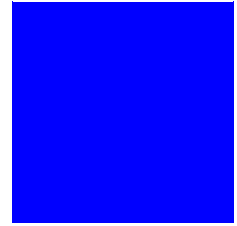


CIMS Lab, Inc.



CIMS Chargeback

NT/UNIX Reference Guide

Version 4.1.1

Title and Publication Number

CIMS Lab Publication Number: NTU-REF-411-01

Printed: April, 2000

Information in this guide is subject to change without notice and does not constitute a commitment on the part of CIMS Lab, Inc. It is supplied on an "as is" basis without any warranty of any kind, either explicit or implied. Information may be changed or updated in this guide at any time.

Copyright Information

CIMS is ©copyright 1974 - 2000 by CIMS Lab, Inc. and its subsidiaries. This guide is ©copyright 1974 - 2000 by CIMS Lab, Inc., and its subsidiaries and may not be reproduced in whole or in part, by any means, without the written permission of CIMS Lab, Inc. and its subsidiaries.

Names marked ™ or ® and other company and product names may be trademarks or registered trademarks of their respective vendors or organizations.

Mailing Address

CIMS Lab, Inc.
3013 Douglas Blvd., Suite 120
Roseville, CA 95661-3842

Table of Contents



Preface

Philosophy	ix
Contacting the CIMS Lab	x
About This Guide	x
Conventions	xiii
Related Publications	xiv

1 • Introduction

2 • CIMS for NT/UNIX Accounting

System Accounting	2-2
UNIX System Accounting	2-2
CIMS for NT/UNIX System Accounting	2-5
CIMS for NT Accounting	2-8
Database System Accounting	2-9
Batch Accounting	2-11
User Accounting	2-13
Project Accounting	2-15
Implementing Project Accounting	2-17

3 • CIMS for NT/UNIX Directory Structure

Distribution Kit File Descriptions	3-2
CIMS for UNIX Accounting Directory	3-4
CIMS for NT/UNIX Bin Directory	3-4
CIMS for NT/UNIX Data Directory	3-5
CIMS for NT/UNIX Description Directory	3-10
CIMS for NT/UNIX Etc Directory	3-10
CIMS for UNIX Examples Directory	3-14
CIMS for NT/UNIX Help Directory	3-14

CIMS for NT/UNIX History Directory 3-15
CIMS for NT/UNIX Log Directory 3-15
CIMS for UNIX Scripts Directory 3-15

4 • CIMS for NT/UNIX Utilities

Introduction 4-3
CIMS for NT/UNIX Authorize Utility 4-3
CIMS for NT/UNIX Format Utility 4-3
CIMS for UNIX FS_Report Utility 4-3
CIMS for UNIX Graph Utility 4-4
CIMS for UNIX GUI 4-4
CIMS for NT/UNIX Login Utility 4-4
CIMS for NT/UNIX Merge Utility 4-4
CIMS for NT/UNIX Rates Utility 4-6
CIMS for NT/UNIX Report Utility 4-6
CIMS for NT/UNIX Sampler Utility 4-6
CIMS for NT/UNIX Select Utility 4-7
CIMS for NT/UNIX Setup Utility 4-11
CIMS for NT/UNIX Super Merge Utility 4-12
CIMS for NT/UNIX Switch Utility 4-16
CIMS for UNIX X-window Login Utility 4-16
CIMS for OpenVMS Utilities 4-16
Using gen_cims_sum, NT_gen_cims_sum, gen_cims_detail, and
NT_gen_cims_detail 4-16

5 • The CIMS for NT/UNIX Files

CIMS for UNIX Configuration Files 5-3
CIMS for NT/UNIX Configuration Parameter Files 5-3
CIMS for NT/UNIX Node Parameter File 5-3
CIMS for NT/UNIX Storage Parameter File 5-4
CIMS for NT/UNIX Parameter File 5-4
CIMS for NT/UNIX Accounting Files 5-5
CIMS for UNIX Activity File 5-5
CIMS for NT/UNIX Authorization File 5-5
CIMS for UNIX Collapsible Terminal File 5-6

CIMS for NT/UNIX Interim Statistics Files5-6

CIMS for NT/UNIX Holiday File5-7

CIMS for NT/UNIX Image Mapping File5-7

CIMS for NT/UNIX Intermediate Statistics File5-8

CIMS for UNIX Login Projects File5-8

CIMS for NT/UNIX Merged Statistics File5-8

CIMS for NT/UNIX Oracle Accounting File5-8

CIMS for NT/UNIX Oracle Basename File5-8

CIMS for NT/UNIX Oracle Username File5-9

CIMS for UNIX Print Parameter File5-9

CIMS for UNIX Queue Mapping File5-9

CIMS for NT/UNIX Rates File5-9

CIMS for NT/UNIX Shift File 5-10

CIMS for NT/UNIX Storage File 5-11

CIMS for UNIX Terminal Parameter File 5-11

CIMS for NT/UNIX Validation File 5-11

CIMS for UNIX X-Application Resource Files 5-11

CIMS for UNIX Report Description Files 5-12

 Select Section 5-12

 Sort Section 5-18

 Body Section 5-19

 Report Description File Considerations 5-27

 Report Description File Examples 5-28

 CIMS for UNIX Graph Description File 5-31

6 • CIMS for NT/UNIX Command Line Interface (CLI) and Commands

CIMS for NT/UNIX Command Line Interface (CLI)6-4

Licensing6-5

User Accounting6-5

 CIMS for NT/UNIX DEFAULT User Record6-5

 Enabling User Accounting6-6

 Adding User Accounts6-6

 User Shell6-7

 User Cost Center6-7

 User Groups6-8

- User Description 6-9
- Project Accounting 6-9
 - CIMS for NT/UNIX DEFAULT Project Record 6-9
 - Enabling Project Accounting 6-10
 - Adding Project Accounts 6-11
 - User Default Project 6-11
 - User Default Jobtype Projects 6-12
 - Custom Project Prompting 6-12
 - Project Switching 6-14
 - Project GID 6-14
 - Project Default Directory 6-15
 - Project Cost Center 6-16
 - Project Groups 6-16
 - Project Description 6-17
 - Project Time Limit 6-18
 - Project Validation 6-18
- Image Accounting 6-22
 - Enabling Proprietary Software 6-22
 - Defining Software Packages 6-23
- Database Accounting 6-23
- Reporting Parameters 6-25
 - Capacity Interval 6-26
 - Rates 6-27
 - Shifts 6-29
 - Holidays 6-30
 - Output Queue 6-31
 - GID Method 6-32
 - Report Header 6-32
 - Digit Separator 6-33
 - Memory Demand Algorithm 6-33
 - Monetary Symbols 6-34
 - Radix Point 6-34
- Data Formatting 6-35
 - CIMS for UNIX Print Accounting Script 6-35
 - CIMS for NT/UNIX Runacct Script 6-36

Processing Multitude Nodes	6-39
7 • CIMS for NT/UNIX Advanced Features	
Advanced Features	7-2
Advanced CIMS for NT/UNIX Operational Scripts	7-2
Data Window	7-4
CIMS for NT/UNIX Disk Space Management	7-5
Out-of-Sequence Management	7-7
Performance Management	7-7
CIMS for UNIX GID Method Implementation	7-8
8 • CIMS for NT/UNIX Messages	
Introduction to CIMS for NT/UNIX Messages	8-3
Format of CIMS for NT/UNIX Messages	8-4
Example Error Message	8-6
Returned Error Status	8-7
CIMS for NT/UNIX RTL Messages	8-7
GEJAC RTL Messages	8-16
CIMS for UNIX DATABASE RTL Messages	8-18
CIMS for UNIX LOGIN Utility Messages	8-22
CIMS for UNIX X-Window LOGIN Utility Messages	8-25
CIMS for UNIX SWITCH Utility Messages	8-27
CIMS for NT/UNIX SAMPLER Utility Messages	8-30
CIMS for NT/UNIX CONVERT Utility Messages	8-32
CIMS for UNIX Oracle Accounting Daemon Messages	8-35
CIMS for UNIX Oracle Project LOGIN Utility Messages	8-36
CIMS for UNIX Oracle Project SWITCH Utility Messages	8-37
CIMS for UNIX IBM DB2 Accounting Daemon Messages	8-38
CIMS for UNIX GUI Messages	8-40
Diagnostics	8-40
Error Messages	8-47
CIMS for UNIX GRAPH Utility Messages	8-48
Diagnostics	8-48
Error Messages	8-52
CIMS for NT LOGIN Utility Messages	8-53

A • The CIMS for NT/UNIX Statistics

UNIX A-2
Oracle A-6
DB2 A-9

B • CIMS for UNIX GUI Files

Resource File B-2
 System Wide B-2
 Environment Variable B-2
 Local Copy B-3
 Colors B-3
 Fonts B-6
 Datatypes B-7
 Line Pattern B-7
 Fill Pattern B-8
 Color B-8
 Line Width B-8
 Point Style B-8
 Point Color B-8
 Point Size B-9
X Resource Files B-9

Index



Preface

As companies continue to integrate computer technology into their business operations, it becomes increasingly important to properly administer the IT function, particularly with respect to performance and cost. And the best way to control costs is to plan for them.

CIMS Chargeback is a comprehensive, flexible software solution that consolidates a wide variety of accounting data for multiple operating systems into a single file that may be accessed from either the mainframe or a workstation. Simply put, CIMS Chargeback is an essential component of an effective financial management system.

The *CIMS for NT/UNIX Reference Guide* provides information on the actions an advanced user would take to use the product as well as command-line instructions. For less advanced users, please consult the *CIMS for NT/UNIX User Guide*.

Philosophy

Originally developed in 1974, CIMS has focused on meeting the financial and resource reporting requirements of Information Services Departments. CIMS has evolved with corporate IT management requirements. Focused commitment to client service and support sets

CIMS apart from competing products. Our goal is to provide the best chargeback and resource reporting software in the world at the lowest possible cost to our customers.

The CIMS Lab strongly believes in and executes the concept of continuous product improvement. Customers have access to CIMS product development personnel to ensure that customer feedback and other critical issues are incorporated into the next release of the product.

Contacting the CIMS Lab

You can contact us with any questions or problems you have. Please use one of the methods below to contact us.

For product assistance or information, contact:

USA & Canada, toll free	(800) 283-4267
International	(916) 783-8525
FAX	(916) 783-2090
World Wide Web	www.cimslab.com

Our Mailing Address is:

CIMS Lab, Inc.
3013 Douglas Blvd., Suite 120
Roseville, CA 95661-3842

About This Guide

The *CIMS for NT/UNIX Reference Guide* provides information about features and capabilities of *CIMS for NT/UNIX*. *CIMS for NT/UNIX* was formerly known as ARSAP for NT and UNIX. There are therefore references to ARSAP throughout this guide. Users of this guide are assumed to be experienced with *CIMS for NT/UNIX*. Less experienced users should begin with the *CIMS for NT/UNIX User Guide*.

This guide assumes that the appropriate *CIMS for NT/UNIX* components have been installed at your site. The instructions for installing the product are in the *CIMS Chargeback NT/UNIX Installation and Getting Started Guide*.

Ch. No.	Chapter Name	Content Description
1	<i>Introduction</i>	
2	<i>CIMS for NT/UNIX Accounting</i>	Discusses system accounting and what resource accounting data CIMS collects from operating system accounting information. Discusses the purpose of user accounting. Discusses the project accounting feature of CIMS for NT/UNIX and how to implement project accounting. Discusses Database Accounting and what information is available from Oracle and DB2.
3	<i>CIMS for NT/UNIX Directory Structure</i>	Describes the directory layout of <i>CIMS for NT/UNIX</i> . Discusses the various data, configuration and script files contained in the directories.
4	<i>CIMS for NT/UNIX Utilities</i>	Discusses the function and features of all the utilities that make up <i>CIMS for NT/UNIX</i> .
5	<i>The CIMS for NT/UNIX Files</i>	Describes each CIMS for NT/UNIX file.
6	<i>CIMS for NT/UNIX Command Line Interface (CLI) and Commands</i>	Introduces the CIMS for NT/UNIX Command Line Interface.

