Prime®

Operator's Guide to the Batch Subsystem

Revision 21.0

DOC9302-3LA

Operator's Guide to the Batch Subsystem

Third Edition

by

Elizabeth Hanes Perry

This guide documents the software operation of the Prime Computer and its supporting systems and utilities as implemented at Master Disk Revision Level 21.0 (Rev. 21.0).

Prime Computer, Inc.
Prime Park
Natick, Massachusetts 01760

PRINTING HISTORY

First Edition (DOC9302-1LA) December 1984 for Release 19.4
Second Edition (DOC9302-2LA) September 1985 for Release 20.0
Third Edition (DOC9302-3LA) July 1987 for Release 21.0

CREDITS

Editorial: Thelma Henner
Production: Judy Gordon
Illustration: Mingling Chang
Document Preparation: Celeste Henry, Kathy Normington

HOW TO ORDER TECHNICAL DOCUMENTS

To order copies of documents, or to obtain a catalog and price list:

United States Customers	International
Call Prime Telemarketing, toll free, at 1-800-343-2533, Monday through Friday, 8:30 a.m. to 5:00 p.m. (EST).	Contact your local Prime subsidiary or distributor.

CUSTOMER SUPPORT

Prime provides the following toll-free numbers for customers in the United States needing service:

1-800-322-2838 (within Massachusetts)
1-800-343-2320 (within other states)
1-800-541-8888 (within Alaska)
1-800-651-1313 (within Hawaii)
For other locations, contact your Prime representative.

SURVEYS AND CORRESPONDENCE

Please comment on this manual using the Reader Response Form provided in the back of this book. Address any additional comments on this or other Prime documents to:

Technical Publications Department
Prime Computer, Inc.
500 Old Connecticut Path
Framingham, MA 01701

Contents

About This Series

This series of Operator's Guides is designed and written to help you, as a system operator or a System Administrator of a Prime computer, do your job. These guides for system operators are logically divided into eight books to make them easy for you to use.

The next few pages describe Prime documentation that will be helpful to you as a system operator or as a System Administrator. To display an online list of Prime documentation, use the HELP DOCUMENTS command.

FOR THE SYSTEM OPERATOR

Before reading this book, you should have some familiarity with Prime systems. A good way to begin is to read the Prime User's Guide (DOC4130-4LA) with its update documents (UPD4130-41A and UPD4130-42A), which explain the PRIMOS® file management system and provide introductory and tutorial information about essential commands and utilities.

Once you are familiar with Prime systems, read this book, the Operator's System Overview, which outlines the material in the Operator's Guide series. Then, select the other books in the series as they apply to the tasks you must perform. As you learn more about system operations, you will use the Operator's Guide to System Commands as a reference for many of the special system commands and arguments that you, as an operator, will need to perform your job. It fully documents most of the commands described in the books in This series.

The Operator's Guide Series

The following books contain detailed information for the system operator.

- Operator's System Overview (DOC9298-2LA) introduces the series and describes computer-room operation of Prime systems.
- Operator's Guide to System Monitoring (DOC9299-3LA) describes how to monitor system activity, messages, and use.
- Operator's Guide to File System Maintenance (DOC9300-3LA) describes the PRIMOS file system and explains how to format disk partitions, run the disk partition maintenance program, determine physical device numbers, and interpret disk error messages.
- Operator's Guide to the Batch Subsystem (DOC9302-3LA) describes how to set up, monitor, and control the Batch subsystem.
- Operator's Guide to the Spooler Subsystem (DOC9303-2LA) describes how to set up, monitor, and control the Spooler subsystem.
- Operator's Guide to System Commands (DOC9304-3LA) serves as a reference guide for most of the commands described in the other books in the series.
- Data Backup and Recovery Guide (DOC10129-1LA) describes how to save information on disk or tape and how to restore that information when it is needed.
- Operator's Guide to Prime Networks (DOC10114-1LA) describes how to set up Prime networking systems and provides reference information about running network-related programs and monitoring network events.

Other Books for the Operator

The following books are also of special interest to system operators.

- Master Index for Operator and System Administrator Guides (DOC10110-2LA) indexes all the Operator and System Administrator Guides. Consulting this index is often the quickest way to find which manual has the information you need.
- The computer handbook for your particular CPU, which explains such topics as booting the system, shutting down PRIMOS, handling halts and hangs (including warm starts), performing tape dumps, and using the Virtual Control Panel (VCP).

- The "Using Your CPU" guide (available only for office machines) is intended for nontechnical users who are acting as system operators, and covers much the same information as the CPU handbooks.

- MAGNET User's Guide (DOC10156-1LA), which describes the MAGNET utility, used to transfer data by magnetic tape from a non-Prime operating system to PRIMOS and vice versa.

FOR THE SYSTEM ADMINISTRATOR

In addition to the documentation in the Operator's Guide Series, be sure to read the System Administrator's Guide Series, which describes how to set up, configure, and maintain PRIMOS:

- System Administrator's Guide, Volume I: System Configuration (DOC10131-1LA) explains how to build a system and allocate resources.

- System Administrator's Guide, Volume II: Communication Lines and Controllers (DOC10132-1LA) explains how to configure communication lines.

- System Administrator's Guide, Volume III: System Access and Security (DOC10133-1LA) explains PRIMOS security features and how to prevent unauthorized use of your system.

- DSM User's Guide (DOC10061-1LA) explains how to use the Distributed System Management software, including how to configure and operate DSM.

These books also provide information for the operator about most of the commands necessary to operate your Prime system.

Other Recommended Reading

In addition to the books listed above, you may find the following books useful:

- New User's Guide to EDITOR and RUNOFF (FDR3104-101B) with its update documents (COR3104-002 and COR3104-001) is a basic reference for any user of a Prime system and provides information about the Prime text editor and formatter.

- PRIMOS Commands Reference Guide (DOC3108-6LA) provides detailed information about user commands.

- PRIMENET Planning and Configuration Guide, (DOC7532-3LA) describes how to plan, configure, and maintain PRIMENET software for a system.

- User's Guide to Prime Network Services (DOC10115-1LA) explains Prime's networking systems.

- NTS User's Guide (DOC10117-1LA) explains the Network Terminal Service.

- Prime 50 Series Technical Summary (DOC6904-2LA) describes the features of the 50 Series systems, including advanced architecture concepts and the software and hardware products the concepts support.

About This Book

This book is one of a series of Operator's Guides, books designed and written to help you, as a system operator, System Administrator, and/or Batch Administrator of Prime computers, do your job. This particular book is a guide to both the operation and the administration of your Batch subsystem.

For the system operator, this book is intended to help you learn how to:

- Monitor the status of the subsystem.
- Control the subsystem when operator intervention is needed.
- Monitor and control users' Batch jobs.
- Detect and repair unexpected problems.

For the System Administrator, this book is intended to help configure the system and set up Batch to:

- Allow Batch to be used on the system.
- Allow the System Administrator to designate one or more Batch Administrators, who can perform most of the administrative duties relating to the Batch subsystem.

For the Batch Administrator, this book is intended to help make the decisions and set up the procedures that:

- Provide users with a smoothly functioning subsystem.
- Enable the Batch Administrator and the system operator to deal with the day-to-day running of the subsystem.

- Help the Administrators and operators resolve unexpected problems.

HOW THIS BOOK IS STRUCTURED

This book is divided into five chapters and three appendixes:

- An introductory chapter for the System Administrator, Batch Administrator, and the system operator (Chapter 1)
- One chapter for the System Administrator, explaining how to set up Batch (Chapter 2)
- One chapter for the Batch Administrator, explaining how to define and modify queues (Chapter 3)
- Two chapters for the system operator, explaining how to control and monitor Batch (Chapters 4 and 5)
- An appendix describing how to maintain the Batch database, including information on how to run the FIXBAT program (used by both the System Administrator and system operator under certain circumstances) (Appendix A)
- An appendix listing and explaining Batch and FIXBAT error messages, including Batch messages that affect the Batch user (Appendix B)

Note

The Prime User's Guide contains a version of Appendix B that lists only those messages that Batch users, rather than Batch Administrators or system operators, are likely to see.

- An appendix explaining how to convert from pre-Rev. 20 Batch to Rev. 21 Batch (Appendix C).

HOW TO USE THIS BOOK

System Administrators, Batch Administrators, and system operators have different responsibilities for Batch. Information on how to perform their respective duties is contained in this one book so that readers who fulfill all three functions can find the appropriate information quickly.

All readers of this guide should read Chapter 1, which introduces Batch concepts and describes various considerations for setting up and maintaining the Batch subsystem. System Administrators should then read Chapter 2; Batch Administrators should read Chapter 3, 4, and 5; system operators should read Chapters 4 and 5. Consult Appendix A when you need specific information on the FIXBAT program or when Batch queues become full. Consult Appendix B if you, or a Batch user, receive a message from Batch that you do not understand. Consult Appendix C if you are upgrading to Rev. 21.0 from a version of Batch which predates Rev. 20.0.

NEW BATCH SUBSYSTEM ENHANCEMENTS AT REVISION 21.0

General: The Batch subsystem was changed extensively at Revision 21.0 to comply with the U.S. Government's C2 security standards. As a result, many Batch files which were previously protected by passwords are now protected by ACLs (Access Control Lists). These files cannot be protected by passwords, nor can the Batch subsystem itself reside on a partition which is password-protected.

Batch Administrators: Before Rev. 21.0, Batch Administrators were designated with the -ADMIN option of the program BATCHQ>INIT. System Administrators now designate Batch Administrators by adding the ACL group .BATCH_ADMIN$ to the new Batch Administrator's user ID with EDIT_PROFILE. By default, there are NO Batch Administrators; in particular, SYSTEM is not a Batch Administrator.

Permissions Issues: The protection scheme for Batch commands has changed. There are now two basic classes of privileged users:

Batch Administrators may issue all of the Batch subsystem commands from any terminal, except for BATCH -START, which may only be issued from the supervisor terminal. In particular, Batch Administrators may:

- Use the BATGEN command to alter the queues.
- Abort, hold, and alter other users' jobs.
- Pause, continue, and stop the Batch monitor.

User 1 (SYSTEM, working at the supervisor terminal) may issue the BATCH -START, BATCH -STOP, BATCH -PAUSE, BATCH -CONTINUE, JOB -HOLD, and JOB -RELEASE commands.

The supervisor terminal has no other special privileges unless the System Administrator has chosen to make SYSTEM a Batch Administrator. This means that the System Administrator must make SYSTEM or individual operators Batch Administrators if the operators are expected to cap or block queues or to abort other users' jobs.

All other users may view and alter only their own jobs.

The FIXBAT command now automatically removes expired jobs from Batch queues; the -DAYS option is no longer necessary but is still accepted.

Phantoms and Queues: If there are fewer phantoms available than the Batch subsystem has queues, there is no way to predict which queue will be allocated a phantom first, or which queue will be allocated a phantom when one becomes available.

PRIME DOCUMENTATION CONVENTIONS

The following conventions are used in command formats, statement formats, and examples throughout this document. Examples illustrate the uses of these commands and statements in typical applications. Terminal input may be entered in either uppercase or lowercase letters.

Convention	Explanation	Example
UPPERCASE	In command formats, words in uppercase letters indicate the actual names of commands, statements, and keywords. These can be entered in either uppercase or lowercase letters.	JOB
lowercase	In command formats, words in lowercase letters indicate items for which the user must substitute a suitable value.	RLEVEL rlv
abbreviations	If a command or statement has an abbreviation, the abbreviation is shown below the full name.	BLOCK BLK
underlining in examples	In examples, user input is underlined, but system prompts and output are not.	OK, BATCH -STATUS [BATCH rev 21.0] No batch jobs. OK,
Brackets [ ]	Brackets enclose a list of two or more optional items. Choose none, one, or more of these items.	FIXBAT [-QUIET / -STARTUP]
Braces { }	Braces enclose a list of items. Choose only one of these items.	BLOCK { queuename / ALL }
Braces within brackets [{ }]	Braces within brackets enclose a list of items. Choose either none or only one of these items; do not choose more than one.	DISPLAY [{ queuename / all }]

Convention	Explanation	Example
Hyphen -	Wherever a hyphen appears as the first letter of an option, it is a required part of that option.	JOB -STATUS
Ellipsis ...	An ellipsis indicates that the preceding item may be repeated.	queuename...

1

Introduction to the Batch Subsystem

This chapter introduces the System Administrator, Batch Administrator, and system operator to Batch concepts and terminology. After you read this chapter, you should understand:

- What Batch provides for the user, and what it does for the System Administrator, Batch Administrator, and system operator
- How users submit Batch jobs into Batch queues to be executed by Batch phantoms
- The duties of the System Administrator (system configuration and planning), the Batch Administrator (queue definition and modification), and the system operator (day-to-day monitoring and control)
- The purposes of three Batch commands (BATCH, JOB, and BATGEN) and one Batch program (FIXBAT)
- Which commands are available only to Batch Administrators, and which are available to all users
- How the needs of a system govern the way Batch queues and phantoms are configured
- What the Batch monitor is and what it does
- Timeslices and scheduler priorities, their purpose, and how to control them

- The importance of defining Batch queues in an appropriate search order
- What job IDs are

BATCH RESPONSIBILITIES

People who work with Batch belong to one or more of these groups:

- Batch users, who submit Batch jobs
- System operators, who monitor and control Batch on a day-to-day basis
- Batch Administrators, who define, monitor, and modify Batch queues as needed
- System Administrators, who ensure that PRIMOS and Batch are both configured to allow correct operation of Batch

In many cases, one person serves as both Batch Administrator and System Administrator; sometimes, that same person serves also as system operator. This section describes the areas of responsibilities for System Administrators, Batch Administrators, and system operators.

System Administrator Responsibilities

Before Batch can be configured or used, the System Administrator

1. Ensures that PRIMOS is configured to accommodate Batch
2. Designates one or more Batch Administrators to perform the administrative duties relating to Batch and uses EDIT_PROFILE to add the ACL group .BATCH_ADMIN$ to their user IDs
3. Builds and installs Batch, if the Customer Support Representative has not already done so
4. Adds the BATCH -START command to PRIMOS.COMI (or C_PRMO) to bring up the Batch subsystem at startup time
5. Modifies the Batch monitor startup command file to fit the needs of the system.

These tasks are described in Chapter 2. After the System Administrator performs these tasks, Batch queues can be configured by the Batch Administrator.

Batch Administrator Responsibilities

Duties of the Batch Administrator include:

- Deciding how many queues to define, and the characteristics of each queue
- Creating the Batch queues and adding the queues to the subsystem in the proper order
- Renewing or replacing the Batch queues when necessary
- Running the FIXBAT or the INIT utility to repair or replace the Batch database if the database becomes damaged
- Monitoring the day-to-day operations of Batch if the system operators are not also Batch Administrators.

These tasks are described in Chapters 1 and 3 and in Appendix A.

System Operator Responsibilities

A system operator's responsibilities for the Batch subsystem generally consist of:

- Monitoring the Batch subsystem
- Starting, pausing, and stopping the Batch monitor
- Delaying submitted jobs until special resources become available

If the system operators are also Batch Administrators, they may also be responsible for:

- Altering submitted Batch jobs when requested to do so by users
- Blocking or capping queues, using the BATGEN utility
- Dealing with any problems that might affect the database, using the FIXBAT utility

Chapters 4 and 5 and Appendix A describe how to handle these tasks.

WHY BATCH?

Batch is among the most flexible of the PRIMOS command file utilities. Batch makes phantom execution of jobs easier for the user, while giving the Batch Administrator and system operator greater control of the environment and execution of the jobs.

Batch allows the Batch Administrator to define from one to sixteen Batch queues. Users submit Batch jobs to these queues, and these jobs later execute as phantoms. The Batch Administrator can set up Batch phantoms to run "in the background" of the system; that is, to run concurrently with interactive jobs, without significantly reducing the response time of interactive jobs.

By default, Batch phantoms run at the same priority as interactive user jobs, but Batch jobs can be set to run at higher or lower priority than other system jobs. For example, the Batch Administrator can set up a special high-priority queue that allows short jobs to be run at high priority, providing fast turnaround. The Batch Administrator can also limit the size of jobs that may be submitted to this special queue, which prevents Batch users from overutilizing this special queue and subsequently degrading overall system performance. (The size of a Batch job is measured in CPU time or elapsed time. CPU time is the amount of time the system actually spends running a particular job, whereas elapsed time is the amount of wall-clock time a job takes to run from start to finish. A job that needs only 20 seconds of CPU time might take 5 minutes of elapsed time to run if the system is heavily loaded.)

The Batch Administrator can also set up one or more low-priority queues that place no limits on the sizes of Batch jobs and that run these jobs at the lowest system priority. In this way, phantoms running jobs from these queues may use small amounts of CPU time when interactive use is heavy and larger amounts of CPU time when interactive use is light or absent.

If the Batch Administrator does not want Batch jobs to execute on the system when interactive use of your system is high (such as between 8 a.m. and 6 p.m.), the Batch Administrator can leave the Batch monitor either shut down or "paused" during these hours. Users may still submit jobs during this time, but these jobs will not be executed until the Batch monitor is either started up again or "continued". The Batch Administrator may define a queue window which prevents the execution of jobs in that queue at certain times of the day. The Batch Administrator may also cap one or more queues to prevent execution of Batch jobs in those queues. (Chapter 4 describes how to start up and shut down the Batch monitor, and also how to pause and continue the Batch monitor. Chapter 3 describes how to define a queue window using BATGEN and how to cap a queue using BATGEN. These functions are typically performed by the system operator.)

Furthermore, if necessary, a Batch Administrator can place Batch jobs in a special "held" state while still in their queues, then later can release these jobs to run at appropriate times. For example, extremely long jobs, such as file updates and backups, can be set up as Batch jobs during the day, then run under operator control at night.

HOW THE BATCH SUBSYSTEM WORKS

This section describes the nature of, and relationships between:

- Batch queues (defined, manipulated, and displayed via the BATGEN command)
- Batch jobs (submitted, manipulated, and displayed via the JOB command)
- Batch phantoms (started by the Batch monitor, displayed via the BATCH command)
- The Batch monitor (started, stopped, and manipulated via the BATCH command)

Batch Queues

Each Batch queue is a separate entity defined by the Batch Administrator to be particularly hospitable to certain types of jobs. Queues are defined by the following nine parameters:

- Name
- Default CPU time limit
- Maximum CPU time limit
- Default elapsed time limit
- Maximum elapsed time limit
- Default PRIMOS file unit for command input files
- Default value for priority of job within queue
- Relative run-time priority
- Timeslice

Queue status can be defined by any combination of the following:

- Capped/Uncapped
- Blocked/Unblocked
- Active/Inactive

The Batch Administrator creates queues and defines their characteristics using the BATGEN command (explained in Chapter 3). The strategy for defining queues is explained in the section of this chapter entitled PLANNING FOR A BATCH SUBSYSTEM.

