR panel.

If a catalog management action is started from the dialog or an immediate action is in progress, you receive a warning message, and the operation will not be marked
complete. If a deferred catalog management action is defined and not yet in progress, the CM action is discarded, and the operation is marked complete.

Modifying a Job That Has Failed

If the job or started task has failed, you might be able to correct the error by changing the job. You can do this from the Tivoli OPC dialog by entering the J row command beside that operation in the HANDLING OPERATIONS ENDED IN ERROR panel. Tivoli OPC uses the ISPF/PDF editor to display the job for you to edit.

Tivoli OPC keeps a separate copy of the job for each run. The job you edit using the J row command is always the job that was used during the failing run of the job (unless you have edited it since). When you end the edit with the END command, Tivoli OPC saves the modified job in the JCL repository and redisplays the ended-in-error list.

If the job contained resolved variables and you want to use the same variables again with new values, perhaps from another JCL variable table, delete the job in the ISPF editor and enter the END command. This forces Tivoli OPC to use a new copy of the job from EQQJBLIB.

JCL variable values are stored in the JCL record and will be used again if referenced in the current copy of the job. This is useful in situations in which you want to set promptable variables in the READY LIST panel, and then refer to them during the submit phase, probably in a COMP statement. For more information about job tailoring, refer to Planning and Scheduling the Workload.

Note: Editing and saving the job does not restart the operation.

Restarting an Ended-in-Error Operation

The simplest way to restart an operation is to enter the SR (step restart) or JR (job restart) row command beside that operation in the HANDLING OPERATIONS ENDED IN ERROR panel (you can use step restart only if your system is set up to do this). You would normally choose this way to restart an operation after the cause of the error is corrected (for instance, by modifying the job).

If job log information is available, you can browse it with the L row command.

If catalog management is defined for the operation and is in progress, you must wait for the CM action to complete.

If you have selected step restart, see “Restarting an Operation from a Certain Step” on page 79.

If you have selected job restart, or have selected step restart and selected the restart options, you see the CONFIRM RESTART OF AN OPERATION panel, shown in Figure 52 on page 78.
The line marked 1 explains the type of recovery action:

**Rebuilt JCL is saved**
This is step restart with no catalog updates.

**Catalog Management actions are initiated**
This is job restart with catalog updates.

**Rebuilt JCL is saved and Catalog Management actions are initiated**
This is step restart with catalog updates.

**Blank (no text)**
This is job restart with no catalog updates.

The recovery includes catalog actions only if deferred catalog management is specified for the operation.

When you restart a job, you can give a reason, which is written to the track log for audit purposes.

When you confirm the restart, Tivoli OPC:

1. Resets the operation status to ready.
2. Initiates any catalog management actions.
3. Resubmits the operation when all conditions are met.

### Taking Action at the Occurrence Level
If the operation cannot be restarted in this way because you need to change the application occurrence, enter the MOD row command beside the operation. This takes you to the MODIFYING AN OCCURRENCE IN THE PLAN panel, where you can change the occurrence.

If you need to rerun the previous operations within the application occurrence before rerunning the failed operation, enter the RER row command beside the operation in the HANDLING OPERATIONS ENDED IN ERROR panel. This command
Displays the RERUNNING AN OCCURRENCE IN THE PLAN panel, where you can specify from which operations you want the occurrence to be restarted.

Handling Operations with the OSEQ Error Code

If Tivoli OPC detects the running of a job with the same name as one in the plan, but out of sequence, it gives the operation in the plan an OSEQ error code, which prevents its running. If the planned operation should be run, use the JR row command to reset it to waiting status. If the planned operation should not run, delete it.

Restarting an Operation from a Certain Step

If the job log is available for the operation, you see the STEP RESTART SELECTION LIST panel, shown in Figure 53.

```
Figure 53. EQMERSL—Step Restart Selection List

From this panel, select the steps that you want included in the restart. Specify the first step (S) and the last step (E) to be executed in the rerun. S defaults to the first step and E defaults to the last step. To have only one step in the rerun, specify S for that step and exclude (X) all the other steps. If you make no selections, a job level restart with no omitted steps is assumed.

When you enter the GO primary command, Tivoli OPC builds JCL to support the restart you have selected. Tivoli OPC then validates the JCL, as described in “How Tivoli OPC Validates the Restart JCL” on page 82. If Tivoli OPC finds inconsistencies, you see the errors in the edit panel shown in Figure 56 on page 83. You can either modify the job to correct the problem, or you may decide that the inconsistency reported by Tivoli OPC will not be a problem for the restart.
```
The codes shown before the Stepname column are intended as a guide and have these meanings:

<table>
<thead>
<tr>
<th>Code</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>:{</td>
<td>Not restartable from this step.</td>
</tr>
<tr>
<td>:)</td>
<td>Restartable from this step.</td>
</tr>
<tr>
<td>:!</td>
<td>Restart may be possible. This step was flushed, or inconsistencies were found.</td>
</tr>
<tr>
<td>=&gt;</td>
<td>This step is the point of failure or the first step.</td>
</tr>
</tbody>
</table>

If catalog management is specified for the operation, enter the CAT primary command to check the action that Tivoli OPC will take for each dataset. You see the MODIFYING CATALOG MANAGEMENT ACTIONS panel, shown in Figure 57 on page 85.

If you decide a step-level restart is not needed after viewing the job, enter the CANCEL command and enter the JR row command from the error list to initiate catalog management actions at job level.

When you have selected the steps, you see the CONFIRM RESTART OF AN OPERATION panel, shown in Figure 52 on page 78.

Enter the JOB primary command to browse the log. You see the BROWSING OPERATION INPUT/OUTPUT STREAM panel, shown in Figure 54.
Simulating Return Codes

Sometimes your restarted job will need the return codes from bypassed steps to complete successfully. The return code simulation function enables Tivoli OPC to simulate return codes together with step-level restart.

To invoke return code simulation, specify Simulate RC = Y on the STEP LEVEL SELECTION LIST panel. In the Simulate retcode field you can set a return code to be used as input to the restarted job. You can edit the rebuilt JCL and change the order of the steps, as long as the added step is placed before the first step that should be excluded from restart.

How Tivoli OPC Builds JCL for Restart

When you select step restart, Tivoli OPC builds the necessary JCL like this:

- All relative GDG references are changed to the corresponding absolute name. References within procedures are resolved by inserting JCL overrides in the appropriate order. The catalog management option in the operation definition must be set to D or Y. Otherwise the GDG dataset cannot be resolved because catalog management information will be missing.

- A RESTART parameter is appended to the job card. If a RESTART card already exists in the JCL, the original card is commented, the new RESTART card is inserted and message EQQM361 is issued to indicate that a statement in the original JCL has been replaced.

- When return code simulation is requested, Tivoli OPC does not insert the RESTART parameter. Instead, Tivoli OPC retrieves the return codes of the bypassed steps from the job log and then adds an additional first step to the job card. The added step refers to a catalogued procedure EQQRCSIM. EQQRCSIM must be installed in all MVS systems that use return code simulation. EQQRCSIM invokes a return code simulation program EQQSWAUP that should be stored in an APF authorized library. The SYSIN requirements for EQQRCSIM are described in Planning and Scheduling the Workload.

- COND=ONLY parameters are inserted in all steps before the restart step if the JCL is for a started task. If a COND parameter is already defined, and does not specify COND=ONLY, the new parameter is inserted as a JCL comment line and message EQQM362 will be issued. If the started task JCL contains nested procedures, Tivoli OPC may not be able to insert the COND statement. If this occurs, it issues message EQQM369.

- COND=(0,LE) parameters are inserted in all steps beyond the last step to be rerun.

- It replaces the JCL in the job repository dataset, so you lose any JCL alterations that you made since the last run of the job.

- INCLUDE JCL statements are considered in the same manner as PROC JCL statements.
How Tivoli OPC Validates the Restart JCL

When Tivoli OPC has rebuilt the JCL for restart, it validates it. If inconsistencies are found, it issues message EQQM339W, and inserts messages as JCL comment cards to direct your attention to the problem. If catalog management is requested, Tivoli OPC inserts messages to identify the datasets that will be deleted, uncataloged, or recataloged by the catalog management function before the job is restarted.

Tivoli OPC checks for the potential problems listed in Figure 55 and inserts message lines in the JCL at the place where the problem is detected:

<table>
<thead>
<tr>
<th>Potential problem</th>
<th>Message number</th>
</tr>
</thead>
<tbody>
<tr>
<td>There are JES3 DJC control statements in the JCL. A JES3 dependent job control net can be destroyed.</td>
<td>EQQM360</td>
</tr>
<tr>
<td>There is already a COND statement where Tivoli OPC needs to insert one.</td>
<td>EQQM362</td>
</tr>
<tr>
<td>There are backward references to a step before the restart step. Tivoli OPC cannot resolve such backward references.</td>
<td>EQQM365</td>
</tr>
<tr>
<td>There are COND or IF statements in the restart scope. MVS ignores COND or IF parameters that refer to a step before the restart step.</td>
<td>EQQM368</td>
</tr>
<tr>
<td>There are nested procedures, and Tivoli OPC needs to modify JCL in a nested procedure. Modifications (insertion of COND parameters and overrides for GDG references) can be made only in the immediate (level 1) procedure.</td>
<td>EQQM369</td>
</tr>
<tr>
<td>There are DISP=MOD references to datasets within the restart scope. Restarts can cause data to be wrongly appended to these datasets.</td>
<td>EQQM371</td>
</tr>
<tr>
<td>Referredenced datasets are deleted, passed, or uncataloged in a step before the restart step. A restart can cause a JCL error.</td>
<td>EQQM372</td>
</tr>
<tr>
<td>Message IEF377I was detected in the joblog. A data set was allocated but not cataloged, because a data set with the same name already exists on another volume. This could lead to the deletion of the cataloged data set instead of the data set just allocated by the job.</td>
<td>EQQM378</td>
</tr>
<tr>
<td>There are Tivoli OPC automatic recovery statements in the JCL. If automatic recovery has already occurred for the job, the recovery statements that have already been matched are now commented out and ignored, but Tivoli OPC will action any further recovery statements that match on subsequent reruns, which may not be what you want when you select step restart from the dialog. This is especially true when there is a RESSTEP keyword on a recovery statement, which can change the scope of the restart.</td>
<td>EQQM379</td>
</tr>
<tr>
<td>Instream data is missing, for jobs not submitted by Tivoli OPC. This includes SYSIN data, passwords on the job statement, and data in a JES3 DATASET statement. Tivoli OPC cannot capture this data for jobs that it does not submit.</td>
<td>EQQM380</td>
</tr>
</tbody>
</table>

Figure 55. How Tivoli OPC validates JCL for step restart

When you specify step restart in the dialog, Tivoli OPC issues a warning message for any of these potential problems and you can edit the JCL.
Figure 56 shows a job prepared for step restart that has potential problems.

Figure 56. Modified JCL for a Step Restart

In edit, you see Tivoli OPC messages that document the inconsistencies ( ). The messages are inserted in the JCL as ISPF message lines in the format:

```
//<msgnumber> msgtext
```

The message lines are temporary ISPF rows. If you want the messages to be kept with the JCL, use the ISPF row command MD to make the temporary row a dataline, which will be saved with the JCL in the job repository. If you use cataloged procedures, Tivoli OPC displays the procedure JCL statement that causes the difficulty as a comment in the job. All datasets that Tivoli OPC will attempt to perform catalog management actions for are indicated ( ).

If you edit the invalid job so that it can run, save it by pressing PF3 (End), and press PF3 again on the STEP RESTART SELECTION LIST panel to submit the job (do not use the GO command again, or Tivoli OPC will rebuild the JCL, losing your changes).

The JCL tailoring and verification performed by Tivoli OPC is explained in more detail in Planning and Scheduling the Workload.
Using Catalog Management

Catalog management helps you recover MVS jobs and started tasks by deleting or uncataloging datasets that were created in the failing job and cataloging datasets that were uncataloged. Without this, a rerun will fail with an error message saying that a dataset already exists, or is already cataloged, or is not found.

Tivoli OPC restores the catalog status of datasets from the restart step up to and including the failing step. It cannot restore the dataset itself—it just deletes or uncatalogs it if it was created and catalogs it if it was uncataloged.

The Tivoli OPC catalog management function works like this:

1. Before the job is run, it scans the JCL for dataset dispositions such as (NEW,CATLG), which will cause a change to the catalog.
2. It records the status of these datasets.
3. When catalog management is required (after a job ends in error), Tivoli OPC retrieves the job log. Refer to Planning and Scheduling the Workload for a full description of job log retrieval. You should not change the output class of the job, because Tivoli OPC might not be able to find the job log if it is needed for a restart (this depends on installation options).
4. It scans the job log for messages showing any change in status for the recorded datasets. For Tivoli OPC to obtain sufficient information from the job log, the MSGLEVEL must be specified as MSGLEVEL=(1,1). This can be defined either as the JES installation default or can be specified on the job card.
5. It makes a list of instructions to restore the data set status. All data sets defined with DISP=NEW are marked as eligible for potential deletion. The operator can refuse the 'deletion' if this action is not in line with the logic of the jobstream to be restarted.
6. The Tivoli OPC controller passes these instructions to the tracker as needed. If an operation specifies immediate catalog management, the controller sends the instructions to the tracker as soon as it has received and processed the job log. If an operation specifies deferred catalog management, the controller sends the instructions when you confirm the restart (either step restart or the whole job) from the panel in Figure 52 on page 78.

Note: Flushed steps datasets are not considered. Datasets that are referenced as DISP=OLD/MOD/SHR in a previous step are considered only if that previous step is excluded from the new run range.

For a detailed description of catalog management, refer to Planning and Scheduling the Workload.
You can modify or browse catalog management information from a number of places in the Tivoli OPC dialog. In most cases, you will probably be using the HANDLING OPERATIONS ENDED IN ERROR panel to browse, modify, execute, or discard catalog management actions. You see the MODIFYING CATALOG MANAGEMENT ACTIONS panel, shown in Figure 57.

![Figure 57. EQQCMMDL—Modifying Catalog Management Actions](image)

Tivoli OPC maintains status information to describe the progress of the catalog management action. If you use CM, include the CM status field in your ended-in-error layout. See “Catalog Management Status Codes” on page 118 for a list of the possible values for catalog management status.

### Specifying Automatic Restart for Operations That Fail

The Tivoli OPC automatic recovery function lets you specify, for each job or started task, automatic recovery actions for specific failures. You specify recovery actions by including Tivoli OPC recovery statements in the job. Automatic recovery parameters that refer to job steps are supported only for jobs that execute on MVS systems. For detailed information, refer to Planning and Scheduling the Workload.

If automatic job recovery fails, or if a job does not have recovery statements when it fails, you can modify the job and create or change the recovery statements. Use the J row command on the HANDLING OPERATIONS ENDED IN ERROR panel to edit the job. The EDITING JCL FOR AN MVS JOB panel in Figure 58 on page 86 shows some Tivoli OPC recovery statements (1 to 3). The beginning of the recovery statement 1 has been changed from ///c5197% to ///c5197> showing that action has already been taken on the statement.
Figure 58. An Example of RECOVER statements

You can add further recovery directives to the job, save the job, and initiate recovery by entering the ARC row command beside the failing operation on the HANDLING OPERATIONS ENDED IN ERROR panel.

You can also use the ARC command to initiate recovery outside the hours specified for automatic recovery.

If a request to start automatic recovery is received while a catalog management action is still in progress, the request is denied.
Chapter 6. Monitoring Special Resources

This chapter shows you how to monitor and change special resources. You can do this using option 7 (SPECRES) in the Modify Current Plan (MCP) dialog. This part of the MCP is called the Special Resource Monitor.

Understanding Special Resources

You can use Tivoli OPC special resources to represent any type of limited resource, such as tape drives, communication lines, or a database. The Tivoli OPC administrator defines resources using the Special Resource Definition dialog, which is described in Planning and Scheduling the Workload. The Special Resource Definition dialog updates the resource database, which has these details of each resource:

- **Name**: Up to 44 characters. This identifies the resource.
- **Availability**: Yes (Y) or no (N).
- **Connected workstations**: A list of the workstations where operations can allocate the resource.
- **Quantity**: 1 through 999999.
- **Used for**: How Tivoli OPC is to use the resource: for planning (P), control (C), both (B), or neither (N).
- **On-error action**: Free all (F), free exclusively-held resources (FX), free shared resources (FS), and keep all (K). Tivoli OPC uses the attribute specified at operation level first. If this is blank, it uses the attribute specified in the resource database. If this is also blank, it uses the ONERROR keyword of the RESOPTS statement.