Ch. No.	Chapter Name	Content Description
7	<i>CIMS for NT/UNIX Advanced Features</i>	Presents an overview of advanced CIMS for NT/UNIX features including operational scripts, the data reporting window, space management, out-of-sequence management, performance management, and GID method implementation.
8	<i>CIMS for NT/UNIX Messages</i>	Provides an explanation and user action for CIMS for NT/UNIX Messages.
9	<i>Client/Server Environment</i>	Describes files and configuration settings used by CIMS for NT/UNIX in client/server environments.
A	<i>The CIMS for NT/UNIX Statistics</i>	Describes the statistics provided by CIMS for NT/UNIX.
B	<i>CIMS for UNIX GUI Files</i>	Describes X and Motif Resources used by the CIMS for UNIX GUI and Graph Utilities.
	<i>Index</i>	Helps you locate information within this manual.

Conventions

Some or all of the following conventions appear in this guide:

Symbol or Type Style	Represents	Example
Bold	a new term	...called a source object .
<i>Alternate color</i>	(online only) hotlinked cross-references to other sections in this guide; if you are viewing this guide online in PDF format, you can click the cross-reference to jump directly to its location	...see <i>Chapter 3, Data Migration</i> .
<i>Italic</i>	words that are emphasized	...the entry <i>after</i> the current entry...
	the titles of other documents	<i>CIMS for NT/UNIX User Guide</i>
	syntax variables	<i>COPY filename</i>
Monospace	directories, file names, command names, computer code	&HIGHLVL.SRCLIB
	computer screen text, system responses, command line commands	Copy file? Y/N
Monospace bold	what a user types	...enter RUN APP.EXE in the Application field
< >	the name of a key on the keyboard	Press <Enter>.

■ Preface

Related Publications

Symbol or Type Style	Represents	Example
▶	choosing a command from a cascading menu	File ▶ Import ▶ Object
Highlighted Screen Text	used to callout screen text on character- based screen captures. (When viewed online, the screen text will be blue.)	Dataset.... Product.... Parmlib...

Related Publications

- *CIMS for NT/UNIX Installation and Getting Started Guide*
- *CIMS for NT/UNIX User Guide*



Introduction

CIMS for NT/UNIX is a resource management and chargeback facility that runs on multiple UNIX platforms and NT servers and workstations. *CIMS for NT/UNIX* collect, process, organize, and report UNIX system accounting information and NT event information, respectively, along with other data generated by *CIMS for NT/UNIX*. *CIMS for NT/UNIX* utilities and scripts provide the flexibility to customize your set-up according to your needs and environment, collect necessary data, consolidate that data, and extract the data in meaningful, custom reports.

CIMS for NT/UNIX Accounting

System Accounting	2-2
UNIX System Accounting	2-2
CIMS for NT/UNIX System Accounting	2-5
CIMS for NT Accounting	2-8
Database System Accounting	2-9
Batch Accounting	2-11
User Accounting	2-13
Project Accounting	2-15
Implementing Project Accounting	2-17

System Accounting

System accounting establishes information about the usage of a computer system. System accounting information includes data such as how many times a user logs into the system, how long they are logged in, how much CPU time they use, how much disk space they are using, and system reboots and shutdowns. This data is provided by the operating system but the information is unsuitable for precise and conclusive resource management and chargeback. In addition, the information pertains to each individual system rather than the entire enterprise.

CIMS for NT/UNIX collects and processes system accounting from various sources and yields information in a consolidated, centralized repository. CIMS for NT/UNIX utilizes system accounting data from the following sources:

- UNIX Accounting
- CIMS for NT/UNIX
- NT Event Logs
- Relational Databases (for example, Oracle and IBM DB2)
- Batch Processing (for example, NQS and Convex)

UNIX System Accounting

UNIX collects resource utilization information and records this data in the following UNIX accounting files:

- Wtmp and Wtmpx (optional) Files
- Acct File
- Print Accounting Files

The UNIX kernel manages processes on the computer system from logins to shutdowns. UNIX logs user logins and logouts in the UNIX `Wtmp` and `Wtmpx` (optional) Files along with records of shutdowns, clock adjustments, and other system information. All these processes consume system resources, such as CPU time, memory, and I/O. UNIX records this process usage in the UNIX `Acct` File. UNIX captures print usage in separate files for each printer. Collectively, these files comprise the resource usage information provided for user accounting.

UNIX `Wtmp` and `Wtmpx` Files

The name of the UNIX `Wtmp` File is `wtmp`. The `wtmp` file holds information about logins, logouts, system shutdowns, system reboots, and system date changes. Normally, the UNIX kernel collects this information, provided the `wtmp` file exists; this file is located in the `/etc` directory on some systems and in `/usr/adm` on other systems.

CIMS for UNIX primarily utilizes user login and logout data. However, the other information is processed. The UNIX kernel stores the following information about each login and logout:

- Username of the user that logged in
- Controlling terminal, when there is one
- Name of the remote system, when there is one
- Time logged in or logged out

Note • Most UNIX systems do not store the usernames in the logout record, only in the login record.

The UNIX `Wtmpx` File is an extension of `wtmp`. The name of the UNIX `Wtmpx` File is `wtmpx` and it is located in the same directory as `wtmp`. UNIX maintains the same information in `wtmpx` and `wtmp` because `wtmpx` is a back-up of `wtmp`. Therefore, should anything happen to `wtmp`, UNIX uses `wtmpx` to restore `wtmp`. CIMS for UNIX does not process `wtmpx` but needs to make sure the `wtmp` and `wtmpx` files stay in synch.

UNIX Acct File

The name of the UNIX Acct File is either `pacct` (most common) or `acct`. The `pacct` file is located in the `/usr/adm` or `/var/adm` directory. The `pacct` file maintains resource information about most processes that are executed on a UNIX system; the UNIX kernel does not store in `pacct` some commands that are handled directly by the shell. In order for UNIX accounting data to be collected, the UNIX accounting subsystem needs to be installed on your system and UNIX accounting must be turned on.

Each time a command is executed, (from simple processes like `ls` to a large application program), the UNIX kernel adds another record to `pacct`. These commands can be processes attached to a controlling terminal (interactive) or processes not attached to a controlling terminal (non-interactive or background). This file can grow rapidly, especially on systems with many users or a heavy workload since an accounting record is logged for each process that is executed.

The UNIX kernel stores the following information about each completed process:

- Command name
- UID and GID of the user that executed the command
- Controlling terminal (when there is one)
- Amount of user and system CPU time that it took to execute that command
- Time the command started executing
- Time that it took to execute the command (the *elapsed* time)
- Average memory usage
- Number of blocks read or written
- On some systems, the number of characters transferred

UNIX Print Accounting Files

The UNIX Print Accounting File contains information about print jobs. Typically, a print accounting file is maintained for each printer. The UNIX kernel stores the following information about each print job:

- Name of the user that printed the job
- Number of pages that were printed

Additional print accounting data might be available. CIMS for UNIX uses only this information that is consistent across the UNIX platforms that provide print accounting data.

Print accounting under UNIX is very limited: very few UNIX systems provide the mechanism to capture print accounting data and the collected data is not very comprehensive.

Normally, this information is not collected on a UNIX system. On some systems, print accounting must be enabled in the printer configuration file (for example, `/etc/printcap`). On other systems, print accounting is enabled through the printer administrative tool. Refer to *Print Accounting Set-Up* in the *CIMS Chargeback NT/UNIX User Guide*

CIMS for NT/UNIX System Accounting

UNIX and NT do not provide all necessary system accounting information for suitable resource management and chargeback. CIMS for NT/UNIX fills these voids. The areas of deficiency are: project tracking, image/package reporting and disk storage.

UNIX and NT do not provide any facilities for collecting information about projects. CIMS for NT/UNIX lets you designate projects and collects this project information. CIMS for UNIX stores this data in the CIMS for UNIX Project Activity File that resides in the CIMS Data Directory. CIMS for NT project implementation utilizes the NT application event log. *Implementing Project Accounting* on page 2-17 discusses project accounting in detail.

UNIX and NT collect image accounting data, however, they do not have a facility to group image data into one package. Typically, proprietary software consists of more than one executable. When a report on an entire software package is required, UNIX and NT could not provide such a report in a straight forward manner. CIMS for NT/UNIX handles the mapping of all software images for proprietary software into one package.

UNIX and NT let you obtain disk storage data but do not collect this vital accounting information. CIMS for NT/UNIX collects disk space utilization into the CIMS for NT/UNIX Disk Storage File that resides in the CIMS Data Directory.

CIMS for UNIX Project Activity File

The CIMS for UNIX Activity File (`A_activity.sys`) contains project information. CIMS for UNIX adds the following information into `A_activity.sys` when a user enters a project manually or automatically:

- UID of the user
- GID of the group/project
- Time the user logged in or switched projects
- New project name
- Terminal name

Implementing Project Accounting on page 2-17 discusses CIMS for UNIX project accounting.

CIMS for NT/UNIX Image Accounting

CIMS for NT/UNIX image accounting lets you track and report usage of important software packages. For example, you might need to analyze the usage of a proprietary software product that consists of numerous programs. UNIX and NT report on individual images and cannot group these images into a single package. CIMS for NT/UNIX image accounting groups any set of software images into one package. Furthermore, CIMS for NT/UNIX tracks and reports the usage of this package to users and projects.

CIMS for NT/UNIX image accounting utilizes the CIMS for NT/UNIX Image Mapping File (`A_image.sys`). `A_image.sys` facilitates the mapping of individual images into a single software package. A software package can consist of any subset of a proprietary software suite or a logical grouping of system commands.

CIMS for NT/UNIX Storage File

The CIMS for NT/UNIX Storage File (`A_storage.sys`) contains information about disk usage on a UNIX system. The CIMS for UNIX Sampler Utility (`A_sampler`) and the CIMS for NT Sampler Utility (`NT_sampler.exe`) collect and store the following information in `A_storage.sys`:

- UID and username of the user owning storage (No UID on NT)
- GID and group name of the project owning the storage (UNIX only)
- Number of bytes of storage
- Number of files

`A_sampler` adds a record to `A_storage.sys` for each unique username/groupname combination. `A_sampler` also records information about the filesystems.

`A_sampler` samples UNIX, Redundant Array of Independent Disks (RAID), Andrews File System (AFS) and Migration filesystems.

NT_sampler samples File Allocation Table (FAT) file items, New Technology File System (NTFS), and High Performance File Systems (HPFS).

CIMS for NT Accounting

NT provides system accounting information through event logs. CIMS for NT utilizes the security and application event logs. In order for event logs to be populated, the NT EventLog service must be started and the following audit policies for success and failure should be enabled:

- Logon and Logoff
- Restart, Shutdown, and System
- Process Tracking

NT Security Log

The NT security log contains the user logins and logoffs, image begin and ends, and system startups and shutdowns. CIMS for NT processes these records to obtain the necessary system accounting information. The CIMS for NT Image Tracking File (A_imglist.sys) maintains the image begin records for images that have not completed yet.

NT Application Log

The NT application log lets applications record application-specific data. CIMS for NT utilizes the application log to store project accounting information. The CIMS for NT Login Shell (NT_login) and Login Script (NT_login_scr) utilities record the project logins in this log. These utilities are tasks that provide a pull down menu option to perform on-line project switching. These utilities also record project switches in this log.

Implementing Project Accounting on page 2-17 section discusses CIMS for NT project accounting.

Database System Accounting

System accounting information is available from various relational database engines. UNIX does not provide a means to gather this accounting data. Therefore, system resources consumed by users within a relational database are lost. However, *CIMS for NT/UNIX* does collect and process the database accounting data from Oracle and IBM DB2.

CIMS for NT/UNIX provides two additional database accounting features: database user groupings and database absolute shift and daily accounting.

CIMS for NT/UNIX assigns database accounting data to the operating system user and database user. The database user grouping feature provides an additional means of grouping users based on the database username. When this feature is enabled, *CIMS for NT/UNIX* will search its Authorization File records for a user record matching the database username and assign groupings from that record. If this account record does not exist, the UNIX username record is used. User records are described in *User Accounting* on page 2-13.

The database absolute shift and daily accounting feature captures interval database accounting data. When database absolute shift accounting is enabled, *CIMS for NT/UNIX* collects database system accounting data on shift boundaries. When database absolute daily accounting is enabled, *CIMS for NT/UNIX* collects database system accounting data at midnight. This feature is useful for long running database connections.