Users submitting jobs can specify the following queue parameters: queue, CPU limit, elapsed time limit, file unit (for command input files only), and job priority within a queue. If users do not specify these parameters, the Batch job submission program (the JOB command) puts the users' jobs in the first suitable queue and assigns the queue's default values to the jobs. The BATGEN -STATUS and BATGEN -DISPLAY commands can show users what queues are available and what their characteristics are. Users can then submit their jobs to the appropriate queues.

The Batch Administrator must either make the first available queue a reasonable default queue or ensure that users know which queues to use and what the time limits and default values of those queues are. If the Batch Administrator does not take one of these two courses of action, the Batch system's behavior may not meet user expectations.

Batch Jobs

Users submit Batch jobs into queues with the JOB command. A batch job consists of a CPL (Command Procedure Language) file or a COMI (command input) file. As described earlier, a user can specify several queue-related parameters. In addition, a user may specify job-specific parameters using JOB command options, such as:

Option	Meaning
-HOME	The home directory of the job if it is different from the home directory of the user submitting the job
-ARGS	The command line for a CPL job
-ACCOUNTING	Accounting information for the job (not used by Batch)
-RESTART	Whether the job is to be automatically restarted if interrupted by an unscheduled system shutdown (such as a power outage)

Option	Meaning
-COMO	The pathname of a COMOUTPUT (COMmand OUTPUT) file of the execution of the job.
-NOTIFY	Whether the user is to be notified that his/her job has ended normally or abnormally
-DEFER	A later date or time, until which a job is to be deferred
-PROJECT	The project name under which the job is to be run if it is different than the project ID under which the user is logged in

Displaying Batch Jobs: The JOB command is also used to display information on jobs in Batch queues, via the -STATUS and -DISPLAY options. A Batch Administrator can display information on all Batch jobs on the system. Other users can display information on only those Batch jobs that they themselves submitted.

Manipulating Batch Jobs: A Batch Administrator can manipulate any user's job with the JOB command; other users can manipulate only their own jobs. (User 1 can manipulate other users' jobs only if SYSTEM is a Batch Administrator.)

A Batch Administrator may use the following special JOB options:

Option	Meaning
-ABORT	Terminates or aborts an executing job, causing the job to be immediately logged out.
-CANCEL	Cancels a waiting or held job, causing the job to be terminated.
-CHANGE	Modifies a job's parameters (except for queue and priority), without affecting the job's position in the queue.
-HOLD	Suspends or holds a waiting job, preventing the job from being initiated by the Batch monitor.
-RELEASE	Releases a held job, allowing the job to be initiated by the Batch monitor
-RESTART	Restarts an executing job, causing the job to be immediately logged out and later restarted by the Batch monitor.

Ordinary Batch users can use any of the above JOB options except -HOLD or -RELEASE; such users, however, are only allowed to modify jobs which they themselves submitted to Batch. Users working at the supervisor terminal can always use the JOB -HOLD and -RELEASE commands, but cannot use any of the other job-modification commands unless SYSTEM has been made a Batch Administrator.

Batch Phantoms and the Batch Monitor

Batch queues are serviced by a special phantom which runs under the user ID BATCH_SERVICE. This phantom is started up by the BATCH -START command. Typically, the System Administrator places the BATCH -START command in the system startup file (PRIMOS.COMI or C_PRMO); this causes the monitor to start automatically when the system is started.

Batch Phantoms vs. User Phantoms: When the Batch monitor (BATCH_SERVICE) finds a waiting job in a queue, the monitor starts up this job by spawning a Batch phantom. There is little difference between a Batch phantom and a phantom started directly by a user via the PHANTOM command. On a STATUS USERS display, user phantoms show up as "phant" whereas Batch phantoms show up as "batch". In addition, the BATCH -STATUS and BATCH -DISPLAY commands display information only for Batch phantoms.

Starting and Ending Batch Phantoms: A Batch phantom invokes the CPL program or command input file submitted by the user. (The Batch subsystem makes a copy of the submitted file, and it is this copy that it invokes.) Unless disabled by the Batch Administrator or system operator via the -HUSH option, the Batch phantom sends a message to the supervisor terminal announcing the starting of the job. When the CPL program or command input file has finished executing, the Batch phantom terminates successfully, and the Batch job is considered to have completed. If the CPL program or command input file encounters a fatal error that causes the Batch phantom to terminate unsuccessfully, the Batch job is considered to have aborted.

When a Batch phantom terminates, whether successfully or unsuccessfully, the Batch monitor deletes the corresponding Batch job from the list of active Batch jobs. In addition, unless disabled by the Batch Administrator or system operator via the -HUSH option, the Batch monitor sends a message to the supervisor terminal announcing the termination of the job and whether it completed or aborted. (If the user who submitted the job used the -NOTIFY option, the Batch monitor also sends that user a message announcing the termination of the job and whether it completed or aborted.) The next Batch job in the queue can then be initiated. Only one job per queue can be running at a time. Therefore, the number of Batch jobs that can be running at any time is limited to the number of Batch queues defined, assuming enough phantoms are available.

Stopping Batch While Jobs are Running: Before you shut the system down, you should first shut down the Batch monitor either by typing BATCH -STOP or LOGOUT ALL at the supervisor terminal. (BATCH -STOP is the preferred method.) Any Batch jobs that were actually running when the system was shut down finish running; after they finish, the phantoms they were using are released. Jobs which were waiting to run do nothing until the Batch monitor is started up again.

ACCESS ISSUES FOR BATCH AND ITS DATABASE

Once you have initialized the Batch subsystem, access to Batch functions and the Batch database is restricted as described below.

Normal Users

Normal users of the Batch system may

- Submit jobs with the JOB command
- Cancel or abort their own jobs
- Check the status of the Batch system with the BATGEN -DISPLAY and -STATUS commands and the BATCH -DISPLAY and -STATUS commands
- Check the status of their own jobs with the JOB -STATUS and -DISPLAY commands

Normal users have no access to jobs which they did not submit themselves.

Operators as Privileged Users

At Rev. 21.0, Batch operators have very few privileges in the Batch system. The SYSTEM ID (user 1) is not automatically a Batch Administrator; you must explicitly add the ACL group .BATCH_ADMIN$ to user SYSTEM if you wish operators to be Batch Administrators. If the SYSTEM ID is not a Batch Administrator, operators have the same privileges under Batch as any normal user. They are also able, using the supervisor terminal, to

- Halt and restart the Batch subsystem with the BATCH -START, -STOP, -PAUSE, and -CONTINUE commands
- Hold and release Batch jobs with the JOB -HOLD and JOB -RELEASE commands

Someone using the supervisor terminal has no other special privileges.

Batch Administrators as Privileged Users

Users who are designated as Batch Administrators have access to the entire Batch subsystem. Batch Administrators have all the rights of any Batch user; they can also

- Interrupt the Batch system with the BATCH -STOP, -PAUSE, and -CONTINUE commands
- Alter other users' jobs with the JOB -ABORT, -CANCEL, and -RESTART commands
- Modify the Batch queues with the BATGEN command
- See all users' jobs with the JOB -STATUS and -DISPLAY commands
- Delay and restart jobs which are already entered in the Batch queues with the JOB -HOLD and -RELEASE commands

Only user 1 (SYSTEM, at the supervisor terminal,) can issue the BATCH -START command. (This is true whether or not SYSTEM is a Batch Administrator.) BATCH -START will work with the RESUS command; see the DSM User's Guide for more information.

As of Rev. 19, the name BATCH_SERVICE should not be listed as a username in the User Validation File (UVF) of the SAD.

Table 1-1
Who May Invoke Batch Commands

Command	Users
BATGEN	A
BATGEN, altering live Batch queues	B
BATGEN -DISPLAY	A
BATGEN -STATUS	A
BATCH -START	S
BATCH -STOP	S,B
BATCH -PAUSE	S,B
BATCH -CONTINUE	S,B
BATCH -STATUS	A
BATCH -DISPLAY	A
JOB	A
JOB -DISPLAY (own jobs)	A
JOB -STATUS (own jobs)	A
JOB -ABORT (own jobs)	A
JOB -CANCEL (own jobs)	A
JOB -RESTART (own jobs)	A
JOB -HOLD	B,S
JOB -RELEASE	B,S
JOB -ABORT (others' jobs)	B
JOB -CANCEL (others' jobs)	B
JOB -RESTART (others' jobs)	B
JOB -DISPLAY (all jobs)	B
JOB -STATUS (all jobs)	B

A -- Any user
S -- Users at the supervisor terminal
B -- Batch Administrators

ACL vs. Password MFD: The Batch subsystem is designed to take advantage of ACL protection. The INIT program grants particular rights to privileged users -- BATCH_SERVICE (the Batch monitor) and Batch Administrators -- and grants varying, lesser rights to other users. This allows other users to use the subsystem, but not to access each other's jobs or to access Batch files and directories.

If the Master File Directory (MFD) in which BATCHQ resides is a password directory, you will not be able to use Rev. 21.0 Batch; you will have to use an earlier version.

User Access for Jobs: Batch jobs submitted by users take on the profiles of the submitting users. Thus, a Batch job is assigned the same group name, project ID, and origin directory that the user has at the time he or she submits the job. These differ only when the user uses the -PROJECT or -HOME options to specify project ID or origin directories for a job different from those under which the user is logged in.

PLANNING FOR A BATCH SUBSYSTEM

The System Administrator and Batch Administrator typically plan for the Batch subsystem. The basic decisions in planning a Batch subsystem are:

- How many queues do you want to define?
- How many phantoms should you allocate to run Batch?
- What scheduler characteristics (timeslices and priorities) do you want each queue to have?
- In what order should queues be searched for job submission and for job initiation?
- Who should be made a Batch Administrator? (discussed in Chapter 2)

The remainder of this chapter provides guidelines to help you make all but the last of these decisions. (The last decision, "Who should be made a Batch Administrator", is discussed in Chapter 2.)

For system operators, the remainder of this chapter is useful in understanding how the System Administrator or Batch Administrator decides to set up Batch, and also in understanding how the behavior of the Batch subsystem reflects these decisions. An understanding of these issues may improve a system operator's ability to monitor and manipulate the system, especially in handling user requests and questions concerning Batch.

How Many Queues Do You Want To Define?

One Batch queue allows one Batch job to execute at a time, and also describes default and maximum job parameters for jobs submitted to that queue. Therefore, you may wish to set up one or more queues depending on the needs of your system (how many Batch jobs should be able to execute at a time?) and the needs of your users (what are good default and maximum parameters for various kinds of Batch jobs on your system, and how many queues do you need to support these various combinations?). Use the BATGEN command, described in Chapter 3, to actually define the queues you wish to set up.

A Batch subsystem can consist of a single queue with no limits (except for user-defined limits) on jobs running within it. The system then simply runs jobs sequentially. If the single queue is set up in this way, all jobs have the same runtime priority. Within the queue, users can still request job priorities (from 9 down to 0) for their jobs in order to affect the order in which these jobs are started. (Priority 9 jobs are initiated before priority 8 jobs in the same queue, and so on.)

Alternatively, a Batch subsystem can contain from two to sixteen queues. In this case, the Batch monitor checks each queue in turn, beginning with queue number one. If it finds a job waiting to run, and a phantom is available, it runs the job. If sixteen queues have jobs, and sixteen phantoms are free, then one job from each queue is started. When the last of these jobs has been started, the monitor begins checking each queue again, to see if any jobs have finished or aborted. If so, the Batch monitor marks the job as completed or aborted, deletes any temporary files, and then checks the queue for another waiting job.

A different situation arises if there are fewer available phantoms than queues. For example, if there are three queues but only one phantom available to run jobs, the monitor runs all jobs that are ready to run (that is, waiting jobs) from queue one before running a job from queue two. The Batch monitor does not run jobs from queue three until queues one and two are both empty or contain only jobs that are held (via the -HOLD option of the JOB command), deferred, or capped.

How Many Phantoms Should You Allocate?

The System Administrator must ensure that Batch requirements for phantoms are met by the system configuration.

The number of phantoms is set by the NPUSR configuration directive in the system configuration file, described in the System Administrator's Guide.

One Phantom for the Batch Monitor: A Batch subsystem requires the availability of at least one phantom exclusively for the Batch monitor. (This phantom runs under the user ID BATCH_SERVICE.) The monitor runs

Batch jobs on whatever other phantoms are available.

Phantoms for Batch Jobs: The number of Batch jobs that can run simultaneously is limited by two factors: the number of queues and the number of available phantoms (that is, the number of phantoms not already taken by the Batch monitor, NETMAN, spooler phantoms, and so on). Because only one job per queue can execute at one time, the number of jobs running at any time is limited to the number of queues. On the other hand, no job can run without a phantom to run it. This means that the number of jobs running is limited to the smaller of two numbers: the number of Batch phantoms and the number of queues.

If many phantoms are available, and you expect that Batch use will be heavy, you can define up to sixteen queues to allow up to sixteen jobs to run at once. If, on the other hand, your system has only two or three free phantoms, you probably do not want to set up more than half a dozen queues.

Although Batch cannot use more available phantoms than there are queues, you do not need to configure enough phantoms to enable each queue to have a Batch job running at the same time. If you configure more Batch queues than phantoms, the queues will contend with each other for phantoms. As one queue finishes a job, one of the other queues will acquire the freed phantom and be able to start a new job. There is no way to predict which queue will be assigned a phantom next. In general, if you want to restrict Batch usage, it is probably wiser to cap excess queues or to allow only a few queues to have active windows during heavy usage periods. Either of these methods will give you better control over job and queue priorities than would limiting the number of phantoms.

Note

Keep in mind that users sometimes spawn phantoms either explicitly, by using the PHANTOM command, or implicitly, such as by using the ^X^E command in EMACS to execute a PRIMOS command. If you configure fewer phantoms than Batch queues, these users may receive the error message:

No phantoms available.

Borrowing Phantoms from Batch: There are two ways to borrow phantoms from the Batch subsystem: by capping an individual queue (which frees only the phantom that is used by that queue); or by pausing the Batch Monitor (which frees all the phantoms that service the queues after currently-executing jobs have terminated). These two methods are used in the following ways:

- If a special job must be run as a phantom, and the Batch subsystem is using all the phantoms, queues can be capped or the

subsystem can be paused (as described in Chapter 4). After a job finishes, the PHANTOM command can be used to run the special job; the phantom that was running the finished job will be used for the special job.

- If there are fewer phantoms than queues, and one job has either been waiting for an unduly long time or simply must be run, you will need to cap all the queues in the Batch system except the queue containing the waiting job. (If you left any other queue uncapped, that queue might snatch the phantom which you intended to free for the waiting job. If you blocked the Batch queues instead of capping them, you would have to wait until the current jobs in those queues finished before the desired job began to run.) If the favored queue contains jobs ahead of the job you wish to run, use the -HOLD option of the JOB command to delay execution of those jobs.

Timeslices and Scheduler Priorities

Every process on the system, including Batch phantoms, has a timeslice and a scheduler priority. Normally, user phantoms run at the same priority level and with the same timeslice as the users themselves. With Batch, you can set up different queues to run Batch jobs at different priority levels and with different timeslices. You can use the priority to cause Batch jobs to tend to receive more or less attention from the scheduler in relation to other processes on the system. At the same time, you can use the timeslice to control the length of time Batch jobs are allowed to run before being rescheduled.

Because timeslices and scheduler priorities are set individually for each queue, you can customize queues for quick, average, or slow jobs. Observe the following general guidelines when you customize queues:

- If a queue is intended for short jobs, it should offer limited CPU time but run at a relatively high priority. Its timeslice can be short or normal.

Note

You can force users to set runtime limits on their own jobs by setting a queue's default CPTIME higher than its maximum CPTIME. (The same could be done for ETIME.) A job submitted without the -CPTIME option could not use such a queue, and would be passed on to the next suitable queue.

- If a queue is intended for average jobs, it should have default timeslice and priority (as assigned by BATGEN when the queue is created). Whether the queue has maximum CPU and elapsed time limits should depend on how heavy you expect Batch usage to be.

- If a queue is intended for large, slow jobs, it should have no CPU time limit and no elapsed time limit. It should have a relatively low priority, but a large timeslice. You *might* decide to set up this type of queue with an extremely low priority such as IDLE (See Chapter 3). Jobs in a queue with a priority of IDLE are run only when there are no higher priority processes (not just Batch processes) waiting for execution.

If queues for short jobs are given a fairly high scheduler priority, jobs in these queues can run fast even when interactive use of the system is fairly heavy. Jobs in the low priority, high timeslice queues tend not to run fast when interactive use is heavy; but these queues can take advantage of any free time when interactive use is light.

Note

To take full advantage of the PRIMOS scheduling mechanism, distribute processes on your system evenly across each of the priority levels you intend to use. There are four levels, numbered 0 (lowest priority) through 3, not including the special priority levels of IDLE and SUSPEND. However, many systems tend to use only levels 0 and 1 or 0, 1, and 2. If a system has many users at priority 1, and only one process (such as a Batch job) at priority 0, the Batch job may actually run faster than any of the interactive users. Setting more processes (such as spooler phantoms) to priority level 0 tends to alleviate this problem. Use the CHAP command, described in the Operator's Guide to System Commands, to change priority and timeslice levels of processes other than Batch jobs and the Batch monitor.

Search Order of Batch Queues

The Batch monitor searches its list of queues either to find a queue into which to put a job, or to find which queues have jobs that need to run. The monitor searches the queues in the order the queues have been added to the system. (Queues are added by the BATGEN utility's ADD command.) The BATGEN -STATUS and BATGEN -DISPLAY commands display queues in the search order. You can utilize the search-order feature of Batch operation in the following ways:

- Queues for extra short jobs should come first in the search order. These queues should not accept jobs without the -CPTIME option or with a -CPTIME option that specifies more than, for example, 20 seconds of CPU time.

- Your default queue -- the one that catches jobs submitted without any -CPTIME or -ETIME option -- should be the first queue that a job can fall into. This means either that it must be the first queue in the search order or that the queues that

precede it must require some option such as the -CPTIME option to be supplied by the user.

- Queues for large, slow, background jobs should be at the bottom of the search list.

Identifying Jobs and Keeping Queues From Becoming Full

The Batch subsystem numbers jobs as it puts them into the queues. These numbers are called job IDs or internal names. Each queue has its own set of job IDs, consisting of a pound sign (#) followed by a number or letter (0-9, A-F) that identifies the queue, followed by a four-digit number. The four-digit number is maintained on a per-queue basis; that is, each queue has its own four-digit counter. For example, job ID #00041 identifies job 0041 in queue 0. A job in another queue might have the job ID #10041, identifying job 0041 in queue 1. (The fact that both job IDs have the same four-digit number is purely coincidental.)