The quantity, availability, and list of workstations can vary with time. The administrator can associate different intervals with the resource.

The administrator also specifies, for each operation, the special resources that it uses: how (shared or exclusive), how many (quantity), and the on-error attribute.

The long-term plan is built without taking the special resources into account, but when you extend the current plan, it schedules operations taking account of all the special resources that are used for planning though the daily planning program does not take manually changed availability, quantity, and deviation into account. This is because they are usually assumed to be temporary changes and the values will be reset to the normal values when, for example, an engineer has repaired a tape unit.
Special Resources

If a special resource is needed in the current plan, Tivoli OPC copies the details from the RD database and stores them in the current plan extension dataset. These details include the information from the resource database, but also have these overriding (global) fields:

**Quantity** 1 through 999999 or blank. If you specify a quantity, this overrides the scheduled quantity from the database.

**Availability** Y or N or blank. If you specify an availability, this overrides the scheduled availability from the database.

**Deviation** -999999 through 999999 or blank. You use the deviation to make a temporary alteration to the scheduled quantity.

You can change the quantity and availability of a special resource, and the connected workstations, using the Special Resource Monitor, which is described in this chapter. You might need to make a resource unavailable (to prevent the submission of all jobs needing a database, if some corruption is suspected), alter the quantity by specifying a deviation (if a tape drive is broken), or change the list of connected workstations (to include a workstation that will take over processing from the normal one).

Other ways of changing resource attributes are:

**EQQUSIN subroutine** Refer to Customization and Tuning.

**SRSTAT command** Refer to Planning and Scheduling the Workload.

If the availability of a resource is known to the Resource Object Data Manager (RODM), Tivoli OPC can, by subscribing to RODM for that resource, be notified automatically of any changes. This is the best alternative where RODM is installed.

Changes to special resources using any of these methods override the scheduled quantity and availability, but you can at any time reset the values to those specified for the current interval.

These are some examples of how you can use the Monitor:

- Browse the special resources in the current plan, and see how they are allocated.
- Deallocate resources from an operation that is running
- Change the resource requirements of an operation that has not yet run
- See what special resource an operation is waiting for, if its extended status is X (for example, status RX), and see what operations are using it.
Example Using Data Sets

The payroll application has many jobs that use the payroll database and therefore must not run together. You could let MVS resolve the contention problem, using DISP=OLD in the JCL, but a job that waits in MVS uses a JES initiator and other resources. This can reduce your batch throughput.

To prevent Tivoli OPC from scheduling or starting an operation that uses the payroll database when it is already being used, define a resource PAYROLL.DATABASE that represents the payroll database. Give each update job, such as PAYDAILY, exclusive control. Jobs that merely read the database, such as PAYQUERY, can have shared control.

In this case, the resource has a quantity of 1 (there is one database). Specify keep on error, because operators want to correct and resubmit a job without another job taking control of the database in the meantime.

When you extend the current plan, Tivoli OPC makes sure the jobs are not scheduled to run together or run at a time when the resources are unavailable. This is how Tivoli OPC makes use of the resource at the planning stage. When Tivoli OPC is ready to submit each job, it checks that the resource is available. This is how Tivoli OPC makes use of the resource at the control stage.

Specify the resource like this:

Name: PAYROLL.DATABASE. Make sure that all operations specify this name exactly.
Quantity: 1
Used for: B (both planning and control).
On-error action: Keep all.
Workstations: Connect to all workstations that can use the dataset.
Availability: The dataset is always available, so no intervals are required.
Example Using Tape Drives

Tape drives are usually owned by only one machine, but they can be used by a started task workstation and a computer workstation on the same machine, so you can, for example, allocate a tape pool to workstations CPU1 and STC1.

Operations that use tapes must allocate them exclusively. Do not let your operations keep this resource on error, because you normally want to release a tape drive for other work while you prepare a rerun of a failed job.

Specify the resource like this:

Name: TAPES
Usage: Give all operations exclusive allocation.
Quantity: 10 (for example). You need not make all the drives available for automatic allocation.
Used for: B (both planning and control).
On-error action: Free all.
Workstations: Connect to all workstations that can use the tapes.
Intervals: Reduce the number available when online systems might need a tape and make the resource unavailable when the workstation is unstaffed.

Example Using Communication Lines

Lines are often shared between processors and can even be shared between operations. Lines are never allocated by MVS, because they are owned by a communication controller, but you might want to ration the number of file transfer jobs, for example. If you have a number of lines from New York to London, with a total capacity of 256 kbaud, you can define a quantity of 256. If you give a file transfer operation exclusive use of 20 units, for example, that gives the lines a limit of 12 file transfers.

You can protect online and voice systems that use the same lines by giving them a shared allocation of, for example, 50 units each. Because they share units, they do not compete with each other—you can have an unlimited number of operations each sharing 50 units, but this limits the quantity available for file transfers to 206.

Specify the resource like this:

Name: LINES.TO.LONDON
Quantity: 256
Used for: B (both planning and control)
On-error action: Free all
Workstations: Connect to all workstations that can use the lines
Intervals: Reduce the number available at peak hours, which will keep more transfer jobs out and improve the performance of online systems.
How Does Tivoli OPC Use Special Resources?

Tivoli OPC keeps a record of the state of each resource and its allocation. Tivoli OPC does not know that PAYROLL.DATABASE is a database and it is unaware that TAPES resources are tape drives. Only you know this and you are responsible for making sure that Tivoli OPC knows the true availability of the objects that the resources represent.

VSAM will not tell Tivoli OPC when the payroll database is opened, and MVS is not going to tell Tivoli OPC when you vary a tape drive offline. This is your responsibility.

The best way to tell Tivoli OPC about changes in special resources is through the RODM interface. System components such as AOC/MVS inform RODM about changes to their resources. You can subscribe to RODM updates by setting the RODMTASK keyword on the OPCOPTS initialization statement, and using the RODMOPTSTS initialization statement. Refer to Customization and Tuning for details of initialization statements.

If you do not have RODM installed, and for resources that RODM does not know about, you can automatically notify Tivoli OPC about changes in resources by intercepting messages (NetView can do this) and issuing the Tivoli OPC SRSTAT command, or calling the EQQUSIN subroutine. If you cannot automatically notify Tivoli OPC of a change in resource status, use the Special Resource Monitor.

Using the Special Resource Monitor

Operations hold and release special resources automatically according to their descriptions in the current plan and the resources are available and connected to workstations as scheduled in the current plan. So why do you need to monitor them?

Resources represent something, such as tape drives, and the current plan is built assuming that a certain number of tape drives will be available. If one breaks down and you do not use RODM or otherwise automatically notify Tivoli OPC, Tivoli OPC no longer has a true picture of the real world—Tivoli OPC continues to allocate the broken tape drive to a job that needs one. The job will wait, because MVS knows that the tape drive is offline. To avoid Tivoli OPC starting a job that will only wait and use operating system resources, use the Special Resource Monitor dialog to reduce the number of tape drives by one. You specify a deviation of -1 (minus one) to show that the quantity is reduced. When the engineer hands it back and you vary it online again, reset the deviation.

If a job is waiting with extended status X, it is waiting for a resource. Use the Special Resource Monitor dialog to check its availability, and see what other operations are using the resource.
Understanding Availability Intervals

Before you use the Special Resource Monitor, be sure that you understand how availability intervals work. Figure 59 shows how planned availability is affected by unplanned events such as input from the Special Resource Monitor, the SRSTAT command, and the EQQUSIN subroutine. Notice that manually altered attributes are honored across an interval boundary and batch planning EXTEND and REPLAN jobs—to make Tivoli OPC revert to the scheduled values after a manual alteration, you must reset the attribute, as at 11.20 in Figure 59.

<table>
<thead>
<tr>
<th>Start of interval / time of event</th>
<th>Planned values</th>
<th>Actual values</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Planned quantity</td>
<td>Planned availability</td>
</tr>
<tr>
<td>08.00</td>
<td>8</td>
<td>N</td>
</tr>
<tr>
<td>08.40</td>
<td>You set the availability to Y with the EQQUSIN subroutine</td>
<td>8</td>
</tr>
<tr>
<td>09.00</td>
<td>8</td>
<td>N</td>
</tr>
<tr>
<td>09.40</td>
<td>You set a deviation of -1 with the SRSTAT command</td>
<td>8</td>
</tr>
<tr>
<td>09.41</td>
<td>You set a deviation of -1 with the SRSTAT command</td>
<td>8</td>
</tr>
<tr>
<td>09.42</td>
<td>You set the deviation to -1 with the Special Resource Monitor</td>
<td>8</td>
</tr>
<tr>
<td>10.00</td>
<td>9</td>
<td>Y</td>
</tr>
<tr>
<td>10.20</td>
<td>You set the quantity to 6 with the SRSTAT command</td>
<td>6</td>
</tr>
<tr>
<td>11.00</td>
<td>8</td>
<td>Y</td>
</tr>
<tr>
<td>11.20</td>
<td>You reset the quantity with the SRSTAT command</td>
<td>8</td>
</tr>
</tbody>
</table>

Figure 59. How Attributes Are Preserved across Intervals

The number available (the last column) is the actual number available for allocation, taking into account the actual quantity, the deviation, and the actual availability.

The three events starting at 09.40 show the difference between altering the deviation with SRSTAT (or a subroutine) and with the Special Resource Monitor. SRSTAT adds the specified deviation to the current deviation, but the dialog replaces the current deviation with the value you specify.

If you change values other than the overriding (global) quantity, availability, and deviation, or the values for an interval, you lose the changes at the next daily
planning run, but the job issues a warning message about any manually changed values that will be lost. For example, if you change the default quantity (the quantity used where intervals are not specified) in the current plan, this is replaced at the next daily planning run with the value from the database.

Accessing the Special Resource Monitor

Follow these steps to use the Special Resource Monitor to remove a resource from an operation:

1. From the Modify Current Plan main menu, select option 7 (SPECRES). You see the SPECIFYING RESOURCE MONITOR LIST CRITERIA panel, shown in Figure 60.

   ![EQQQMSEP—Specifying Resource Monitor List Criteria](image_url)

2. Use the panel to limit the resources displayed. The SPECIAL RESOURCE and SPECRES GROUP ID fields can have Tivoli OPC filter characters such as * (asterisk) to specify a range of resources. Specify the values shown, for example, to show all available resources whose names begin with PAY.
3 Press Enter to list the resources. You see the SPECIAL RESOURCE MONITOR panel, shown in Figure 61.

```
EQQOMLSL --------------- SPECIAL RESOURCE MONITOR -------- ROW 1 TO 1 OF 1
Command ==> Scroll ==> PAGE

Enter any of the row commands below:
B - Browse, M - Modify, I - In use list, W - Waiting queue

<table>
<thead>
<tr>
<th>Resource</th>
<th>RDM</th>
<th>Adjust</th>
<th>Used</th>
<th>Used</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAYROLL.DATABASE</td>
<td>Y</td>
<td>NNN</td>
<td>1/zerodot/zerodotN</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

**Figure 61. EQQOMLSL—Special Resource Monitor**

Each resource is described with these fields:

**A** Availability. Y or N. When a “global” availability has been set (for example, with the SRSTAT command), this is shown, rather than the value specified for the interval.

**RDM** RODM can update three resource fields:

- **A** Availability
- **D** Deviation
- **Q** Quantity

The panel shows a separate status for each of the RODM fields:

- **N** Not monitored. Tivoli OPC does not subscribe to RODM for this field.
- **I** Inactive. RODM is inactive, or communication has been lost. Tivoli OPC tries to reconnect every 5 minutes.
- **P** Pending. A status request has been sent to RODM, but no reply has yet been received.
- **A** Active. RODM monitoring is active and has updated the field.

**Adjust Qty** The current quantity, taking any deviation into account. When a 'global' quantity has been set (for example, with the SRSTAT command), this is used, rather than the value specified for the interval or the default value.

**Used Shared** Quantity allocated as shared.

**Used Excl** Quantity allocated as exclusive.

**W** Indicates that there are operations waiting to allocate the resource (Y), or no waiting operations (N).

The available row commands are:

- **B** Browse the resource.
- **M** Modify the resource.
- **I** Display the operations that have allocated this resource.
- **W** Display the operations that are waiting to allocate this resource.
4 Enter the I row command to see the operations that are using the resource. You see the SPECIAL RESOURCE MONITOR - IN USE LIST panel, shown in Figure 62.

![Special Resource Monitor - In Use List](image1)

Figure 62. EQQQMIML—Special Resource Monitor - In Use List

The operations at the top of the list have been using the resource for the longest time and are more likely to release their allocation soon. The duration helps you estimate when the operations will release their allocation of the resource.

The list has these columns:

- Actual start date and time
- Operation ws (workstation)
- Operation no. (operation number)
- Jobname
- Est Dur (the estimated operation duration)
- S (status of the operation)
- Qty (number allocated to this operation)
- Type (type of allocation—shared (S) or exclusive (X)).

5 Enter the D row command beside the operation to release the resources from the operation.

Note: This is a logical release. In Figure 62, for example, the D command causes PAYDAILY to free the Tivoli OPC resource PAYROLL.DATABASE, but if its status is S, it can still be writing to the real payroll database.

6 You see the CONFIRMING DELETION OF AN OPERATION FROM QUEUE OR LIST panel. Enter Y to deallocate the resource from the operation.
Looking at the Operations Waiting for a Resource

To see the operations that are waiting for a resource, enter the W row command beside the resource in the SPECIAL RESOURCE MONITOR panel. You see this panel:

```
EQQQWMML -------- SPECIAL RESOURCE MONITOR - WAITING QUEUE  ROW 1 TO 1 OF 1
Command ==> Scroll ==> PAGE

Enter any of the row commands below:
S - Select details, D - Delete from queue

Special resource : PAYROLL.DATABASE
Text : serializes access to the Paymore database

Row Latest Out Operation Jobname Pri Qty Type Reason
cmd Date Time ws no. Wait
'' 95/06/08 10.40 CPU1 050 PAYQUERY 5 1 S *
*******************************************************************************
```

Figure 63. EQQQWMML—Special Resource Monitor - Waiting Queue

The operations at the top of the list have earlier latest-out times and are therefore more likely to get the resource when a sufficient quantity becomes available. The list has these columns:

- Latest out date and time
- Operation ws (workstation)
- Operation no. (operation number)
- Jobname and priority
- Qty (the amount of the resource that the operation needs)
- Type (type of allocation—shared (S) or exclusive (X))
- Reason Wait (the reason that this operation must wait). It can have these codes:

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>FEWINF</td>
<td>According to the values of the LOOKAHEAD initialization parameter and the planned job duration, there will not be enough quantity to satisfy the waiting operation.</td>
</tr>
<tr>
<td>INVRES</td>
<td>The resource is not valid.</td>
</tr>
<tr>
<td>NOWSC</td>
<td>No workstation is connected.</td>
</tr>
<tr>
<td>OTHRES</td>
<td>The operation needs this resource, but is waiting for another special resource.</td>
</tr>
<tr>
<td>RODMP</td>
<td>Tivoli OPC is waiting for a status update from RODM.</td>
</tr>
<tr>
<td>TOOFEW</td>
<td>There is not enough quantity to satisfy the operation that is waiting.</td>
</tr>
<tr>
<td>UNAVL</td>
<td>The resource is not available.</td>
</tr>
<tr>
<td>UNAVLF</td>
<td>According to the values of the LOOKAHEAD initialization parameter and the planned job duration, the resource will not be available for the required time.</td>
</tr>
</tbody>
</table>

* An operation is taking all the resource.
Enter the D row command beside a row to remove the dependency of the operation on the resource so that it can start (but it might be waiting for other resources, too). Do this with care, because the operation might not have access to the resource that it needs to run successfully. In the PAYQUERY example, if you remove the dependency of PAYQUERY on the payroll database resource, the PAYQUERY job will start, but it will wait for the database if it is still allocated to PAYDAILY. You see the CONFIRMING DELETION OF AN OPERATION FROM QUEUE OR LIST panel. Enter Y to remove the resource dependency from the operation.

Modifying a Special Resource

From the SPECIAL RESOURCE MONITOR panel (Figure 61 on page 94), you can browse or modify special resource details using the B or M row commands. To modify a resource, enter the M row command beside a resource. You see the MODIFYING A SPECIAL RESOURCE panel, shown in Figure 64.