CIMS for NT and UNIX Oracle Accounting

The Oracle Relational Database Management System (RDBMS) records resource statistics in internal tables when the timed statistics variable is set. You set this variable in the initialization file of an Oracle instance. The Oracle RDBMS has several UNIX daemons/NT processes that perform the work on behalf of database users connecting to an Oracle instance. Therefore, the only resource usage UNIX/NT can gather is the work done by these daemons/processes and cannot assign this work to the individual users. Users access an Oracle server locally or remotely.

Oracle SQL*Net controls remote access through the TNS listener. When accessing an Oracle server from a remote multi-user machine, the Oracle TNS listener daemon on some platforms lumps all users under the operating system username of the listener process. This situation has been corrected when using SQL*Net V2.3 or higher. When users are coming in from PCs or MacIntoshes, you can designate a unique username by using the `USERNAME` parameter in the SQL*Net configuration file.

CIMS for NT and UNIX periodically samples the internal tables of an Oracle instance to attribute the resource usage to Oracle users. In addition, CIMS for NT and UNIX project accounting is extended, letting you track this information under projects (refer to *Implementing Project Accounting* on page 2-17). The CIMS for NT and UNIX Oracle Accounting File (`A_dbacct.sys`) contains resource usage information for users accessing Oracle databases. The CIMS for NT and UNIX Oracle Accounting Daemon (`A_dbao`) collects and stores the following information in `A_dbacct.sys`:

- UID and username of the user (UNIX only)
- GID and group of the user (UNIX only)
- Oracle username used to access the database
- Oracle database name
- Statistics reflecting the resources used

`A_dbao` adds a record to `A_dbacct.sys` each time it detects a user has disconnected from a database instance. When the CIMS for NT and UNIX Oracle project accounting is implemented, `A_dbao` gathers the project data and associates the proper project with the resource accounting data.

CIMS for NT and UNIX IBM DB2 Accounting

IBM DB2 provides system monitors to obtain database accounting data. *CIMS for NT/UNIX* retrieves this system monitor data to record resource usage for a user. *CIMS for NT/UNIX* supports IBM DB2 Common Server (cs) and Universal Data Base (UDB) V5. CIMS for UNIX also supports PE (Parallel Edition). The *CIMS for NT/UNIX* DB2 Accounting File (A_db2acct.sys) contains resource usage information for users accessing DB2 databases. The *CIMS for NT/UNIX* DB2 Accounting Daemon (A_dbadb2) collects and stores the following information in A_db2acct.sys:

- UID and username of the user (UNIX only)
- GID and group of the user (UNIX only)
- DB2 authorization ID used to access the database
- DB2 database name
- DB2 application ID
- Statistics reflecting the resources used

A_dbadb2 adds a record to A_db2acct.sys each time it detects a user has finished a DB2 application.

Batch Accounting

Batch processing is not a standard feature under UNIX. However, batch processing lets you conveniently submit long running, repetitive jobs. CIMS for UNIX processes batch system accounting data that is currently available under UNIX. Network Queuing System (NQS) is a batch processing product that is available under most UNIX platforms. CIMS for UNIX supports Monsanto's Release 3.3.6 or higher. In addition, batch processing is available under ConvexOS through the Convex CXbatch system.

NQS Batch Accounting File

The NQS Batch Accounting File stores information about NQS batch jobs. This file is named `nqs` and is located in `/usr/adm` or `/var/adm`. Once NQS accounting has been enabled, the NQS batch system writes a batch accounting record in the same NQS Batch Accounting File for each job that terminates on the NQS queues. The batch accounting record contains the following information about each batch job:

- Batch queue on which the job ran
- Host where the job ran
- Time the job was submitted
- Time the job completed
- UID and GID of the user that ran the job
- Activity ID
- Job sequence number
- Host where the job originated
- Resources used by the job
- Priorities of the job

Convex Batch Accounting File

The Convex Batch Accounting File stores information about Convex CXbatch jobs. On most Convex platforms, this file is given the name `batch-acct` and is located in `/usr/adm`. Once CXbatch accounting has been enabled, the ConvexOS kernel writes a batch accounting record in the same Convex Batch Accounting File for each job that terminates on the CXbatch queues. The batch accounting record contains the following information about each batch job:

- Batch queue on which the job ran
- Host where the job ran
- Time the job was submitted
- Time the job completed
- UID and GID of the user that ran the job
- Activity ID
- Job sequence number
- Host where the job originated
- Resources used by the job
- Priorities of the job

User Accounting

UNIX, NT, and databases do not provide easy reporting on resource usage for the user account as a whole. Combining all types of usage for individual accounts can be a very time consuming proposition. In addition, UNIX and databases do not provide a method for assigning portions of a user's resource usage to projects or defining sets of users to be tracked as a group.

The CIMS for NT/UNIX user accounting feature solves these problems. The CIMS for NT/UNIX implementation extends UNIX, NT, and database user accounting by collecting additional information and providing control and reporting features by user. The CIMS for NT/UNIX user accounting features lets you define the following information for each user:

- Implement project accounting (discussed in *Implementing Project Accounting* on page 2-17)
- Define groups of users
- Designate the shell environment (UNIX only)
- Describe the user
- Specify charge rates

CIMS for NT/UNIX groups users through grouping fields or a cost center. The grouping fields can be locational (country, state, and so forth), or hierarchical (for example, department, agency, division, etc.). The grouping fields can be used to produce multi-tiered user group reports. These groupings are statically assigned to each username.

The user environment shell pertains to CIMS for UNIX only. The user shell defines the UNIX shell program that CIMS for UNIX invokes on behalf of the user. This shell must be a valid UNIX shell. When an invalid shell is provided, CIMS for UNIX defaults to the Bourne shell (`/bin/sh`).

A user description can be designated in *CIMS for NT/UNIX* to bypass potentially cryptic usernames. A user description usually provides a narrative or comment describing the user account.

User accounts can be assigned a rate table. Rate tables are used to define the charges that apply to resource usage statistics for chargeback purposes or as a means of factoring usage metrics from platforms of varying computing power. User accounts can be assigned differing rate tables in order to assign different charges to individual user accounts.

Project Accounting

Managing computer resources or charging for resource usage requires more than just assigning an invoicing or reporting scheme to individual user accounts. Tracking computer resource usage and charge information to a special account accessible to multiple users is often necessary. The UNIX and NT accounting systems do not provide efficient or convenient computer usage costs tracking.

The *CIMS for NT/UNIX* project accounting feature lets you track computer resource usage to a project that can be accessed by multiple users. A project can be defined in any manner, such as: a task, a charge number, a contract or sales order, a customer, or any other reference that follows your site's nomenclature. *CIMS for NT/UNIX* project accounting also lets you impose access control on a project account while maintaining separate user accounts on the computer system.

You can assign projects to all user accounts or just selected users. A default project can be assigned to a user letting you automatically and transparently assign a project to the user's resource usage. This method is useful when a user always begins with the same project, or the *CIMS for NT/UNIX* Administrator wants to conceal the fact that project accounting is being used on a particular user. When a default project is not assigned, *CIMS for NT/UNIX* prompts the user to enter a project when logging in. When a user is prompted for a project, he must enter a valid project name in order to access the system.

The *CIMS for NT/UNIX* project accounting feature lets you switch projects on-line. Once a project has been associated with a session, a new project instance can be started without terminating the current session. This on-line project switching lets you work with multiple projects alleviating numerous time-consuming logins and logouts. The project account validation settings apply to project switching just as they do for initial projects.

When default project is disabled, the user is prompted for a project during login. The user is always prompted for a project when switching projects. *CIMS for NT/UNIX* allows customization of project prompts.

CIMS for NT/UNIX reports on jobtypes. The jobtypes are: interactive, background, batch, storage, Oracle and DB2. Default or fallback projects can be designated for each jobtype. Rates are also based on jobtype.

CIMS for NT/UNIX project and user accounting features are similarly implemented. The *CIMS for NT/UNIX* project accounting features lets you define the following information for each project:

- Define groups of users
- Designate a project GID (UNIX only)
- Designate a project directory
- Describe the project
- Define access control
- Specify jobtypes

CIMS for NT/UNIX allows for grouping projects through grouping fields or a cost center. The grouping fields can be locational (country, state, and so forth), or hierarchical (for example, department, agency, division, etc.). The grouping fields can be used to produce multi-tiered project group reports.

You can associate a project Group ID (GID) with the project by setting the user's real GID to the project's GID.

Note • Changing a GID can have security implications and should be used with care.

This feature can be used to restrict access to files owned by a particular project, and lets CIMS for UNIX assign storage information to a project account.

You can associate a default directory with the project by setting the user's current directory to the project's directory. When that project is used, the user is placed into that directory upon entering the project. This lets all users working on the project store their files in the same directory.

A project description is available to give additional narrative or comments about the project.

CIMS for NT/UNIX allows project account control through project time limits and project validation methods. You can impose a time period on a project to indicate when a project becomes effective and when the project expires. A project can be validated by user account authorization, password verification, customized confirmation, a combination of these three, or project existence.

You can assign a project account to a rate table. This lets you define the charges that apply to resource usage statistics for chargeback purposes. Project accounts can be assigned different rate tables in order to assign differing charges for individual project accounts.

Implementing Project Accounting

Project accounting lets you track resource usage to an entity other than user. Projects provide a dynamic entity as opposed to grouping fields that are static.

In order to use the *CIMS for NT/UNIX* project accounting features, you need to introduce project accounting components into your environment. The project accounting components are the *CIMS for NT/UNIX* utilities that interface with the user to prompt and validate or provide project information. One *CIMS for NT/UNIX* component establishes the project login mechanism for the user to gain access to your system(s) and databases. The introduction of this component means some additional changes to your environment to ensure proper logins as well as logouts. In addition, projects can be associated with batch jobs. Another component provides the user with the ability to switch projects from a login session and database connection.

The *CIMS for NT/UNIX* project components are:

UNIX login—A_login, A_login_xm

UNIX switch—A_switch

NT login and switch—NT_login and NT_login_scr

Convex Batch—A_login

NQS Batch—A_login

Oracle login—A_dbplo

Oracle switch—A_dbpso

Project Accounting for UNIX

The CIMS for UNIX project accounting components are: Login (A_login), X-window Login (A_login_xm) and Switch (A_switch). These CIMS for UNIX utilities write a project record to the CIMS for UNIX Activity File (A_activity.sys) when the user supplies a valid project. The environment variable ARSAP_CURRPROJ is set in accordance with this project.

The CIMS for UNIX X-window Login Utility (A_login_xm) provides the means to associate a project with the entire X-window session. This mechanism alleviates project prompting for all subsequent terminal windows brought up in that session.

These utilities allow the user to query for a list of their authorized projects. When using either the LOGIN or SWITCH Utilities, the user enters a question mark (?) in place of the project name. The list of projects for which the user is authorized is displayed followed by the project prompt.

The implementation for X-window LOGIN utility is slightly different. The A_login_xm data entry window has a **Display Validated Projects** button. A_login_xm displays a list box with the valid projects when this button is selected. The user can either double-click the appropriate project or enter the project name in the project prompt text field box. [Figure 2-1](#) presents the A_login_xm data entry window. [Figure 2-2](#) presents the A_login_xm valid project list window.



Figure 2-1 • A_login_xm Data Entry Window



Figure 2-2 • A_login_xm Valid Project List Window

The set UID bit must be set on A_login, A_login_xm, and A_switch. This is accomplished with the following command:

```
# chmod 4555 A_login* A_switch
```

The above executables must also be owned by root.

CIMS for UNIX Login Utility

The CIMS for UNIX Login Utility (A_login) is similar to but not to be confused with UNIX login. You implement A_login in two ways: the UNIX shell or an exec'ed process.

You use A_login as a UNIX shell by specifying A_login as the user's shell in the UNIX passwd file (/etc/passwd). When you are using Network Information Service (NIS), you need to have CIMS for UNIX on all your machines. In addition, you might need to create symbolic links to ensure the path to A_login is uniform on all machines.

You also need to add A_login to the UNIX shells file, which is /etc/shells on most UNIX platforms, and /etc/security/login.cfg under IBM AIX. When you use DARPA Internet File Transfer Protocol (ftp), you must modify the shells file.