Whenever a job is submitted to a queue, it is given the lowest job ID which is not currently in use in that queue. For instance, if a queue contains jobs #00001 and #00002, the next job added to that queue will get ID #00003. However, if a queue contains jobs #00002 and #00003, the next job added to the queue will get job ID #00001, since that ID is not currently a member of the queue. As soon as a job finishes, it is automatically removed from the queue and its job ID is available for reuse. This numbering scheme means that there can be a maximum of 10,000 jobs simultaneously in one queue; this limit is practically impossible to reach.

WHAT TO READ NEXT

This chapter introduced you to Batch concepts and terminology.

The System Administrator should now read Chapter 2, which describes installing and setting up Batch.

The Batch Administrator should now read Chapter 3, which describes defining and modifying Batch queues (as should a System Administrator who is also acting as a Batch Administrator). If SYSTEM has not been configured as a Batch Administrator, the Batch Administrator will also need to read Chapter 4 to be able to control Batch jobs.

The system operator should now read Chapter 4, which describes starting, stopping, and manipulating the Batch monitor, controlling Batch jobs, and handling Batch queues. The system operator should then read Chapter 5, which describes monitoring the Batch subsystem.

All readers should skim Appendix A for a description of the FIXBAT utility and Appendix B for a description of Batch messages. After

that, consult Appendix A if your Batch database becomes corrupted and Appendix B when you need help interpreting an error message.

2

Setting Up the Batch Subsystem

Before Batch queues can be configured, you, as the System Administrator, must set up Batch. This chapter explains setting up Batch prior to queue configuration.

To set up Batch you must:

1. Ensure that there are enough phantoms for Batch to run (by specifying a sufficiently large number via the NPUSR configuration directive).

2. Decide which users should be designated as Batch Administrators. Batch Administrators are responsible for the day-to-day management of the Batch system: adding and deleting queues, pausing the Batch system when necessary, and so on. If you do not designate any Batch Administrators, you will not be able to delegate any of Batch's management tasks. In particular, operators using the supervisor terminal will not be able to manage Batch's queues unless you explicitly make SYSTEM a Batch Administrator.

3. Have a Batch Administrator run BATCHQ>INIT.RUN to initialize the Batch subsystem.

4. Add the BATCH -START command to PRIMOS.COMI (or C_PRMO) to bring up the Batch subsystem at startup time.

Either the System Administrator or a Batch Administrator may perform a subsequent step:

5. Modify the Batch monitor startup command file to allow job initiation and termination messages to be sent to the supervisor terminal by the Batch monitor.

This chapter first explains the directory structures used by the Batch subsystem and then explains in detail each of the steps listed above. The steps are also listed in Table 2-1.

DIRECTORY STRUCTURE OF BATCH

The Batch subsystem directly involves two top-level directories on each system. These directories are:

- CMDNCO, the system-wide command directory, which is used by Batch only to hold the commands $$, BATCH, BATGEN, and JOB
- BATCHQ, which contains the Batch database, the Batch monitor startup command file, and three programs:
  - INIT, which initializes the database
  - FIXBAT, which repairs and cleans up the database
  - MONITOR, which is the Batch monitor program

CMDNCO and BATCHQ are present on the U1 partition of the Master Disk.

Generally, you will be working with the contents of the BATCHQ directory. The CMDNCO directory is updated only during the installation of Batch.

Note

A Batch database from a revision of PRIMOS prior to Rev. 20.0 is incompatible with Rev. 21.0 Batch. Specifically, pre-Rev. 20.0 queue files and pre-Rev. 20.0 BATDEF files (which contain queue definition information) cannot be used with Rev. 21.0 Batch; you will need to reenter your queue definitions under Rev. 21.0. (See Appendix C for details.)

If you are converting from a version of Rev. 20.0 to Rev. 21.0, running the INIT program converts your Batch database to Rev. 21.0 format. However, you should remove all jobs from the Batch queues before you run INIT.

Important Files in the BATCHQ Directory

The BATCHQ directory contains several files that are important to you, as System or Batch Administrator:

- START_BATCH_MONITOR.COMI, the Batch monitor startup command file. This file runs the Batch database repair and cleanup program, FIXBAT, and then runs the Batch monitor program, MONITOR.

- O_LOG, the Batch monitor command output (log) file. This file is generated by the Batch monitor whenever it starts up; it is opened when the Batch monitor startup command file, START_BATCH_MONITOR.COMI, runs the FIXBAT program.

  As released, the startup command file specifies that whenever the Batch monitor starts up, the existing (previous) O_LOG file is to be copied to a file named OLDLOG (also in BATCHQ), overwriting the previous contents of OLDLOG. You have the option of specifying instead that the existing O_LOG file be spooled to a printer, that it be deleted, or that no log file be kept at all. You modify the Batch monitor startup command file to change the way the Batch monitor handles log files.

- ERROR., the Batch database error information file. The Batch subsystem generates this file when it encounters a fatal database error. This file is designed to be read only by your Customer Support Center; it describes the error encountered and the particular portion of Batch that encountered the error.

  If you experience Batch database problems, print the contents of this file and the output from the following command sequence:

```
ATTACH BATCHQ
COMOUTPUT BATCH.COMO
LD -DETAIL -SORT_DTM
LD *>@@>@@ -DETAIL -SORT_DTM
COMOUTPUT -END
```

  (Only Batch Administrators can issue the above commands.) If you contact your Customer Support Center, make available the printed contents of ERROR. and the BATCH.COMO file you generated using the above commands.

- INIT, FIXBAT, and MONITOR, three program files.

START_BATCH_MONITOR.COMI, O_LOG, the three program files, and ERROR. are the only files you ever need to access directly in the BATCHQ directory. You access other files, such as BATDEF, through Batch using programs such as BATGEN or FIXBAT.

Table 2-1
Batch Upgrade and Installation Checklist for Rev. 21.0

Upgrading Systems Before Rev. 20.0	*Upgrading Systems at Rev. 20.0 or Later*	*Initial Installation at Rev. 21.0*
Check to make sure that at least two phantoms are available for Batch.	Check to make sure that at least two phantoms are available for Batch.	Check to make sure that at least two phantoms are available for Batch.
Print old Batch queue definitions for later reference.		
Block all batch queues and wait until no jobs are in queues.	Block all batch queues and wait until no jobs are in queues.	
Issue BATCH -STOP command.	Issue BATCH -STOP command.	
Have Rev. 21.0 installed.	Have Rev. 21.0 installed.	Have Rev. 21.0 installed.
Use EDIT_PROFILE to assign Batch Administrators.	Use EDIT_PROFILE to assign Batch Administrators.	Use EDIT_PROFILE to assign Batch Administrators.
Resume BATCHQ>INIT.RUN.	Resume BATCHQ>INIT.RUN.	Resume BATCHQ>INIT.RUN.
Use BATGEN to re-enter your old queue definitions.		Use BATGEN to set up queues.
Add BATCH -START to PRIMOS.COMI (or C_PRMO).	Add BATCH -START to PRIMOS.COMI (or C_PRMO).	Add BATCH -START to PRIMOS.COMI (or C_PRMO).
(Optional) Modify START_BATCH_MONITOR.COMI.	(Optional) Modify START_BATCH_MONITOR.COMI.	(Optional) Modify START_BATCH_MONITOR.COMI.
Use BATCH -START to start Batch subsystem.	Use BATCH -START to start Batch subsystem.	Use BATCH -START to start Batch subsystem.

Passwords And Batch

Before Rev. 21.0, the Batch system required certain of its subdirectories to be password directories. It now forbids passwords; neither the directory BATCHQ itself nor any of the BATCHQ subdirectories may have passwords.

PHANTOMS AND BATCH

As is explained in Chapter 1, the Batch subsystem requires the use of at least two phantoms; without them, the Batch system can do no useful work. Make certain that the system configuration file, typically named CONFIG in the directory CMDNCO, contains an NPUSR directive that specifies enough phantoms so that Batch can use at least two of them; one for the monitor and at least one for executing Batch jobs. (You must allocate one additional phantom for every queue which you want to be available independently of other queues.) At Rev. 21.0, PRIMOS itself uses more system phantoms than in earlier releases, so that you may need to raise the number of phantoms available on your system in order to keep Batch running.

DESIGNATING A BATCH ADMINISTRATOR

If you wish, you may decide to delegate most of the administrative duties involving Batch to another person, called a Batch Administrator. (In fact, you may delegate these responsibilities to several different Batch Administrators.) We strongly recommend that you make SYSTEM a Batch Administrator. If either SYSTEM or the individual operators are not registered as Batch Administrators, then the system operators will be unable to perform many of the day-to-day queue administration tasks -- they will be able to observe the Batch system, but not to alter its behavior. (In Rev. 20.0 and earlier Batch systems, the SYSTEM user ID (User 1) was automatically a Batch Administrator; this is no longer true.)

Batch Administrators are designated with the EDIT_PROFILE command. To make a given person a Batch Administrator, add the ACL group ".BATCH_ADMIN$" to his or her user ID. By default there are no user IDs associated with this ACL group.

Designating a user as a Batch Administrator gives the user ALL access to the BATCHQ directory and to all its subdirectories and files. (The user BATCH_SERVICE is automatically assigned these privileges.)

Batch Administrators can run the FIXBAT and INIT programs in BATCHQ. They can also use the BATCH command options -STOP, -PAUSE, and -CONTINUE, and the JOB command options -HOLD, -RELEASE, -ABORT, -CANCEL, and -RESTART. Moreover, Batch Administrators have full access to the BATDEF file in the BATCHQ directory, enabling them to change queue configurations at any time. All other users are given only Read access to BATDEF. (Read access is necessary for all users to submit Batch jobs, because the JOB program must have access to the BATDEF file.)

Even if you designate a Batch Administrator, you, as System Administrator, are still responsible for two ongoing tasks:

- Ensuring that the system configuration continues to be appropriate for the Batch usage on your system
- Updating the system startup file (PRIMOS.COMI or C_PRMO) to reflect changes in the way Batch is started up, such as when the maximum timeslice or the scheduler priority of Batch jobs is changed

In large installations, Batch Administrators should be aware of the configurations of the various systems. Keep in mind that anyone using the supervisor terminal is considered by Batch (and all of PRIMOS) to be an operator, and has some privileges not available to the average user. An advantage of being a Batch Administrator is that you may administer Batch without having to log in under a separate ID or move to the supervisor terminal.

INITIALIZING THE BATCH DATABASE

The program BATCHQ>INIT initializes the Batch database by creating (or updating, if you have an earlier version of Batch) the files, directories, and access categories that constitute the Batch database. You cannot use the Batch subsystem until INIT has been run, and only a Batch Administrator can run INIT. If the BATCHQ directory is kept on a password-protected partition, INIT aborts with the message "BATCHQ cannot be in a password MFD".

Before initializing the Batch database, you must make sure that the system date and time are set; if they are not, use the SETIME command to set them. (The INIT program, which you use to initialize the database, does not run unless the date and time are set.)

Note

BATCHQ>INIT overwrites the queue definition file BATCHQ>BATDEF. If you are installing Batch for the first time, BATCHQ>INIT creates an empty queue definition file. If you are upgrading from a version of Rev. 20.0 to Rev. 21.0, BATCHQ>INIT converts the old BATDEF file into a Rev. 21.0 BATDEF file, preserving

your current queue definitions. However, if you are upgrading from a revision of Batch before Rev. 20.0, your old queue definitions cannot be converted by the Batch system, and running BATCHQ>INIT destroys your Batch definitions. In that case, you should read Appendix C before you run INIT.

The format of the INIT command is:

RESUME BATCHQ>INIT [{ -RESET_QUEUES / -RSTQ }]

The -RESET_QUEUES Option

The -RESET_QUEUES option creates an empty BATDEF file replacing any existing BATDEF file and thus destroying any existing queue configurations. (If -RESET_QUEUES is not specified, the existing BATDEF file, with its queue definitions, is left as is.) The queues are also automatically reset if the BATDEF file is not present when INIT is run. This option is run automatically the first time that you run Rev. 20.0 or later Batch on a pre-Rev. 20.0 system; it destroys any information which was kept in the old BATDEF file.

If you are converting from a pre-Rev. 20.0 Batch system to Rev. 21.0, you must reenter your queue definitions; before installing or initializing the new Batch system, read Appendix C.

MODIFYING THE SYSTEM STARTUP FILE

The Batch subsystem is usually started at system startup by the system startup file (CMDNCO>PRIMOS.COMI). The command format for starting Batch in the system startup file is:

BATCH -START [{ -RLEVEL / -RLV } rlv] [{ -TIMESLICE / -TS } ts]

For example:

OK, BATCH -START -RLEVEL 1 -TIMESLICE 20

In this command, rlv and ts are decimal integers: rlv is the PRIMOS scheduler priority of the Batch monitor; ts is the timeslice in tenths of a second for the Batch monitor.

When editing PRIMOS.COMI, you must place the BATCH command after the COMINPUT and SHARE commands that share system libraries. An appropriate place for the BATCH -START command is just before the MAXUSR command.

When the BATCH -START command is issued, a phantom named BATCH_SERVICE is started using the file BATCHQ>START_BATCH_MONITOR.COMI as its command input file. BATCH -START may only be issued from the supervisor terminal or from the PRIMOS startup file PRIMOS.COMI; it is ignored from any other terminal.

Changing the Batch System's Priority

As options to the BATCH command, -RLEVEL and -TIMESLICE are used to set absolute values (such as the PRIMOS priority level) of the Batch monitor and, thus, the Batch subsystem. If you have a large number of Batch users running short jobs, you may want to use a value for rlv that is larger than the default value of 1 to broaden the range of priorities at which Batch jobs can run. (To do this, use the BATGEN RLEVEL subcommand, explained below.) To broaden the range of timeslices at which Batch jobs can run (explained below), you may want to use a value for ts that is larger than the default value of 20.

The range of values for rlv is 0 through 3, where 3 is the highest priority. The default value for rlv is 1; this is the same priority at which users normally run. If you specify an rlv for Batch which is higher than normal users' priorities, interactive system response degrades noticeably. You cannot specify the IDLE or SUSPEND priority levels for rlv. The range of values for ts is 1 through 99. The default value for ts is 20 tenths of a second, or two seconds. Specifying rlv and ts is optional.

Note

Using less than the default values for rlv or ts is not recommended.

RLEVEL and TIMESLICE are BATGEN ADD and MODIFY subcommands (explained in Chapter 3) as well as BATCH command options. As a BATGEN ADD or MODIFY subcommand, RLEVEL is used to set the PRIMOS scheduler priority for jobs from individual queues relative to the scheduler priority of the Batch monitor. (For example, you may set a queue's priority so that that queue's jobs run with a priority that is one less than the priority of the Batch monitor.) You can set individual queues' priorities to be the same as or lower than the priority of the Batch monitor; you cannot set a queue's priority higher than that of the Batch monitor. Thus, you may want to give the Batch monitor a high priority to broaden the possible range of priorities of individual queues in a Batch subsystem.

As a BATGEN ADD or MODIFY subcommand, TIMESLICE is used to set the timeslice for jobs from individual queues. However, jobs never receive a timeslice larger than the timeslice of the Batch monitor. Thus, again, you may want to give the Batch monitor a large timeslice (using the BATCH command's -TIMESLICE option, explained above) to broaden the possible range of timeslices at which Batch jobs can run.

See Chapter 3 for more information about RLEVEL and TIMESLICE when used as ADD and/or MODIFY subcommands of BATGEN.

MODIFYING THE BATCH MONITOR STARTUP COMMAND FILE

You may wish to modify the Batch monitor startup command file, named BATCHQ>START_BATCH_MONITOR.COMI, for one or more of the following reasons:

- To change the way in which the Batch monitor command output (log) file is handled (via the -STARTUP option of FIXBAT)
- To cause information on removed inactive jobs to be sent to the Batch monitor command output (log) file (by removing the -QUIET option of FIXBAT)
- To cause the Batch monitor and Batch phantoms to send messages concerning the initiation and termination of Batch jobs to the supervisor terminal (by removing the -HUSH option of MONITOR)

To do any of the above, edit the file named START_BATCH_MONITOR.COMI in the BATCHQ directory. In it you will find a command that runs the FIXBAT program and a command that runs the MONITOR program. Make the desired changes, as listed below, and write out the changed copy of START_BATCH_MONITOR.COMI. The changes will take effect the next time the Batch monitor is started (either by you or during system startup).

Removing Inactive Jobs From the Database

In Rev. 21.0 and later, the Batch System automatically removes inactive jobs from the database; no special effort by the user is needed.

Changing the Way the Batch Monitor Handles Its Log File

As supplied by Prime, START_BATCH_MONITOR.COMI runs the FIXBAT program with the -STARTUP SAVE option. This specifies that a command output file named O_LOG in BATCHQ is to be opened. Before FIXBAT opens the command output file, it checks to see if a file named O_LOG already exists (from a previous time the Batch monitor was running). If so, FIXBAT copies the O_LOG file into a file named OLDLOG (also in BATCHQ), overwriting the previous contents of OLDLOG. Then, FIXBAT opens the command output file O_LOG. This provides two levels of log file retention -- the current log file and the previous log file.

Log files are useful because you can use these files to help track down problems with Batch. In particular, if you need to consult your Customer Support Center about problems with Batch or Batch jobs, you should provide printouts of the Batch monitor log files for the appropriate times. It is possible, however, to turn off log file generation completely or to print the old log file instead of preserving it.

Spooling the Existing O_LOG File: To cause FIXBAT to spool the existing O_LOG file, change the line in START_BATCH_MONITOR.COMI that runs FIXBAT to read:

```
RESUME FIXBAT [other-options] -STARTUP SPOOL
```

Deleting the Existing O_LOG File: To cause FIXBAT to delete the existing O_LOG file, which is not recommended, change the line that runs FIXBAT to read:

```
RESUME FIXBAT [other-options] -STARTUP DELETE
```

Generating No O_LOG File: You can, if you choose, prevent FIXBAT from opening any log file at all. Be warned, however, that this invites disaster; if something goes wrong with your Batch subsystem, your Customer Support Representative may well be unable to diagnose the problem without the log file. If you are certain that you don't want a log file, change the line that runs FIXBAT to read:

```
RESUME FIXBAT [other-options] -STARTUP NOLOG
```

Note

You must preserve the -STARTUP option on the FIXBAT command line in the START_BATCH_MONITOR.COMI file, or the Batch monitor will not successfully start up.

Suppressing the Logging of Inactive Job Information

Whenever FIXBAT is run, it automatically removes inactive jobs (jobs which have been terminated or have run to completion) from its queues. FIXBAT normally does not log information on those jobs to the Batch monitor log file.