```
EQQMMOP ---------------- MODIFYING A SPECIAL RESOURCE ----------------
Option ===>
Select one of the following:
1 INTERVALS - Specify intervals
2 WS - Modify default connected work stations
Special resource : TAPES
Text : tape drives on CPU1 and STC1
Specres group id : SAMPLE
Hiperbatch ===> N DLF object Y or N
USED FOR ===> B Planning and control C , P , B or N
ON ERROR ===> F_ On error action F , FX , FS , K or blank
DEV DEVIATION ===> _______ Number to deviate -999999 to 999999 or blank
AVAILABLE ===> _ Global availability Y or N or blank
QUANTITY ===> ______ Global quantity 1 to 999999 or blank

Defaults
QUANTITY ===> 8_____ Number available 1-999999
AVAILABLE ===> Y Available Y or N

Last updated by XRAYNER on 94/06/10 at 10.21
```

Figure 64. EQQMMOP—Modifying a Special Resource

This is similar to the Special Resource Definition dialog, which is described in Planning and Scheduling the Workload, except that here you are not updating the Resource Description database. When you change resource details with the MCP Resource Monitor, you are updating the current plan resource information, which is stored in the current plan extension (CX) dataset.

That is why this panel has extra fields that are not present in the Special Resource Definition dialog:

- Deviation (1)
- Available (2)
- Quantity (3)

If no change has been made to the current plan, Deviation is zero or blank (no deviation) and Available and Quantity are blank (they are specified in the current interval data, if any, or take the default value). If you have changed these fields, either with this panel, with the SRSTAT command, or with the EQQUSIN subroutine, the deviation is added to the quantity to give the actual quantity.
available and the changed availability overrides the scheduled availability until it is reset (set to blank).

**Note:** The resource details are retained in the current plan as long as any of the fields 1 to 3 are set (quantity or availability are non-blank, or deviation is not zero or blank). If you set TAPES unavailable, for example, using field 2, TAPES remains unavailable indefinitely, past daily planning EXTENDs and REPLANs, for weeks or even years. You must set them to blank (or, in the case of deviation, to zero) manually for the database values to take effect. When you change other fields on the panel, such as the default quantity, the value will be replaced by the value from the database the next time a daily planning EXTEND or REPLAN is run. Daily planning issues a warning message when it replaces a manually changed value with a value from the database.

The last line of the MODIFYING A SPECIAL RESOURCE panel in Figure 64 on page 97 shows how the resource came to be in the current plan:

**Last updated by**  
**How the resource was added**

**User ID**  
The resource details may have been added during batch daily planning because a planned operation references the resource. If there have been no manual alterations to the resource, this shows who last updated the database record.

If the resource details have been changed with the SRSTAT command, this shows who issued the command.

If an occurrence is added to the plan (using the MCP dialog, for example), and the resource was added to the plan because an operation in the occurrence references the resource, this shows who added the occurrence.

**SUBMIT**  
Tivoli OPC added the resource to the plan when it submitted an operation, and the resource is not in the database.

**DYNADD**  
Batch daily planning added the resource to the plan, because an operation referenced it, but it is not in the database.

To change the intervals, select option 1 (Intervals). You see this panel:

```
EQQQDIML ------- MODIFYING INTERVALS FOR A SPECIAL RESOURCE  ROW 1 TO 4 OF 4
Command ====> Scroll ====> PAGE

Enter any of the row commands below:
I(nn) - Insert, R(nn),RR(nn) - Repeat, D(nn),DD - Delete, or,
S - Work stations

Special resource : TAPES
Text : tape drives on CPU1 and STC1

Row Date From To Qty A
  cmd Time Time

```

Figure 65. EQQQDIML—Modifying Intervals for a Special Resource
When you change interval data in the current plan, this is not replaced by values from the database when you next run daily planning—a manually altered interval remains in the plan.

To change the workstations where operations can allocate the resource, enter the S row command beside the interval. You see the MODIFYING CONNECTED WORKSTATIONS FOR A SPECIAL RESOURCE panel, shown in Figure 66.

![Figure 66. EQQDWML—Modifying Connected Workstations for a Special Resource](image)

Change the connected workstations as required and press PF3 (End).
Chapter 7. Managing Tivoli OPC with Tivoli Global Enterprise Manager

This chapter describes how to use the OPC view in the Tivoli Global Enterprise Manager (Tivoli GEM) console, to monitor and manage the status of your Tivoli OPC installation. For information about how to install and configure the OPC instrumentation for Tivoli GEM, refer to Tivoli OPC Customization and Tuning.

What is Global Enterprise Manager

Tivoli Global Enterprise Manager is a unified solution for managing the large, cross-platform business applications that are used to run a company. It focusses on the way you perform on-going management tasks by ensuring the highest levels of service, enabling you to easily deploy, operate, and maintain the business-critical applications.

The most strategic business applications are commonly combinations of application components that run in both host and distributed environments. For example, an enterprise order-entry and processing system might have an order-entry application running on desktop PCs, a local processing application running on Windows NT servers, and a batch processing fulfillment application running on a mainframe. Tivoli GEM provides a management view that shows the relationship, data flow, and interdependencies among these application components. It also provides high-level management information about the IT resources, based on how they are performing in a business context.

Tivoli GEM uses a new business-focused approach to applications management, enabling you to manage the strategic business systems and to view and control all application components—and the underlying platforms—in a unified manner.

From the Tivoli GEM console, you can see all the application components that reside in your distributed and host (System/390) environments and the application relationships that make up a “business system.” This kind of view (known as a topology view) is dynamically updated by Tivoli GEM. It gives you at a glance the status of all the components and relationships of a business system. From that view, you can monitor, control, and configure application components, at a true enterprise level, without needing to understand the details of the underlying platforms.

To extend the benefits provided by Tivoli GEM many Tivoli modules, including Tivoli OPC, have been extended to provide “out-of-the-box” integration with Tivoli GEM.

Tivoli OPC Instrumentation for Tivoli GEM

Tivoli OPC has been instrumented to be managed by Tivoli GEM. This section describes how to manage Tivoli OPC from the Tivoli GEM console.

Tivoli GEM provides a specific business system for system management. Part of this business system consists of the Event Management subsystem and the Problem Management subsystem. Tivoli OPC instrumentation adds a new Scheduling Management tab to the systems management business system.
Regardless of the platforms on which they run, all scheduling products that are integrated with Tivoli GEM, including Tivoli OPC, can be managed from that Scheduling Management view. Tivoli GEM thus provides a complete overview of all the OPC components. It also provides a unique point from which you can monitor and manage them all.

The Tivoli OPC instrumentation for Tivoli GEM enables you to do the following:

- Display all the Tivoli OPC components, including controllers, stand-by controllers, MVS trackers, AS/400 trackers, TCP/IP-connected trackers, such as trackers for Microsoft Windows NT, IBM AIX, Sun Solaris, SunOS, HP-UX, and IBM OS/2.

- Show the different links between the above components. This provides, at a glance, a check on the status of the connections. For example, an MVS tracker that is running might have no open connection to the controller.

- Have a graphical management of monitors specific to each component being able to do the following:
  - Ask for the value of the monitor
  - Be notified when the value of the monitor changes
  - Associate a severity value (NORMAL, WARNING, INFORMATIONAL, SEVERE, or FATAL) with each monitor value

- Issue commands specific to each components such as:
  - Start or stop Tivoli OPC controller and trackers without logging them on.
  - Query the status of components.
  - Issue Tivoli OPC MVS modify commands.
  - Issue tracker agent commands.

- Know at a glance, in a sysplex environment, which is the active controller and which is the stand-by controller

Before you can use the Tivoli OPC instrumentation for Tivoli GEM, the Tivoli OPC instrumentation code must be installed and configured, as described in the Tivoli OPC Customization and Tuning manual.

Managing Tivoli OPC from the Tivoli GEM Console

Tivoli OPC instrumentation for Tivoli GEM adds the new Scheduling Management subsystem to the Systems Management business system.

For each heartbeat message sent by Tivoli OPC to the MVS system log, NetView routes a corresponding heartbeat event to Tivoli GEM, which creates a new icon in the Scheduling Subsystem view.

Topology View

The Scheduling Subsystem view shows the Tivoli OPC topology with all the OPC components, such as OPC controllers, OPC MVS trackers, OPC stand-by controllers, and OPC TCP/IP tracker agents, together with the controller–tracker links, independent of the platform on which they reside. Figure 67 on page 103 shows the Tivoli OPC components and controller–tracker links.
In the topology view, the status of each Tivoli OPC component is indicated by the component's background color. For example, a red background indicates Severe status and a yellow background indicates Warning status. The same is true for connections. Tracker agents connected to the Tivoli OPC controller by a black link are normal, whereas yellow links indicate a Warning condition. When a tracker agent is not connected to a Tivoli OPC controller, the OPC tracker agent is active but has not yet established a connection with a Tivoli OPC controller.

**Component Monitors**

Each Tivoli OPC component shown in the Scheduling topology view has a defined set of monitors. This set of monitors can change, depending on the component type, but it always contains at least the monitor for the component status.

You can display the set of monitors defined for a Tivoli OPC component by double-clicking on the component icon in the OPC topology view. Figure 68 on page 104 shows the view of monitors defined for a Tivoli OPC controller.
For the Tivoli OPC controller, seven monitors have been defined to monitor the general status of the OPC controller and the status of such OPC datasets as the Current Plan and the Database.

A red background to the Status and Current Plan Monitors would indicate a Severe condition, meaning that the Current Plan VSAM files may have been corrupted and that OPC is not able to run properly. The values of these monitors determine the status of the OPC components and are reflected in the Topology view of the OPC installation.

For each monitor, you can also set the polling interval and the threshold, and query monitor value and its threshold value, as shown in the pop-up menu in Figure 68.

### Connection Monitors

You can also define monitors for links between components. In the Tivoli OPC topology view, two monitors are defined for each link between a Tivoli OPC controller and an OPC tracker: the first monitors the number of completed jobs on that connection, the second monitors the number of in-error jobs on the connection. Figure 69 on page 105 shows the monitors defined in Tivoli OPC instrumentation for controller–tracker relationships.
Figure 69. Monitors Defined in Tivoli OPC Instrumentation for Controller–Tracker Relationships

As with other monitors, you can set thresholds on link monitors, so that for example if the number of In Error jobs on a connection exceed the defined threshold the status of the link changes accordingly.

Operational Tasks

Each Tivoli OPC component shown in the Scheduling topology view provides a set of operational tasks. These tasks represent commands you can execute on the selected component. Click with the right-mouse button on the selected component a pop-up menu to display the set of operational commands that you can execute for that component. The command you choose is executed on the selected component, regardless of the component type and of where the component resides if it is in the OS/390 or in a distributed environment.

Figure 70 on page 106 shows the set of operational tasks defined for the Tivoli OPC controller component.
Figure 70. The Set of Operational Tasks Defined for the Tivoli OPC Controller Component

If you click on the highlighted choice in the pop-up menu, an input dialog box is shown if the task requires additional input. Otherwise, the command is sent to the selected component to be executed in the target system. You can display command results or messages from the component by selecting the NetView console window from the toolbar.

In the Tivoli OPC instrumentation, the operational commands for all the Tivoli OPC components are sent first by Tivoli GEM to NetView in the OS/390 environment, then from NetView to the target component.

Operational commands for MVS controller and MVS tracker components are executed by NetView in the target OS/390 system by means of CLISTs or MVS modify commands.

Operational commands for Tivoli OPC tracker agents are sent by NetView to the target systems by means of a TCP/IP remote exec protocol.

Therefore, to run operational commands and get back the result, make sure that a connection between Tivoli GEM and NetView is active and that a TCP/IP connection is available between NetView and the target tracker agent system.

For this reason, no operational tasks are available for AS/400 tracker agents. Only the Query State task is available to check whether the component is up and running.
Appendix A. MVS Commands Supported by Tivoli OPC

You can start, stop, cancel, or modify Tivoli OPC using the following MVS operator commands:

S  START
P  STOP
C  CANCEL
F  MODIFY

In addition, you can use the MODIFY (F) command to start and stop individual subtasks.

You can enter these commands from a multiple console support (MCS) console or from a program such as the spool display and search facility (SDSF). In both cases, the terminal or console operator must have the required authority to enter operator commands.

Starting Tivoli OPC

To start Tivoli OPC, enter this MVS operator command:

```
START Tivoli OPC
S procname (procname = Tivoli OPC JCL procedure name)
```

If a Tivoli OPC started task with this name is already active, the second attempt to invoke it ends with an error message. If this happens, the started task in error cannot write an error message to the Tivoli OPC message log (ddname EQQMLOG) because the message log is already being used by the active started task. If Tivoli OPC is to run as a batch job, do not start it with an operator command. Instead, submit a batch job with the same name as the Tivoli OPC subsystem. JES starts this job in the same manner as any ordinary job.

**Note:** Because Tivoli OPC uses JES exits, among other things, to track the progress of MVS jobs, it does not start before JES is active.

Stopping Tivoli OPC

To stop Tivoli OPC, enter the following MVS operator command:

```
STOP Tivoli OPC
P procname (procname = Tivoli OPC JCL procedure name)
```

If you are stopping a controller, the controller creates a backup copy of the current plan dataset (if required) and ends all active functions.
When Tivoli OPC ends, it writes this message to the message log:

```
STOP message log
EQQZ086I NO ACTIVE Tivoli OPC SUBTASKS.
Tivoli OPC IS ENDING
```

## Canceling Tivoli OPC

If Tivoli OPC is still active 5 minutes after you enter the STOP operator command, you must cancel Tivoli OPC.

You may also need to cancel Tivoli OPC if the current plan is corrupt, because a normal shutdown causes a backup to the alternate file (refer to *Customization and Tuning* for details of current plan recovery). There are two ways to do this. The first is to enter:

```
(1) CANCEL Tivoli OPC
    C procname,DUMP (procname = Tivoli OPC JCL procedure name)
```

This causes Tivoli OPC to end with a dump on the SYSMDUMP file (if the ddname is in the started-task JCL). The second way is to enter:

```
(2) CANCEL Tivoli OPC
    C procname (procname = Tivoli OPC JCL procedure name)
```

This causes Tivoli OPC to end without a dump.

If the STOP command is ineffective and you have no earlier documentation of the problem, cancel Tivoli OPC with a dump so that the error can be located.

## Modifying Tivoli OPC

Using the MODIFY command, you can supply information to Tivoli OPC after it has started. The syntax of the MODIFY command is:

```
MODIFY Tivoli OPC
    F procname,modifyoption
```

where:

- `procname` Is the Tivoli OPC JCL procedure name
- `modifyoption` Is one of the following:

```
S=taskname
```

Start the specified Tivoli OPC subtask.
P=taskname
Stop the specified Tivoli OPC subtask.

taskname can be one of the following:

**APPC**  APPC subtask.
**AR**  Automatic recovery subtask.
**A4**  APPC tracker router subtask.
**DC**  Catalog management subtask.
**DRT**  Data router subtask.
**EMGR**  Event manager subtask.
**ERDR**  All active event-reader subtasks.
**EWTR**  Event writer subtask.
**EXA**  External router subtask.
**FL**  Fetch joblog task.
**GEN**  General service subtask.
**JCC**  Job-completion-checker subtask.
**NMM**  Normal-mode-manager subtask. The normal mode manager must be restarted as soon as possible after it has stopped. Many Tivoli OPC functions require an active NMM task to execute successfully.
**RODM**  RODM subtask.
**SUB**  Submit subtask.
**TA**  TCP/IP router subtask.
**VTAM**  Network communication function (NCF) subtask.
**WSA**  Workstation analyzer subtask.

Only the tasks in the Tivoli OPC subtask table can be activated by a MODIFY command. The subtask table is built when Tivoli OPC is started. This means that you can only start a task that has stopped earlier in the current session. If you attempt to start a started subtask or stop a stopped subtask, error message EQQZ049W is issued, and no action is taken.

**CPQSTA=ON**
Activates the STATMSG(CPLOCK) message.

**CPQSTA=OFF**
Deactivates the STATMSG(CPLOCK) message.

**DSPSTA**
Displays, in message EQQZ095, the status of statistics messaging. The message indicates whether messaging is active for EVENTS, CPLOCKS, GENSERV, and WSATASK. It also gives the values currently set for EVELIM and STATIM. For details, refer to Tivoli OPC Messages and Codes.

**EVELIM=nnnn**
Sets the new value of the EVELIM keyword of the JTOPTS statement. Allowed values are 0 to 9999.
EVESTA=ON
Activates the STATMSG(EVENTS) message.

EVESTA=OFF
Deactivates the STATMSG(EVENTS) message.

GENSTA=ON
Activates the STATMSG(GENSERV) message.

GENSTA=OFF
Deactivates the STATMSG(GENSERV) message.

HB(, TRK)
Sends a heartbeat message on the MVS system log for the OPC subsystem or for all trackers connected to that subsystem if it is an OPC controller. If GEM/OPC instrumentation is active, a heartbeat event is sent to the Tivoli GEM console and an icon is shown for the subsystem and for each active tracker agent connected to that subsystem.

JCLDBG=ON
Activates the single JCL trace. For each job handled by WSA task information, such as the elapsed time in milliseconds needed to handle the job, retrieve the JCL, access the JS VSAM, or whatever else, will be shown.

This is a powerful trace and should be activated only for short periods of time to identify possible performance problems.

JCLDBG=OFF
Deactivate the single JCL trace.

NEWDSLST
Order a Tivoli OPC tracker to rebuild the dataset triggering filter table EQQDSLST. The new table is read from member EQQDSLST of the dataset referenced to by the EQQJCLIB ddname in the started task JCL for the tracker. The new table replaces the table in ECSA.

NEWNOERR
Order a Tivoli OPC tracker to rebuild the NOERROR table, in the case NOERROR statements have been modified in the parameter library member that contains the JTOPTS statement.

NOERRMEM(member)
Order a Tivoli OPC tracker to rebuild the NOERROR table, in the case NOERROR statements have been modified in a parameter library member that was specified in an INCLUDE statement.

NOERRMEM(M1)
Order a Tivoli OPC tracker to delete all NOERROR codes defined by member M1, once you have previously changed M1 to contain only comments. The modified member can contain a different number of NOERROR codes than the original member.

Note: Tivoli OPC opens the EQQPARM library when Tivoli OPC is started and parameter library members (residing in library extents), that have been created, cannot be accessed, after have been opened. To avoid this problem, the datasets that define the EQQPARM library should be allocated without any secondary extents.
**QUELEN=nnnn**
Sets the new value of the QUEULEN keyword of the JTOPTS statement. Allowed values are 0 to 9999, but a minimum value of 5 is forced.

**STATIM=nn**
Sets the new value of the STATIM keyword of the JTOPTS statement. Allowed values are 0 to 99.

**STATUS**
Returns a message on the system log with the status of the OPC subsystem. The status can be one of the following:

- **FULLY_OPERATIONAL**
  Everything is active and is working properly.

- **PARTIALLY_OPERATIONAL**
  The OPC subsystem has limited functionality. For example, if a controller ER is stopped, the controller can still schedule jobs but cannot receive their statuses.

- **NOT_OPERATIONAL**
  The major subsystem functionality is not available. For example, a controller is not able to execute a plan or to submit a job.

**STATUS, DD=ddname**
Checks for the status of the Tivoli OPC dataset associated with the specified ddname. ddbname can be a specific DD name, such as EQQWSDS, EQQCP1DS, or EQQLTDS, or it can assume the value ALL, CP, DB, LTP, or JTL. It returns the return code of the last I/O operation performed on that ddbname. The status of the dataset can be one of the following:

- **NORMAL**
- **WARNING**
- **SEVERE**
- **CRITICAL**
- **UNKNOWN**

**STATUS, {OP_COMP | OP_ERR}, "destination name"**
Returns the number of completed operations (OP_COMP) or the number of ended-in-error operations (OP_ERR) for the specified tracker (destination name).

**Notes:**

1. If more than one workstation is defined for tracker tracker name, the number of completed or in-error operations is the sum of the operations on all the workstations defined on that tracker.

2. destination name is the destination name of a Tivoli OPC tracker, as specified in the ROUTOPTS keyword in the initialization statements.

**STATUS, SUBTASK**
Lists all Tivoli OPC subtasks with their statuses. The status can be **ACTIVE** or **INACTIVE**.
STATUS, TRK=(trkname | trktype)
Returns the status of a Tivoli OPC tracker agent defined for that controller. The status can be ACTIVE or INACTIVE, indicating whether an active session exists for that tracker agent.

trkname The tracker agent destination name defined in the Tivoli OPC PARM member, or the TCP/IP or SNA address of the tracker agent.

trktype The type of tracker agent. It can be TCP, SNA, APPC or ALL. If you specify one of these values, you will get the status for each defined tracker agent of that type. For example, if you specify TCP, you will get the status of all TCP/IP tracker agents you have defined for that controller; if you specify ALL, you will get the status of all trackers defined to that controller.

TAKEOVER
Order a standby controller to take over the functions of the controller. This command is valid only when both systems are part of the same XCF group, and no controller is active. You can use this command only for Tivoli OPC address spaces where OPCHOST(STANDBY) is specified on the OPCOPTS initialization statement.

Note: Takeover can occur automatically if you have specified the TAKEOVER keyword on the XCOPTS initialization statement of a standby system. Refer to Customization and Tuning for more information.

VSTRC=START
Start a trace on the message log of all VSAM I/O requests. In a busy Tivoli OPC system, you will need a large message-log dataset, and the trace will affect the performance of the Tivoli OPC system.

VSTRC=STOP
Stop a VSAM I/O request trace on the message log.

WSASTA=ON
Activates the STATMSG(WSATASK) message.

WSASTA=OFF
Deactivates the STATMSG(WSATASK) message.

After the STOP command is entered, the MODIFY command no longer functions, and gives this message on SYSLOG:
IEE324I MODIFY REJECTED - TASK BUSY

Modifying Tivoli OPC Data Store
Using the MODIFY command, you can supply information to the Tivoli OPC Data Store after it has started. The syntax of the MODIFY command is:

MODIFY Tivoli OPC
F procname,modifyoption

where:
modifyoption
  Is one of the following:

S=taskname
  Start the specified Data Store subtask.

P=taskname
  Stop the specified Data Store subtask.

  taskname can be one of the following:

  ARRD  Reader task.
  ARCU  Cleanup task.
  ARCM  Communication.
  ARDYWR Display number of active writers.
  ARDYTW Display WINTERVAL value.
  ARDYNY Display MAXSTOL value.
  ARDYNS Display MAXSYSL value.
  ARDYTU Display CINTERVAL value.
  ARDYN  Display MAXSYSL value.
  ARDYP  Display all initialization parameters values.
  ARMDWR Modify number of active writers.
  ARMDTW=n Modify WINTERVAL value (seconds).
  ARMDNY=n Modify MAXSTOL value (number of lines).
  ARMDNS=n Modify MAXSYSL value (number of lines).
  ARMDTU=n Modify CINTERVAL value (seconds).
  ARMDNS=n Modify MAXSYSL value (number of lines).
  ARDGCM=on/off Activate/deactivate Communication task traces.
  ARDGWR=on/off Activate/deactivate Writer task traces.
  ARDGRD=on/off Activate/deactivate Reader task traces.
  ARDGJQ=on/off Activate/deactivate JES Queue task traces.
  ARDGDB=on/off Activate/deactivate Data Base task traces.

Appendix A. MVS Commands Supported by Tivoli OPC 113
Appendix B. Status, Error, and Reason Codes

Tivoli OPC assigns a status code to every occurrence and every operation in the current plan. An error code is also assigned for any operation that ends in error. You can see the status of operations by using the Workstation Communication dialog or the Query Current Plan dialog. When the catalog management function is used, Tivoli OPC maintains status information that reports the progress of the catalog management action.

The codes assigned by Tivoli OPC are not just for documentation purposes. They report the real status of the operation and are used by several Tivoli OPC functions to make important decisions about the running of the operation.

Occurrence Status Codes

The occurrence status codes are:

- **C** Complete
- **D** Deleted
- **E** An operation in the occurrence has ended-in-error
- **P** A pending predecessor exists for the occurrence
- **S** Started
- **U** Undecided (the status is not known)
- **W** No operations in the occurrence have started.

Operation Status and Extended Status Codes

When Tivoli OPC displays the status of an operation, it uses the format xy, where x is the status code and y, if present, is the extended status code.

Operation Status Codes

The operation status codes are:

- **A** Arriving—the operation is ready for processing; no predecessors were defined
- **R** Ready for processing; all predecessors are complete
- **S** Started
- **C** Complete
- **D** Deleted
- **I** The operation is interrupted
- **E** The operation has ended-in-error
- **W** The operation is waiting for a predecessor to complete
- **U** Undecided—the operation status is not known.
Extended Status Codes

Together with the normal status codes, Tivoli OPC maintains extended status codes that provide additional information about the status of operations. The extended status code is not always present.

The following extended status codes are valid, depending on the type and status of the operation:

- Valid for all operations that have a status of arriving (A) or ready (* or R):
  - X Waiting for resource
  - H A dialog user has used the HOLD command on the operation
  - N A dialog user has used the NOP command on the operation

- Valid for all operations that have a status of arriving (A), ready (* or R), started (S), or error (E):
  - M The status of the operation has been manually set by a dialog user from the ready list.

- Valid only for computer workstation operations that have a status of arrived (A) or ready (* or R):
  - T Waiting until a particular time
  - L The operation is a late time-dependent operation with the suppress-if-late attribute
  - R The operation has ended in error but was automatically reset (the completion code is defined in the installation options to be automatically reset)
  - E An error occurred during job submission or release
  - D Closedown in progress.
  - Blank OPC is in the process of submitting this job. OPC is waiting for the availability of a parallel server or a critical resource, or the operation is not to be submitted automatically.

- Valid only for computer workstation operations that have a status of started (S):
  - Q The job has been added to the JES job queue
  - S The job or started task is executing
  - M The status of the job or started task has been manually set to S
  - U Submit in progress
  - Blank The job has been successfully submitted but has not yet been reported as added to the JES job queue.

- Valid only for computer workstation operations that have a status of ready (R) or error (E):
  - A The job is waiting for a deferred catalog management action to be initiated or discarded by a dialog user
  - C The job is waiting for the catalog management action to be completed.
Error Codes

Tivoli OPC assigns error codes to certain operations and to job and started task steps. These codes are used by the automatic job recovery function to decide a recovery action.

**CAN**  The job or started task was canceled by the operator or by a TSO user before execution. This code is also possible if the job-termination event (type 3P) is missing.

**CCUN**  The completion code is unknown. The job or started task has ended, but no completion code is available. This code is also possible if the job-end event (type 3J) is missing.

Check the job log and SYSLOG.

**JCCE**  An error during JCC (job completion checker) processing prevented the JCC from determining an error code for the operation.

**JCL**  A JCL error was recognized after the job or started task began to execute, or a JCL error was recognized after syntax checking in the internal reader.

**JCLI**  A JCL error occurred immediately; that is, the error was detected before the job or started task began. This code is also possible when both the job-start event (type 2) and the job-end event (type 3J) are missing.

**MCP**  The operation was manually set to error in the MCP dialog.

**OFxx**  The system that the operation is defined on has gone offline. The WSOFFLINE parameter on the JTOPTS initialization statement specifies that started operations should be marked as ended-in-error. xx is the status and extended status of the failing operation. Operations that were running (status SS) have a step-code error status of OFFL.

**OJCV**  An error occurred during JCL-variable substitution when the job or started task was submitted, or Tivoli OPC detected an error in the RECOVER statement during automatic recovery. Browse the JCL for the operation or the EQQMLOG dataset to find more information about the failure.

**OSEQ**  A job or started task began to execute before all its predecessors had completed. This can occur only if the job was not submitted by Tivoli OPC and if either HOLDJOB(NO) or HOLDJOB(USER) is specified for the Tivoli OPC event writer options. Refer to Customization and Tuning.

**OSUB**  A failure occurred when Tivoli OPC attempted to submit a job or start a started task. In the case of a started task, it could be that the started task is a subsystem that is not started by JES, or the Tivoli OPC subsystem EQQSTC ddname is not allocated to a JES-defined procedure library. The operation should be marked as ended-in-error.

**OSUF**  A failure occurred when Tivoli OPC attempted to retrieve the JCL for a job or started task. This code is set if the SUBFAILACTION keyword of the JTOPTS initialization statement specifies that the operation should be marked as ended-in-error. This code is also caused if you have JOBCHECK(SAME) and the job name in the application description does not match the one on the job card.

**OSUP**  A time operation is late, and the SUPPRESSACTION parameter of the JTOPTS initialization statement specified that the operation should be marked as ended-in-error.
OSxx  The system on which the operation is defined has failed. The WSFAILURE parameter on the JTOPTS initialization statement specifies that started operations should be marked as ended-in-error. xx is the status and extended status of the failing operation. Operations that were running (status SS) have a step-code error of OSYS.

PCAN  A print operation was canceled by the operator.

nnnn  Step return code.

Sxxx  System abend code.

Uxxx  User abend code in hexadecimal notation. For example user abend 2750 is represented in Tivoli OPC as UABE.

xxxx  User-defined error code.

---

**Catalog Management Status Codes**

When the catalog management function is used, Tivoli OPC maintains status information to report on the progress of the catalog management action. The following CM status codes are possible:

- **Z** Cancelled—catalog management status is not available. Examine the Tivoli OPC message log for CM errors; one possibility is that the dataset collection at the tracker has failed.

- **E** Catalog management action has failed.

- **I** Catalog management has been initiated but has not yet started.

- **S** The catalog management action has started.

- **M** The catalog management action is in progress but is delayed waiting for a response from DFHSM.

- **R** A catalog management action has been restarted. This status is possible if either the controller or tracker stopped while a CM action was in progress.

- **C** Actions completed normally.

- **D** The catalog management action has been discarded by a dialog user or the pre-catalog management exit, EQQUX008.

- **X** Excluded—Tivoli OPC has determined that there are no datasets defined by DD statements in the JCL that require catalog management actions. This status has the same meaning as complete.
Job Log Retrieval Status Codes

When the job log retrieval function is used, Tivoli OPC maintains status information to report on the retrieval of the log. The following status codes are possible:

- **C** Completed—the controller has received the log.
- **E** Error. There was an error retrieving the log.
- **I** Initiated. The controller has sent a retrieval request to the tracker, but the tracker has not yet processed the request.
- **S** Started. The controller has sent a retrieval request to the tracker, and the tracker has started to retrieve the log.
- **blank** The controller has not sent any retrieval request to the tracker.

Operation reason codes

If your ready list layout includes the RSNC field, you can see these operation reason codes. Note that the codes are listed in hierarchical order. For example, if job submission failed, and job submission is deactivated, code **D** is obtained—not code **F**.

- **D** Job submission deactivated
- **C** Workstation is closed
- **P** All parallel servers in use
- **A** Automatic reset error condition
- **F** Job submission failed
- **J** No automatic job submission
- **L** Job is late
- **T** Start time not reached
- **1** Not enough free WS resource 1