The UNIX rsh command is used to perform non-interactive remote commands. When rsh is invoked, UNIX does not pass the local environment to the remote shell executing the remote command. Due to this UNIX limitation, CIMS for UNIX cannot track these commands to a project.

Alternatively, A_login could be executed from the user's .login, .profile or /etc/profile, depending on the user's UNIX shell. When the user's login shell is csh, you place the following entry in the user's .login:

```
#
# Get Accounting Login data for CIMS for UNIX
#
if ( "$USER" != "root" ) then
    if ( ${?ARSAP_CURRPROJ} = 0 ) then
        exec /usr/ARSAP/bin/A_login
    endif
endif
```

When the user's login shell is either sh or ksh, you place the following entry in the system-wide profile, /etc/profile, or the user's .profile:

```
#
# Get Accounting Login data for CIMS for UNIX
#
if [ ${USER} != "root" ]
then
    : ${ARSAP_CURRPROJ :=`echo "NEW"`}
    if [ $ARSAP_CURRPROJ = "NEW" ]
    then
        exec /usr/arsap/bin/A_login
    fi
fi
```

Note • In the examples above, USER might need to be LOGNAME on some UNIX platforms (for example, HP-UX).

CIMS for UNIX X-window Login Utility

Several X Display Managers are available under UNIX: xdm, gdm (the Gnome Display Manager), and Common Desktop Environment (CDE). These managers perform UNIX login validation and start the user's Xsession through the session manager resource. Under xdm the session manager resource is defined in `xdm-config`. Under CDE, this resource is defined in `Xconfig`. Under gdm, the user's Xsession is started from scripts contained in the gdm SessionDir directory. SessionDir is defined in the gdm configuration file `/etc/X11/gdm/gdm.conf`.

You implement the CIMS for UNIX X-window Login Utility (`A_login_xm`) by specifying `A_login_xm` as the session manager. This in effect interjects the project login between the UNIX login validation and the start of the Xsession. When project accounting is enabled for the user, `A_login_xm` displays the project login window to prompt the user for a project. When a password is required for the project or if the user has entered an invalid project, `A_login_xm` provides the project password window. When the user has entered an invalid project or password, `A_login_xm` exits, and the X Display Manager login box is redisplayed. When the project is validated, `A_login_xm` starts the Xsession script.

The location of Xsession differs between X-window managers and platforms. CIMS for UNIX provides a system-wide parameter to define the location of Xsession. You specify the location of Xsession by setting the Xsession Location text field box on the ARSAP System Defaults data entry window (see Figure 2-4 in the CIMS for NT/UNIX User Guide).

You specify the location of Xsession through the CIMS for UNIX CLI as follows:

```
# A_setup
A_SETUP> DEFAULT/XSESSION_LOCATION="/etc/dt/Xsession"
%SETUP-S-MODIFIED, 1 node record successfully modified
A_SETUP> exit
#
```

To complete the project accounting X-window implementation, you need to call `A_login_xm` with the **logout** argument at the end of the user's session. The placement of this call is dependent on your platform type. In the `Xsession` file, the user's local X-startup (for example, `.xsession` or `.xinitrc`) file is exec'ed. This local file is exec'ed in order to reduce the number of processes. When additional processes are not a concern, remove the `exec` command in the `Xsession` file and add the `A_login_xm` command with the `logout` argument after the call to the user's `.xsession` (or `.xinitrc`). For example, the following is from a typical `Xsession` file:

```
if [ -f $startup ]; then
    exec /bin/sh $startup
else
    if [ -f $resources ]; then
        xrdp -load $resources
    fi
    twm &
    exec xterm -geometry 80x24+10+10 -ls
fi
```

This should be modified as follows:

```
if [ -f $startup ]; then
    /bin/sh $startup
    /usr/arsap/bin/A_login_xm logout
else
    if [ -f $resources ]; then
        xrdp -load $resources
    fi
    twm &
    xterm -geometry 80x24+10+10 -ls
    /usr/arsap/bin/A_login_xm logout
fi
```

When you do not want to remove the `exec` part of the call to the user's `.xsession` in `Xsession`, you must place the call to `A_login_xm` with the `logout` argument at the end of each individual user's `.xsession` file. The example above could appear as the last image called in the user's `.xsession` file.

What follows are some notes on implementing `A_login_xm` on particular platforms. Please review this section on platform considerations before installing `A_login_xm`.

Where to find the Xconfiguration file (`xdm_config`):

CDE -	<code>/usr/dt/config</code>	<code>Xconfig</code>
Sun -	<code>/usr/openwin/lib/xdm</code>	<code>xdm-config</code>
SGI -	<code>/usr/lib/X11/xdm</code>	<code>xdm-config</code>

CDE - on HP-UX, AIX, and Digital UNIX

In CDE environments the `Xconfig` file is located in `/usr/dt/config`. This file needs to have the following modification made in order to call the CIMS for Unix X-window LOGIN Utility during the login process. Look for the line where `Dtlogin*session` is defined:

```
Dtlogin*session:/usr/dt/bin/Xsession
```

Change the definition to call the CIMS for Unix X-window Login Utility.

```
Dtlogin*session:/usr/arsap/bin/A_login_xm
```

Next, you will need to copy the `Xsession` script from `/usr/dt/bin` to `/etc/dt`. This is done for consistency with CDE documentation.

```
cp /usr/dt/bin/Xsession /etc/dt/Xsession
```

Near the end of the `Xsession` script, you will find the section where the users `$startup` script is exec'ed. This section needs to be modified to remove all exec's. The following example is from the `Xsession` file delivered on HP-UX platforms.

Example

```

if [ $shellprofile ]
then
Log "execing $startup using $shellprofile..."
  case ${SHELL##*/} in
    #
    # Modified for A_login_xm
    #
    sh | ksh) exec $SHELL -c "DT=true; export DT; \
    . $shellprofile; \
    unset DT; \
    PATH=/usr/dt/bin:\$PATH $tooltalk; \
    exec $startup >$dtstart_sessionlogfile 2>&1" ;;
    csh) exec $SHELL -c "unsetenv _ PWD; \
    setenv DT true; \
    source $shellprofile; \
    unsetenv DT; \
    (set path = ( $DT_BINPATH \$path );$tooltalk); \
    exec $startup >&! $dtstart_sessionlogfile" ;;
    sh | ksh) $SHELL -c "DT=true; export DT; \
    . $shellprofile; \
    unset DT; \
    PATH=/usr/dt/bin:\$PATH $tooltalk; \
    $startup >$dtstart_sessionlogfile 2>&1" ;;
    csh) $SHELL -c "unsetenv _ PWD; \
    setenv DT true; \
    source $shellprofile; \
    unsetenv DT; \
    (set path = ( $DT_BINPATH \$path );$tooltalk); \
    $startup >&! $dtstart_sessionlogfile" ;;

  esac
else
  PATH=/usr/dt/bin:$PATH $tooltalk
  Log "execing $startup..."
  #
  # Modified for A_login_xm
  #
  # exec $startup >$dtstart_sessionlogfile 2>&1
  $startup >$dtstart_sessionlogfile 2>&1
  fi
sleep 2
/usr/local/arsap/bin/A_login_xm logout
##### eof #####

```

Now the location of the Xsession file needs to be defined in the CIMS for UNIX Parameter File. This can be done using the CLI as follows;

```
> /usr/arsap/bin/A_setup
A_SETUP> DEFAULT/XSESSION_LOCATION="/etc/dt/Xsession"
%SETUP-S-MODIFIED, 1 node record successfully modified
A_SETUP> exit
>
```

CDE - on SOLARIS 7

On SOLARIS, the Xconfig file does not define a value for Dtlogin*session. The login session is unconditionally /usr/dt/bin/Xsession. In addition, this version of Xsession does not "exec" the users startup. Instead the user's startup is started as a separate process. For this reason, A_login_xm must be called directly from the Xsession file. The call to A_login_xm with the "logout" argument is called in /usr/dt/bin/Xsession as well.

First add /usr/dt/bin/Xsession to the XSESSION_LOCATION field of the CIMS for UNIX Parameter File:

```
# A_setup
A_SETUP> DEFAULT/XSESSION_LOCATION="/usr/dt/bin/Xsession"
%SETUP-S-MODIFIED, 1 node record successfully modified
A_SETUP> exit
```

Next, add the following lines at the start of /usr/dt/bin/Xsession:

```
#####
# ADD THIS SECTION FOR CIMS FOR UNIX A_login_xm FOR LOGIN
#
#####

if [ -z "$ARSAP_CURRPROJ" -a -z "$ARSAP_LOGIN_RUN" ]
then
  export ARSAP_LOGIN_RUN="TRUE"
  export ARSAP_CURRPROJ="TRUE"
  /usr/arsap/bin/A_login_xm
  exit
fi

#####
```

Next, add the following lines to the end of `/usr/dt/bin/Xsession`:

```
#####
#
# ADD THIS SECTION FOR CIMS FOR UNIX A_login_xm FOR LOGOUT
#
#####
sleep 2
/usr/arsap/bin/A_login_xm logout
#####
##### eof #####
```

Gnome Display Manager (gdm) on Linux

The gdm login screen, `gdmgreeter` allows the user to select from several environments prior to logging in. Unless `gdmgreeter` has been modified, the user may select from the Gnome, KDE, Default, Failsafe, or Anotherlevel environments. For each environment, a script exists in the `gdm SessionDir` directory which calls `/etc/X11/xdm/Xsession` with an argument of the selected environment. The `SessionDir` directory is `/etc/X11/gdm/Sessions`. For example, for the Gnome environment, `Xsession` is called from `/etc/X11/gdm/Sessions/Gnome` as follows;

```
#!/bin/bash

/etc/X11/xdm/Xsession gnome
```

To implement the CIMS for UNIX X-windows Login Utility, modify `/etc/X11/gdm/Sessions/Gnome` as follows, (assume CIMS for UNIX is installed in `/usr/cims`);

```
#!/bin/bash

/usr/cims/bin/A_login_xm gnome
```

A similar modification would be required for each of the scripts in `/etc/X11/gdm/Sessions` except the Failsafe script. Failsafe should be left as an emergency login environment.

Next, set the `Xsession_Location` field in the CIMS for UNIX Parameter file.

```
> /usr/cims/bin/A_setup
A_SETUP> DEFAULT/XSESSION_LOCATION="/etc/X11/xdm/Xsession"
%SETUP-S-MODIFIED, 1 node record successfully modified
A_SETUP> exit
>
```

Finally, the `Xsession` script needs to be modified to remove the use of "exec" and to call the CIMS X-window Login Utility with the argument "logout" when the user's session is complete. The following example shows an `Xsession` script, which has been modified. Note that the "failsafe" case has been left unchanged.

```
case $# in
1)
    case $1 in
    failsafe)
        exec xterm -geometry 80x24-0-0
        ;;
    gnome)
        gnome-session
        ;;
    kde)
        startkde
        ;;
    anotherlevel)
        # we assume that switchdesk is installed.
        /usr/share/apps/switchdesk/Xclients.anotherlevel
        ;;
    esac
esac

# otherwise, take default action
if [ -x "$HOME/.xsession" ]; then
    "$HOME/.xsession"
elif [ -x "$HOME/.Xclients" ]; then
    "$HOME/.Xclients"
elif [ -x /etc/X11/xinit/Xclients ]; then
    /etc/X11/xinit/Xclients
else
```

```

        xsm
fi

sleep 2
/usr/cims/bin/A_login_xm logout

```

Data General

- The Xsession file uses \$HOME/.Xsession instead of \$HOME/.xsession.
- The default system .mwmrc file is named system.mwmrc and is located in /usr/lib/x11. Individual .mwmrc files are located in the user's home directory. Depending upon which of these files is in use, the following change needs to be made in order to write a CIMS for UNIX logout record.