If you wish to log this information, which takes up five through seven lines of text for each job, remove the -QUIET option from the FIXBAT command line in START_BATCH_MONITOR.COMI. For example, in the Batch monitor startup command file, you might change the command line that currently reads:

```
RESUME FIXBAT -STARTUP SAVE -QUIET
```

so that it now reads:

```
RESUME FIXBAT -STARTUP SAVE
```

This new command line causes FIXBAT to be run whenever the Batch monitor is started up; if FIXBAT removes any jobs from the queues, it logs information about them to the log file.

Suppressing Job Initiation and Termination Messages

Normally, the Batch monitor does not send messages to the supervisor terminal about job initiation or termination. If you would like such messages to be seen at the terminal, change the Batch monitor startup command file, START_BATCH_MONITOR.COMI in the BATCHQ directory, as described below.

If your system tends to run many small Batch jobs, you will probably not want to get job initiation and termination messages on the supervisor terminal, since they will be so common as to obscure more vital diagnostic information.

To see job initiation and termination messages the next time the Batch monitor is started up, edit the startup command file and change the line that runs the Batch monitor via the RESUME MONITOR command from:

```
RESUME MONITOR -HUSH
```

to:

```
RESUME MONITOR
```

Notes

The Batch monitor sends job initiation and termination messages to the Batch monitor log file even when they are suppressed at the supervisor terminal.

The MONITOR program can only be run from Batch's startup file; it cannot be run from a terminal.

For an overview of the Batch installation process, see Table 2-1, Batch Upgrade and Installation Checklist. A System Administrator who is also a Batch Administrator should now read Chapter 3 to learn how to define and modify Batch queues.

3

Defining and Modifying Batch Queues

Once Batch has been set up, as described in Chapter 2, you, as the Batch Administrator, define Batch queues. This chapter describes the BATGEN command, used to define and modify Batch queues, in detail.

THE BATCH DEFINITION FILE

Information on Batch queues and their configuration is stored in a Batch definition file. This file contains configuration information for up to sixteen Batch queues.

Typically, there is only one Batch definition file per system. This file is named BATDEF, resides in the BATCHQ directory, and is referred to as the live Batch definition file because this file holds the live queue configuration for the Batch subsystem.

You create and modify a Batch definition file with the BATGEN command, as described below. Only people who have been previously validated as Batch Administrators can modify the official Batch Definition file. (See Chapter 2, Setting up the Batch Subsystem.)

By using BATGEN, as described below, you can create another Batch definition file to change the search order of queues on your system, and then copy the new Batch definition file into the live Batch definition file. You may also use BATGEN to create other Batch definition files for experimental purposes.

> Caution
>
> Do not use a text editor, such as ED or EMACS, to modify a Batch definition file. Such an attempt will likely result in the corruption of the file, destroying all queue configuration information in the file. If the corrupted file is the live Batch definition file, BATCHQ>BATDEF, the Batch monitor will shut itself down within ten or fifteen minutes.

INVOKING BATGEN

The command format for defining and modifying queues using BATGEN is:

```
BATGEN pathname
```

Usually, pathname is BATCHQ>BATDEF, as the BATDEF file is the only file that the Batch monitor reads in its search for queues in which to place jobs. Therefore, you typically invoke BATGEN by typing:

```
BATGEN BATCHQ>BATDEF
```

It is possible, however, to create queues in other files and then transfer them into the BATDEF file. You can create new queues and transfer them into BATDEF by:

1. Typing BATGEN pathname, where pathname is something other than BATCHQ>BATDEF
2. Doing whatever work you want within BATGEN
3. Exiting BATGEN with the command FILE BATCHQ>BATDEF

For an example of this, see CLEANING UP QUEUES in Chapter 4.

> Caution
>
> Only BATGEN can copy new queue configurations correctly into BATCHQ>BATDEF. If you try to copy in new configurations with COPY, you will disturb the ACLs set on BATDEF by INIT. If this happens, users will be unable to use the Batch subsystem. (They will get the error message "Insufficient access rights. BATCHQ>BATDEF") In order to remedy this situation, you must use BATGEN to recopy the desired queues into BATCHQ>BATDEF. (You must be a Batch Administrator in order to modify BATCHQ>BATDEF.)

BATGEN COMMANDS

Once pathname has been read and validated, BATGEN displays a prompt (>) and waits for a BATGEN command. Available commands are:

```
ACTIVE_WINDOW { queuename } { hh:mm-hh:mm }
AW            { ALL       } { FULL        }

ADD   queuename

BLOCK { queuename }
BLK   { ALL       }

CAP { queuename }
    { ALL       }

DELETE { queuename }
DL     { ALL       }

DISPLAY [{ queuename }]
DP      [{ ALL       }]

FILE   [pathname]
FI

MODIFY   queuename
MOD
```

QUIT
Q

STATUS
ST

UNBLOCK { queuename | ALL }
UNBLK

UNCAP { queuename | ALL }

Note

Until the FILE command is issued, no changes you make while using BATGEN take effect on the system. For example, if you block a queue by using the BLOCK command, a DISPLAY command shows that queue as blocked even though the Batch definition file and the Batch subsystem still continues to show that queue as unblocked; another user displaying the same Batch definition file sees the queue as unblocked. Not until you FILE the Batch definition file does the blocking of the queue take effect. If you QUIT without FILEing the Batch definition, the queue will never actually be blocked at all.

A queue name (queuename) is an alphanumeric name with as many as 32 characters. It is created by the ADD command and is the only name by which the queue may be referenced. Queue names must conform to standard PRIMOS filename rules. The name ALL is not accepted, as it would cause ambiguity in commands such as BLOCK ALL or DELETE ALL, where ALL means all queues.

The name of the queue has nothing to do with the queue's number or with the order in which queues are searched for jobs. The ID number (which becomes the first digit after the job number for jobs submitted to that queue) is assigned by the Batch subsystem and reflects the order in which queues are used when those queues are first defined. The search order reflects the order in which the queues are created, or added, to the Batch subsystem. To establish a queue as the number-one queue for searching, ADD it first; ADD the number-two queue second, and so on.

The BATGEN commands, and their subcommands, are defined on the following pages.

▶ ACTIVE_WINDOW { queuename / ALL } { hh:mm-hh:mm / FULL }
AW

Creates a time window for an existing queue (or all queues) during which the specified queue is available to execute new jobs. Either queuename or the keyword ALL must be specified. hh must not be larger than 23 and mm must not be larger than 59. If the keyword FULL is specified or if the beginning and ending times for the active window are identical, the queue will be active at all times. The time window (in hours and minutes) or the keyword FULL must be specified.

Jobs waiting in the queue or submitted to the queue during its inactive period do not execute until the next active window begins. Jobs executing when the ending time of the active window is reached, are allowed to finish executing.

▶ ADD queuename

Creates a new queue. If queuename is valid and not already in use as a queue name, BATGEN responds with "Enter queue characteristics:", displays a prompt ($), and waits for a subcommand (described below). If there is already a queue named queuename, BATGEN displays the fatal error message "Queue queuename already exists."

ADD and MODIFY subcommands are described after this list of commands, immediately following the UNCAP command.

▶ BLOCK { queuename / ALL }
BLK

Blocks an existing queue (or all queues). Jobs cannot be submitted to a blocked queue; jobs already in the queue are not affected.

▶ CAP { queuename / ALL }

Caps an existing queue (or all queues). This command allows the currently executing job(s) in the affected queue(s) to finish executing. It then halts execution of jobs from the affected queue. Execution of jobs does not begin again until the queue is uncapped with the BATGEN UNCAP command. Jobs may still be submitted to a capped queue.

▶ DELETE { queuename }
DL { ALL }

Prepares an existing queue (or all queues) for deletion. A deleted queue remains defined as a queue but accepts no more job submissions. The queue is not actually removed from the Batch definition file until all currently waiting, deferred, held, and executing jobs have been run or cancelled. The Batch monitor performs the actual removal of the queue; therefore, a queue is not actually removed unless the Batch monitor is running.

▶ DISPLAY [{ queuename }]
DP [{ ALL }]

Displays the name, the status, and the characteristics of the named queue (or of all queues). If you type DISPLAY without any argument, BATGEN displays information for all queues.

▶ FILE [pathname]
FI

Writes out the new, updated queue configuration to the Batch definition file named pathname to include commands given during this session. If you do not specify pathname, the file you specified when you invoked BATGEN is used. After the new configuration is written out, you are returned to command level.

▶ MODIFY queuename
MOD

Modifies an existing queue and, thus, all jobs waiting in that queue. If queue queuename exists, BATGEN responds with "Enter queue characteristics:", displays a prompt ($), and waits for subcommands (described below). If queue queuename does not exist or if it is flagged for deletion, BATGEN displays a fatal error message. The message is either "Illegal queuename", if the queue does not exist, or "Unknown queuename", if the queue is flagged for deletion.

MODIFY subcommands are discussed immediately following this list of commands.

▶ QUIT
Q

Terminates the BATGEN session without changing the Batch definition file. If you modified anything during the session, BATGEN asks, "Environment modified, OK to quit?" Type YES or a carriage return to tell BATGEN to QUIT. If you QUIT, all your changes during that BATGEN session will be lost.

▶ STATUS
ST

Shows the name and the status of all queues in tabular form. The order in which queues are displayed is the search order for queues (affecting the order of job submission and job initiation).

▶ UNBLOCK { queuename }
UNBLK { ALL }

Unblocks a blocked queue (or all queues). Normally, queues are not blocked. Unblocking a queue allows jobs to be submitted to the queue once again.

▶ UNCAP { queuename }
{ ALL }

Uncaps a capped queue (or all queues). Normally, queues are not capped. Uncapping a queue allows jobs in the queue to be eligible for execution once again.

ADD AND MODIFY SUBCOMMANDS

Subcommands for the ADD command and MODIFY command are identical. Six of them -- CPTIME, ETIME, FUNIT, PRIORITY, RLEVEL, and TIMESLICE -- define queue characteristics. Two others -- RETURN and QUIT -- tell BATGEN to save or ignore the preceding subcommands. The ADD/MODIFY subcommands function as follows. (All numeric values must be decimal integers.)

▶ CPTIME default [maximum]
CPT

Sets CPU time limits for jobs that run in a queue. The default limit is used for any job that is submitted without a CPU time limit (using the -CPTIME option of the JOB command). The maximum is an absolute limit. Jobs asking for greater CPU time than the maximum are not allowed into the queue.

The values for CPTIME are given in decimal seconds. The keyword NONE can also be used to signify that no time limit is to be set. Thus, the subcommand CPTIME 30 NONE causes jobs submitted without CPU time limits to be limited to 30 seconds of CPU time, but allows unlimited time to those jobs requesting it.

The default value can exceed the maximum. If this is true, jobs which are not assigned a CPTIME by the user will not be allowed into the queue. For example, CPTIME NONE 60 closes the queue to jobs that do not specify CPU time limits. Such jobs are given the queue's default limit of NONE and then denied admission to the queue because their CPU time limit is greater than the queue's maximum. (Jobs specifying a CPU time limit of more than 60 seconds are also denied admission to the queue.) Specify a default value larger than the maximum value if you want to force users always to define their own CPU time limits.

When a queue is first created, the default and maximum CPTIME values are both set to NONE. When you modify an existing queue, you can change only the default value or both the default and maximum values. For example, the command CPTIME default maximum changes both values, but the command CPTIME default changes only the default value.

▶ ETIME default [maximum]
ET

Sets elapsed time limits. ETIME functions in exactly the same way as CPTIME does, except that its values are given in minutes rather than seconds, and it pertains to the -ETIME option of the JOB command (see the Prime User's Guide for information on the JOB command) rather than the -CPTIME option. When a queue is first created, these values are both set to NONE.

▶ FUNIT number
FU

Sets the default file unit number for Batch jobs which are command input files. This option is ignored for CPL jobs; for command input files, a queue's FUNIT defines the file unit number which is assigned to the command input file itself. If the user's JOB command specifies a file unit number, the user-specified file unit number overrides the FUNIT assigned to the queue. Number can range from 1 to 127. When a queue is created without FUNIT's being specified, the queue's FUNIT is set to 6.

▶ PRIORITY value
PRI

Sets the default value for a job's priority within the queue itself -- that is, its priority vis-a-vis other jobs in the same queue. Any job not specifying its own priority, via the -PRIORITY option of the JOB command, is given this default value. Permissible values are 0 through 9; 9 is the highest (most favored) priority and 0 is the lowest priority. When a queue is first created, this value is set to 5.

Note

This priority affects only the order in which jobs within a single queue are initiated. It does not determine how fast they run. Use the RLEVEL and TIMESLICE subcommands to determine runtime priority.

▶ QUIT
Q

Discards the work done at subcommand level. If you are modifying an old queue, QUIT leaves that queue unchanged. If you are adding a new one, QUIT throws away the new queue's name as well as its characteristics. If you modified anything before quitting, BATGEN asks, "Queue definition modified, ok to quit?" If it does not receive an answer of YES (or a carriage return), it prompts you to save work with "Please return". (Note that the START command restarts the monitor with all changes that you have specified.)

▶ RETURN
RTN

Saves the new characteristics for future display and/or filing.

▶ RLEVEL { delta-value }
RLV { IDLE }

Does not set the runtime priority for a job submitted to a queue, except if RLEVEL IDLE is specified. Rather, it determines the amount the priority of a job is lowered from the priority of the Batch monitor. (The Batch monitor's priority is set using the -RLEVEL option of the BATCH -START command.) The range of delta-value is 0 through 7 with 0 meaning that the queue's jobs run at the same priority as the monitor, and 7 representing the maximum lowering. If you specify IDLE, jobs from this queue are processed by the CPU only when there are no higher priority processes waiting for execution. On an active system, jobs at IDLE priority will proceed very slowly. (Note that delta-value is a value that submitting users cannot specify for themselves.) When a queue is first created, this value is set to 0.

PRIMOS currently allows a process to have a priority of 0, 1, 2, or 3 (not including the IDLE and SUSPEND priority levels explained in the Operator's Guide to System Commands), with 3 being the highest PRIMOS priority level. Therefore, if the Batch monitor is running at PRIMOS priority 3, RLEVEL values of 3 through 7 have the same effect because the maximum number of levels that the Batch monitor may be lowered is 3 (from 3 to 0). If the monitor is running at PRIMOS priority 1, RLEVEL values of 1 through 7 have the same effect because the Batch monitor may not be lowered more than one level (below 0).

▶ TIMESLICE value
TS

Sets the timeslice value in tenths of a second for jobs in the queue. A queue's timeslice may be smaller than the Batch monitor's timeslice, and may be used as the effective timeslice of jobs in the queue. However, if the queue's timeslice is larger than the monitor's, the queue's timeslice is ignored, and the monitor's timeslice is used as the effective timeslice for each job in the queue. (The submitting user has no control over this value.)

The timeslice value ranges from 1 through 99, but probably should not go below 20 unless job priority is unusually high. When a queue is first created, this value is set to 20, equaling 2 seconds.

SAMPLE BATGEN SESSION

In the following sample BATGEN session, two typical Batch queues are defined. The first, QUICK.QUEUE, is intended for use by a large number of users submitting short jobs. The second queue, PAYROLL, is intended solely for the processing of a payroll. Except for the illustrative errors, all input could come from a command file. Note that > and $ are prompts, and that all user input is underlined.

```
OK, BATGEN BATCHQ>BATDEF
[BATGEN Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]
Creating new batch definition file: batchq>batdef  (BATGEN)
> ADD QUICK>QUEUE
Illegal queue name. QUICK>QUEUE (BATGEN)
> ADD QUICK.QUEUE
Enter queue characteristics:
$ CPTIME 20
$ ETIME 5
$ PRIORITY 4
$ RETURN
> ADD QUICK.QUEUE
Queue QUICK.QUEUE already exists (unblocked, uncapped).
> ADD PAYROLL
Enter queue characteristics:
$ CPTIME NONE
$ ETIME NONE
$ FUNIT 31
$ PRIORITY 9
$ RLEVEL 2
$ RETURN
> DISPLAY

Queue name = quick.queue, unblocked, uncapped.
Active window = FULL;
Default cptime=20, etime=5, priority=4;
Maximum cptime=None, etime=None; Funit=6;
Delta rlevel=0; Timeslice=20;

Queue name = payroll, unblocked, uncapped.
Active window = FULL;
Default cptime=None, etime=None, priority=9;
Maximum cptime=None, etime=None; Funit=126;
Delta rlevel=2; Timeslice=20;

> MODIFY QUICK.QUEUE
Enter queue characteristics:
$ CPTIME 25
$ RLEVEL 1
$ TIMESLICE 10
$ RETURN
```

```
> DISPLAY QUICK.QUEUE
Queue name = quick.queue, unblocked, uncapped.
Active window = FULL;
Default cptime=25, etime=5, priority=4;
Maximum cptime=None, etime=None; Funit=6;
Delta rlevel=1; Timeslice=10;

> ACTIVE_WINDOW QUICK.QUEUE 06:00-23:59
> DISPLAY QUICK.QUEUE

Queue name = quick.queue, unblocked, uncapped.
Active window = 06:00-23:59;
Default cptime=3, etime=5, priority=4;
Maximum cptime=None, etime=None; Funit=6;
Delta rlevel=1; Timeslice=10;

> FILE
OK,
```

Unless you have given the system operator Batch Administrator status, you should now read Chapter 4 to learn how to control the Batch subsystem.

As Batch Administrator, you should skim Appendix A to understand how to run the FIXBAT program, should that prove necessary.

4

Controlling the Batch Subsystem

As an operator, you have very few special privileges under the Batch system. Many of the Batch maintenance commands are available only to Batch Administrators, who are determined by your System Administrator; User 1 is not automatically a Batch Administrator. If you do not have access to a user ID which has Batch Administrator rights, you will be able to issue some Batch commands only from the supervisor terminal; others you will not be able to issue at all. This chapter distinguishes between commands which can be issued only by Batch Administrators and those which can be issued only from the supervisor terminal. Table 1-1 also contains a complete summary of which Batch commands are available to the various classes of users.

Once a Batch subsystem is set up and started, any system operator can control Batch by:

- Starting, stopping, pausing, or continuing the monitor. Pausing the monitor temporarily prevents Batch jobs from being initiated, although jobs already executing when you pause the monitor continue executing.

- Holding or releasing individual jobs in the Batch queues. (A held job remains in the queue but cannot be executed until you release it.)