- **2** Not enough free WS resource 2
- **H** Closedown in progress.
- **S** Waiting for special resource.
Appendix C. Fields Displayed in Ready and Error Lists

This appendix contains a list of fields that you can display in the ready list and the error handling list. In this table, **Length** is the maximum length (in bytes) of the data in the field.

<table>
<thead>
<tr>
<th>Column title</th>
<th>Length</th>
<th>Description of column contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>PR#</td>
<td>04</td>
<td>Number of predecessor operations</td>
</tr>
<tr>
<td>PS#</td>
<td>03</td>
<td>Number of parallel servers required by the operation</td>
</tr>
<tr>
<td>UPR#</td>
<td>04</td>
<td>Number of uncompleted predecessor operations</td>
</tr>
<tr>
<td>R1#</td>
<td>03</td>
<td>Number of 1st WS resources required by the operation</td>
</tr>
<tr>
<td>R2#</td>
<td>03</td>
<td>Number of 2nd WS resources required by the operation</td>
</tr>
<tr>
<td>SR#</td>
<td>03</td>
<td>Special resources referenced by the operation</td>
</tr>
<tr>
<td>SU#</td>
<td>04</td>
<td>Number of successor operations</td>
</tr>
<tr>
<td>Arrived</td>
<td>08</td>
<td>Operation arrival date, actual if arrived</td>
</tr>
<tr>
<td>Ard</td>
<td>03</td>
<td>Operation arrival day, actual if arrived</td>
</tr>
<tr>
<td>time</td>
<td>05</td>
<td>Operation arrival time, actual if arrived</td>
</tr>
<tr>
<td>Application</td>
<td>16</td>
<td>Application ID</td>
</tr>
<tr>
<td>Act dur</td>
<td>07</td>
<td>Actual duration of the operation</td>
</tr>
<tr>
<td>A</td>
<td>01</td>
<td>Automatic error-completion indicator, Y or N</td>
</tr>
<tr>
<td>Ended</td>
<td>08</td>
<td>End date of the operation, or blank</td>
</tr>
<tr>
<td>End</td>
<td>03</td>
<td>End day of the operation, or blank</td>
</tr>
<tr>
<td>time</td>
<td>05</td>
<td>End time of the operation, or blank</td>
</tr>
<tr>
<td>H</td>
<td>01</td>
<td>Automatic hold/release indicator, Y or N</td>
</tr>
<tr>
<td>Owner</td>
<td>16</td>
<td>Application owner ID</td>
</tr>
<tr>
<td>Started</td>
<td>08</td>
<td>Actual start date of the operation</td>
</tr>
<tr>
<td>Std</td>
<td>03</td>
<td>Actual start day of the operation</td>
</tr>
<tr>
<td>time</td>
<td>05</td>
<td>Actual start time of the operation</td>
</tr>
<tr>
<td>S</td>
<td>01</td>
<td>Automatic job submission, Y or N</td>
</tr>
<tr>
<td>Application text</td>
<td>24</td>
<td>Verbal description of the application</td>
</tr>
<tr>
<td>CM-A</td>
<td>01</td>
<td>Catalog management action for operation</td>
</tr>
<tr>
<td>JLOG</td>
<td>01</td>
<td>Status of joblog retrieval</td>
</tr>
<tr>
<td>CM-S</td>
<td>01</td>
<td>Catalog management status for operation</td>
</tr>
<tr>
<td>C</td>
<td>01</td>
<td>Critical path, F-1st (Y=on path, N=not)</td>
</tr>
<tr>
<td>S</td>
<td>01</td>
<td>Current status of the operation</td>
</tr>
<tr>
<td>T</td>
<td>01</td>
<td>Dependency type, P=predecessor, S=successor</td>
</tr>
<tr>
<td>W</td>
<td>01</td>
<td>Deadline WTO, Y or N</td>
</tr>
<tr>
<td>E dur</td>
<td>05</td>
<td>Estimated duration of the operation</td>
</tr>
<tr>
<td>Errc</td>
<td>04</td>
<td>Error code of the operation, otherwise blank</td>
</tr>
<tr>
<td>Destination</td>
<td>08</td>
<td>Execution destination ID</td>
</tr>
<tr>
<td>Form no.</td>
<td>08</td>
<td>Form number or blank</td>
</tr>
<tr>
<td>Authgrp</td>
<td>08</td>
<td>Authority group</td>
</tr>
<tr>
<td>Ia date</td>
<td>08</td>
<td>Application input arrival date (after MCP)</td>
</tr>
<tr>
<td>Ia day</td>
<td>03</td>
<td>Application IA day, (&lt;) preceding month, (&gt;) next</td>
</tr>
<tr>
<td>Ia time</td>
<td>05</td>
<td>Application input arrival time (after MCP)</td>
</tr>
<tr>
<td>I start</td>
<td>08</td>
<td>Intermediate start date after interrupt</td>
</tr>
<tr>
<td>IST</td>
<td>03</td>
<td>Intermediate start day after interrupt</td>
</tr>
<tr>
<td>time</td>
<td>05</td>
<td>Intermediate start time after interrupt</td>
</tr>
<tr>
<td>Jobname</td>
<td>08</td>
<td>Job name</td>
</tr>
<tr>
<td>C</td>
<td>01</td>
<td>Job, SYSOUT class, or blank</td>
</tr>
<tr>
<td>Job id</td>
<td>08</td>
<td>JES job ID</td>
</tr>
<tr>
<td>S</td>
<td>01</td>
<td>Job status (I) init, (H) held, (Q) released</td>
</tr>
<tr>
<td>L</td>
<td>01</td>
<td>Latest out time passed</td>
</tr>
<tr>
<td>Last out</td>
<td>08</td>
<td>Latest out date of the operation</td>
</tr>
<tr>
<td>Lo</td>
<td>03</td>
<td>Latest out day of the operation</td>
</tr>
</tbody>
</table>

© Copyright IBM Corp. 1991, 1999
<table>
<thead>
<tr>
<th>Column title</th>
<th>Length</th>
<th>Description of column contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>time</td>
<td>05</td>
<td>Latest out time of the operation</td>
</tr>
<tr>
<td>Last upd</td>
<td>08</td>
<td>Time stamp of last MCP update in the format MMDDHHmm (month, day, hour and minutes)</td>
</tr>
<tr>
<td>H</td>
<td>01</td>
<td>Operation manually HELD by user</td>
</tr>
<tr>
<td>N</td>
<td>01</td>
<td>Operation NOPed by user</td>
</tr>
<tr>
<td>no.</td>
<td>03</td>
<td>Operation number</td>
</tr>
<tr>
<td>Deadline</td>
<td>08</td>
<td>Deadline date of the operation</td>
</tr>
<tr>
<td>DI</td>
<td>03</td>
<td>Deadline day of the operation</td>
</tr>
<tr>
<td>time</td>
<td>05</td>
<td>Deadline time of the operation</td>
</tr>
<tr>
<td>Plan ia</td>
<td>08</td>
<td>Planned input arrival date of operation</td>
</tr>
<tr>
<td>ia</td>
<td>03</td>
<td>Planned input arrival day of operation</td>
</tr>
<tr>
<td>Oi</td>
<td>01</td>
<td>Operator instructions available</td>
</tr>
<tr>
<td>time</td>
<td>05</td>
<td>Planned input arrival time of operation</td>
</tr>
<tr>
<td>Plan end</td>
<td>08</td>
<td>Planned end date of the operation</td>
</tr>
<tr>
<td>End</td>
<td>03</td>
<td>Planned end day of the operation</td>
</tr>
<tr>
<td>time</td>
<td>05</td>
<td>Planned end time of the operation</td>
</tr>
<tr>
<td>P</td>
<td>01</td>
<td>On preparation workstation, Y or N</td>
</tr>
<tr>
<td>P</td>
<td>01</td>
<td>Priority of the operation</td>
</tr>
<tr>
<td>Pl start</td>
<td>08</td>
<td>Planned start date of the operation</td>
</tr>
<tr>
<td>St.</td>
<td>03</td>
<td>Planned start day of the operation</td>
</tr>
<tr>
<td>time</td>
<td>05</td>
<td>Planned start time of the operation</td>
</tr>
<tr>
<td>Transp</td>
<td>06</td>
<td>Transport time, from previous WS to this WS</td>
</tr>
<tr>
<td>Rdr date</td>
<td>08</td>
<td>Date the reader recognized the job card</td>
</tr>
<tr>
<td>Rdr t</td>
<td>05</td>
<td>Time the reader recognized the job card</td>
</tr>
<tr>
<td>L</td>
<td>01</td>
<td>Time job suppressed if late, Y or N</td>
</tr>
<tr>
<td>R</td>
<td>01</td>
<td>Reroutable operation, Y or N</td>
</tr>
<tr>
<td>R</td>
<td>01</td>
<td>Restartable operation, Y or N</td>
</tr>
<tr>
<td>RSNC</td>
<td>01</td>
<td>Reason code</td>
</tr>
<tr>
<td>St</td>
<td>02</td>
<td>Operation status, current and extended status</td>
</tr>
<tr>
<td>T</td>
<td>01</td>
<td>Time job indicator, Y or N</td>
</tr>
<tr>
<td>Operation text</td>
<td>24</td>
<td>Verbal description of the operation</td>
</tr>
<tr>
<td>User field</td>
<td>16</td>
<td>Operation user field</td>
</tr>
<tr>
<td>U</td>
<td>01</td>
<td>Urgency indicator</td>
</tr>
<tr>
<td>RE</td>
<td>01</td>
<td>Operation rerouted, Y or N</td>
</tr>
<tr>
<td>ws</td>
<td>04</td>
<td>Workstation</td>
</tr>
<tr>
<td>T</td>
<td>01</td>
<td>Workstation type</td>
</tr>
<tr>
<td>WTO ws</td>
<td>01</td>
<td>Workstation is a WTO</td>
</tr>
<tr>
<td>Dur.</td>
<td>07</td>
<td>Actual duration, if completed, otherwise estimated</td>
</tr>
<tr>
<td>Inp arr</td>
<td>08</td>
<td>Operation IA date, actual if arrived</td>
</tr>
<tr>
<td>ia</td>
<td>03</td>
<td>Operation IA day, actual if arrived, otherwise planned</td>
</tr>
<tr>
<td>time</td>
<td>05</td>
<td>Operation IA time, actual if arrived</td>
</tr>
<tr>
<td>X</td>
<td>01</td>
<td>Extended status of the operation</td>
</tr>
<tr>
<td>Ended</td>
<td>08</td>
<td>End date, actual if ended, otherwise planned</td>
</tr>
<tr>
<td>End</td>
<td>3</td>
<td>End day, actual if ended, otherwise planned</td>
</tr>
<tr>
<td>time</td>
<td>05</td>
<td>End time, actual if ended, otherwise planned</td>
</tr>
<tr>
<td>Start</td>
<td>08</td>
<td>Operation start date, actual if started</td>
</tr>
<tr>
<td>St</td>
<td>03</td>
<td>Operation start day, actual if started</td>
</tr>
<tr>
<td>time</td>
<td>05</td>
<td>Operation start time, actual if started</td>
</tr>
</tbody>
</table>
Glossary

A

ABARS. See Aggregate Backup and Recovery Support.

active application description. An application description that is complete and ready for use in planning or scheduling.

actual duration. At a workstation, the actual time in hours and minutes it takes to process an operation from start to finish.

adjusted quantity. The current quantity of a special resource, taking the deviation into account.

AD. See application description.

Aggregate Backup and Recovery Support (ABARS). A DFHSM facility that manages backup and recovery of user-defined data set groups (aggregates). Aggregate backup copies and related control information are written as portable data and control files on 3480 or 3420 volumes.

Advanced Program-to-Program Communications (APPC). An implementation of the Systems Network Architecture (SNA), logical unit (LU) 6.2 protocol that allows interconnected systems to communicate and share the processing of programs.

all-days cyclic period. A cyclic period where all days are counted when calculating the interval.

alert. Two Workload Monitor/2 objects, Operations List and Workstations List, can be used to monitor a Tivoli OPC subsystem and notify you if alert conditions are met. The alert can be a sound (Beep), or a message in a window (Message). The Details view of the Plan object must be open to monitor for plan alerts. The List or Icons views of the Operations List object must be open to monitor for operation alerts.

APAR. Authorized program analysis report. A report of a problem that is suspected to be caused by a defect in a current, unaltered release of a program.

API. See application programming interface.

APPC. See Advanced Program-to-Program Communications.

application. A measurable and controllable unit of work that completes a specific user task, such as the running of payroll or financial statements. The smallest entity that an application can be broken down into is an operation. Generally, several related operations make up an application.

application description (AD). A database description of an application.

application group. Type of application description which holds run cycle and calendar information for standard applications or job descriptions which have been defined as a member of the group.

application ID. The name of an application. (For example, PAYROLL or DAILYJOBS.)

application programming interface (API). A formally-defined programming language interface between an IBM system control program or a licensed program and the user of a program.

application transaction program (ATP). A program that uses the Advanced Program-to-Program Communications (APPC) application programming interface (API) to communicate with a partner program at a remote node.

application version. See versions.

ATP. See application transaction program.

authority. The ability to access a protected resource.

authority group. A name used to generate a RACF resource name for authority checking.

automatic events. Events recognized by or triggered by an executing program. Automatic events are usually generated by Tivoli OPC tracking programs but can also be created by a user-defined program.

automatic hold/release. Function used to control jobs that are submitted outside Tivoli OPC. It allows you to define whether such jobs should be automatically released at the appropriate time if placed in HOLD status when submitted.

automatic job and started-task recovery. A Tivoli OPC function that lets you specify, in advance, alternative recovery strategies for operations that end in error.

automatic-reporting workstation. A workstation (for example, a processor or printer) that reports events (the starting and stopping of operations) in real time to Tivoli OPC.
availability. The degree to which a system (and in Tivoli OPC, an application) or resource is ready when needed to process data.

B

batch loader. A Tivoli OPC batch program that you can use to create and update information in the application-description and operator-instruction databases.

buffer. A memory area reserved for performing input/output (I/O) operations.

BMP. Batch message processing.

C

calendar. The data that defines the operation department's work time in terms of work days and free days.

capacity. The actual number of parallel servers and workstation resources available during a specified open interval.

capacity ceiling. The maximum number of operations that a workstation can handle simultaneously.

catalog. A directory of files and libraries, with reference to their locations. A catalog may contain other information such as the types of devices in which the files are stored, passwords, blocking factors.

catalog management. Catalog management is a recovery function of Tivoli OPC, which handles the deleting or uncataloging of datasets created in a job operation that ends in error.

CICS. Customer Information Control System.

closed workstation. A workstation that is unavailable to process work for a specific time, day, or period.

Common Programming Interface (CPI). A consistent set of specifications for languages, commands, and calls to enable applications to be developed across all Systems Application Architecture (SAA) environments.

complete (C). The status of an operation indicating that it has finished processing.

completion code. A Tivoli OPC system code that indicates how the processing of an operation ended at a workstation. See error code.

complex of processors. A JES2 Multi-Access Spool system or a JES3 system with more than one processor.

computer workstation. (1) A workstation that performs MVS processing of jobs and started-task operations, and that usually reports status to Tivoli OPC automatically. (2) A processor used as a workstation. It can refer to single processors or multiprocessor complexes serving a single job queue (for example, JES2 or JES3 systems).

contingency plan. A plan for emergency response, backup procedures, and post-disaster recovery. Synonymous with disaster recovery plan, emergency plan.

controller. The Tivoli OPC component that runs on the controlling system, and that contains the Tivoli OPC tasks that manage the Tivoli OPC plans and databases.

controlling system. The system that the controller runs on.

control on servers. If a workstation is defined with control on servers, OPC/ESA will not start more operations at the workstation than there are available servers.

conversation. In Advanced Program-to-Program Communications (APPC), a connection between two transaction programs over a logical unit-logical unit (LU-LU) session that allows them to communicate with each other while processing a transaction.

conversation verb. In Advanced Program-to-Program Communications (APPC), one of the verbs a transaction program issues to perform transactions with a remote program.

CP. See current plan.

CPI. See Common Programming Interface.

CPI-C. Common Programming Interface for Communications. See also Common Programming Interface.

cross-system coupling facility (XCF). MVS components and licensed programs use the XCF services to provide additional functions in a SYSPLEX.

critical path. The route, within a network, with the least slack time.

current plan (CP). A detailed plan of system activity that covers a period of at least 1 minute, and not more than 21 days. A current plan typically covers 1 or 2 days.

cyclic interval. The number of days in a cyclic period.

cyclic period. A period that represents a constant number of days. There are two types of cyclic periods:
• Work-days-only cyclic period, where only the work days are counted when calculating the number of days in the period.
• All-days cyclic period, where all days are counted.

D

daily planning. The process of creating a current plan.

DASD. Direct access storage device.

database. A collection of data that is fundamental to a system. Tivoli OPC uses six databases: calendar, period, workstation description, JCL variable table, application description, and operator instruction.

Data Facility Hierarchical Storage Manager (DFHSM). A licensed MVS program which provides automatic and command functions that manage user storage space and data recovery.

Data Facility Systems Management Subsystem/MVS (DFSMS/MVS). A group of licensed MVS programs which transform system environments from user-managed DASD volumes to administrator-controlled, system-managed data sets.

Data Lookaside Facility (DLF). The MVS/ESA component that manages Hiperbatch objects.

data processing center (DP center). A center or department, including computer systems and associated personnel, that performs input, processing, storage, output, and control functions to accomplish a sequence of operations on data.

Data Store. The Tivoli OPC component managing the job runtime information at the tracked system. It is dedicated to the storing and possible retrieval of sysout datasets belonging to OPC-submitted jobs, to optimize the sysout availability.

DB2. DATABASE 2.

DBCS. Double-byte character set.

ddname. Data definition name.

deadline. See deadline date and deadline time.

deadline date. The latest date by which an occurrence must be complete.

deadline time. The latest time by which an occurrence must be complete.

deadline WTO message. You can specify that Tivoli OPC issue an operator message (EQQW776I) when a started operation has not been marked as completed before the deadline time. In addition to the standard message, the user-defined text that describes the operation is issued as part of the WTO.

default calendar. (1) A calendar that you have defined for Tivoli OPC to use when you do not specify a calendar in an application description. (2) A calendar that Tivoli OPC uses if you have neither specified a calendar in an application description, nor defined your own default calendar.

dependency. A relationship between two operations in which the first operation must successfully finish before the second operation can begin.

descriptive text. User-written text describing the operation. This text is also issued as part of the write-to-operator message if the operation has been started, exceeds its deadline, and has the deadline write-to-operator (WTO) option specified.

Details notebook. See Details view.