The menu item Root Menu in .mwmrc is defined as follows:

```

Menu Root Menu{
.
.
"logout"f.exec "DG_KILL -TERM $XSESSIONWINDOW"
}

```

This should be changed to:

```

Menu Root Menu
{
.
.
"logout"f.exec "/usr/arsap/bin/A_login_xm logout; DG_KILL -TERM
$XSESSIONWINDOW"
}

```

IBM

- For 3.2.x, on IBM RS6000 AIX platforms in addition to modifying the Xsession file, the default system .mwmrc file must also be changed. This can usually be found in /usr/lpp/X11/Motif1.2/app_defaults directory. The menu item End_Session in .mwmrc is defined as follows:

```
Menu "end_session"  
  
{  
  "End_Session"f.title  
  no-labelf.separator  
  "OK"f.exec "kill -9 0; kill -9 $$"  
}
```

This should be changed to:

```
Menu "end_session"  
  
{  
  "End_Session"f.title  
  no-labelf.separator  
  "OK"f.exec "/usr/arsap/bin/A_login_xm logout; kill -9 0; kill -  
9 $$"  
}
```

If users have their own version of .mwmrc they must include this change.

SGI

- IRIX has more than one DisplayManager*session in the X configuration file. Each one must be replaced with a call to A_login_xm.

```
DisplayManager._0.session: /usr/arsap/bin/A_login_xm  
DisplayManager._1.session: /usr/arsap/bin/A_login_xm
```

- On SGI IRIX systems, the call to `A_login_xm logout` is not put in the `Xsession` file. IRIX uses a special routine `/usr/bin/X11/endsession` to end a window session. The following should be the last executed line in the file after the call to:

```
/usr/bin/X11/xprop -remove . . .  
sleep 2  
/usr/arsap/bin/A_login_xm logout
```

SUN (SunOS and Solaris)

- SUN systems running OpenWindows probably need to create a link:

```
ln -s $OPENWINHOME/bin/openwin /usr/lib/X11/xdm
```

CIMS for UNIX Switch Utility

The CIMS for UNIX Switch Utility (`A_switch`) performs online project switching under UNIX. `A_switch` can be used in two ways: `exec'ed process` or `activity-only`.

UNIX `exec` lets the specified command replace the current shell that is then terminated. Therefore, `A_switch` replaces the current shell, prompts for the project, validates the project and then `exec's` the user's shell using the `SHELL` environment variable. UNIX `exec` passes the user's environment to the new process. The best way to implement `A_switch` is to define an alias (not available under Bourne shell). By using an alias, the user does not have to remember to `exec A_switch`. For example, when the alias for `A_switch` is defined as (under `csh`):

```
# alias sw "exec $ARSAP_HOME/bin/A_switch"
```

The user simply executes the command `sw` to perform on-line project switching.

You can also exec `A_switch` from within a script. This capability lets project within background or batch jobs assure this work is tracked to the proper project. You should use the user/project authorization validation method because passwords would be in plain text. The following example illustrates the use of `A_switch` within a script:

```
#!/bin/csh
exec A_switch home <<xxFUNNYxx
<commands ...>
xxFUNNYxx
```

Another way to provide a project for background or batch jobs is the activity-only switch. `A_switch` creates an activity-only switch when a minus sign (-) precedes the project. `A_switch` enters the project into `A_activity.sys` and does not exec a new shell.

The syntax of an activity only switch is:

```
#A_switch -<project>
```

If you use activity-only switching, you must also set the value of `SELECT_QUALS` to include `/UNKNOWN=LAST_PROJECT` in the CIMS for UNIX Configuration File, `A_config.par`.

Project Accounting for NT

The CIMS for NT Login Utility is a Windows application. This utility manages project accounting on NT as a session task. The task prompts the user for a project. After the project is validated, this task remains until the user logouts. This task is identified by the ARSAP Resource Accounting icon on your desktop. When you open this icon, the ARSAP Resource Accounting window appears with three pulldown menus: File, Project and Help.

The File menu contains four items: **Start-Up Program Manager**, **Logoff**, **Reboot** and **Shutdown**. The **Start-Up Program Manager** menu item manages the Program Manager. If Program Manager is running, CIMS for NT pops it to the top; else, CIMS for NT starts the Program Manager. The

Logoff menu item lets the user end the session. The **Reboot** menu item invokes the reboot sequence. The **Shutdown** menu item invokes the shutdown sequence.

The **Project** menu contains two items: **Switch** and **Current Project**. The **Switch** menu item provides the on-line project switching function. The **Current Project** menu item displays the user's current project.

The **Help** menu provides the Windows on-line help facility.

The CIMS for NT Login Utility is implemented two ways: `NT_login` and `NT_login_scr`. They provide the same functionality but are invoked through different means. `NT_login` supports login shell capabilities. You replace the default login shell with `NT_login`. Under NT 3.51, the default login shell is Program Manager (`progman`). Under NT 4.0, the default shell is Explorer (`explorer`). `NT_login_scr` supports start-up script capabilities. You invoke `NT_login_scr` as the first task from your start-up script. Under NT 4.0, you can only use `NT_login_scr`.

CIMS for NT Login Shell (NT 3.51 only)

In order to use `NT_login`, you must edit the NT Registry to modify the login shell. The Windows Registry Editor (Regedit) icon should be located in the Administrative Tools program group. If this icon has not been set-up, you should find the registry editor executable (`regedt32.exe`) under `winnnt35\system32`. After starting up RegEdit, you need to follow the following path that contains the shell definition:

```
HKEY_LOCAL_MACHINE
  SOFTWARE
    MICROSOFT
      WINDOWSNT
        CURRENT VERSION
          WINLOGON
```

The shell variable is defined in WINLOGON. You modify this variable from:

```
Shell: REG_SZ: progman.exe to
```

```
Shell: REG_SZ: nt_login.exe
```

You need to move NT_login.exe into the NT system directory. For example, if NT and CIMS for NT are installed on your D: drive, move NT_login.exe to D:\WINNT35\SYSTEM32.

For example under MS-DOS, you use the following commands to move NT_login to the NT system area:

```
D:\> cd arsap\bin
D:\arsap\bin> move NT_login.exe D:\WINNT35\SYSTEM32
```

Two anomalies occur when you use NT_login:

- Program Manager pop-down file menu no longer has logoff, shutdown, or reboot options.
- When performing any NT installation, Program Manager exits.

To correct the Program Manager logoff menu selection anomaly, log off, shutdown, and reboot options are provided under the File menu from the ARSAP Resource Accounting icon.

You can easily restart the program manager from the ARSAP Resource Accounting icon. Click the icon and select File ► Restart Program Manager.

For NT 3.51, you modify the user profile through the User Profile Editor program item. The User Profile Editor is usually located in the Administrative Tools program group.

To modify the user profile for NT 3.51

- 1** Click the **User Profile Editor** icon.
- 2** In the User Profile window, turn on the **Wait for logon script to complete before starting program manager** toggle setting. (If marked, it is on.)
- 3** If this solution is system-wide, we recommend you save the profile as the system default

CIMS for NT Login Script (NT 4)

You invoke NT_login_scr as the start-up script or as a call to NT_login_scr from a start-up script.

Start-up scripts are specified in the user environment profile. You modify the user environment profile through the User Manager program item usually located in the Administrative Tools program group. After invoking User Manager, choose a user that brings up the User Properties window. You click **Profile** displaying the User Environment Profile window. You modify the **Logon Script Name** data entry box to specify your start-up script that could be NT_login_scr.exe. The behavior of scripts is controlled by the user profile.

For NT 4.0, you modify the user profile by setting the synch logon scripts in the Registry. You use the System Policy Editor that is usually located in the Administrative Tools menu group.

To modify the user profile for NT 4.0

- 1** Click **System Policy Editor**.
- 2** Click **File**.
- 3** Click **Open Registry**.
- 4** Double-click **Local Computer**.
- 5** Double-click **Windows NT System**.
- 6** Double-click **Logon**.
- 7** Turn on the **Logon Scripts Synchronously** toggle setting. (If checked, it is on.)
- 8** Save Registry settings.

If you use the default script area, you then need to move NT_login_scr to the script area. For example, if NT and CIMS for NT are installed on your D: drive, move NT_login_scr.scr to
D:\WINNT\SYSTEM32\REPL\IMPORT\SCRIPTS.

For example under MS-DOS, you use the following commands to move NT_login_scr to the NT default script area:

```
D:\> cd arsap\bin  
  
D:\arsap\bin> move NT_login_scr.scr  
D:\WINNT\SYSTEM32\REPL\IMPORT\SCRIPTS
```

If you have established user profiles, you might need to modify each profile to invoke NT_login_scr as the start-up script.

If you call NT_login_scr from a start-up script, you need to add the call to NT_login_scr.exe into the script. NT_login_scr should be the first call to assure all work is tracked to the proper project.

If you decide to change from NT_login to NT_login_scr, make sure the login shell in the Registry is progman.exe. If you already use a start-up script, we recommend you use NT_login_scr instead of NT_login.

Project Accounting for NQS Batch Processing

You implement project accounting for Network Queuing System (NQS) batch processing by also implementing CIMS for UNIX project accounting.

CIMS for UNIX associates the resource statistics consumed for an NQS batch job with a project. The NQS submit command, qsub, includes an -x option. The -x option exports the user's environment variables to the batch job. In addition, if the batch shell strategy is set to either **login** or **free**, NQS uses the user's login shell when running the job.

If you are using CIMS for UNIX project accounting, the environment variable ARSAP_CURRPROJ should be set in the user's environment and the user's shell should be A_login. When the user submits a job to NQS with the -x option, NQS supplies the environment variable ARSAP_CURRPROJ and executes A_login.

A_login checks for the environment variable ENVIRONMENT that NQS sets to BATCH for all jobs. A_login then checks for the environment variable ARSAP_CURRPROJ, that indicates the user's current project. If

ARSAP_CURRPROJ is not present but the project accounting option is enabled for the user, A_login creates a project name from the environment variable QSUB_REQNAME, that is the name of the job. A_login writes a batch project record to A_activity.sys for the job, thus, tracking the job to a project.

When the batch login strategy is set to either **login** or **free**, the following records are written to the various accounting files:

- Batch Login record: written by A_login to the CIMS for UNIX Activity File
- Image record(s): Image termination records written to the UNIX Acct File
- Batch Logout record: batch job termination record written to the defined batch accounting log file

If the batch shell strategy is set to **fixed**, A_login is not executed when the job begins. Therefore, A_login cannot determine the project and writes a Batch Login record to A_activity.sys. In this case, each image record is credited as a single batch job executed on the AT batch queue. *CIMS for NT/UNIX* reports the resource statistics consumed by these batch jobs to the user's default batch project. For this reason, you should set the batch login strategy to either **login** or **free**.

The `qsub -x` option is essential to batch processing project accounting. You must ensure `qsub` is run with this option. This is accomplished by redefining the `qsub` command. When you have designated `csch` in the user account's shell, you can define an alias for `qsub` to include the `-x` option. You would place the following entry in the user's `.cshrc` file: the `-x` option in all subsequent `qsub` commands:

```
#  
# Define qsub to include "-x" option.  
#  
alias qsub "qsub -x"  
#
```

CIMS for UNIX Oracle Project Accounting

With CIMS for UNIX, Oracle resources can be tracked per UNIX/Oracle user. In addition, CIMS for UNIX lets you obtain statistics on a project basis. The *project* can specify another individual entity, whether it is an application name, a task, or the initiator of the process.

CIMS for UNIX Oracle Project accounting is maintained through two CIMS for UNIX utilities: CIMS for UNIX Oracle Login and CIMS for UNIX Oracle Project Switch. These utilities require a new table ARSAP_PROJECT table.

The ARSAP_PROJECT tablespace is created during the installation process through an SQL script. This table consists of fields for project information. The CIMS for UNIX Oracle utilities update the table by adding a new record each time they are invoked.

The CIMS for UNIX Oracle daemon samples the proper Oracle tables to gather resource statistics. The daemon also analyzes the ARSAP_PROJECT table to determine the resources utilized per project.

Since this table is part of the schema of an instance, it can be updated by other means depending on the roles/privileges granted to that mechanism. If the approach is taken, the ARSAP_PROJECT table must be updated with the proper project information letting the CIMS for UNIX Oracle daemon operate correctly.

CIMS for UNIX project accounting can be used for on-line transaction processing (OLTP) applications where all transactions come in through a single, continuously connected Oracle user. The identity of the initiator is not normally passed into Oracle. For example, in the case of a transaction manager like Tuxedo.