A system operator who is also a Batch Administrator can also control Batch by:

- Aborting, canceling, or restarting individual jobs.
- Blocking individual queues, keeping those queues from accepting the submission of new jobs while letting the rest of the subsystem continue running. (Jobs already in a blocked queue are not affected.)
- Capping individual queues, keeping those queues from executing any new jobs while letting the rest of the subsystem continue running. (Jobs already executing when you cap a queue continue executing.)
- Adding new queues. (as explained in Chapter 3)
- Deleting queues. (as explained in Chapter 3)

Note

> Queues are deleted in an orderly fashion. This allows all jobs in the queue to finish before the queue is deleted. If an emergency situation requires an immediate cessation of all activity in a queue, cap and block that queue. Later you can decide to allow the queue to finish executing its jobs, or you can use the -CANCEL option with the JOB command to empty a queue of waiting jobs and allow it to be deleted immediately.

Permissions Issues: Someone working at the supervisor terminal can start, pause, and stop the Batch monitor and hold and release Batch jobs. If the SYSTEM user ID is a Batch Administrator, a person working at the supervisor terminal can also control other users' jobs, modify the queues, and cap and block the queues from the supervisor terminal. A Batch Administrator working at a user terminal can do all of the above except start the Batch monitor.

STARTING THE BATCH MONITOR

The Batch monitor is normally started up during system cold start. Therefore, you will rarely need to start up the Batch monitor separately. However, you should understand how the Batch monitor is started. The command file CMDNCO>PRIMOS.COMI usually contains a command to start the Batch monitor. This command is in the following form:

```
BATCH -START -RLEVEL rlv -TIMESLICE ts
```

(Although -RLEVEL and -TIMESLICE are optional, many installations use them. Thus, when you start the Batch monitor yourself, you should use the same options and values for rlv and ts as does your system startup file. The -RLEVEL and -TIMESLICE options are explained in Chapter 2.)

If the Batch monitor does not start when the system starts, you can issue the identical command from the supervisor terminal to start the Batch monitor. BATCH -START may only be issued from the supervisor terminal; it will not work on any other terminal. When you type BATCH -START, the following message appears:

```
[BATCH Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]
Monitor started up.
OK,
```

This indicates that the Batch monitor has been initiated. It does not indicate that the Batch monitor has begun processing Batch jobs; this happens later, as described next.

How BATCH -START Works

The BATCH -START command creates a phantom named BATCH_SERVICE. This phantom serves as the Batch monitor.

The monitor cannot begin work until the system time and date have been set. Therefore, if the BATCH -START command is issued before time and date are set (as can happen on older machines if the BATCH -START command is included in the PRIMOS.COMI file), the monitor waits, doing nothing, until the SETIME command is given.

After the system time and date have been set, the monitor runs a program called FIXBAT. FIXBAT ensures that a valid database exists for the processing of user jobs. The monitor then sends the following message to the supervisor terminal, notifying you that it is ready to process Batch jobs:

```
*** BATCH_SERVICE (user nnn on sysname) at hh:mm
Monitor in Operation.
```

If you issue the BATCH -START command while the monitor is running, the Batch subsystem ignores the command and sends you the message:

```
[BATCH Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]
Monitor already started.
ER!
```

If users submit Batch jobs when the monitor is not running, they receive the warning message:

```
Warning: jobs are not being processed at this time.
```

You receive this warning message when using most forms of the BATCH command while the Batch monitor is not running. Jobs can be submitted despite the message. However, jobs are not executed until the monitor begins to process them.

PAUSING THE BATCH MONITOR

You can make the monitor pause, preventing it from starting the execution of any new jobs, but allowing it to:

- Complete currently executing jobs
- Signal the completion or abortion of executing jobs
- Delete queues

To pause the Batch monitor, type the command:

```
BATCH -PAUSE
```

If you are a Batch Administrator, you may type this command from any terminal; otherwise, you may only type this command on the supervisor terminal. Normally, the response is:

```
[BATCH Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]
Monitor paused.
OK,
```

Shortly thereafter, if the Batch monitor is running, it recognizes the pause request, and sends the following message to the supervisor terminal:

```
*** BATCH_SERVICE (user nnn on sysname) at hh:mm
Monitor paused.
```

When the monitor is paused, users who type the JOB or BATCH commands -- including you, when you use some forms of the BATCH command -- receive the message:

```
Note: the batch monitor is currently not starting up jobs.
```

CONTINUING THE BATCH MONITOR

To direct the monitor to start executing jobs again, type the command:

```
BATCH -CONTINUE
```

As with the -PAUSE subcommand, BATCH -CONTINUE may be issued from any terminal by a Batch Administrator, or from the supervisor terminal by anyone else. Typically, the response is:

```
[BATCH Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]
Monitor continued.
OK,
```

When the Batch monitor is running and recognizes that it has been continued, it sends the following message to the supervisor terminal:

```
*** BATCH_SERVICE (user nnn on sysname) at hh:mm
Monitor continued.
```

STOPPING THE BATCH MONITOR

To stop the Batch monitor, type the command:

```
BATCH -STOP
```

Normally, the response is:

```
[BATCH Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]
Stop request issued.
OK,
```

When the Batch monitor receives the stop request, it sends the following message to the supervisor terminal:

```
*** BATCH_SERVICE (user nnn on sysname) at hh:mm
Operator stop.
```

After displaying that message, the monitor logs itself out. The act of logging out causes a phantom logout message to be sent to the supervisor terminal:

```
Phantom nnn: Normal logout at hh:mm
Time used: hhh mmm connect, mmm sss CPU, mmm sss I/O.
```

In this message, nnn represents the phantom's number; hh:mm represents the time of day; and mm ss represent minutes and seconds of elapsed time.

If you type the BATCH -STOP command when the Batch monitor is not running or is just starting up, the following message is displayed:

```
[BATCH Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]
Monitor not running. -stop (BATCH)
ER!
```

Although a Batch Administrator may issue the BATCH -STOP command from any terminal, BATCH -START may be issued only from the supervisor terminal. If you try to START Batch from any other terminal, you will get the message:

```
[BATCH Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]
No right. Must be at system console (BATCH)
```

Note

Avoid stopping and restarting the monitor while jobs are running. Stopping and restarting the monitor in this way slows down the completion of jobs that are running at the time the BATCH -STOP command is issued.

Forced Logouts

As the operator, you can log out the monitor with the LOGOUT ALL or LOGOUT -nnn command. The monitor does not log out immediately; instead, it logs itself out gracefully -- keeping the Batch database intact -- and sends the message:

```
*** BATCH_SERVICE (user nnn on sysname) at hh:mm
Forced logout by operator.
```

After displaying this message, the monitor logs itself out. The act of logging out causes a phantom logout message to be sent to the supervisor terminal:

```
Phantom nnn: Normal logout at hh:mm
Time used: hhh mmm connect, mmm sss CPU, mmm sss I/O.
```

If these messages are not sent within a short period of time, you can repeat the LOGOUT command. This second forced logout causes the monitor to log itself out immediately without sending the "Forced logout by operator" message. (It still sends the "Phantom nnn: Normal logout" message.)

However, using the second LOGOUT command may leave the database in an unknown state. If the database does become scrambled by such a LOGOUT, and if the next BATCH -START does not repair the damage, you must run FIXBAT yourself interactively and, if that fails, run INIT to reinitialize the database.

> **Caution**
>
> If you run INIT, all Batch job data will be lost. Batch users will have to resubmit any waiting, deferred, held, or executing jobs.

Example of Stopping the Batch Monitor

Following is an example of stopping and restarting the Batch monitor, showing how the commands and messages are displayed at the supervisor terminal. BATCH -STATUS commands have been included to show the warning message sent when Batch is not running or when the Batch monitor is paused. Notice that at the time the BATCH -STATUS command is given, there are no jobs in the Batch system.

```
OK, BATCH -STOP
[BATCH Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]
Stop request issued.
OK,
*** BATCH_SERVICE (user 101 on SYA.A) at 15:45
Operator stop.

Phantom 101: Normal logout at 15:45
Time used: 10h 18m connect 12m 06s CPU, 18m 21s I/O
BATCH -STATUS
[BATCH Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]
Warning: jobs are not being processed at this time.

No batch jobs.
OK, BATCH -START
[BATCH Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]
Monitor started up.
OK,
*** BATCH_SERVICE (user 111 on SYA.A) at 15:46
Monitor in operation.
OK, BATCH -PAUSE
[BATCH Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]
Monitor paused.
OK,
*** BATCH_SERVICE (user 111 on SYA.A) at 15:47
Monitor paused.
BATCH -STATUS
[BATCH Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]
Note: the batch monitor is currently not starting up jobs.

No batch jobs.
OK, BATCH -CONTINUE
[BATCH Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]
Monitor continued.
OK,
*** BATCH_SERVICE (user 111 on SYA.A) at 15:47
Monitor continued.
BATCH -STATUS
[BATCH Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]
No batch jobs.
OK,
```

CONTROLLING BATCH JOBS

(This section applies only to operators who are also Batch Administrators; if your own user ID or the SYSTEM ID is not a Batch Administrator, you will be unable to issue many of the commands in this section.) While logged in as a Batch Administrator you can perform any operation on a user job that the user could perform, except that you must refer to each user job by its job ID instead of its job name.

An example of aborting a job is shown below. The example comes from a system in which the MONITOR program is being run without the -HUSH option. (The MONITOR program is run by the Batch monitor startup command file, START_BATCH_MONITOR.COMI.) If you choose to use the -HUSH option, job execution messages will not be logged to the supervisor terminal; they will, however, appear in the Batch monitor log file.

There is a brief interval between the time the JOB command acknowledges the -ABORT option and the time it informs the supervisor terminal that the job has been aborted. Immediately following this acknowledgement is the message that the next waiting job in the queue (if any) has begun executing.

```
OK, JOB #00003 -ABORT
[JOB Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]
Job diamonds.file(#00003) aborted.

OK, DATE
27 Aug 87 14:02:20 Thursday

*** BATCH_SERVICE (user 104 on SYS.A) at 14:02
Job DIAMONDS.FILE for HOOVER(#00003) aborted.

*** BATCH_SERVICE (user 104 on SYS.A) at 14:02
*BATCH* Executing WHITEWASH for HOOVER(#00004).
```

A Batch Administrator can control user jobs just as users are able to do; a Batch Administrator can also HOLD and RELEASE user jobs, something users cannot do. For example, you might HOLD a job if you know that the job needs a resource (such as a tape drive) that is temporarily unavailable. When the resource becomes available, you would then RELEASE the job. JOB -HOLD and -RELEASE are the only JOB commands (other than -DISPLAY and -STATUS, which are available to every user) which may be issued from the supervisor terminal even if SYSTEM is not a Batch Administrator.

See the PRIMOS Commands Reference Guide or the Prime User's Guide for complete information on the following JOB options which you can use to manipulate Batch jobs:

Option	Action
-ABORT -AB	Aborts an executing job. Aborting a waiting or held job has the same effect as cancelling it.
-CANCEL -CAN	Cancels a waiting or held job. Cancelling an executing job does not stop the job from executing, but does mark the job as "not restartable" to prevent the job from starting again (via the -RESTART option or after a system shutdown and startup).
-RESTART -RST	Aborts and restarts an executing job if the job is marked "to be restarted". A job marked "not restartable" (via specification of the -RESTART NO option during job submission or an attempted -CANCEL while executing) is aborted but not restarted.
-CHANGE -CHG	Changes the characteristics of a waiting or held job. Although you can change the characteristics of an executing job, such changes do not take effect unless (and until) the job is restarted.
-HOLD	Holds a waiting job. The job will not be initiated until after it is released.
-RELEASE -RLS	Releases a held job. The job will then be considered waiting (eligible for execution).

HANDLING BATCH QUEUES

Like any user, you can use the BATGEN -STATUS and BATGEN -DISPLAY commands to check the status or parameters of all currently defined queues. If you are a Batch Administrator, you can also use the BATGEN command to block or cap a queue (thus temporarily closing it to new jobs or executions) and to unblock or uncap a queue (reopening it to jobs or executions).

Monitoring Batch Queues

To determine the status of all currently defined queues, issue the command:

```
BATGEN -STATUS
```

The queues and their statuses are displayed in tabular format. For example:

```
[BATGEN Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]

Queue:          Status:
--------------- --------------------
Express-1       unblocked  uncapped
Express-2       unblocked  uncapped
Normal-1        unblocked  uncapped
Normal-2        unblocked    capped
Background-1    unblocked  uncapped
Background-2    unblocked  uncapped (inactive)

OK,
```

The status of a queue can be:

Status	Meaning
unblocked	The queue is available for job submission. This is the typical state for a queue.
uncapped	The queue is available for job execution. This is the typical state for a queue.
blocked	A Batch Administrator has closed the queue to new entries. Jobs which are already in the queue will be processed by the Batch monitor, but the queue will not accept any further jobs until it is unblocked.
capped	A Batch Administrator has shut off processing in the queue. New jobs may be added to the queue, but none of them will be processed until the queue is uncapped. The job which was executing when the queue was capped will continue to completion.

inactive	The queue is outside of its active window (the time period during which the queue is operational) and is unavailable for job execution. Jobs may be submitted to the queue and will be processed when the queue reenters its active window.
flagged for deletion	A Batch Administrator has deleted the queue. The queue is in the process of being deleted, but there are still jobs in it that are waiting, held, or executing. When there are no more jobs in the queue, the Batch monitor removes the queue from the list of currently defined queues, and sends a message to the supervisor terminal indicating that the queue has been deleted.

To display the parameters of all currently defined queues, issue the command:

```
BATGEN -DISPLAY
```

A sample response is:

```
[BATGEN Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]

Queue name = Express-1, unblocked, uncapped.
Active window = FULL;
Default cptime=None, etime=None, priority=9;
Maximum cptime=30, etime=5; Funit=6;
Delta rlevel=0; Timeslice=10;

Queue name = Express-2, unblocked, uncapped.
Active window = FULL;
Default cptime=None, etime=None, priority=9;
Maximum cptime=30, etime=5; Funit=6;
Delta rlevel=0; Timeslice=10;

Queue name = Normal-1, unblocked, uncapped.
Active window = FULL;
Default cptime=None, etime=None, priority=5;
Maximum cptime=None, etime=None; Funit=6;
Delta rlevel=1; Timeslice=20;

Queue name = Normal-2, unblocked, capped.
Active window = FULL;
Default cptime=None, etime=None, priority=5;
Maximum cptime=None, etime=None; Funit=6;
Delta rlevel=1; Timeslice=20;
```

```
Queue name = Background-1, unblocked, uncapped.
Active window = FULL;
Default cptime=None, etime=None, priority=5;
Maximum cptime=None, etime=None; Funit=6;
Delta rlevel=2; Timeslice=40;

Queue name = Background-2, unblocked, uncapped.
Active window = 18:00-23:59;
Default cptime=None, etime=None, priority=5;
Maximum cptime=None, etime=None; Funit=6;
Delta rlevel=2; Timeslice=40;

OK,
```

See Chapter 3 for complete information on the BATGEN -DISPLAY command and its output.

Blocking, Capping, and Setting the Active Window for Queues

To block, unblock, cap, uncap, or set the active window for a queue, (for which you must be a Batch Administrator), type the command:

```
BATGEN BATCHQ>BATDEF
```

Once the file BATCHQ>BATDEF has been read and validated, BATGEN displays a prompt character and waits for a BATGEN command. For example:

```
OK, BATGEN BATCHQ>BATDEF
[BATGEN Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]
>
```

Useful BATGEN Commands

```
ACTIVE_WINDOW { queuename } { hh:mm-hh:mm }
AW            { ALL       } { FULL        }

BLOCK { queuename }
BLK   { ALL       }

CAP { queuename }
    { ALL       }

DISPLAY [{ queuename }]
DP      [{ ALL       }]

FILE
FI

QUIT
Q

STATUS
ST

UNBLOCK { queuename }
UNBLK   { ALL       }

UNCAP { queuename }
      { ALL       }
```

To display a list of queues, type:

```
STATUS
```

To block a queue called EXPRESS4, type:

```
BLOCK EXPRESS4
```

To unblock this queue, type:

```
UNBLOCK EXPRESS4
```

To cap this queue, type:

```
CAP EXPRESS4
```

To uncap this queue, type:

```
UNCAP EXPRESS4
```

If you want to set a time window on this queue of 8 a.m. to 5 p.m. for example, type:

```
ACTIVE_WINDOW EXPRESS4 08:00-17:00
```

Issue another STATUS command to verify that you changed the right queue. Then, type:

```
FILE
```

After you type the FILE command, the system writes the changes to the queue that you have requested.

When to Cap Queues: Use the CAP command to borrow phantoms from Batch queues. (See Chapter 1.)

You can also use the CAP command to prevent jobs which depend on a particular resource (such as a tape drive) from executing until that resource is available. Jobs requiring that resource could be submitted to a particular queue which would be uncapped only when the resource became available.

When to Block Queues: Use the BLOCK command an hour or so before the Batch subsystem is shut down. This ensures that new jobs will not be submitted, but allows jobs which are already in the queue to be processed.

You can also use the BLOCK command to do Batch load balancing. Suppose a site has two queues, one for express jobs and one for longer jobs. Suppose also that the express queue is backed up while the queue for longer jobs is empty. The operator or Batch administrator could block the express queue in order to force new jobs, provided they met the queue requirements, into the longer queue, thereby distributing the Batch work load more evenly among the available Batch phantoms.

Sample BATGEN Session

Here is a sample session using BATGEN to block the queue named Normal-1 and cap the queue named EXPRESS-1. Notice that the queue BACKGROUND-2 is outside of its active window:

```
OK, BATGEN BATCHQ>BATDEF
[BATGEN Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]
> STATUS

Queue:        Status:
------------ ----------------
Express-1     unblocked uncapped
Express-2     unblocked uncapped
Normal-1      unblocked uncapped
Normal-2      unblocked uncapped
Background-1 unblocked uncapped
Background-2 unblocked uncapped (inactive)

> BLOCK NORMAL-1
> CAP EXPRESS-1
> STATUS
Queue:        Status:
------------ ----------------
Express-1     unblocked   capped
Express-2     unblocked uncapped
Normal-1        blocked uncapped
Normal-2      unblocked uncapped
Background-1 unblocked uncapped
Background-2 unblocked uncapped (inactive)

> FILE
OK,
```

5

Monitoring the Batch Subsystem

The operator monitors the status of the Batch subsystem by two methods: by explicitly requesting information, and by reading messages spontaneously displayed by Batch.