Details view. A view of a Workload Monitor/2 object showing details about the object. The Details view of the Plan object shows information about the current plan. The Details view of the Operation object shows information about the selected operation. The Details view of the Workstation object shows information about the selected workstation.

deviation. A temporary variation in the quantity of a special resource.

DFHSM. See Data Facility Hierarchical Storage Manager.

DFSMS/MVS. See Data Facility Storage Management Subsystem.

dialog. The user's online interface with Tivoli OPC.

Disaster Recovery Plan (DRP). A plan for emergency response, backup procedures, and post-disaster recovery. Synonymous with contingency plan, emergency plan.

DLF. See Data Lookaside Facility.

DP center. See data processing center.

DRP. See Disaster Recovery Plan.

duration. The length of time an operation is active at a workstation.
end user. A person who uses the services of the data processing center.

ended-in-error (E). The Tivoli OPC reporting status for an operation that has ended in error at a workstation.

error code. A code set by Tivoli OPC to describe how the processing of an operation ended at a computer workstation.

ETT. See event-triggered tracking.

estimated duration. The estimated length of time an operation will use a workstation. This is initially based on a value that is provided when the operation is defined, but can be adjusted automatically by Tivoli OPC’s feedback mechanism to reflect actual durations.

event. An action that changes an operation’s status and changes the current plan.

event manager. The Tivoli OPC function that processes all tracking events and determines which of these are Tivoli OPC-related.

event reader. A Tivoli OPC task that reads event records from an event dataset.

event tracking. A function of Tivoli OPC that follows events in the operations department in real time and records status changes in the current plan.

event-triggered tracking (ETT). A component of Tivoli OPC that waits for specific events to occur, and then adds a predefined application to the current plan. ETT recognizes two types of events: the reader event, which occurs when a job enters the JES reader, and the resource event, which occurs when the availability status of a special resource is set to “yes”.

event writer. A Tivoli OPC task that writes event records in an event dataset.

exclusive resource. A resource that can be used by only one operation at a time.

expected arrival time. The time when an operation is expected to arrive at a workstation. It can be calculated by daily planning or specified in the long-term plan.

extended status code. Together with the normal status codes, Tivoli OPC maintains extended status codes that provide additional information about the status of operations. The extended status code is not always present.

external dependency. A relationship between two occurrences, in which an operation in the first occurrence (the predecessor) must successfully finish before an operation in the second occurrence (the successor) can begin processing.

feedback limit. A numeric value in the range 100–999 that defines the limits within which actual data that is collected in tracking is fed back and used by Tivoli OPC.

filter criteria. Input values that are used to limit the mass update of applications to only those specified. This term is used in the Tivoli OPC ISPF dialogs.

first critical operation. An operation of an occurrence that has the earliest latest-start-time. The first critical operation of an occurrence determines the critical path.

first operation. (1) An operation in an occurrence that has no internal predecessor. (2) The start node in a network.

fixed resources. A set of resource names used to check the authority of users to access the Tivoli OPC dialogs.

form number. A user-defined code that identifies the type of paper to be used for an operation on a printer workstation. Tivoli OPC can use the form number to identify the different print operations belonging to one job.

free day. Any day that is not a work day.

free-day rule. A rule that determines how Tivoli OPC will treat free days when the application run day falls on a free day.

general workstation. A workstation where activities other than printing and processing are carried out. A general workstation reporting to Tivoli OPC is usually manual, but it can also be automatic. Manual activities can include data entry and job setup.

generic alert. An alert that is broadcast by Tivoli OPC, and collected by NetView, when an operation ends in error. You can specify this as an option when defining application descriptions.

global search character. In Tivoli OPC, a percent sign (%), which represents any single character, or an asterisk (*), which represents any character string of any length.

global variable table. The JCL variable table that Tivoli OPC checks for a variable substitution value if no
value is found in the specific JCL variable table that is associated with the operation.

**Graph view.** (1) A view of the Workload Monitor/2 Workstation object. Shows the total number of operations with different statuses for a single workstation. (2) In the Graphical User Interface for Application Description, a view of the operations that make up an application. It shows the workstation where each operation is run, and dependencies between the operations.

**Graphs view.** A view of the Workload Monitor/2 Workstations List object. Shows the total number of operations with different statuses for each of the workstations that are included in the object.

**group definition.** The application group to which the application description or job description is a member.

**H**

**highest return code.** A numeric value in the range 0–4095. If this return code is exceeded during job processing, the job will be reported as ended-in-error.

**Hiperbatch.** The MVS/ESA facility that stores VSAM and QSAM data in Hiperspace for access by multiple jobs. The facility can significantly reduce the execution time of certain batch streams that access VSAM and QSAM data sets.

**Hot standby.** Using the MVS/ESA cross-system coupling facility (XCF), you can include one or more standby controllers in your configuration. A standby system can take over the functions of a controller if the controller fails or if the MVS/ESA system that it was active on fails.

**I**

**Icons view.** The Workload Monitor/2 objects, Workstations List and Operations List, contain other objects. The Icons view shows an icon for each contained object.

**IMS.** Information Management System.

**incident log.** An optional function available under the job completion checker.

**initiator/terminator.** The job scheduler function that selects jobs and job steps to be executed, allocates input/output devices for them, places them under task control, and at completion of the job, supplies control information for writing job output on a system output unit.

**in-progress operation.** An operation with a status of A, R, *, I, E, or S.

**input arrival time (IAT).** The user-defined date and time when an operation or an application is planned to be ready for processing.

**interrupted (I).** A Tivoli OPC reporting status for an operation that indicates that the operation has been interrupted while processing.

**ISPF.** Interactive System Productivity Facility.

**J**

**JCC.** See job completion checker.

**JCL.** Job control language. A problem-oriented language designed to express statements in a job that are used to identify the job or describe its requirements to an operating system.

**JCL tailoring.** Tivoli OPC provides automatic JCL tailoring facilities, which enable jobs to be automatically edited using information that is provided at job setup or submit.

**JCL variable table.** A group of related JCL variables. See variable table.

**JES.** Job entry subsystem. A system facility for spooling, job queuing, and managing I/O.

**job.** (1) A set of data that completely defines a unit of work for a computer. A job usually includes all necessary computer programs, linkages, files, and instructions to the operating system. (2) In Tivoli OPC, an operation performed at a computer workstation.

**job class.** Any one of a number of job categories that can be defined. By classifying jobs and directing initiators to initiate specific classes of jobs, it is possible to control a mixture of jobs that can be run concurrently.
job-completion checker (JCC). An optional function of Tivoli OPC that allows extended checking of the results from CPU operations.

job description. A single processor (job or started-task) operation and its dependencies.

Job Description dialog. The ISPF dialog used to create job descriptions.

job ID. The JES job ID of the job associated with the operation.

job name. The name of the job associated with an operation. The job name is assigned in the JOB statement of a job. It identifies the job to the system.

job preparation. Job preparation involves modifying jobs in preparation for processing. This can be performed manually, by a job preparer, or automatically by Tivoli OPC JCL tailoring functions.

job setup. The preparation of a set of JCL statements for a job at a job setup workstation. Job setup can be performed manually by an operator, or automatically by Tivoli OPC.

job setup workstation. A general workstation defined with the job setup option. A job setup workstation lets you modify your job or STC JCL before execution.

job submission. A Tivoli OPC process that presents jobs to MVS for running on a Tivoli OPC-defined workstation once the scheduling criteria for the operation is met.

job tracking. A Tivoli OPC process that communicates with operating systems that control computer workstations.

JS. The JCL repository dataset.

layout. In the Graphical User Interface for Application Description, a user-created file that determines which information about each application is displayed when you view a list of application descriptions. An application description contains many details about the application, such as application ID, valid to date, application status, and last user. A layout specifies which details the user wishes to view.

layout ID. A unique name that identifies a specific ready or error list layout.

limit for feedback. See feedback limit.

list, application. In the Graphical User Interface for Application Description, a list of application definitions from which the user can select one to work with. It consists of application definitions selected according to user-specified criteria.

List view. The Workload Monitor/2 objects Workstations List and Operations List contain other objects. The List view shows a list of the contained object and displays data about each contained object.

local. Synonym for channel-attached.

local processor. (1) In a complex of processors under JES3, a processor that executes users' jobs and that can assume global functions if the global processor fails. (2) In Tivoli OPC, a processor in the same installation that communicates with the controlling Tivoli OPC processor through shared DASD or XCF communication links.

logical unit (LU). In Systems Network Architecture (SNA), a port through which an end user accesses the SNA network in order to communicate with another end user and through which the end user accesses the functions provided by system services control points (SSCPs).

logical unit 6.2 (LU 6.2). A type of Systems Network Architecture (SNA) logical unit (LU) for communication between peer systems. Synonymous with APPC protocol, see Advanced Program-to-Program Communications (APPC).

long-term plan (LTP). A high-level plan of system activity that covers a period of at least 1 day, and not more than 4 years. It serves as the basis for a service level agreement with your users, and as input to daily planning.

LU. See logical unit.

LU-LU session type 6.2. See logical unit 6.2.

LTP. See long-term plan.
**M**

**manipulation button.** One of the two mouse buttons. With default mouse settings, the manipulation button is mouse button 2, the button on the right. You press and hold this button to move an object, for example, to drag an object to a printer. Pressing the manipulation button once when the pointer is on an object, opens the object’s pop-menu.

**manual reporting.** A type of workstation reporting in which events, once they have taken place, are manually reported to Tivoli OPC. This type of reporting requires that some action be taken by a workstation operator. Manual reporting is usually performed from a list of ready operations.

**mass updating.** A function of the Application Description dialog in which a large update to the application database can be requested.

**MCU.** Multiple Console Support.

**Merged Graph view.** A view of the Workload Monitor/2 Workstations List object. Shows the total number of operations with different statuses for all the workstations that are included in the object. The information is shown in a single graph.

**modify current plan (MCP).** A Tivoli OPC dialog function used to dynamically change the contents of the current plan to respond to changes in the operation environment. Examples of special events that would cause alteration of the current plan are: a rerun, a deadline change, or the arrival of an unplanned application.

**most critical application occurrences.** Those unfinished applications whose latest start time is less than or equal to the current time.

**N**

**NCF.** See Network Communication Function.

**NCP.** Network Control Program.

**NetView operations.** Operations that consist of an operator instruction that Tivoli OPC passes to NetView. These operations are run at a general workstation with the WTO option specified.

**Network Communication Function (NCF).** A VTAM application that submits work to remote systems and passes events back to the Tivoli OPC tracker subsystem on the Tivoli OPC controlling system.

**noncyclic period.** A period that does not represent a constant number of days or work days. Examples: quarter, academic semester.

**nonreporting.** A reporting attribute of a workstation, which means that information is not fed back to Tivoli OPC.

**O**

**occurrence.** An instance of an application in the long-term plan or current plan.

An application occurrence is one attempt to process that application. Occurrences are distinguished from one another by run date, input arrival time, and application ID. For example, an application that runs four times a day is said to have four occurrences per day.

**occurrence group.** Consists of one or more application occurrences added to the long-term plan or current plan, where such occurrences are defined as belonging to a particular application group specified in the group definition field of the application description or job description.

**offset.** Values, in the ranges 1 to 999 and –1 to –999, that indicate which days of a calendar period an application runs on. This is sometimes called displacement.

**OI.** See operator instruction.

**OPC/ESA.** Operations Planning and Control/ESA

**OPC host.** The processor where Tivoli OPC updates the current plan database.

**OPC local processor.** A processor that connects to the Tivoli OPC host or remote processor through shared event datasets or XCF communication links.

**OPC remote processor.** A processor connected to the Tivoli OPC host processor via an SNA network. A Tivoli OPC event writer and an event transmitter (Tivoli OPC Network Communication Function) are installed on the remote processor and transmit events to the Tivoli OPC host processor via VTAM.

**open interval.** The time interval during which a workstation is active and can process work.

**operation.** A unit of work that is part of an application and that is processed at a workstation.

**operation deadline.** The latest time when the operation must be complete.

**operation latest out.** For an operation that has predecessors, the latest out date and time are the latest
start time for the first critical operation in the application occurrence. If the first critical operation has not started by this date and time, then the operation is flagged as late, because it will be impossible for it to start on time based on the sum of the planned durations of all the operations on its critical path.

**operation number.** The number of the operation. This uniquely identifies each operation in an application.

**Operation object.** An object contained in the Workload Monitor/2 Operations List object. It represents one operation in the current plan.

**operation status.** The status of an operation at a workstation.

**operation waiting for arrival.** The status of an operation that cannot begin processing because the necessary input has not arrived at a workstation. This status is applicable only for operations without predecessors.

**Operations List object.** A Workload Monitor/2 object that can be used to display information about operations in the current plan. It contains Operation objects.

**operator instruction (OI).** An instruction that an operator can view when the operator must manually intervene in Tivoli OPC operations.

**origin date.** The date that a period (cyclic or noncyclic) starts on.

**owner ID.** Owner ID is an identifier that represents the application owner.

P

**parallel operations.** Operations that are not dependent on one another and that can, therefore, run at the same time.

**parallel servers.** These represent the number of operations that can be processed concurrently by that workstation.

**partner transaction program.** An Advanced Program-to-Program Communications (APPC) transaction program located at the remote partner.

**PDF.** Program Development Facility.

**pending application description.** An application description that is incomplete and not ready for use in planning or scheduling. See active application description.

**pending occurrence.** The dummy occurrence created by the daily planning process to honor a dependency that has been resolved in the long-term plan but cannot be resolved in the current plan because the predecessor's input arrival time is not within the current plan end time.

**pending predecessor.** A predecessor dependency to an occurrence which is defined in the long-term plan but not yet included in the current plan. See also pending occurrence.

**period.** A time period defined in the Tivoli OPC calendar.

**personal workstation.** In Tivoli OPC documentation this term is used to refer to a computer that runs IBM Operating System/2.

**PIF.** See program interface (PIF).

**plan.** See current plan.

**Plan object.** A Workload Monitor/2 object that can be used to get information about the status of the current plan. When the Details view of the Plan object is open, the object monitors for current plan alerts if alert conditions have been specified.

**predecessor.** An operation in an internal or external dependency that must finish successfully before its successor operation can begin.

**print workstation.** A workstation that prints output and usually reports status to Tivoli OPC automatically.

**printout routing.** The ddname of the daily planning printout dataset.

**priority.** The priority of an operation is a value from 1 to 9 (where 1=low, 8=high, and 9=urgent). It is one of the factors that determines how Tivoli OPC schedules applications.

**program interface (PIF).** A Tivoli OPC interface that lets user-written programs issue various requests to Tivoli OPC.

Q