In the case of a transaction manager, the Oracle user logs into Oracle using A_dbp10 to specify the initial project. After the user is logged in and messages are received, the manager (or bulletin board) must extract the identity of the initiator of each transaction and updates the project table. This identity provides the project with the gathered resource associated statistics.

CIMS for UNIX Oracle Login

The CIMS for UNIX Oracle Login Utility (`A_dbplo`) logs the initial project you are working on. This utility is executed at the UNIX command line to replace the typical means for connecting to an Oracle application.

The command line format for the CIMS for UNIX Oracle Login Utility is as follows:

```
% A_dbplo <application> [-project] [Oracle parameters]
```

where `application` is the Oracle application and, as with any Oracle application, the additional Oracle parameters can be provided (for example, `username/password`).

The project can be specified on the command line. The project designation is optional. If a project is not specified, the utility retrieves the project from the current CIMS for UNIX project defined in the UNIX environment.

For example, Oracle `sqlplus` can be started with this command:

```
sqlplus username/password
```

To utilize project accounting, the same Oracle application is started by this command:

```
% A_dbplo sqlplus -<project> username/password
```

In addition, an alias can be defined to let you more conveniently invoke the CIMS for UNIX Oracle Login Utility. Of course, defining an alias is dependent on the UNIX shell you are using. For example, an alias to define CIMS for UNIX Oracle Login to invoke `sqlplus` would be (for `csh`),

```
% alias A_sql+="/usr/arsap/bin/A_dbplo $ORACLE_HOME/bin/sqlplus"
```

With an alias, `sqlplus` can be started with `A_sql+`, invoking CIMS for UNIX Oracle login and `sqlplus`.

The same would be true for sql_ldba shown in this example,

```
% alias A_sqlldba "/usr/arsap/bin/A_dbplo $ORACLE_HOME/bin/sqlldba"
```

as well as, with Oracle applications developed through forms.

CIMS for UNIX Oracle Project Switch

The CIMS for UNIX Oracle Project Switch Utility (A_dbps0) modifies the current project to the specified project. You switch to another project while running the Oracle application. At any time, you change projects using the command:

```
!A_dbps0 project
```

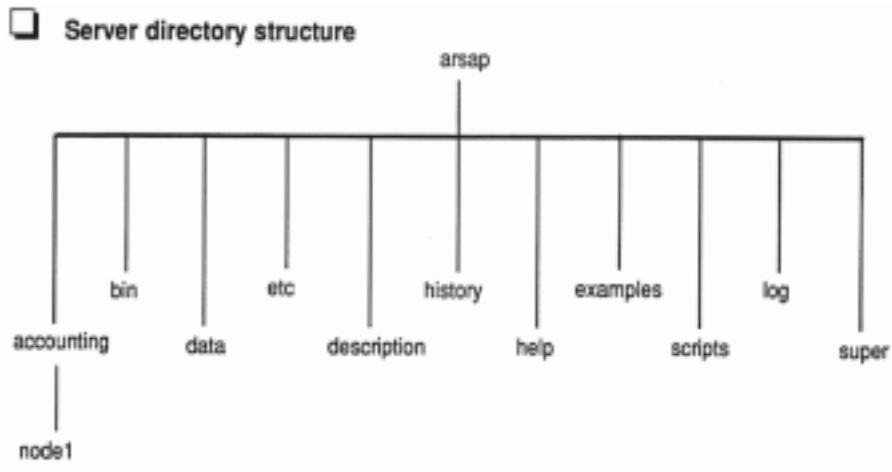
After the new record is entered into the ARSAP_PROJECT table, all activity is tracked for this new project.

CIMS for NT/UNIX Directory Structure

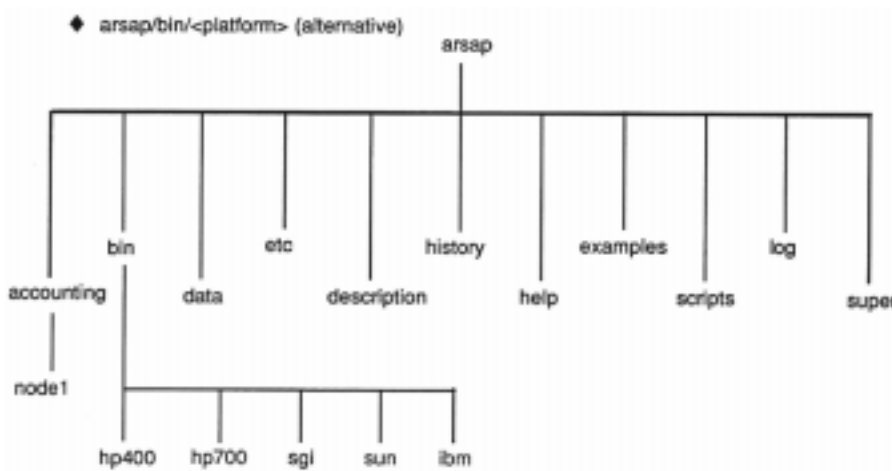
Distribution Kit File Descriptions	3-2
CIMS for UNIX Accounting Directory	3-4
CIMS for NT/UNIX Bin Directory	3-4
CIMS for NT/UNIX Data Directory	3-5
CIMS for NT/UNIX Description Directory	3-10
CIMS for NT/UNIX Etc Directory	3-10
CIMS for UNIX Examples Directory	3-14
CIMS for NT/UNIX Help Directory	3-14
CIMS for NT/UNIX History Directory	3-15
CIMS for NT/UNIX Log Directory	3-15
CIMS for UNIX Scripts Directory	3-15

Distribution Kit File Descriptions

The CIMS for NT/UNIX installation sets up the directory structure. The top-level directory is the location where CIMS for NT/UNIX is installed and is referred to as ARSAP_HOME. The ARSAP_HOME directory is as follows:



The CIMS for UNIX NFS directory structure with shared binaries is slightly different. This ARSAP_HOME directory is as follows:



The CIMS for NT/UNIX directories are as follows:

- The CIMS for NT/UNIX Accounting directory (accounting)
- The CIMS for NT/UNIX Bin directory (bin)
- The CIMS for NT/UNIX Data directory (data)
- The CIMS for NT/UNIX Description directory (description)
- The CIMS for NT/UNIX Etc directory (etc)
- The CIMS for NT/UNIX Examples directory (examples)
- The CIMS for NT/UNIX Help directory (help)
- The CIMS for NT/UNIX History directory (history)
- The CIMS for NT/UNIX Log directory (log)
- The CIMS for UNIX Scripts directory (scripts)

The CIMS for UNIX Admin directory (admin)

The CIMS for UNIX DB2 directory (db2)

The CIMS for UNIX Oracle directory (oracle)

The CIMS for UNIX Reports directory (reports)

The CIMS for UNIX Cost Center Report directory (cost_center)

The CIMS for UNIX Node Report directory (node)

The CIMS for UNIX Oracle Report directory (oracle)

The CIMS for UNIX Package Report directory (package)

The CIMS for UNIX Project Report directory (project)

The CIMS for UNIX User Report directory (user)

■ CIMS for NT/UNIX Directory Structure

Distribution Kit File Descriptions

The CIMS for NT/UNIX Data directory is an important directory. This directory contains the data files needed during the operation of CIMS for NT/UNIX. CIMS for NT/UNIX uses an environment variable named `ARSAP_DATA` to locate the directory.

CIMS for UNIX Accounting Directory

The CIMS for NT/UNIX Accounting directory (`accounting`) only exists on the `ACCOUNTING_SERVER`. This directory contains subdirectories for each `ACCOUNTING_CLIENT` node that transfers CIMS for NT/UNIX accounting and storage files to the server each night. Each node subdirectory contains the `ACCOUNTING_CLIENT` node's accounting and storage files.

This directory is only created on the `ACCOUNTING_SERVER`.

CIMS for NT/UNIX Bin Directory

The CIMS for NT/UNIX Bin directory (`bin`) contains binaries of the CIMS for NT/UNIX utilities and the UNIX libraries needed to build the CIMS for UNIX Oracle and DB2 utilities.

Note • All these files are mandatory and cannot be deleted.

The CIMS for UNIX Oracle libraries are: `libdbao.a`, `libdbplo.a` and `libdbpso.a`, that build the CIMS for UNIX Oracle utilities: `A_dbao`, `A_dbplo` and `A_dbpso`, respectively. If you do not use Oracle and do not plan on using Oracle, you can delete these libraries.

The CIMS for UNIX DB2 libraries are `libdbadb2.a`, `libdbadb2.v5.a` and `libdbadb2_pe` that build the CIMS for UNIX DB2 utilities `A_dbadb2`, `A_dbadb2_v5` and `A_dbadb2_pe`, respectively. If you do not use DB2 and do not plan on using DB2, you can delete these libraries.

CIMS for NT/UNIX Data Directory

The CIMS for NT/UNIX Data directory (data) under the home directory contains Windows NT and data files. This directory is the main processing area for CIMS for UNIX. The files contained in this data area consist of input, parameter, output and statistics files for the CIMS for Windows NT and UNIX utilities.

Files that end with “.sys” are active CIMS for NT/UNIX files. The CIMS for NT/UNIX system files (.sys) are formatted files and must only be modified using CIMS for NT/UNIX utilities. The CIMS for NT/UNIX data files (.dat) are processed files created by CIMS for NT/UNIX utilities. The CIMS for NT/UNIX parameter files (.par) are text files that you modify according to your configuration.

CIMS for NT/UNIX Accounting File

The CIMS for NT/UNIX Accounting File (acc_YYYYMMDD.dat, where YYYY is the year, MM is the month, DD is the day) is created at the end of the CIMS for NT/UNIX Nightly script. It contains the formatted accounting data. It is a mandatory file to the CIMS for NT/UNIX product but can be deleted after the data is processed.

This file can be moved or copied to the CIMS for NT/UNIX accounting directory on the ACCOUNTING_SERVER depending on options supplied to the Nightly script.

CIMS for UNIX Activity File

The CIMS for UNIX Activity File (A_activity.sys) is used by the CIMS for UNIX LOGIN and SWITCH Utilities to support the CIMS for UNIX project accounting process. This file is required for the CIMS for UNIX product to operate correctly and cannot be deleted.

CIMS for NT/UNIX Authorization File

The CIMS for NT/UNIX Authorization File (A_uaf.sys) contains USER and PROJECT account flags and parameters. This file is required for the CIMS for NT/UNIX product to operate correctly and cannot be deleted.

CIMS for UNIX Batch Statistics File

The CIMS for UNIX Batch Statistics File (`A_batstat.sys`) is created during the data formatting phase of the CIMS for UNIX nightly script. It is a mandatory file for the formatting phase to account for batch statistics, but can be deleted after the formatted CIMS for UNIX Accounting File has been created.

CIMS for UNIX Collapsible Terminal File

The CIMS for UNIX Collapsible Terminal File (`A_terminals.par`) contains a list of terminals that are collapsed by the CIMS for NT/UNIX SELECT Utility when the `TERMINAL_BY_USER` option is enabled. When this file is not present when installing CIMS for UNIX, it is created with these terminals:

```
ftp    X:    1tu    Xty    ptm
uucp   NT:   tty    pts
```

These entries specify all terminals starting with “ftp” (for example, “ftp1256”) are collapsed to simply “ftp”.

This is a mandatory file and cannot be deleted.

CIMS for NT/UNIX Holiday File

The CIMS for NT/UNIX Holiday File (`A_holiday.sys`) contains the dates of those days that you want to consider holidays. This is a mandatory file and cannot be deleted.

CIMS for NT/UNIX Image Mapping File

The CIMS for NT/UNIX Image Mapping File (`A_imgmap.sys`) contains the names of images that you want to track, and package names for the images. This is a mandatory file and cannot be deleted.

CIMS for NT/UNIX Intermediate Statistics File

The CIMS for NT/UNIX Intermediate Statistics File (`A_stat.sys`) is created by the CIMS for NT/UNIX `SELECT` Utility as a result of processing the CIMS for NT/UNIX Accounting File(s). This file is then input to the CIMS for NT/UNIX `MERGE` Utility, to produce the CIMS for NT/UNIX Merged Statistics File.

CIMS for NT Login Projects File

The CIMS for NT Login Projects file (`A_ntappstat.sys`) is created by the CIMS for NT `FORMAT` Utility. It records the date/time, node name, user name, and project name for each entry detected in the NT Application Event Log.