REQUESTING GENERAL STATUS INFORMATION

To determine the general status of the Batch system and monitor, issue the command:

```
BATCH -DISPLAY
      -DP
```

The following information is displayed:

- The number of waiting, deferred, and held jobs per queue
- The total number of waiting, deferred, and held jobs (if more than one queue contains waiting, deferred, or held jobs)
- The number of queues containing waiting, deferred, or held jobs (if more than one queue exists)
- The user ID, job ID, user number, and queuename for each currently executing job

For example:

```
OK, BATCH -DISPLAY
[BATCH Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]

Number of waiting, deferred, and held jobs:

Queue    Jobs
-------- ----
Normal-1 1
Normal-2 3

Total=   4 (2 queues)

2 currently running jobs:

  User   Jobid#   # Queue
-------- ------ --- ----------
CLOTHO   #10032 114 Normal-2
CLIO     #00172 117 Normal-1
```

For a brief summary of this information, issue the command:

```
BATCH -STATUS
      -ST
```

BATCH -STATUS displays one line of information that shows the total number of active jobs, the number of waiting, deferred, and held jobs, the number of queues with waiting, deferred, or held jobs, and the number of executing jobs. (The number of queues with executing jobs is the same as the number of executing jobs, because a queue can have only one job executing at one time.) If there are no active jobs, BATCH -STATUS displays the message "No batch jobs".

For example:

```
[BATCH Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]
2 batch jobs; 1 waiting, deferred, or held job in 1 queue;
1 executing job.
```

REQUESTING INFORMATION ON USER JOBS

Two options to the JOB command are useful for obtaining information on user jobs:

```
JOB -STATUS
    -ST
```

and

```
JOB -DISPLAY
    -DP
```

When the JOB -STATUS or JOB -DISPLAY commands are issued by a normal user (non-Batch Administrator), they give information about all jobs which that user has submitted. When issued by a Batch Administrator, the JOB -STATUS command displays the user ID, job ID, status, external name, and queue of all active (that is, executing, deferred, held, or waiting) jobs in the system. The JOB -DISPLAY command, when issued by a Batch Administrator, displays the job name, job ID, user name, queue name, submission time, status, priority, cpu limit, elapsed limit, project, and home directory for all active jobs in the system.

Consider the following JOB -STATUS command, as issued by a Batch Administrator:

```
OK, JOB -STATUS
[JOB Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]

  User    Jobid#    State          External name          Queue
--------  ------  ---------  -------------------------  --------
ACCT      #00002  executing  WEEKLYLEDGER                Normal-1
ACCT      #00003   waiting   RECEIVABLES
ACCT      #00004   waiting   PAYABLES

PERS      #10001     held    NEWHIRES                   Normal-2
PERS1     #10003   waiting   EXITS

SECU      #20005  executing  ENCRYPT                    Express
SECU      #20001   waiting   BADGES

AC1       #30003   waiting   MASTERPLAN                 Background-1
AC2       #30004  deferred   STEP1
MDC       #30005   waiting   STEP2

MDC       #40012     held    $PLAN                      Background-2
MDC       #40013     held    ROADBLOCK
AC2       #40016   waiting   STOPN.SEARCH
```

Note

The results of the JOB -DISPLAY command are displayed, not in job ID order, but in order of execution. Thus, in queue EXPRESS job #20005 appears before job #20001. This ordering happens because Batch always assigns the lowest unused number as a job ID. For instance, if jobs #10002 and #10003 are in the queue, the next arriving job will be assigned #10001, but will be run after the two jobs which arrived earlier.

A Batch Administrator can monitor a specific active job, no matter which user submitted it, by providing the job ID in the JOB -STATUS or -DISPLAY command. For example:

```
JOB #10003 -DISPLAY
```

In addition, you can also monitor any of user SYSTEM's jobs by its job name. For example:

```
JOB USAGE -DISPLAY
```

The following example illustrates the use of the JOB -DISPLAY command with a job ID:

```
OK, JOB #00003 -DISPLAY
[JOB Rev. 21.0 Copyright (c) Prime Computer, Inc. 1986]

Job RECEIVABLES(#00003), user ACCT waiting (queue Normal-1).
Submitted today at 1:39:24 p.m., initiated today at 1:55:11 p.m.
Funit=6, priority=5, cpu limit=None, elapsed limit=None.
Project=DEFAULT, Notify=No
Home ufd=<SYS.A>ACCT
```

The STATUS USERS command includes running Batch jobs in its list of users. Batch jobs can be easily distinguished by the word "batch" in the Line column of the STATUS USERS display.

UNDERSTANDING SPONTANEOUS MESSAGES

The Batch subsystem sends spontaneous messages to the supervisor terminal during operation. Some of these messages are sent by the Batch monitor, some are sent by running Batch jobs, and, rarely, some are sent by Batch users.

The categories of spontaneous messages are:

- Job initiation and termination messages
- Batch monitor status change messages
- Batch queue deletion messages
- Batch subsystem error messages

Messages that are always sent by the Batch monitor come from user BATCH_SERVICE. Other messages may be sent by BATCH_SERVICE, Batch users, or Batch jobs; because these messages may be sent by users with usernames other than BATCH_SERVICE, the messages themselves begin with *BATCH*.

Job Initiation and Termination Messages

If the System Administrator has not chosen to run Batch with the MONITOR -HUSH option (see chapter 2), the Batch system sends messages to the supervisor terminal whenever a job is begun, aborted, restarted, or completed.

Whenever the Batch monitor initiates a user's job, the Batch job itself sends the following message to the supervisor terminal:

BATCH Executing job name for username(job-id).

job name is the filename of the job being run; username is the name of the user who submitted the job; and job-id is the number given the job by the Batch subsystem.

When a job is completed (or aborted), the monitor sends the message:

Job job name for username(job-id) completed.

or

Job job name for username(job-id) aborted.

These messages help you monitor Batch usage and load without having to make explicit inquiries using BATCH -DISPLAY.

When the Batch monitor starts up, it checks for jobs that terminated after the Batch monitor was last shut down, such as during a system shutdown. It treats these terminated jobs much like jobs that terminate while the monitor is running, except that it also considers making such jobs restartable. If it makes a terminated job restartable, it sends the job-aborted message and then sends the following message:

```
*** BATCH_SERVICE (user nnn on sysname) at hh:mm
Job will be restarted.
```

This message indicates that the Batch monitor has automatically marked this job as "to be restarted". If the job is not restartable, this message is not sent and the job is not restarted. The message is displayed for each job marked as restartable. You see this message only when the Batch monitor first starts up (after it sends the "Monitor in operation" message).

The System Administrator can prevent all job initiation, restart, and termination messages from displaying at the supervisor terminal. On a system with heavy Batch use, job initiation and termination messages can produce much wasted paper on a hardcopy supervisor terminal and also could prevent the reception of more important messages by constantly monopolizing the supervisor terminal. See Chapter 2 for more information on preventing the monitor from sending job initiation and termination messages.

Batch Monitor Status Change Messages

The following messages are sent by the Batch monitor to indicate that the monitor has recognized an operator command. These messages are self-explanatory; see Chapter 4 for information on how they are caused.

```
*** BATCH_SERVICE (user nnn on sysname) at hh:mm
Monitor in operation.

*** BATCH_SERVICE (user nnn on sysname) at hh:mm
Monitor paused.

*** BATCH_SERVICE (user nnn on sysname) at hh:mm
Monitor continued.

*** BATCH_SERVICE (user nnn on sysname) at hh:mm
Operator stop.

*** BATCH_SERVICE (user nnn on sysname) at hh:mm
Forced logout by operator.
```

Batch Queue Deletion Messages

If Batch is not being run with the MONITOR -HUSH command, the following messages are sent when a queue is removed from the BATDEF file in BATCHQ by the Batch monitor:

*** BATCH_SERVICE (user nnn on sysname) at hh:mm
Queue queue-name deleted.

*** BATCH_SERVICE (user nnn on sysname) at hh:mm
Removed queue-name from BATDEF.

Both of these messages ultimately result from a BATGEN DELETE command being given to delete the Batch queue named queue-name. The first message is sent if the queue has had at least one job submitted to it since its creation; the second message is sent if the queue has never had any jobs submitted to it.

Batch Subsystem Error Messages

If a condition occurs that prevents the Batch subsystem from functioning correctly (for example, a full disk or a damaged database), the Batch monitor sends a warning or error message to the supervisor terminal. (Often these messages also ring the supervisor terminal's bell.) Refer to Appendix B for information on what these messages mean, and what you should do about them.

APPENDICES

A

Maintaining the Batch Database

This appendix describes how to use FIXBAT and how to clean up Batch queues.

USING FIXBAT

FIXBAT (fix batch) is an offline utility designed to:

- Handle the startup protocol for the Batch monitor, making sure the database is valid before starting the monitor
- Fix any broken pointers within the queue files

These functions can be performed by a Batch Administrator or a system operator.

FIXBAT runs automatically whenever the Batch monitor is started up, to make sure that the database is valid. If the Batch monitor was shut down normally, FIXBAT does nothing. Chapter 2 describes how to configure FIXBAT when it runs at Batch startup.

You can also run FIXBAT at a user terminal or at the supervisor terminal. (If the Batch database becomes invalid, for instance, you run FIXBAT interactively from any terminal to repair it.)

Running FIXBAT at Startup Time

FIXBAT runs automatically whenever the Batch monitor is started up by the BATCH -START command. The command that runs FIXBAT is found in the command file BATCHQ>START_BATCH_MONITOR.COMI:

```
RESUME FIXBAT -STARTUP SAVE
```

This command checks to see that the database is valid before starting the monitor.

When BATCH -START Runs FIXBAT

This section contains a brief explanation of what happens when the BATCH -START command is executed. Details are provided in the sections that follow.

1. The BATCH -START command starts up a phantom (logged in as BATCH_SERVICE) that runs the command file in BATCHQ named START_BATCH_MONITOR.COMI.

2. The command file runs FIXBAT.

3. FIXBAT checks to see if Batch terminated normally the last time it was run (that is, to see if BATCH -STOP was the last command given to the Batch subsystem). If Batch terminated normally, FIXBAT stops. If not, FIXBAT performs the steps below.

4. FIXBAT checks to be sure the Batch database is valid.

5. If the database is not valid, FIXBAT attempts to fix the database. If FIXBAT cannot repair the database, it aborts with an error message.

6. When FIXBAT finishes its tasks, the command file executes the command RESUME MONITOR. This starts the Batch monitor and sends the following message to the supervisor terminal:

   *** BATCH_SERVICE (user nnn on sysname) at hh:mm
   Monitor in operation.

Note

The "Monitor in operation" message that appears after you type the BATCH -START command can take some time to appear. The length of time between starting the monitor and the display of the message depends on the amount of work FIXBAT has to do. Note also that FIXBAT cannot begin to work until the system time and date are set.

Invoking FIXBAT Interactively

FIXBAT resides as a program, FIXBAT.RUN, in the BATCHQ top-level directory. To run FIXBAT interactively:

1. Use the command BATCH -STOP to log out the Batch monitor (if it is running).
2. Log in as a Batch Administrator.
3. Type RESUME BATCHQ>FIXBAT, specifying the desired options (explained below).

The FIXBAT Command and its Options

The format for the FIXBAT command is:

```
RESUME FIXBAT [options]
```

The FIXBAT command has three options:

Option	Meaning
-DAYS n	This is an obsolete command option; it is no longer necessary.
-QUIET	Suppresses the message to the terminal when FIXBAT removes a job from the queue.

-STARTUP argument

Tells FIXBAT to start the BATCH monitor. When this option is specified, FIXBAT assumes it is being run by the BATCH -START command. That is, FIXBAT assumes it is being run as a phantom started from the supervisor terminal. As a result, the -STARTUP option CANNOT be specified as an interactive command; FIXBAT may be run with the -STARTUP option only if it is being run from the file START_BATCH_MONITOR.COMI. When the -STARTUP option is specified, the phantom running FIXBAT becomes the Batch monitor after FIXBAT is finished.

The -STARTUP option takes one of four arguments: SAVE, SPOOL, DELETE, or NOLOG. These arguments tell FIXBAT what to do with the Batch monitor command output (log) file.

Argument	Meaning
SAVE	Renames O_LOG, the current command output log file, to OLDLOG (deleting OLDLOG first, if it already exists). Creates a new copy of O_LOG, and sends all future command output there.
SPOOL	Spools O_LOG, the current command output log file, calling it BATCH.LOG. Erases the old O_LOG file and opens a new O_LOG file.
DELETE	Opens O_LOG as a command output log file. (If a copy of O_LOG already exists, it is truncated when FIXBAT starts up, destroying the existing contents.)
NOLOG	Takes no action with regard to command output files; no log file is kept.

Notes

Changing the argument used with -STARTUP can result in old comoutput files (OLDLOG or O_LOG) being left around or recent log files being deleted. Whenever a change like this is made, the System Administrator must ensure that any existing files are taken care of before the Batch monitor is started up with the new START_BATCH_MONITOR.COMI file.

The DELETE and NOLOG arguments to the -STARTUP option are not recommended, as they make it difficult for the System Administrator to analyze any Batch problems that may occur. However, if problems do occur, the Batch subsystem tries to create a file named ERROR. in the BATCHQ directory giving some information on the error.

The O_LOG file, when generated by -STARTUP, contains an identifying first line, suitable as a header line for a spool file. It contains the time of day, day of the week, date, and the FIXBAT revision number. After that information, the file contains two blank lines. The log trail of what both FIXBAT and the Batch monitor did begins on line 4.

The log trail can include such comments as "Fixing database", "Deleted T$0001", "Deleted C00041", which are not errors, simply notifications that certain files deemed useless by FIXBAT were deleted.

The log trail can also include information on any jobs that were deleted from the queues (unless -QUIET was specified). In addition, any strange file formats (such as partial queue entries) are noted here.

One thing to note is that most of FIXBAT's output when -STARTUP is specified is preceded by the time of day on each output line. This is useful for determining the timing of operations related to a user complaint that some undefined thing went wrong. For example, if you run FIXBAT interactively, the following message might be displayed:

```
Deleted T$0000
```

On the other hand, if FIXBAT is running as part of BATCH -START, the same message might appear as follows:

```
06:43:52 Deleted T$0000
```

Both the Batch monitor and FIXBAT write the time of day into the log file in this manner. However, if more than one line is sent to the log file during a particular time, the time is sent with only the first line; subsequent lines are indented with spaces accordingly. For example:

```
06:44:42 Deleted C10001
         Deleted C50233
06:44:44 FIXBAT finished.
```

If FIXBAT or the Batch monitor runs past midnight, it sends the new date to the log file along with the first line it sends on the new day. For example:

```
Date: 02/14/85
02:20:18 Operator stop.
```

If FIXBAT aborts, the cause can generally be found by looking at the log file. Usually, deleting the offending file and restarting the Batch monitor (and therefore FIXBAT) is the fastest way to fix any problems.

If FIXBAT is run by the BATCH -START command, it runs as the Batch monitor. In this case, when FIXBAT finishes, the Batch monitor startup command file issues a RESUME MONITOR command and the monitor revision number is displayed, followed by a log trail of the Batch monitor's activities.

How FIXBAT Works

FIXBAT begins by deleting temporary files out of BATCHQ. Here, temporary files are defined as files with six-character names beginning with T and with the last five characters being decimal digits, the first representing the queue number and the last four being the job ID.

These files are generated by the Batch monitor to bootstrap Batch jobs, including attaching to the home directory. Under some conditions, these temporary files may not be deleted; the Batch monitor usually deletes them itself. When a temporary file is deleted, the message "Deleted <name>" is displayed.

If FIXBAT decides to fix the database, it displays the message "Fixing database", removes any extraneous temporary files, and checks the database's integrity.

FIXBAT Error Messages and Responses

While FIXBAT is running, it displays certain messages describing what it is doing. If something goes wrong, FIXBAT displays an error message. Refer to Appendix B for a complete list of Batch and FIXBAT error messages.

In general, if FIXBAT aborts, it means that certain parts of the Batch database (usually job data) are irretrievably lost. Deleting the offending file and rerunning FIXBAT may resolve the particular error. However, FIXBAT may abort again on a different file. If FIXBAT cannot fix the database, run the INIT program.

CLEANING UP QUEUES

Under Rev. 21.0 Batch, a Batch queue is considered full only if it contains 10,000 Batch jobs simultaneously. (Under earlier versions of Batch, a queue was considered full after it had processed 10,000 Batch jobs.) It should now be extremely rare for a Batch queue ever to become full.

When full queues exist, the following things happen:

- When users submit jobs to the full queue (using the JOB command's -QUEUE option), they receive the error message:

  Queue full.

- When users submit jobs without specifying queues, the monitor conducts its usual search for queues. However, it ignores the full queue, treating it as if it were blocked. If the full queue is the only queue that meets a user's requirements, that user receives the error message "No queue available for job." (If some other queue is acceptable, the monitor simply submits the jobs to that queue.)

If a queue should become full in your Batch system, we recommend simply adding another queue with the same characteristics and encouraging users to submit their jobs to that queue. No other action is necessary, since a full queue would cease to be full as soon as the job which was executing at the top of the queue ran to completion.

B

Batch Messages

Messages in this appendix include both those sent to user terminals and those most often seen at the supervisor terminal. Some messages merely report the progress of a job. Others report mild or serious errors.

When a serious problem occurs (for example, when the Batch monitor discovers that the Batch database has been damaged), three things generally happen:

- A message is sent to the supervisor terminal.
- The bell at the supervisor terminal rings.
- The Batch monitor logs itself out.

When these occur, you should look at the Batch monitor log file (if one exists) and at the error message sent to the supervisor terminal. By looking at these two sources of information, you can discover whether the error being reported is the result of an earlier error. If you cannot discover what happened to the Batch subsystem, print out a copy of the file ERROR. and consult your Customer Service Representative.

BATCH MESSAGES

Following is a list of Batch messages and their meanings, not including FIXBAT messages. (FIXBAT messages are listed at the end of this Appendix.) The nature of each message is indicated in parentheses at the beginning of each explanation.

The following types of messages exist:

Response: Displayed after you type something to the Batch subsystem. These are informative messages only; they do not indicate that anything is wrong with your command or the Batch subsystem.

Warning: Displayed to indicate that some part of your request or something about the current state of the Batch subsystem may affect the successful honoring of your request.

Fatal: Displayed to indicate that your request failed either due to an error on your part or to the temporary inability of the Batch subsystem to honor your request.

Severe: Displayed to indicate a severe error involving the Batch database. Typically, you have to run FIXBAT, INIT, or even FIX_DISK to repair the problem. (Running INIT causes all Batch job data -- and possibly all Batch queue definitions -- to be lost.)

Supervisor: Any message with this note is sent to the supervisor terminal; the severity of the message will be indicated separately.

Query: Displayed to elicit a response from you. Answer YES or NO, as appropriate.

The messages, listed in alphabetical order, start with those messages that begin with variable names.

- <extra-text> seen when end-of-line expected.

(Fatal) You have typed in more text than was expected; extra-text is the part of your command which was not understood. If this message appears after you enter the BATGEN command, you are returned to PRIMOS, and PRIMOS displays the ER! prompt. If this message appears in response to a BATGEN command or subcommand, an interactive user is left in command/subcommand mode, but a command file or CPL program is aborted.