**query current plan (QCP) dialog.** An ISPF dialog that displays information taken directly from the current plan. The information includes information on operations, workstations, and application occurrences.

**QSAM.** Queued Sequential Access Method.
RACF. Resource Access Control Facility.

read authority. Access authority that lets a user read the contents of a dataset, file, or storage area, but not change it.

ready (R). The status of an operation indicating that predecessor operations are complete and that the operation is ready for processing.

ready list. An ISPF display list of all the operations ready to be processed at a workstation. Ready lists are the means by which workstation operators manually report on the progress of work.

receive. (1) To obtain a message or file from another computer. Contrast with send. (2) In Communications Manager, the command used to transfer a file from a host.

record format. The definition of how data is structured in the records contained within a file. The definition includes record names, field names, and field attributes, such as length and data type.

recovery. See automatic job and started-task recovery.

remote job tracking. The function of tracking jobs on remote processors connected by VTAM links to a Tivoli OPC controlling processor. This function enables a central site to control the submitting, scheduling, and tracking of jobs at remote sites.

remote processor. A processor connected to the Tivoli OPC host processor via a VTAM network.

replan current period. A Tivoli OPC function that recalculates planned start times for all occurrences to reflect the actual situation.

reporting attribute. A code that specifies how a workstation will report events to Tivoli OPC. A workstation can have one of four reporting attributes:

A Automatic
C Completion only
N Nonreporting
S Manual start and completion.

reroutable. Tivoli OPC can reroute operations if the workstation that they are scheduled to run on is inactive. An example of this can be if communication links to the system where the workstation is located fail. This option applies to operations only when they have status R (ready) or W (waiting). When you define an operation, you can specify one of the following reroutable options:

Y The operation is eligible to be rerouted if the workstation becomes inactive.
N The operation will not be rerouted, even though the workstation has an alternate destination.
blank The operation will be rerouted according to the WSFAILURE parameter on the JTOPTS initialization statement. This is the default.

rerun. A Tivoli OPC function that lets an application or part of an application that ended in error be run again.

Resource Object Data Manager. A licensed program that monitors resources and informs subscribing applications of their availability.

restartable. If an operation is defined as restartable, Tivoli OPC can automatically restart that operation if the workstation that it is using becomes inactive. This option applies only to the operation while it has status S (started). The operation will be reset to status R (ready).

return code. An error code that is issued by Tivoli OPC for automatic-reporting workstations.

RODM. See Resource Object Data Manager.

row command. An ISPF dialog command used to manipulate data in a table.

rule. A named definition of a run cycle that determines when an application will run.

run cycle. A specification of when an application is to run. The specification may be in the form of a rule or as a combination of period and offset.

SAA. See Systems Application Architecture.

SAF. System Authorization Facility.

schedule. (1) The current or long-term plan. (2) To determine the input arrival date and time of an occurrence or operation.

selection button. One of the two mouse buttons. With default mouse settings, the selection button is mouse button 1, the button on the left. You use this button to select windows, menu choices, pages in a notebook, and buttons. Pressing the selection button twice when the pointer is on an object opens the object to the default view.

send. (1) To send a message or file to another computer. Contrast with receive. (2) In
Communications Manager, the command used to transfer a file to the host.

server. The optional Tivoli OPC component that runs on the controlling system and handles requests from remote ISPF dialogs, remote PIF applications, and the Graphical User Interface for Application Description.

service functions. Functions of Tivoli OPC that let the user deal with exceptional conditions, such as investigating problems, preparing APAR tapes, and testing Tivoli OPC during implementation.

service level agreement. An agreement made between the data processing center and its user groups indicating the service hours and levels, as well as the kind of service the DP center will provide.

Settings notebook. See Settings view

Settings view. A view of an object that is used to specify properties of the object itself.

shared DASD. Direct access storage devices that can be accessed from more than one processor.

shared resource. A special resource or workstation resource that can be used simultaneously by more than one operation.

slack. Refers to 'spare' time. This extra time can be calculated for the critical path by taking 'Deadline less the Input Arrival less the sum of Operation Durations'.

SMF. System Management Facilities. An MVS component that collects and records system and job-related information.

smoothing factor. A value in the range 0-100 that controls the extent to which actual durations are fed back into the application description database.

SMP. System Modification Program.

SNA. See Systems Network Architecture.

special resource. A resource that is not associated with a particular workstation, such as a dataset.

splittable. Refers to a workstation where operations can be interrupted while being processed.

standard. User-specified open intervals for a typical day at a workstation.

started (S). A Tivoli OPC reporting status, for an operation or an application, indicating that an operation or an occurrence is started.

started-task computer workstation. You can specify that a computer workstation will support started tasks by giving the workstation the STC option. Operations defined to this workstation will be treated as started tasks, not as jobs.

started-task operations. Operations that start or stop started tasks. These operations are run at a computer workstation with the STC option specified.

status. The current state of an operation or occurrence.

status code. Codes that represent the current state of an operation. The status code is often associated with an extended status code.

The status of an operation can be one of the following:

A The operation is waiting for input to arrive.
R The operation is ready for processing (all predecessors have been reported as complete).
S Operation processing has started.
C Operation processing has completed.
D The operation has been deleted from the current plan.
I Operation processing has been interrupted.
* The operation is ready for processing. There is a predecessor at a nonreporting workstation, but all other predecessors are reported as complete.
E The operation has ended in error.
W The operation is waiting for a predecessor to complete.
U The operation status is not known.

submit/release dataset. A dataset shared between the Tivoli OPC host and a local Tivoli OPC processor that is used to send job-stream data and job-release commands from the host to the local processor.

subresources. A set of resource names and rules for the construction of resource names. Tivoli OPC uses these names when checking a user's authority to access individual Tivoli OPC data records.

subsystem. A secondary or subordinate system, usually capable of operating independently of, or asynchronously with, a controlling system.

successor. An operation in an internal or external dependency that cannot begin until its predecessor completes processing.

SYSOUT. A system output stream, also an indicator used in data definition statements to signify that a dataset is to be written on a system output unit.
SYSOUT class. An indicator used in data definition statements to signify that a dataset is to be written on a system output unit. It applies only to print workstations.

SYSPLEX. An MVS/ESA systems complex provides systems management enhancements for coordinating and controlling the data processing facility across multiple systems, while minimizing complexity. Implemented using the 9037 Sysplex Timer and the cross-system coupling facility (XCF) component of MVS/ESA.

Systems Application Architecture (SAA). A formal set of rules that enable applications to be run without modification, in different computer environments.

Systems Network Architecture (SNA). The description of the logical structure, formats, protocols, and operational sequences for transmitting information units through the networks and also operation sequences for controlling the configuration and operations of networks.

T

tail plan. Created during the daily planning process, includes only tail work; that is, work that started during or before the current planning period and that extends beyond its end.


temporary operator instructions. Operator instructions that have a specific time limit during which they are valid. They will be displayed to the workstation operator only during that time period.

time dependent. Tivoli OPC attempts to start operations as soon as possible, when all dependencies have been resolved and processing resources are available. However, you can specify that an operation is time-dependent, so Tivoli OPC will not start it until a specific time.

time zone support. A feature of Tivoli OPC that lets applications be planned and run with respect to the local time of the processor that runs the application. Some networks might have processors in different time zones. The controlling processor will make allowances for differences in time during planning activities to ensure that interacting activities are correctly coordinated.

TP. See application transaction program.

tracker. The Tivoli OPC component that runs on every system in your complex. It acts as the communication link between the MVS system that it runs on and the controller.

tracking event log. A log of job-tracking events and updates to the current schedule.

transport time. The time allotted for transporting materials from the workstation where the preceding operation took place to the workstation where the current operation is to occur. The transport time is used only for planning purposes. Operations will be started irrespective of the transport time specified.

TSO. Time Sharing Option.

turnover. A subfunction of Tivoli OPC that is activated when Tivoli OPC creates an updated version of the current plan.

U

undecided (U). A Tivoli OPC reporting status, for an operation or an application, indicating that the status is not known.

update authority. (1) Access authority to use the ISPF/PDF edit functions of the Tivoli OPC dialog. The authority is given to the user via RACF. (2) Access authority to modify a master file or dataset with the current information.

V

validity period. The time interval defined by an origin date and an end date within which a run cycle or an application description is valid.

variable table. A group of related JCL variables. Tivoli OPC can check these variable tables for substitution values for variables that occur in JCL. This substitution can occur during job setup or at job submit.

versions. Applications with the same ID but different validity dates.

VSAM. Virtual Storage Access Method.

VTAM. Virtual Telecommunications Access Method.

W

waiting (W). A status indicating that an application is waiting for a predecessor operation to complete.

waiting list. A list of jobs that have been submitted but still have uncompleted predecessors. Operations will be included in the waiting list if the JCL is not
submitted by the Tivoli OPC controller and the Tivoli OPC tracker has been started with HOLDJOB(YES).

**work day.** A day on which applications can normally be scheduled to start.

**work-days-only cyclic period.** A cyclic period where only work days are counted when calculating the interval.

**work-day end time.** The time when one Tivoli OPC work day ends and the next day begins. By default, this time is midnight.

For example, if the work-day end time is 02:00, work for Friday can continue until 02:00 on Saturday morning, even if Saturday is a free day. If Saturday and Sunday are free days, no new work will be started until 02:00 on Monday.

**Workload Monitor/2.** A part of Tivoli OPC. It runs on OS/2 Version 2 (or later) and communicates with a Tivoli OPC controller subsystem. It carries data about the subsystem's current plan from the host to a workstation, and can update operation status.

**workstation.** (1) A unit, place, or group that performs a specific data processing function. (2) A logical place where work occurs in an operations department.

Tivoli OPC requires that you define the following characteristics for each workstation: the type of work it does, the quantity of work it can handle at any particular time, and the times it is active. The activity that occurs at each workstation is called an operation. (3) See also personal workstation.

**workstation description database.** A Tivoli OPC database containing descriptions of the Tivoli OPC workstations in the operations department.

**workstation resource.** A physical resource, such as a tape drive, that must be allocated among jobs. When you define a workstation, you can specify the quantity of each of two resources (R1 and R2) that are available to operations. When defining operations to that workstation, you can specify the number of these resources that must be available for the operation to start on that workstation.

**workstation type.** Each workstation can be one of three types: computer, printer, or general.

**write-to-operator workstation.** A general workstation that lets you use Tivoli OPC scheduling facilities to issue a write-to-operator (WTO) message at a specific operator console defined by the workstation destination. NetView can intercept the WTO message and take necessary action.

**WTO message.** Write-to-operator message.

**WTO operations.** Operations that consist of an operator instruction that Tivoli OPC passes to NetView. These operations are run at a general workstation with the WTO option specified.

**X**

**XCF.** MVS/ESA cross-system coupling facility.

**XRF.** Extended recovery facility.
Index

Special Characters
* (asterisk) operation status code 115
* (asterisk) use in generic search arguments 16
% (percent) use in generic search arguments 16
= (equals) ISPF quick return command 14

A
A extended status code 116
A operation status code 115
actual quantity 92
AD/OI checks
  setting options 12
Adding an Application to the Current Plan panel 48
Adding an Occurrence Group to the CP panel 54
adding and deleting operations 62
Adding Applications to the Current Plan panel 47
adding dependencies 51
adjusted quantity 92
All Dependencies of an Operation panel 40
allocated special resources 95
altering dependencies 61
altering resources 97
alternate workstations 68
application groups
  adding to the current plan 46
applications
  adding to the current plan 46
  excluding 55
ARC row command 85
asterisk (*) operation status code 115
ATTR command 17
automatic
  restart 85
availability 87
  intervals 92

B
batch programs 20
blank, extended status code 116
Browsing General Current Plan Information panel 42
Browsing Most Critical Occurrences panel 39
Browsing Operation Input/Output Stream panel 80
Browsing Summary of Activities at a Workstation panel 41
browsing system information 41, 71
browsing the job log 77

C
C extended status code 116
C occurrence status code 115
C operation status code 115
C row command 60, 76
calendar 10
CAN error code 117
CANCEL command, MVS 108
CAT row command 77
catalog management 77, 84
  rerunning occurrences 58
  status codes 118
CCUN error code 117
changing dependencies in the current plan 61
changing error-list layouts 76
changing occurrences added to the current plan 56
changing operations in the current plan 61
changing resource availability 66, 93
changing resources 97
changing the current plan 45, 47
changing workstation availability 67
changing workstation details 68
checking current plan status 42
checking dependencies 40
checking occurrences 38
checking operations 40
checking workstation status 41
codes
  catalog management 118
  error 117
  extended status codes 116
  job log retrieval 119
  status codes for occurrences 115
  status codes for operations 115
  color and highlight attributes for panels 10
  command delimiter, specifying 11
  commands
    MVS
      CANCEL 108
      MODIFY 108
      START 107
      STOP 107
  commands, concatenating 13
  commands, ISPF in dialog 14
Communicating with Workstations menu 21
completing an ended-in-error operation 76
completing an occurrence 60
concatenating commands 13
Confirm Restart of an Operation panel 77
connected workstations 87, 99
dialog (continued)
  Modify Current Plan 43
  options, setting 8
  overview 7
  parameters, setting 8
  query current plan 37
  Ready List 21
discarding actions 84
double-byte character set (DBCS) 16
dummy application 46

e
  E extended status code 116
  E occurrence status code 115
  E operation status code 115
  Editing JCL for an MVS Job panel 83, 85
  editing JCL for an operation 30
  Editing JCL for an Operation panel 30, 32
  ended-in-error list
    layout 74
  ended-in-error operations 73
  ended-in-error status 29
  EQQELDEF member 76
  EQQELOUT member 76
  EQQELYCL panel 75
  EQQELYLL panel 75
  EQQHIPUP panel 65
  EQQHISTL panel 64
  EQQMAADL panel 48
  EQQMAAGL panel 54
  EQQMADDP panel 47
  EQQMAMOL panel 56
  EQQMAOCP panel 48
  EQQMCMDL panel 85
  EQQMEP1L panel 73
  EQQMERRP panel 74
  EQQMERTRP panel 77