CIMS for NT Image File

The CIMS for NT Image file (`A_ntsecstat.sys`) is created by the CIMS for NT `FORMAT` Utility. It records the date/time, node name, user name, and images for each entry detected in the NT Security Event Log.

CIMS for NT Image List File

The CIMS for NT Image List file (`A_ntimglist.sys`) is created by the CIMS for NT `FORMAT` Utility. It maintains a list of images that are still running when the NT Security Event Log is formatted.

CIMS for UNIX Login Projects File

The CIMS for UNIX Login Projects File (`A_lgiproj.sys`) is created by the CIMS for UNIX `FORMAT` Utility. It records the date/time, node name, user name, and project name for each entry detected in the most recently processed CIMS for UNIX Activity File. It is an optional file for CIMS for UNIX.

CIMS for UNIX Login Statistics File

The CIMS for UNIX Login Statistics File (`A_lgstat.sys`) is created by the CIMS for UNIX `FORMAT` Utility. It is a mandatory file to the formatting phase, but can be deleted after the formatted CIMS for UNIX Accounting File has been created.

CIMS for NT/UNIX Merged Statistics File

The CIMS for NT/UNIX Merged Statistics File (`A_stat.dat`) is created by the CIMS for NT/UNIX `MERGE` Utility as a result of processing the CIMS for NT/UNIX Intermediate Statistics File (`A_stat.sys`) and the CIMS for NT/UNIX Storage File(s). This file is then used as input to the CIMS for NT/UNIX `REPORT` Utility.

CIMS for UNIX Print Parameter File

The CIMS for UNIX Print Parameter File (`A_print.par`) contains a list of print accounting files associated with the various print queues. To account for print jobs and pages printed, this file is required for the CIMS for UNIX product to operate correctly and cannot be deleted.

CIMS for UNIX Print Statistics File

The CIMS for UNIX Print Statistics File (`A_prtstat.sys`) is created by the `FORMAT` Utility. It is a mandatory file to the formatting phase, but can be deleted after the formatted CIMS for UNIX Accounting File has been created.

CIMS for UNIX Process Statistics File

The CIMS for UNIX Process Statistics File (`A_procstat.sys`) is created by the CIMS for UNIX `FORMAT` Utility. It is a mandatory file to the formatting phase, but can be deleted after the formatted CIMS for UNIX Accounting File has been created.

CIMS for NT/UNIX Queue Mapping File

The CIMS for NT/UNIX Queue Mapping File (`A_queuemap.sys`) contains a list of batch and print queues. This is a mandatory file and cannot be deleted.

CIMS for NT/UNIX Rates File

The CIMS for NT/UNIX Rates File (`A_rates.sys`) contains the rates used to compute the charges for all the resource statistics. This is a mandatory file and cannot be deleted.

CIMS for NT/UNIX Shift File

The CIMS for NT/UNIX Shift File (`A_shift.sys`) contains definitions for shifts. This is a mandatory file and cannot be deleted.

CIMS for NT/UNIX Storage Parameter File

The CIMS for NT/UNIX Storage Parameter File (`A_storage.par`) contains a list of file systems monitored by CIMS for NT/UNIX. This is a required file for the CIMS for NT/UNIX Sampler Utility to operate correctly and should not be deleted when you are doing storage sampling.

CIMS for NT/UNIX Storage File

The CIMS for NT/UNIX Storage File (`A_storage.sys`) contains the disk space information for USER and PROJECT account collected and logged by the CIMS for NT/UNIX Sampler Utility. This file is copied into a daily storage file (`str_YYYYMMDD.dat`, where YYYY is the year, MM is the month, DD is the day) by the CIMS for NT/UNIX Nightly script.

CIMS for UNIX Terminal Parameter File

The CIMS for UNIX Terminal Parameter File (`A_term_par.sys`) contains terminal names, groups, and factor values. This is a mandatory file and cannot be deleted.

CIMS for NT/UNIX Validation File

The CIMS for UNIX Validation File (`A_validate.sys`) contains user and project name associations used to implement the user/project validation method. It is a mandatory file for the user/project validation method.

CIMS for UNIX Wtmp Statistics File

The CIMS for UNIX Wtmp Statistics File (`A_wtmpstat.sys`) is created by the FORMAT Utility. It is a mandatory file to the formatting phase, but can be deleted after the CIMS for UNIX Accounting File has been created.

CIMS for NT/UNIX Description Directory

The CIMS for NT/UNIX Description directory (`description`) contains the CIMS for UNIX Report Description Files. The Report Description Files fall into one of these categories that are identified by the file name extension:

<code>adf</code>	Audit Description File
<code>cdf</code>	Capacity Description File
<code>gdf</code>	Graphic Description File
<code>rdf</code>	Resource Description File

CIMS for NT/UNIX Etc Directory

The CIMS for NT/UNIX Etc directory (`etc`) contains operational scripts used in maintaining and operating CIMS for NT/UNIX.

The CIMS for UNIX Date Script

The CIMS for UNIX Date script is an awk script that handles all date calculations used by CIMS for UNIX scripts. The date calculations are Year 2000 ready.

The CIMS for UNIX Environment Script

The CIMS for UNIX Environment script (`arsap_env`) sets up the CIMS for UNIX environment using the CIMS for UNIX environment configuration file (`$ARSAP_DATA/A_config.par`).

The CIMS for UNIX Platform Script

The CIMS for UNIX Platform script (`arsap_platform`) sets up the proper environment variables needed for each specific UNIX platform supported by CIMS for UNIX.

The CIMS for UNIX Check Nightly Script

The CIMS for NT/UNIX Check Nightly script (`check_nightly`) checks the CIMS for UNIX accounting directory on the `ACCOUNTING_SERVER`. Each `ACCOUNTING_CLIENT` subdirectory is searched to determine the existence of the current day's CIMS for UNIX accounting and storage files. Any failures are logged in the exception log for the current day. The name of this script has been changed from `acct_check.scr`.

The CIMS for UNIX Check Pacct Script

The CIMS for UNIX Check Pacct script (`check_pacct`) checks the size of the `pacct` file. This script performs periodic checks when the script is performed as a scheduled job through the clock daemon (`cron`). When the `pacct` file exceeds the desired limit, the `pacct` is moved into a holding area in CIMS for UNIX History directory until `turnacct` is run as part of the CIMS Nightly Script. The `check_pacct` was originally developed for HP-UX customers because of the limit set under HP-UX. However, `pacct` can grow large on any platform and there is a limit set.

The CIMS for UNIX Redo Print Script

The CIMS for UNIX Redo Print script (`redo_print`) formats the backup copies of the print accounting files listed in the CIMS for UNIX Print Parameter file (`A_print.par`). The backup copies are maintained in the CIMS for UNIX History directory. This script is part of the `redo_nightly` script.

The CIMS for NT/UNIX Nightly Script

On UNIX, the script is named `arsap_nightly`. On NT, the script is named `nt_arsap_nightly`.

The CIMS for NT/UNIX Nightly script performs the following operations:

- Turns over the raw accounting data files (UNIX raw accounting files, CIMS for UNIX raw project accounting file, Oracle statistics gathered by CIMS for UNIX Oracle daemon, Oracle statistics gathered by the CIMS for NT Oracle service, and NQS (Network Queue System) raw accounting file) into the CIMS for UNIX history directory.
- Runs the CIMS for NT/UNIX Utilities necessary to generate the CIMS for NT/UNIX Accounting File.
- Runs the CIMS for NT/UNIX SAMPLER Utility.
- Creates a backup of the raw accounting data files (optional) and sends the CIMS for NT/UNIX accounting and storage files to the designated destination.

This script automates the system accounting process when the script is performed as a scheduled job through the clock daemon (cron). This script turns the raw accounting files over, formats the accounting data into the CIMS accounting file, samples disk space to develop the CIMS storage file, and places the accounting and storage files in a designated area on the `ACCOUNTING_SERVER` node.

The CIMS for NT/UNIX Turn Accounting Script

The CIMS for NT/UNIX Turn Accounting script simplifies the process of turning over the various raw accounting data files that are used during the process of formatting the data. This file is required for the CIMS for NT/UNIX product to operate correctly and cannot be deleted.

The `turnacct` script turns over the raw accounting files for the day and starts accounting for the next day. Under UNIX this script also can be used interactively to turn UNIX accounting ON or OFF.

The /roll qualifier for A_format is important for processing image records. The roll qualifier consolidates background image records (as in, same process, user and terminal). We strongly recommend that the /roll qualifier be removed when doing image accounting under AIX 3.2.5.

The CIMS for NT/UNIX Runacct Script

The CIMS for NT/UNIX Runacct script formats the collected raw accounting files into the CIMS accounting file. This script processes UNIX raw accounting files (pacct(s) and wtmp), the CIMS for UNIX Activity file (A_activity.sys), the CIMS for NT/UNIX Oracle statistics file (A_dbacct.sys), the NQS batch accounting file (bacct) and the NT Event Logs. This file is required for the CIMS for NT/UNIX product to operate correctly and cannot be deleted.

The CIMS for NT/UNIX Storage Sampling Script

The CIMS for NT/UNIX Storage Sampling script (sampler) executes each night and performs the disk space sampling. The sampler script performs the storage data collection by executing the CIMS for NT/UNIX SAMPLER Utility. This file is required for the CIMS for NT/UNIX product to operate correctly and cannot be deleted.

The CIMS for UNIX Nightly Send Script

The CIMS for UNIX Nightly Send script (an_send) transfers the CIMS accounting and storage files generated by the CIMS for UNIX Nightly script to the ACCOUNTING_SERVER for processing and reporting. The type of transfer methods are:

- ftp (File Transfer Protocol)
- rcp
- dcp (DECnet copy)
- fmu

The CIMS for UNIX Prtacct Script

The CIMS for UNIX Prtacct script (`prttacct`) file processes the print accounting files listed in the CIMS for UNIX Print Parameter file (`A_print.par`). This file is required only for Print Accounting.

The CIMS for NT/UNIX Redo Nightly Script

The CIMS for NT/UNIX Redo Nightly script (`redo_nightly`) reprocesses the raw accounting files for a node. The raw accounting files have been saved in the `$ARSAP_HOME/history` directory. The resulting CIMS for NT/UNIX accounting file is placed on the CIMS for NT/UNIX accounting server. This script is useful when `arsap_nightly` does not run to completion.

The CIMS for NT/UNIX Process Multiple Nodes Script

The CIMS for NT/UNIX Process Multiple Nodes script processes the CIMS for NT/UNIX accounting and storage files for multiple nodes. This script is run on the CIMS for NT/UNIX accounting server that contains the accounting and storage files for all accounting nodes in the proper subdirectories of the `arsap/accounting` directory. The accounting nodes are specified in CIMS for NT/UNIX Node Parameter file (`A_node.par`).

CIMS for UNIX Examples Directory

The CIMS for UNIX Examples directory (`examples`) contains miscellaneous information that includes `A_gui` and `A_graph` resource files. The files in this directory are provided as a courtesy by the CIMS Lab and are not intended to be part of the supported product.

CIMS for NT/UNIX Help Directory

The CIMS for NT/UNIX Help directory (`help`) contains the CIMS for NT/UNIX utility on-line help files.

CIMS for NT/UNIX History Directory

The CIMS for NT/UNIX History directory (`history`) is created during installation to prepare for CIMS for UNIX system accounting. This directory retains the historical raw accounting files: UNIX accounting files (`wtmp`, `wtmpx` and `pacct`), CIMS for UNIX project accounting file (`actvty`), CIMS for UNIX Oracle accounting file (`dbacct`), CIMS for UNIX DB2 accounting files (`db2acct`), batch accounting file, and NT Event Logs. In addition, when the CIMS for NT/UNIX accounting and storage files are not transferred to an CIMS for NT/UNIX accounting directory (on this node or a remote node), they are kept in this directory.

CIMS for NT/UNIX Log Directory

The CIMS for NT/UNIX Log directory (`log`) is a repository for log files created by various CIMS for NT/UNIX utilities and scripts.

CIMS for UNIX Scripts Directory

The CIMS for UNIX Scripts directory (`scripts`) contains convenience scripts. These scripts are organized as administrative, oracle, db2 and report scripts.