- <queue-priority> is out of range. -PRIORITY

(Fatal) The number queue-priority, supplied with the -PRIORITY option, was out of range. The range for queue-priority is 0 through 9. Resubmit the job using a valid value for queue-priority.

- <unit-number> is out of range for -FUNIT

(Fatal) The number unit-number, supplied with the -FUNIT option, was out of range. The range for unit-number is normally 1 through 32. If you wish to supply an FUNIT between 33 and 128, first return to PRIMOS command level; next use the PRIMOS OPEN command to open file unit 127; next, OPEN file unit 128. Now resubmit the job, using a valid value for unit-number.

- Another BATCH monitor may be running.

(Supervisor, Fatal) The Batch monitor cannot start up because another copy of the program is already running. (You might also get this message if another user were running FIXBAT or INIT.) If you know who is running the program, wait until the program finishes running, then use BATCH -START to restart the Batch monitor. If you do not know who is running the program, periodically attempt to restart the Batch monitor.

- Bad $$ JOB line.

(Fatal) A file submitted using the JOB command has a $$ line as the first noncomment line, but the $$ command is not a $$ JOB command. The file should be changed so that the $$ line is valid. The use of $$ is reserved for future expansion by BATCH.

- Bad queue control file.

(Severe) One of the Batch subsystem database files is inaccessible or has a bad format. The Batch subsystem is inoperative until the database is fixed.

- Bad queue definition file.

(Fatal) A file referenced by BATGEN does not comply to format requirements; it is not a valid queue definition file. If this error occurs anywhere other than the BATGEN program, then the system Batch definition file has been overwritten with invalid data, and the Batch subsystem will not work until the definition file is repaired.

- *BATCH* Database invalid.

(Supervisor, Severe) The monitor logs itself out after sending this message, and the Batch system is left inoperative. (Users receive error messages if they try to invoke JOB or BATCH.) The System Administrator should determine what the error is and fix it if possible. If the Batch monitor generates a command output (log) file, that file should reveal the source of the error. The file is named O_LOG in BATCHQ; if the file BATCHQ>START_BATCH_MONITOR.COMI runs FIXBAT.SAVE with the -STARTUP NOLOG argument, there will be no log file available.

In general, if the exact cause of the problem is not known (such as a Pointer mismatch error in the database, or a disk write-protected error), run FIXBAT. If that fails, you should first list all your queue definitions with BATGEN, then resume the BATCHQ>INIT program with the -RESET_QUEUES option. This will reinitialize the database, deleting all your queue definitions as it goes. If running BATCHQ>INIT doesn't work, there are probably disk errors. If it does work, redefine the Batch queues using BATGEN and start the Batch monitor up again. (All job data is deleted by the INIT program.)

- *BATCH* Info in "BATCHQ>ERROR.". (BILD$B)

(Severe) The source of an error has been successfully written to the BATCHQ>ERROR. file. (Note that the period is included in the pathname.) This message is usually preceded and followed by other severe error messages. (Note that the contents of the file ERROR. are primarily useful to a representative from your Customer Support Center in discovering what sort of error occurred; they are unlikely to be meaningful to the typical System Administrator.)

- BATCHQ cannot be in a password MFD.

(Fatal) Rev. 21.0 Batch cannot run if the directory BATCHQ is kept on a password MFD. Either change the MFD on which BATCHQ is kept to ACL protection or run Rev. 20.0 Batch. (To change an MFD to ACL protection, attach to that directory and type "SAC", after checking that no other program on that MFD depends on the password.)

- Cannot run from console

(Fatal) You have tried to run the Monitor program from the console; it can only be run from the Batch startup file START_BATCH_MONITOR.COMI.

- Can't start batch job!

(Supervisor, Fatal) The Batch monitor has not been started from the supervisor terminal, and it cannot log in processes under different login names or log out other processes. The monitor logs itself out

gracefully after sending this message. Issue the BATCH -START command from the supervisor terminal if this situation occurs.

- (Changes made)

(Response) The changes specified in a JOB -CHANGE operation have been made. If the job is initiated after the changes are made, then it executes with the specified changes in place. The job status is displayed after the (Changes made) message is displayed.

- Command or CPL file required as first arg. on submission.

(Fatal) A JOB command has been given with job options (such as -HOME, -PRIORITY, -CPTIME, and so on) but no CPL program name or command file name was specified. The command format is:

```
JOB pathname [options]
```

- CPU time cannot exceed 1000000

(Fatal) You have asked for a CPU time greater than one million seconds either when submitting a job or defining a queue. Enter a lower number of seconds.

- Cpu limit must be specified.

(Fatal) A job has been submitted without a -CPTIME option, and all appropriate queues in the BATCH system require jobs to specify maximum CPTIMEs. Resubmit the job, specifying a maximum CPU time limit.

- Creating new batch definition file: pathname (BATGEN)

(Response) The pathname specified does not exist. When the FILE command is given, it creates the specified file and puts the Batch queue definitions in it. BATGEN initializes its environment when it cannot find pathname so that no queues are defined.

- Date and time not set. (Batch)

(Fatal) A BATGEN or JOB command, or a RESUME BATCHQ>INIT command, has been issued from the supervisor terminal before the system date and time were set. These parts of the Batch system cannot be run until the system date and time are set using the SETIME command from the supervisor terminal.

- Deferred until <date> at <hh:mm:ss>.

(Response) You have submitted a job for a later date and time. If date is not specified the job will be executed the day it is submitted. If no time is specified, the job execution defaults to midnight of the date specfied. If the specified date and time has already passed, a warning message will be sent to the user, and the job will be immediately available for execution.

- Elapsed time limit must be specified.

(Fatal) The -ETIME option has not been specified for a queue which requires that option. (That is, the default elapsed time limit for that queue is greater than the maximum elapsed time limit for that queue.) Resubmit the job with the -ETIME option specified. To determine the maximum limits for queues, use BATGEN -DISPLAY.

- End of line.

(Fatal) A required keyword or option was not present on the command line. The message generally contains more information on what is expected. Reenter the command with the additional requested information.

- End of line. Illegal <option-name> argument

(Fatal) A job parameter option, option-name, was specified last on the JOB command line or on the $$ JOB line, but had no argument (end of line). Supply the information required by option when you reenter the command or modify the $$ JOB line accordingly.

- End of line. Queue name required

(Fatal) A BATGEN command which requires a queue name has been issued without specifying any queue. (ADD, MODIFY, BLOCK, UNBLOCK, CAP, UNCAP, ACTIVE_WINDOW, and DELETE all require queue names.) Reenter the command with the queue name desired.

- End of line. Value required

(Fatal) While in BATGEN subcommand mode, a subcommand that requires at least one numeric parameter has been issued, but no number was given. Subcommands requiring at least one numeric parameter are CPTIME, ETIME, FUNIT, PRIORITY, TIMESLICE, and RLEVEL. Note that the CPTIME and ETIME subcommands accept two parameters, either or both of which can be the keyword NONE, indicating no limits. Reenter the subcommand with the value desired. For example: TIMESLICE 10.

- Enter queue characteristics:
  $

(Response) The ADD or MODIFY command, given while in BATGEN command mode, has succeeded. The user is now in BATGEN subcommand mode, identified by the $ prompt instead of the > prompt used in BATGEN command mode. To reenter command mode from subcommand mode, type QUIT or RETURN. RETURN saves the information modified while in subcommand mode; QUIT discards it, asking for verification if any of it was changed.

- Environment modified, ok to quit?

(Query) A QUIT command has been issued while in BATGEN command mode, after the environment was modified. Valid responses to this question are YES, NO, and OK. If you type "YES" or "OK", all the changes made during this BATGEN session will be lost.

- Error: <message-text> (<program>) err=<nnnn>

(Supervisor, Severe) An error occurred in the Batch subsystem, encountered either by the Batch monitor, a Batch job, or a Batch user. Typically, other fatal error messages are sent to ERROR. and to the supervisor terminal and the database is invalidated. Use FIXBAT to fix the database. If this fails, get a Batch Administrator to run BATCHQ>INIT.RUN. nnnn is the error code, program is the program or subroutine in Batch that encountered the error, and message-text is additional information on the error.

- Extraneous text on command line. (MONITOR)

(Fatal) A bad command line exists in BATCHQ>START_BATCH_MONITOR.COMI. The command line should read RESUME MONITOR or RESUME MONITOR -HUSH, but some excess information currently follows the -HUSH option. Fix the command line in the file and restart the Batch monitor.

- File has no non-comment lines. <pathname> (JOB)

(Fatal) A user has submitted a command file or CPL program pathname which either is empty or is made up entirely of comment lines.

- Forced logout by operator.

(Supervisor, Response) The Batch monitor has been forcibly logged out. It sends this message to indicate that it has successfully logged out without leaving the Batch data base in an indeterminate state.

- Home directory required.

(Fatal) The -HOME option was not present on the JOB command line or on the optional $$ JOB line during submission, and JOB was unable to determine the attach point of the submitting job. Resubmit the job, and include the -HOME option followed by the absolute pathname indicating where the job is to execute. If the pathname is too long to fit, use a shorter version of it when you resubmit the file. First, edit the file to include an ATTACH command with a relative pathname that attaches down through the remaining subdirectories to reach the destination. Then, resubmit the job, using the shortened version of the pathname.

- Home=<pathname>

(Response) During job submission, the -HOME option has not been specified on the command line or in the file ($$ JOB), but the job was successfully submitted. The JOB command determined the home attach point of the submitting user to be pathname, and used this as the home attach point of the submitted job.

Note

JOB does not attempt to determine whether the user can attach to the home directory as owner. If the user cannot attach because of a bad password error or an insufficient access rights error, the job terminates, and a requested command output file is not produced.

- Illegal -CHANGE option.

(Fatal) The options -QUEUE and -PRIORITY are invalid during a -CHANGE operation using the JOB command, because queue and queue priority of a job cannot be changed. Cancel or abort the job and resubmit it to the appropriate queue with the desired queue priority.

- Illegal answer.

(Warning) This warning is displayed when the answer to a question is not YES, NO, or OK. The question is asked again. These questions are asked when a user tries to QUIT out of BATGEN command or subcommand mode after modifying the environment or queue.

- Illegal combination. <option>

(Fatal) A job submission parameter (such as -ACCT, -HOME, -QUEUE, and so on) has been specified on the same JOB command line as an option to perform an incompatible JOB command such as -CANCEL, -DISPLAY, -ABORT,

and so on. option is the second (conflicting) option. Use separate JOB commands to perform separate functions.

- Illegal combination. -FUNIT (JOB)

(Fatal) A CPL job has been submitted using the -FUNIT option. The FUNIT option is meaningless for CPL jobs. Resubmit the job without the -FUNIT option.

- Illegal limit.

(Fatal) A parameter supplied to the -CPTIME or -ETIME option during job submission/changing is not a valid limit. That is, it is less than or equal to zero or is not a valid decimal number, and it is not the keyword NONE. Reenter the command with valid limits.

- Illegal name.

(Fatal) One of the Batch programs is expecting a name or command, but it instead reads a character string beginning with a dash (-), indicating that an option is present. (If you must include a dash in your character string for some reason, enclose it in quotation marks.)

- Illegal number. <n> (BATGEN)

(Warning) The numeric parameter n supplied for a BATGEN subcommand is not a valid decimal number. Reenter the line with a valid decimal number. (All numbers input by the Batch subsystem are decimal.) Subcommands that can return this error are CPTIME, ETIME, FUNIT, PRIORITY, TIMESLICE, and RLEVEL. Note that the CPTIME and ETIME subcommands accepts the keyword NONE indicating no limits, but flags the number 0 as an invalid number. Also, these two subcommands interpret the numbers as numbers ranging from 1 through 1000000, whereas the numbers for the other subcommands range from 0 through 32767.

- Illegal number. <n> (JOB)

(Fatal) The numeric argument n supplied for the -FUNIT or -PRIORITY option during job submission using the JOB command is not a valid decimal number. Reenter the command line with valid numeric parameters.

- Illegal option.

(Fatal) One of the Batch programs is expecting an option, namely, an unquoted character string beginning with a dash (-). Reenter the command line with a valid format.

- Illegal queue name. <queue-name> (BATGEN)

(Warning) An attempt has been made to add a queue with a name (queue-name) that does not comply with filename rules. (These rules are: the first character must not be a digit; and the character set is limited to alphabetics, digits, and selected special characters). Reenter the command with a valid queue name. Note that a queue name of ALL is invalid, so that the DELETE ALL is not issued except when deleting all queues is desired.

- Illegal queue name. <queue-name> (JOB)

(Fatal) The queue name (queue-name) specified after a -QUEUE option while submitting or changing a job does not comply with queue name format rules. Use BATGEN -STATUS or -DISPLAY to determine the names of valid queues.

- Illegal value. <illegal-rlevel-value> (BATGEN)

(Response, Fatal) An illegal value has been entered for the RLEVEL subcommand. Legal values for the RLEVEL command are 0 through 7 and IDLE.

- In <pathname>:

(Fatal) This opening phrase precedes JOB error messages when the errors originate in a $$ JOB line within the file pathname. The error message also includes the $$ JOB line itself.

- In the submission file:

(Fatal) This opening phrase precedes JOB error messages when the errors originate in the $$ JOB line of a file, and the submission program cannot determine the file's pathname to display it.

- Incorrect username.

(Fatal) The username of the submitting user does not match the username in the $$ JOB line of a file submitted using the JOB command. Edit the

file and change the username in the $$ JOB line to the username of the submitter. Note that a username of * means that any user may submit the file.

- Insufficient access rights. BATCHQ>BATDEF

(Fatal) This message usually means that the Batch installation file did not run to completion; as a result, nobody but a Batch Administrator can use the Batch subsystem. Ask your Batch Administrator to check to see whether INIT finished running when Batch was installed.

- *** Invalid batch database, please contact your system administrator.

(Severe) This message means that the Batch subsystem program being run detected an error (such as disk failure, pointer mismatch, or misprotected file) in the Batch system database, and flagged the database as invalid. Notify the System Administrator, who has the responsibility for reinitializing the database, running FIXBAT, or running FIX_DISK, as appropriate. The BATCH and JOB commands will not work until the situation is resolved.

- Invalid -COMO pathname. <invalid-pathname> (JOB)

(Fatal) The format of the pathname specified with the -COMOUTPUT option to the JOB command is invalid. Re-issue the command.

- Invalid -DEFER option time. <invalid-time> (JOB)

(Fatal) The format of the time or date specified with the -DEFER option to the JOB command is invalid. Re-issue the command.

- Invalid project id. <invalid-project-id> (JOB)

(Fatal) Either the format of the project ID specified with the -PROJECT option to the JOB command is invalid, the user does not belong to the specified project, or the project does not exist.

- ?Job <jobname> (<jobid>) <job-status>.

(Warning) An attempt has been made to use the JOB command on a job named <jobname> with an internal job ID of <jobid>, but its status (<job-status>) prevented such an operation. Examples of such attempts include trying to restart a completed job and attempting to release a job that is not held.

- Job <jobname> for <username>(<jobid>) <job-status>.

(Supervisor, Response) The Batch monitor has changed the status of a job. (This message is not displayed when the monitor changes a restarted job back to waiting). jobname is the external name of the job, username is the submitting user, jobid is the internal job ID, and job-status is either aborted or completed.

- Job name required.

(Fatal) A required job identifier (an internal job ID or external name) has not been specified with one of the following options: -CHANGE, -CANCEL, -ABORT, -RESTART, -HOLD, and -RELEASE. Reenter the command with the job identifier. Examples:

```
JOB TOP -CHANGE -PRIORITY 9
JOB #10032 -ABORT
```

- (Job no longer restartable)

(Response) A JOB -CANCEL has been performed on an executing job. The job itself was not canceled, but it has been flagged as being unrestartable. In this state, use of the -RESTART option aborts the job but does not restart it.

- (Job not changed.) Queue not found. <queue-name> (JOB)

(Fatal) A request to change the characteristics of a job cannot be honored because the queue to which the job was submitted (queue queue-name) cannot be found in BATCHQ>BATDEF. This is an unusual error, but it can result if a queue is deleted at a particular moment during the JOB -CHANGE operation.

- Job not found.

(Fatal) The job referred to in a JOB -CHANGE, -CANCEL, -ABORT, -RESTART, -HOLD, or -RELEASE command cannot be found by searching the active jobs list. This can mean one of three things: that no job exists with that name; that all jobs with that name have completed, aborted, or canceled; or that a job exists with that external name but the user making the request is not the same user who originally submitted the job.

- (Job not restartable)

(Warning) A JOB -RESTART has been performed on an unrestartable job. An attempt is made to abort the job after this message is displayed.

- (Job restarted)

(Response) A JOB -RESTART has been performed on a restartable job. Although an error message can appear after this message, the job is generally restarted unless a JOB -CANCEL or JOB -CHANGE -RESTART NO command is issued. One possible error message, "Insufficient access rights", may appear if the user is logged in as SYSTEM or BATCH_SERVICE and has restarted another user's job from a user terminal, or if the process has recently logged out. "Not found" can also be displayed if the process is logged out.

- Monitor already started. (or Monitor already paused.)

(Response) Informs the operator that the monitor is already in the state requested.

- Monitor continued.

(Supervisor, Response) The Batch monitor has been continued. Jobs may now be started up. This message is displayed as the result of a BATCH -CONTINUE command following an earlier BATCH -PAUSE command.

- Monitor in operation.

(Supervisor, Response) Tells the operator that the Batch monitor has finished fixing the database (by running FIXBAT) and is ready to process jobs.

- Monitor not paused.

(Fatal) The Batch monitor was not paused, possibly because the user did not have the authority to interrupt it.

- Monitor paused.

(Response and Supervisor both) The Batch monitor has been paused or has been started up in the paused state. No jobs will be started up. Use BATCH -CONTINUE to continue the monitor.

- Monitor started up.

(Response) Tells the operator that the monitor has been started up and is now going through an initialization phase.

- Multiple jobs with this name (use internal name).

(Fatal) The job name used in the JOB command belongs to at least two jobs submitted by this user. The job ID (internal name) must be used in this case. Use JOB -STATUS to determine the internal and external names of all active jobs belonging to the user issuing the command.

- Multiple occurrance.

(Fatal) An option has been specified twice either on the JOB command line or on the $$ JOB line during job submission or job changing (for example, JOB TEST -HOME HERE -HOME THERE). (If an option is specified once on the JOB line and once on the $$ JOB line, no error results; the parameter on the JOB line takes precedence.) Reenter the command, but specify each option only once.

- Must be batch administrator. (INIT)

(Fatal) The user has tried to issue a command which only Batch Administrators may issue. Ask a Batch Administrator to issue the command, or ask the System Administrator to make you a Batch Administrator.

- Must be first option.