  EQQMRSVP panel 69
  EQQMWDOL panel 70
  EQQMWSOL panel 70
  EQQMWSRP panel 68
  EQQMWSVP panel 69
  EQQQDIML panel 98
  EQQQDWML panel 99
  EQQQMIML panel 95
EQQQMLSL panel 94
EQQQMOP panel 97
EQQQMSEP panel 93
EQQQMWML panel 96
EQQRCSIM procedure 81
EQQRJCLE panel 30, 32
EQQRRLRM panel 27
EQQRVAL panel 31
EQQRYCL panel 24
EQQRYLL panel 23
EQQRSRLP panel 22
EQQRTOPP panel 21
EQQSAOSP panel 38
EQQSGCPP panel 42
EQQJCLB panel 80
EQQSMC1P panel 39
EQQSSPSP panel 36, 40
EQQPG1L panel 40
EQQSTOPP panel 37
EQQSWAUP program 81
EQQSWSSP panel 41
EQQUSIN subroutine 67
EQQXSUBP panel 20
error codes 117
catalog management 118
job log retrieval 119
specifying 49
error list 73
EX command 62
excluding applications 55
EXECUTE command 35
extended status codes 116
extending plans 2

F
FAILED workstation status 67
fast path 14, 15, 45
files currently in use 42
filter characters 16
filter panel 15, 45
finding data in a list 17
freeing a resource on error 87
function keys 19

G
GDDM 18
attributes, setting
ATTR command 15
command 15
GEM
See Tivoli Global Enterprise Manager (Tivoli GEM)
Generating JCL for a Batch Job panel 20
generic search arguments 16
GRAPH command 15, 17
Graphical Data Display Manager, see GDDM
graphically displaying data in a list 17
group definitions
adding an occurrence to a group 54
adding to the current plan 46, 54

H
Handling Operations Ended in Error panel 73
highlight attributes, on panel 73
HIIST command 62
history database 27, 63
history function 63
holding operations 33

I
I operation status code 115
I row command 62
in-use list 95
including dependencies 50
information about jobs 29
initiating deferred actions 84
input arrival date 48
input field pad character, specifying 11
instructions 76
Interactive System Productivity Facility (ISPF)
See ISPF (Interactive System Productivity Facility)
intervals
for special resources 87, 92
introduction 1
ISPF (Interactive System Productivity Facility)
command delimiter 13
dialog commands 14
dialog manager 7
setting options 7, 11
table services 7
using 7
ISPF select service 24

J
J row command 61, 77
Japanese language environment 16
JCCE error code 116
JCL
editing 77
error code 117
preparing 30
repository
file currently in use 42
variables 31
JCL Edit
setting options 12
JCLI error code 117
job log
  browsing 77
job log retrieval
status codes 119
job name
  changing 61
job statement information for system printer,
specifying 11
job submission
  criteria 36
delays 36
jobs
  delaying the start 33
  executing 35
  holding 33
  NOPing 34
  releasing 33
JR row command 77
JTOPTS initialization statement
  SUBFAILACTION keyword 117
  SUPPRESSACTION keyword 117
  WSFAILURE keyword 118
  WSOFFLINE keyword 117
K
  Kanji characters 16
  keep on error 87
  KEYS command 18
L
  L extended status code 116
  language support, REINIT 8
  last command 18
  latest start time 38
  layouts 22
  list criteria, specifying 15
  list dataset options, specifying 11
  List Dependency Status Change panel 59
  List of JCL Preparation Variables to Be Set panel 31
  lists, working with in Tivoli OPC 15
  local time offset 9
  LOCATE command 15, 17
  locating data in a list 17
  log dataset options, specifying 11
  long-term plan 2
  LTP (long-term plan) 2
M
  M extended status code 116
  M row command 60
  main menu, Tivoli OPC, panel EQQOPCAP 7
manually holding operations 33, 62
manually releasing operations 33, 62
MCP dialog 43
MCP error code 117
MH command 33, 62
MOD row command 77
MODIFY command, MVS 108
Modify Current Plan (MCP) dialog 43
  adding applications 46
  adding group applications 46
Modifying a Special Resource panel 97
Modifying a Workstation in the Current Plan panel 68
Modifying an Operation in the Current Plan menu 52
Modifying Catalog Management Actions panel 85
Modifying Connected Workstations for a Special
  Resource panel 99
Modifying Intervals for a Special Resource panel 98
Modifying Occurrences Added to the Current Plan
  panel 56
Modifying Occurrences in the Current Plan panel 57
Modifying Open Time Intervals in the CP panel 70
Modifying Operations in the Current Plan panel 52, 62
modifying resources 97
Modifying the Current Plan menu 45
Modifying Workstation Status in the Current Plan
  panel 69
monitoring resources 91
monitoring special resources 87
MR command 33, 62
MVS
  CANCEL command 108
  MODIFY command 108
  operator commands 107
  START command 107
  STOP command 107
N
  nnnn error code 118
  NOPing an operation 34, 62
  normal mode manager 109
  NP command 34, 62
O
  O row command 76
occurrences
  adding to the current plan 46
  changing 60
  completing 60
  deleting 60
  excluding 55
  grouping 54
  querying 38
  rerunning 58
  restarting 57
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>occurrences (continued)</td>
<td>115</td>
</tr>
<tr>
<td>offline actions pending</td>
<td>68</td>
</tr>
<tr>
<td>OFFLINE workstation status</td>
<td>67</td>
</tr>
<tr>
<td>OFxx error code</td>
<td>117</td>
</tr>
<tr>
<td>OJCV error code</td>
<td>117</td>
</tr>
<tr>
<td>on error attribute</td>
<td>87</td>
</tr>
<tr>
<td>OPCODES initialization statement</td>
<td>91</td>
</tr>
<tr>
<td>open intervals for workstations</td>
<td>67</td>
</tr>
<tr>
<td>OPER command</td>
<td>61</td>
</tr>
<tr>
<td>operation reason codes</td>
<td>119</td>
</tr>
<tr>
<td>operations</td>
<td></td>
</tr>
<tr>
<td>automatic restart</td>
<td>85</td>
</tr>
<tr>
<td>browsing</td>
<td>62</td>
</tr>
<tr>
<td>changing</td>
<td>61</td>
</tr>
<tr>
<td>checking</td>
<td>40</td>
</tr>
<tr>
<td>delay start</td>
<td>62</td>
</tr>
<tr>
<td>delaying the start</td>
<td>33</td>
</tr>
<tr>
<td>deleting</td>
<td>62</td>
</tr>
<tr>
<td>diagnosing delays</td>
<td>36</td>
</tr>
<tr>
<td>ended-in-error status</td>
<td>29</td>
</tr>
<tr>
<td>EX command</td>
<td>62</td>
</tr>
<tr>
<td>EXECUTE command</td>
<td>35</td>
</tr>
<tr>
<td>extended status codes</td>
<td>116</td>
</tr>
<tr>
<td>history</td>
<td>62, 63</td>
</tr>
<tr>
<td>holding</td>
<td>33, 62</td>
</tr>
<tr>
<td>in error</td>
<td>73</td>
</tr>
<tr>
<td>interrupting</td>
<td>29</td>
</tr>
<tr>
<td>MH command</td>
<td>62</td>
</tr>
<tr>
<td>MR command</td>
<td>62</td>
</tr>
<tr>
<td>NOPing</td>
<td>34</td>
</tr>
<tr>
<td>NOPing an operation</td>
<td>62</td>
</tr>
<tr>
<td>NP command</td>
<td>34, 62</td>
</tr>
<tr>
<td>querying</td>
<td>40</td>
</tr>
<tr>
<td>releasing</td>
<td>33, 62</td>
</tr>
<tr>
<td>removing a resource</td>
<td>93</td>
</tr>
<tr>
<td>removing from group</td>
<td>62</td>
</tr>
<tr>
<td>removing from schedule</td>
<td>62</td>
</tr>
<tr>
<td>requesting immediate execute</td>
<td>35, 62</td>
</tr>
<tr>
<td>rerunning</td>
<td>58</td>
</tr>
<tr>
<td>restarting</td>
<td>77</td>
</tr>
<tr>
<td>reversing a NO-OP</td>
<td>34, 62</td>
</tr>
<tr>
<td>setting status</td>
<td>28</td>
</tr>
<tr>
<td>starting immediately</td>
<td>35, 62</td>
</tr>
<tr>
<td>status codes</td>
<td>115</td>
</tr>
<tr>
<td>step restart</td>
<td>79</td>
</tr>
<tr>
<td>submission criteria</td>
<td>36</td>
</tr>
<tr>
<td>UN command</td>
<td>34, 62</td>
</tr>
<tr>
<td>using special resources</td>
<td>95</td>
</tr>
<tr>
<td>waiting for resources</td>
<td>96</td>
</tr>
<tr>
<td>operator commands, MVS</td>
<td>107</td>
</tr>
<tr>
<td>operator instructions</td>
<td>29, 76</td>
</tr>
<tr>
<td>options and parameters, defining</td>
<td>8</td>
</tr>
<tr>
<td>OS/2</td>
<td>19</td>
</tr>
<tr>
<td>OSEQ error code</td>
<td>79, 117</td>
</tr>
<tr>
<td>OSUB error code</td>
<td>117</td>
</tr>
<tr>
<td>OSUF error code</td>
<td>117</td>
</tr>
<tr>
<td>OSUP error code</td>
<td>117</td>
</tr>
<tr>
<td>OSxx error code</td>
<td>118</td>
</tr>
<tr>
<td>P</td>
<td></td>
</tr>
<tr>
<td>P occurrence status code</td>
<td>115</td>
</tr>
<tr>
<td>pad character, specifying</td>
<td>11</td>
</tr>
<tr>
<td>panel highlight attributes</td>
<td>10</td>
</tr>
<tr>
<td>adding an Application to the Current Plan</td>
<td>48</td>
</tr>
<tr>
<td>adding an Occurrence Group to the CP</td>
<td>54</td>
</tr>
<tr>
<td>adding applications to the current plan</td>
<td>47</td>
</tr>
<tr>
<td>all dependencies of an operation</td>
<td>40</td>
</tr>
<tr>
<td>browsing general current plan information</td>
<td>42</td>
</tr>
<tr>
<td>browsing most critical occurrences</td>
<td>39</td>
</tr>
<tr>
<td>browsing operation input/output stream</td>
<td>80</td>
</tr>
<tr>
<td>browsing summary of activities at a workstation</td>
<td>41</td>
</tr>
<tr>
<td>communicating with workstations</td>
<td>21</td>
</tr>
<tr>
<td>confirm restart of an operation</td>
<td>77</td>
</tr>
<tr>
<td>creating a dependency in the current plan</td>
<td>53</td>
</tr>
<tr>
<td>creating a ready list layout</td>
<td>24</td>
</tr>
<tr>
<td>creating an error list layout</td>
<td>75</td>
</tr>
<tr>
<td>current plan and status inquiry</td>
<td>37</td>
</tr>
<tr>
<td>defining dependencies in the current plan</td>
<td>53</td>
</tr>
<tr>
<td>editing JCL for an MVS Job</td>
<td>83, 85</td>
</tr>
<tr>
<td>editing JCL for an operation</td>
<td>30, 32</td>
</tr>
<tr>
<td>EQQELYCL</td>
<td>75</td>
</tr>
<tr>
<td>EQQELYLL</td>
<td>75</td>
</tr>
<tr>
<td>EQQHIPUP</td>
<td>65</td>
</tr>
<tr>
<td>EQQHISTL</td>
<td>64</td>
</tr>
<tr>
<td>EQQMAADL</td>
<td>48</td>
</tr>
<tr>
<td>EQQMAAGL</td>
<td>54</td>
</tr>
<tr>
<td>EQQMADDP</td>
<td>47</td>
</tr>
<tr>
<td>EQQMAOMOL</td>
<td>56</td>
</tr>
<tr>
<td>EQQMAOCP</td>
<td>48</td>
</tr>
<tr>
<td>EQQMCMADL</td>
<td>85</td>
</tr>
<tr>
<td>EQQMEP1L</td>
<td>73</td>
</tr>
<tr>
<td>EQQMERRP</td>
<td>74</td>
</tr>
<tr>
<td>EQQMERSL</td>
<td>79</td>
</tr>
<tr>
<td>EQQMERTP</td>
<td>77</td>
</tr>
<tr>
<td>EQQMUCL</td>
<td>83, 85</td>
</tr>
<tr>
<td>EQQMMADP</td>
<td>53</td>
</tr>
<tr>
<td>EQQMMADLP</td>
<td>53</td>
</tr>
<tr>
<td>EQQMMODP</td>
<td>52</td>
</tr>
<tr>
<td>EQQMMOPL</td>
<td>52</td>
</tr>
<tr>
<td>EQQMOCL</td>
<td>57</td>
</tr>
<tr>
<td>EQQMOPL</td>
<td>62</td>
</tr>
<tr>
<td>EQQMOSTL</td>
<td>59</td>
</tr>
<tr>
<td>EQQMROCL</td>
<td>58</td>
</tr>
<tr>
<td>EQQMTOPP</td>
<td>45</td>
</tr>
<tr>
<td>EQQMWSOL</td>
<td>70</td>
</tr>
<tr>
<td>EQQMWSRP</td>
<td>68</td>
</tr>
</tbody>
</table>
Generating JCL for a Batch Job 20
Handling Operations Ended in Error 73
List Dependency Status Change 59
List of JCL Preparation Variables to Be Set 31
Modifying a Special Resource 97
Modifying a Workstation in the Current Plan 68
Modifying an Operation in the Current Plan 52
Modifying Catalog Management Actions 85
Modifying Connected Workstations for a Special Resource 99
Modifying Intervals for a Special Resource 98
Modifying Occurrences Added to the Current Plan 56
Modifying Occurrences in the Current Plan 57
Modifying Open Time Intervals in the CP 70
Modifying Operations in the Current Plan 52, 62
Modifying the Current Plan 45
Modifying Workstation Status in the Current Plan 69
Ready List 27
Ready List Layouts 23
Rerunning an Occurrence in the Current Plan 58
Selecting an Error List Layout 75
Selecting Application Occurrence and Operation Information 36, 40
Selecting Application Occurrence Information 38
Selecting Applications to Add to the CP 48
Special Resource Monitor 94
Special Resource Monitor - In Use List 95
Special Resource Monitor - Waiting Queue 96
Specifying Ended In Error List Criteria 74
Specifying Ready List Criteria 22
Specifying Resource Monitor List Criteria 93
Step Restart Selection List 79
parameters and options, defining 8
PCAN error code 118
PF keys, assigning 18
PF keys, specifying number of 11
PFSHOW command 19
plan
current 2
long-term 2
potential predecessor 50
predecessors
changing 61
checking 40
including 50
potential 50
preparing JCL 31
primary commands 61
print process options, specifying 11
program function (PF) keys 19
promptable variables 31
Q extended status code 116
QCP dialog 37
quantity of resources 94
Query Current Plan (QCP) dialog 37
R extended status code 116
R operation status code 115
R row command 58, 62
R1 changing resource 61
R2 changing resource 61
ready list using 27
Ready List layout user exit 24
Ready List Layouts panel 23
Ready List panel 27
reason codes 119
reason for restart 78
rebuilding the current plan 67
recalling the last command 18
recovery when automatic restart fails 85
redirecting work to an alternate workstation 68
REINIT option 8
releasing operations 33, 62
removing a resource from a operation 93
removing an operation from the schedule 34, 62
REPLAN option 67
RER row command 77
Rerunning an Occurrence in the Current Plan panel 58
RESET request 79
Resource Monitor 91
Resource Object Data Manager (RODM) 91
resources
  changing 97
  deviation 94
  monitoring 87
  quantity 94
  unplanned changes 66, 93
restart
  automatic 85
  reason 78
restarting from a step 79
restarting occurrences 57
RETRIEVE command 18
return code simulation 81
RG command 62
RODM (Resource Object Data Manager) 91
RODMOPTS initialization statement 91
RODMTASK keyword of OCPOPTS 91
row commands 57—62, 76, 77, 85
  for special resources 96, 97
S
  S extended status code 116
  S occurrence status code 115
  S operation status code 115
  saving error-list layouts 76
  screen format, specifying 11
search order
  calendar 10
searching 16
Selecting an Error List Layout panel 75
Selecting Application Occurrence and Operation Information menu 40
Selecting Application Occurrence and Operation Information panel 36
Selecting Application Occurrence Information panel 38
Selecting Applications to Add to the CP panel 48
selection criteria 45
setting an occurrence to complete 60
setting up JCL 30
simulating return codes 81
SORT command 15, 16
sorting items in a list 16
Special Resource Monitor 91
Special Resource Monitor - In Use List panel 95
Special Resource Monitor - Waiting Queue panel 96
Special Resource Monitor panel 94
special resources 87
  See also resources
special resources (continued)
  changing 97
  list of allocations 95
  specifying a default calendar 10
  specifying dependencies 51
Specifying Ended In Error List Criteria panel 74
Specifying list criteria 15
Specifying Ready List Criteria panel 22
Specifying Resource Monitor List Criteria panel 93
SR row command 77
SRSTAT command 88
START command, MVS 107
start time
  latest 38
started task
  delaying the start 33
  holding 33
  releasing 33
starting operations
  diagnosing delays 36
  submission criteria 36
starting operations immediately 35, 62
status
  current plan 42
  setting 28, 61
  workstation 41, 68
  workstation status 68
status codes
  catalog management 118
  E 29
  extended 116
  job log retrieval 119
  occurrence 115
  operation 115
step restart 79
  return code simulation 81
Step Restart Selection List panel 79
STOP command, MVS 107
SUBFAILACTION keyword of JTOPTS 117
submit options
  changing 61
subsystem
  cancelling with MVS 108
  modifying with MVS 108
  starting with MVS 107
  stopping with MVS 107
subsystem name, setting or changing 8
successors
  changing 61
  checking 40
  including 50
SUPPRESSACTION keyword of JTOPTS 117
Sxxx error code 118
SYSOUT
  browsing 77
workstation (continued)
  changing 61
  changing details in the plan 68
  changing fixed resources 61
  connected 87, 99
  setup 30
  status 41, 67
    offline actions pending 68
    status set by EQQUXO09 68
    status set by WSSTAT 68
    status was set manually 68
    waiting for connection 68
    waiting for manual intervention 68
WSFAILURE keyword of JTOPTS 118
WSOFFLINE keyword of JTOPTS 117
WSSTAT command 67

X
X extended status code 116
xxx error code 118

V
viewing operator instructions 29
viewing the job log 77

W
W occurrence status code 115
W operation status code 115
W row command 57
  waiting for connection 68
  waiting for manual intervention 68
  waiting for resources 96
  wildcard characters 16
  work run on request 46
  working with lists in Tivoli OPC 15
Workload Monitor/2 19
workstation
  alternate 68
  availability 67

U
U extended status code 116
U occurrence status code 115
U operation status code 115
UN command 34, 62
UNKNOWN workstation status 67
  updating the current plan 67
user exit 24
  Ready List layout 24
using ISPF 7
utilities 20
Uxxx error code 118

T
T extended status code 116
terminal type, specifying 11
time
  modifying time options 61
  specifying format 8
time zones
  handled by Tivoli OPC dialog 9
Tivoli GEM
  See Tivoli Global Enterprise Manager (Tivoli GEM)
Tivoli Global Enterprise Manager (Tivoli GEM)
  component monitors 103
  connection monitors 104
  managing Tivoli OPC 102
  operational tasks 105
  Tivoli OPC instrumentation for 101
  topology view. 102
Tivoli OPC main menu, panel EQQOPCAP 7
tracking log
  file currently in use 42
type of resource allocation 95
If you especially like or dislike anything about this book, please use one of the methods listed below to send your comments to IBM. Whichever method you choose, make sure you send your name, address, and telephone number if you would like a reply.

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

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

If you prefer to send comments by mail, use the reader's comment form (RCF) at the back of this book. If you wish, you can give the RCF to the local branch office or IBM representative for postage-paid mailing.

If you prefer to send comments by fax, use this number, which is in Italy:

39+06+596+62077

If you prefer to send comments electronically, use this network ID:

ROMERCF at VNET.IBM.COM

Make sure to include the following in your note:

- Title and publication number of this book
- Page number or topic to which your comment applies
Help us help you!

Tivoli Operations Planning and Control
Controlling and Monitoring the Workload
Version 2 Release 3
Publication No. SH19-4377-02

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

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

Satisfied □ Dissatisfied □

How satisfied are you that the information in this book is:

Satisfied □ Dissatisfied □

Accurate □ Complete □
Easy to find □ Easy to understand □
Well organized □ Applicable to your task □

Specific Comments or Problems:

__________________________________________________________________________
__________________________________________________________________________
__________________________________________________________________________
__________________________________________________________________________
__________________________________________________________________________

Please tell us how we can improve this book:

__________________________________________________________________________
__________________________________________________________________________
__________________________________________________________________________

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

Name ___________________________ Address ___________________________

Company or Organization ___________________________

Phone No. ___________________________