(Fatal) The option -CHANGE, -CANCEL, -ABORT, -RESTART, -STATUS, -DISPLAY, -HOLD, or -RELEASE must be the first option on the JOB command line (a job identifier may proceed it for some options.)

- Must run as BATCH_SERVICE.

(Fatal) You have tried to run the MONITOR program from a terminal; it can only be run from START_BATCH_MONITOR.COMI, the Batch startup program.

- My disk is full. Please help me.

(Supervisor, Warning) The Batch monitor has encountered a "Disk Full" condition while trying to initiate a job. It retries the job initiation every five minutes, sending this message after each unsuccessful attempt. This message causes a bell to ring at the supervisor terminal. You must delete some files from the disk to free up space.

- My quota is exceeded. Please help me.

(Supervisor, Warning) The Batch monitor has encountered a "Quota Full" condition while trying to initiate a job. It retries the initiation every five minutes, sending this message after each unsuccessful attempt. This message causes a bell to ring at the supervisor terminal. You should delete some files from the disk to free up space or raise the quota set on the BATCHQ directory.

- No active jobs

(Response) Displayed by a JOB -DISPLAY or -STATUS command, this message indicates that there are no waiting, deferred, held, or executing jobs belonging to the user. If the user is SYSTEM, a Batch Administrator, or BATCH_SERVICE, then there are no jobs that are waiting, deferred, held, or executing in the entire system. The JOB command may also display "named jobname", with either "for user username" or "in system" appended to the message, depending on whether the user specified a job name and whether the user is privileged (logged in as SYSTEM, a Batch Administrator or BATCH_SERVICE).

- No batch jobs.

(Response) You have issued a BATCH -STATUS command, but there are no jobs in the Batch queues.

- No configured queues.

(Response) A BATGEN -STATUS or -DISPLAY command has been issued for a Batch definition file that has no defined queues.

- No job changes specified.

(Fatal) The -CHANGE option has been given to the JOB command, but no actual changes were specified on the command line. Reenter the command, specifying changes to be made following the -CHANGE option on the command line.

- No longer executing.

(Fatal) A JOB -ABORT or JOB -RESTART has been performed on a job that had execution status, but by the time the execution file was read in to determine the usernumber of the process, the process had disappeared. If the message "(Job restarted)" is displayed, then the job has restarted. Although the operation itself was unsuccessful, the desired results have been achieved.

- No phantoms.

(Warning) The Batch system cannot process jobs because it has no phantoms allocated to process them. Batch will try again later. Ask your System Administrator when Batch will next be allocated some phantoms.

- No queue active windows set.

(Response) The BATGEN command ACTIVE_WINDOW ALL has been issued when all queues already had the same active window set.

- No queue available for job.

(Fatal) A job submitted with the JOB command did not specify a -QUEUE option, and no suitable queue can be found. Suitability requirements include CPU and elapsed time limits within the confines of the queue, queue unblocked, and so on. Use the BATGEN -STATUS or -DISPLAY command to display a list of valid queues and their status.

- No queues have waiting, deferred or held jobs.

(Response) A BATCH -DISPLAY command has been issued when there were no queues with any waiting, deferred, or held jobs. A queue can have one executing job not considered to be waiting, deferred or held.

- No queues were blocked.

(Response) The BLOCK ALL command has been issued when all queues were already blocked.

- No queues were capped.

(Response) The CAP ALL command has been issued when all queues were already capped.

- No queues were deleted.

(Response) The DELETE ALL command has been issued when all queues were already deleted.

- No queues were unblocked.

(Response) The UNBLOCK ALL command has been issued when all queues were already unblocked.

- No queues were uncapped.

(Response) The UNCAP ALL command has been issued when all queues were already uncapped.

- No right. Must be at system console.

(Fatal) You have issued the BATCH -START command from a user terminal. Go to the supervisor terminal and issue it again.

- No right. Must be Batch Administrator or Console (JOB)

(Fatal) A -HOLD or -RELEASE operation has been attempted using the JOB command, and the user was neither logged in as a Batch Administrator nor using the supervisor terminal.

- No running jobs.

(Response) A BATCH -DISPLAY command has been issued, and no jobs were running. Jobs can be waiting when there are no running jobs even when the monitor is running and there are free phantoms. There is always a small amount of turnaround time between submittal and execution of a job.

- Not an absolute treename.

(Fatal) The home directory specified with the -HOME option of the JOB command is a relative pathname. That is, it began with *>. Resubmit the job, giving an absolute pathname after the -HOME option.

- Not found. Checking database status. (CHEK$B)

(Fatal) A file which Batch uses to determine which jobs are in operation is missing. A Batch Administrator should attach to BATCHQ and run INIT. All jobs in the Batch queues will be lost, but the Batch system should return to normal operation.

- Not your job.

(Fatal) A job has been referenced using an internal name in the JOB command, but the job does not belong to the user making the reference. Use the JOB -STATUS command to obtain a list of all active jobs submitted by the requesting user.

- Note: the batch monitor is currently not starting up jobs.

(Response) A Batch subsystem command has been issued while the Batch monitor is paused. No jobs are started up while the monitor is paused.

- Null home directory.

(Fatal) The home directory specified with the -HOME option of the JOB command is a null string. Resubmit the job with an absolute pathname after the -HOME option.

- Operator stop.

(Supervisor, Response) The monitor received a stop request via a BATCH -STOP command. The monitor logs out after sending this message.

- Out of range.

(Warning) A BATGEN subcommand has been given a numeric parameter that is out of range for that subcommand. The ranges are: 1 through 128 for FUNIT, 0 through 9 for PRIORITY, 1 through 99 for TIMESLICE, and 0 through 7 for RLEVEL. Reenter the subcommand with the correct parameter.

- Please answer "YES" or "NO".

(Fatal) You have typed something other than "YES" or "NO" in answer to a question. Type one of those two words again.

- Please FILE.

(Warning) A QUIT command has been issued while in BATGEN command mode, after the environment has been modified; the question "Environment modified, ok to quit?" has been asked, and the answer was NO. This message is a reminder to use FILE to create a modified environment.

- Please RETURN.

(Warning) A QUIT subcommand has been given while in BATGEN subcommand mode, after the queue characteristics have been modified; the question "Queue definition modified, ok to quit?" has been asked, and the response was NO. This message is a reminder that the proper way to leave a subcommand session is to use the RETURN subcommand.

- Queue <queue-name> already exists (<queue-status>).

(Warning) While in BATGEN command mode, an attempt has been made to add a queue named queue-name that already existed. queue-status is blocked, unblocked, or flagged for deletion. To change the queue definition, use the MODIFY subcommand. However, if the queue is flagged for deletion, any attempt to block, unblock, modify, or display it causes the "Unknown queue name" message to be displayed.

- Queue <queue-name> deleted.

(Supervisor, Response) Queue queue-name, flagged for deletion in the BATDEF file, has just been deleted by the Batch monitor because the queue became empty.

- Queue <queue-name> flagged for deletion.

(Warning) While in BATGEN command mode, an attempt has been made to delete a queue named queue-name that was already flagged for deletion. To allow the queue to disappear, use FILE to write out the BATDEF file. The queue disappears when it contains no more waiting, held, or executing jobs. It can then be added again.

- Queue active window(s) set:

(Response) This is an informational response to a successful invocation of the BATGEN ACTIVE_WINDOW command.

- Queue blocked.

(Fatal) The queue referred to by a -QUEUE option during job submission is currently blocked to new submissions. Try it again later, or use another queue. Use the BATGEN -STATUS to display a list of all queues.

- Queue(s) blocked:
  queue-name, queue-name...

(Response) Your BATGEN BLOCK command successfully blocked the named queues.

- Queue(s) capped:
  queue-name, queue-name...

(Response) Your BATGEN CAP command successfully capped the named queues.

- Queue(s) deleted:
  queue-name, queue-name...

(Response) Your BATGEN DELETE command successfully marked the named queues for deletion.

- Queue definition modified, ok to quit?

(Query) A QUIT subcommand has been issued while in BATGEN subcommand mode, and the characteristics of the queue being added or modified have been changed. Valid answers to this question are YES (or carriage return), NO, and OK.

- Queue does not exist.

(Fatal) The -QUEUE option on the JOB command line or the (optional) $$ JOB line referred to a queue that either did not exist or was in the process of being deleted (flagged for deletion). The BATGEN -STATUS command provides a list of currently available queues and the status of each queue.

- Queue files not correct rev.

(Severe) The queue control files in the Batch database are not up-to-date with the current PRIMOS release. Reread the section of Chapter 2 which deals with installing Batch, and run the INIT program as described.

- Queue full.

(Fatal) The queue to which the user has tried to submit the job already contains 10,000 jobs. This should never happen, since jobs are removed from queues as soon as they complete. If you really do have 10,000 jobs simultaneously in one queue, you should set up another queue and encourage users to submit jobs there.

- Queue status is already <status>.

(Response) You have tried to change a queue to its present state. No changes were made.

- Queue name and window times required. (BATGEN)

(Response, Fatal) Both the queue name and the active window times must be specified with the BATGEN ACTIVE_WINDOW command. Re-issue the command.

- Queue(s) unblocked:
  queue-name, queue-name...

(Response) The queues listed were unblocked successfully.

- Queue(s) uncapped:
  queue-name, queue-name ...

(Response) The queues listed were uncapped successfully.

- Register setting.

(Fatal) Register settings are invalid in the Batch subsystem, except as part of a submitted file. Reenter the command line without the register setting. (A register setting is two octal numbers separated by a slash, such as 1/15.)

- Removed <queue-name> from BATDEF

(Supervisor, Response) This message is sent to the supervisor terminal when the Batch monitor finds, in the BATDEF file, a queue that is flagged for deletion but that has never been used. The message indicates that queue queue-name has been deleted from BATDEF, and no job data has been lost as a result.

- Specified value is out of range.

(Fatal) The -CPTIME or -ETIME option specified during job submission or a -CHANGE operation is greater than the maximum allowed by the queue to which the job was submitted. This message is preceded by a message indicating the maximum limit for that queue ("Cpu limit is n" or "Elapsed time limit is n"). If the limits cannot be lowered so that the job can be successfully run, try a queue with higher limits.

- Stop request issued.

(Response) The BATCH -STOP command has requested that the Batch monitor stop. Within 20 seconds the monitor should send an "Operator Stop." message to the supervisor terminal and log out.

- Syntax error. Register settings are illegal

(Warning) A register setting (such as 1/15) has been found when no more information was expected. Reenter the command without register settings.

- This job cannot be restarted.

(Response) This message is displayed by a JOB -DISPLAY command if the job being displayed cannot be restarted. A job is not restartable if a JOB -CANCEL command is issued for that job while it is executing, or if it is submitted with the -RESTART NO option. An attempt to restart the job aborts it without restarting it.

- (This job has already executed <n> time(s).)

(Response) This message is displayed by a JOB -DISPLAY command if the job being displayed is executing and has already been executed (n times). The condition is caused by a JOB -RESTART or by a system cold start after shutdown while the job is executing.

- This job will be restarted.

(Supervisor, Response) This message can be displayed under two circumstances. It will be displayed in response to a JOB -DISPLAY command if a JOB -RESTART has been done but the job is still executing. When the monitor sees that the job has aborted or completed, it returns the job to the waiting state.

This message can also be displayed when Batch is first started. The message is sent to the supervisor terminal after a "Job jobname for username(jobid) aborted/completed" message is sent. It means that the job is eligible for restarting, and is therefore being reset to the waiting state. This second case usually means that the job was terminated prematurely by a system shutdown.

- Too many options.

(Fatal) At least two conflicting options are entered, such as JOB -DISPLAY -CHANGE or JOB TEST -ABORT -CANCEL. Use separate JOB commands to perform separate operations.

- Too many queues.

(Warning) An attempt has been made, using the ADD command in BATGEN, to add a queue when there are already 16 defined queues (blocked, unblocked, capped, uncapped, or flagged for deletion).

- Unable to validate project. project-id (JOB)
  Please contact System Administrator. (JOB)

(Fatal) There may be some problem accessing the SAD. Contact the System Administrator.

- Unit in use. Checking database status. (CHEK$B)

A file which Batch is using is temporarily in use. If this message appears only once or twice, ignore it. If it appears persistently in reply to many Batch commands, a Batch Administrator should attach to the directory BATCHQ, delete the file SEMFIL, and then run INIT. All jobs which are currently in the Batch queues will be lost, but the Batch system should return to full operation. If this does not work, the next cold start should cure the problem.

- Unknown command.

(Warning) An unrecognized command has been entered while in BATGEN command mode. The user is left in BATGEN command mode.

- Unknown option. -option (JOB)

```
Usage: JOB treename [-batch_options] [-ARGS cpl_args]
                    -HOme homedirectory
                    -QUEue queuename
                    -CPTime cpulimit
                    -ETime elapsedlimit
                    -DeFer date.time/-No_DeFer
                    -COMOutput comopathname/-No_COMOutput
                    -NotiFY/-No_NotiFY
                    -PROJect projid
          jobid -CHanGe [-batch_options] [-ARGS cpl_args]
                -CANcel
                -ABort
                -ReSTart
                -STatus
                -DisPlay
```

(Fatal) An option used with the BATCH or JOB command is not a recognized option.

- Unknown queue name.

(Warning) A command entered while in BATGEN command mode refers to a queue that either does not exist or is flagged for deletion by the DELETE command.

- Unknown -STARTUP argument. <keyword> (FIXBAT)

(Supervisor, Fatal) The keyword supplied to the -STARTUP option is not SAVE, DELETE, SPOOL, or NOLOG.

- Unknown subcommand.

(Warning) While in BATGEN subcommand mode, an unrecognized subcommand has been given. The user is left in subcommand mode.

- Unrecognized command line option.

(Supervisor, Fatal) The Batch monitor has been invoked with an unrecognized option on the command line. The only valid MONITOR option is -HUSH. Fix the file START_BATCH_MONITOR.COMI in the BATCHQ directory accordingly, and restart the Batch monitor using BATCH -START.

- Unrecognized option.

(Fatal) BATGEN has been invoked with an unrecognized option on the command line. The only valid BATGEN options are -STATUS and -DISPLAY.

- Warning: -DEFER option time already passed.
  Job is now waiting (not deferred). (JOB)

(Warning, Response) A job has been submitted with a defer time that is earlier than the current time. The job will not be deferred.

- Warning: jobs are not being processed at this time.

(Severe, Warning) If followed by "Please contact your system administrator immediately", this message indicates that the Batch database has not been initialized or that something (such as a disk head crash) has damaged the database. If "Please contact..." appears, the operation the user attempted was not performed.

If the "Please contact..." message does not appear, this message means that the Batch monitor is not currently processing jobs; it is either paused or stopped. Any submitted jobs will not be executed until Batch is started up. At that time, the requested operation(s) will be performed. In either case, the user is immediately returned to PRIMOS command level. This message may be displayed by the BATCH command or by the JOB command.

- Window format is HH:MM-HH:MM. (BATGEN)

(Response, Fatal) The active window specified could not be set because of an incorrect time format. HH can not be more than 23 and MM can not be more than 59.

- Your job, <job-id>, was submitted to queue queue-name.

(Response) This is an informational response to a successful invocation of the JOB command. This message can be followed by: (capped), (inactive), or (capped, inactive).

FIXBAT MESSAGES

The following messages are displayed by FIXBAT, but not sent to the supervisor terminal.

- <filename> leftover words=<n>

(Warning) The queue file named filename had n words at the end of it, not enough for a full queue entry. This is not a fatal error. The queue file is truncated. The message could indicate that a process submitting a job has been forcibly logged out while creating a new queue entry.

- Can't process batch jobs from system console. (FIXBAT)

(Fatal) This problem has two possible causes. Someone has either tried to run FIXBAT with the -STARTUP option interactively from the supervisor terminal, or has tried to run FIXBAT as a phantom logged in as SYSTEM. FIXBAT with the -STARTUP option must be run as a phantom. The phantom must be started by issuing the BATCH -START command from the supervisor terminal.

- Deleted <filename>.

(Response) This message means that FIXBAT found and deleted a temporary (T$nnnn) file, an inactive job file (Cqnnnn), or a queue file (QCTRq) in which all entries have been deleted because of the -DAYS option.

- Execute/data username mismatch (reinitialize). <job-id>

(Severe) FIXBAT found a job that is supposedly executing, but the corresponding job ID in the EXECUT file is owned by a different user. Run BATCHQ>INIT to reinitialize.

- FIXBAT finished.

(Response) The process of fixing the Batch database has been successfully completed. FIXBAT now exits to PRIMOS.

- Fixing database.

(Response) This message indicates that FIXBAT has decided to fix the entire BATCHQ database.

C
Converting Pre-Rev. 20.0 Systems

Because pre-Rev. 20.0 BATDEF files are incompatible with Rev. 20.0 and later Batch, you will need to create a new BATDEF file the first time you initialize Batch on your system. However, you may want to retain the same queue configurations you had previously. To do this, take the following three steps:

1. Before you install Rev. 21.0 Batch or Rev. 21.0 PRIMOS on your system, make a COMO file of the pre-Rev. 20.0 BATDEF file using the following commands:

```
ATTACH BATCHQ
COMO OLD_BATDEF.COMO
BATGEN -DISPLAY
COMO -END
```

   Now print out a copy of OLD_BATDEF.COMO, and save it for use in recreating your queues.

2. After Rev. 21.0 is installed, make yourself a Batch Administrator and RESUME BATCHQ>INIT. This will automatically create new queue definition files in Rev. 21.0 format, overwriting your old queues.

3. Modify your new BATDEF file with the command:

```
BATGEN BATCHQ>BATDEF
```

Using the file you created in step 1 as a guide, recreate your Batch queues.

INDEX

Index

SURVEY

READER RESPONSE FORM

Operator's Guide to the Batch Subsystem DOC9302-3LA

Your feedback will help us continue to improve the quality, accuracy, and organization of our publications.

1. How do you rate this document for overall usefulness?

   ☐ *excellent* ☐ *very good* ☐ *good* ☐ *fair* ☐ *poor*

2. What features of this manual did you find most useful?

   _____
   _____
   _____
   _____
   _____
   _____

3. What faults or errors in this manual gave you problems?

   _____
   _____
   _____
   _____
   _____
   _____

4. How does this manual compare to equivalent manuals produced by other computer companies?

   ☐ *Much better* ☐ *Slightly better* ☐ *About the same*
   ☐ *Much worse* ☐ *Slightly worse* ☐ *Can't judge*

5. Which other companies' manuals have you read?

   _____
   _____

Name:_____ Position:_____

Company:_____

Address:_____

_____

_____ Postal Code:_____

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

First Class Permit #531 Natick, Massachusetts 01760

BUSINESS REPLY MAIL

Postage will be paid by:

Prime™

Attention: Technical Publications
Bldg 10
Prime Park, Natick, Ma. 01760