The admin subdirectory contains scripts to assist the administration of CIMS for UNIX.

The db2 subdirectory contains CIMS for UNIX DB2 set-up scripts. These scripts should be utilized by your database administrator (DBA) to set-up the role and table needed for CIMS for UNIX DB2 utilities to operate properly.

The oracle subdirectory contains CIMS for UNIX Oracle SQL set-up scripts. These scripts should be utilized by your database administrator (DBA) to set-up the role and table needed for CIMS for UNIX Oracle utilities to operate properly.

■ CIMS for NT/UNIX Directory Structure

Distribution Kit File Descriptions

The reports subdirectory is organized according to the report classification. The user reports are in the user subdirectory, the Oracle reports are in the oracle subdirectory and so forth. These reports are textual.

CIMS for NT/UNIX Utilities

Introduction	4-3
CIMS for NT/UNIX Authorize Utility	4-3
CIMS for NT/UNIX Format Utility	4-3
CIMS for UNIX FS_Report Utility	4-3
CIMS for UNIX Graph Utility	4-4
CIMS for UNIX GUI	4-4
CIMS for NT/UNIX Login Utility	4-4
CIMS for NT/UNIX Merge Utility	4-4
CIMS for NT/UNIX Rates Utility	4-6
CIMS for NT/UNIX Report Utility	4-6
CIMS for NT/UNIX Sampler Utility	4-6
CIMS for NT/UNIX Select Utility	4-7
CIMS for NT/UNIX Setup Utility	4-11
CIMS for NT/UNIX Super Merge Utility	4-12
CIMS for NT/UNIX Switch Utility	4-16

CIMS for UNIX X-window Login Utility4-16

CIMS for OpenVMS Utilities4-16

**Using gen_cims_sum, NT_gen_cims_sum, gen_cims_detail, and
NT_gen_cims_detail4-16**

Introduction

The following descriptions summarize the various CIMS for NT/UNIX utilities. Detailed information about each utility can be found in the CIMS for NT/UNIX on-line help facility.

CIMS for NT/UNIX Authorize Utility

The CIMS for NT/UNIX **Authorize** Utility lets the CIMS Administrator initialize, modify, and examine CIMS for NT/UNIX user and project parameters, options, and validation combinations. These parameters and options control the operation of CIMS for each particular account, including the default accounts. This utility creates and maintains the CIMS for NT/UNIX Authorization File (`A_uaf.sys`) and Validation file (`A_validate.sys`).

Several CIMS for NT/UNIX utilities access `A_uaf.sys` and `A_validate.sys`. Changes to these files take effect the next time a utility reads the files.

CIMS for NT/UNIX Format Utility

The CIMS for NT/UNIX **FORMAT** Utility lets the CIMS Administrator format the various raw system and CIMS accounting files to produce new accounting files. This utility creates a CIMS formatted accounting file for each raw accounting file. The resulting formatted files are sorted and merged to produce the CIMS Nightly Accounting File. This utility is normally run as part of the CIMS for NT/UNIX Nightly Script.

CIMS for UNIX FS_Report Utility

The CIMS for UNIX **Filesystem** Report Utility generates two types of storage utilization reports, a Daily Report and a Trend Report. Both reports can identify individual users and projects/groups that are using the storage space sampled by the CIMS for UNIX Sampler Utility. The sampled directories determine the number of columns in the report.

CIMS for UNIX Graph Utility

The CIMS for UNIX **Graph** Utility generates graphic resource management and chargeback reports from the data contained in the CIMS for UNIX Merged Statistics File (`A_stat.dat`). This utility uses the CIMS for UNIX Report Utility to generate a data report, which is then processed and displayed graphically. You can specify the chart description interactively or load it from a file.

CIMS for UNIX GUI

The CIMS for UNIX GUI is a graphical user interface that builds and executes CIMS for UNIX utility commands. The GUI provides an easy to use and understand interface to CIMS for UNIX. The GUI lets a CIMS Administrator manage all functions within CIMS for UNIX from one common interface.

CIMS for NT/UNIX Login Utility

The CIMS for NT/UNIX **LOGIN** Utility executes each time a user logs onto the computer system. Depending upon the CIMS options in effect when invoked, this utility retrieves project information, creates a project environment, and retrieves special terminal information for use by other CIMS for NT/UNIX utilities. This utility writes this information to the CIMS for UNIX Activity File (`A_activity.sys`) or the NT Application Event log. The information is then formatted and merged with other raw accounting data.

CIMS for NT/UNIX Merge Utility

The CIMS for NT/UNIX **MERGE** Utility merges the CIMS Intermediate Statistics File (`A_stat.sys`) with information from the CIMS Authorization File (`A_uaf.sys`) and the CIMS Storage File (`A_storage.sys`). The Merge Utility can be invoked interactively after the CIMS for NT/UNIX Select Utility produces `A_stat.sys`, or invoked from

the CIMS for NT/UNIX Process Multiple Node(s) script (proc_multi). The resulting CIMS Merged Statistics File (A_stat.dat) is used to produce reports through the CIMS for NT/UNIX Report, Graph, and GUI utilities.

The CIMS for NT/UNIX MERGE Utility first copies all records from the CIMS Intermediate Statistics File to the CIMS Merged Statistics File. This utility then applies any grouping information that has been set up in the CIMS Authorization File (refer to *User Accounting* on page 2-13). Finally, this utility merges in data from the CIMS Storage Files.

The storage data that the CIMS for NT/UNIX Merge Utility reads must be located in the ARSAP_DATA Directory. If the data is not normally stored in this directory, it must be copied or linked to the directory before running the CIMS for NT/UNIX Merge Utility.

There is only one command in the MERGE Utility, MERGE. There are several miscellaneous qualifiers available with the MERGE command. These qualifiers have no effect on the data itself, just on how the MERGE command runs.

The /STATISTICS qualifier outputs resource statistics that show how much CPU time, elapsed time, I/O, and so forth, the MERGE command itself used. This option is valid only on certain UNIX platforms.

The /VERBOSE qualifier causes Merge to output more verbose messages during the MERGE command.

You do not need to tell the MERGE Utility the reporting period dates. The CIMS for NT/UNIX Select Utility stores these dates in the CIMS for NT/UNIX Intermediate Statistics File, and the MERGE Utility copies them to the CIMS Merged Statistics File.

The MERGE Utility does not act in a cumulative fashion. It runs on a given time period (the reporting period). During the phase in which the data is copied from the Intermediate Statistics File to the Merged Statistics File all the storage statistics records are zeroed. The utility then figures out the storage statistics for the given time period, and adds them to the CIMS Merged Statistics File.

Because of this, you cannot run the MERGE Utility on a nightly basis to accumulate storage statistics for the whole month. You must run the MERGE Utility once at the end of the month.

On the other hand, if you want to generate reports on a daily basis, you need to run both the SELECT and MERGE Utilities on a daily basis. This is most easily done by executing the CIMS Process Multiple Nodes Script, `proc_multi`.

Basically, a good rule of thumb is to run the MERGE Utility as often as you want to generate your reports.

CIMS for NT/UNIX Rates Utility

The CIMS for NT/UNIX RATES Utility lets the CIMS Administrator create, modify, and examine rate table records. The CIMS Rates File (`A_rates.sys`) is initially created with the `DEFAULT_RATES` rate table, which is used when adding additional rate tables to the file. This file is read by other CIMS utilities when requiring rate information.

CIMS for NT/UNIX Report Utility

The CIMS for NT/UNIX REPORT Utility lets the CIMS Administrator generate resource management and chargeback reports from the data contained in the CIMS for NT/UNIX Merged Statistics File (`A_stat.dat`), the CIMS Audit Statistics File (`A_audit.dat`), and the CIMS Capacity Statistics File (`A_cap.sys`). This utility generates resource, audit and capacity reports by utilizing a resource description file (`rdf`), an audit description file (`adf`) and a capacity description file (`cdf`), respectively.

CIMS for NT/UNIX Sampler Utility

The CIMS for NT/UNIX SAMPLER Utility scans specified directories and logs disk space utilization information in the CIMS Storage File (`A_storage.sys`). This utility samples all directories that have an entry in the CIMS Storage Parameter File (`A_storage.par`).

CIMS for NT/UNIX Select Utility

The CIMS for NT/UNIX SELECT Utility processes selected accounting data generated by the CIMS FORMAT Utility and writes that data to files more conducive to processing by other CIMS utilities. This utility does not process raw accounting files. The CIMS SELECT Utility supports writing resource statistical information to the CIMS Intermediate Statistics File (A_stat.sys) for processing by the CIMS MERGE Utility. This utility also writes session audit information to the CIMS Audit Statistics File (A_audit.dat) and node utilization information to the CIMS Capacity Statistics File (A_cap.sys).

The CIMS SELECT Utility reads the accounting data that has been gathered for each day of the reporting period and writes it to the appropriate statistics file based on the types of reports for which you are processing the data. These indexed ASCII statistics files contain records for every combination of user, project, node, and so forth, plus the statistics accumulated for each record.

The CIMS SELECT Utility also checks the date and time of each session against the information you previously entered in the CIMS Holiday Mapping File and the CIMS Shift Mapping File. If the date is a holiday that you defined, it assigns the holiday shift to that session. If it is not a holiday, then based on the time of the session this utility assigns the appropriate shift from your defined shifts to that session.

If you have mapped images to software packages, each time the SELECT Utility encounters one of those images, it adds the statistics associated with the image process to a software package record in the CIMS Intermediate Statistics File as well as the session (interactive or background).

The CIMS SELECT Utility can work cumulatively. You can have it add resource statistics from the accounting file to the CIMS Intermediate Statistics File from a previous run of Select. As long as you do not use the /ZERO qualifier (see below), the new accounting data is added to what is

already there. You also have the choice of removing the CIMS Intermediate Statistics File so that the CIMS SELECT Utility creates a new file as it begins consolidating the data for the reporting period.

The accounting data that the Select Utility reads must be located in the CIMS Data Directory. If the data is not normally stored in this directory, it must be copied or linked to the directory before running the Select Utility. If the SELECT Utility is run as part of the CIMS Process Multiple Nodes Script, `proc_multi`, the linking of the Nightly Accounting Files to the CIMS Data Directory is performed by the script.

The CIMS SELECT Utility has three commands, plus several qualifiers, that dictate how the data is processed. The main command in the utility is SELECT. This command produces the CIMS Intermediate Statistics File for resource reporting. The AUDIT and CAPACITY commands process the CIMS Nightly Accounting Files for audit and capacity planning reports, respectively. The AUDIT command produces the CIMS Audit Statistics File (`A_audit.dat`). The CAPACITY command produces the CIMS Capacity Statistics File (`A_cap.sys`).

If you are using a combination of the commands at one time, the SELECT command should come first, the audit and capacity commands should follow, delimited by a "/". Examples of how the command should be input are as follows.

```
A_SELECT> SELECT ...  
A_SELECT> AUDIT/CAPACITY ...  
A_SELECT> SELECT/AUDIT/CAPACITY ...
```

The command is then followed by the qualifiers that set the reporting period. The reporting period is one of the most important things to decide before processing the data. The reporting period you select can depend on any number of factors: such as company policies, government regulations, financial requirements, and so forth. The reporting period can range from daily to weekly to monthly. The reporting period can be an interim report, or you might want to reprocess the data for the past two years.

The reporting period is defined by a starting date and an ending date. The starting date is defined by the `/SINCE` qualifier. The ending date is defined by the `/BEFORE` qualifier. For instance, you might say the starting date is 1-mar-1999 and the ending date is 1-apr-1999. Note that neither date includes a time. The time is always assumed to be 00:00. Thus, this reporting period goes from 00:00 on the first of March to 00:00 on the first of April and includes all the time in March.

The next qualifier, `/NODE`, is only used if **Clump Support** is enabled. (See the *Operating in a Client/Server Environment* section). This qualifier indicates the nodename where the data that is currently being processed originated. The nodename is included in the CIMS Intermediate Statistics File.

The remaining qualifiers are optional, and you decide whether you need them, based on your own requirements.

You can use the `/UNKOWN_PROJECT` qualifier with the `SELECT` command to determine the assigned project when the CIMS `SELECT` Utility encounters a process with no previously assigned project. Normally, the utility uses CIMS Activity records to determine the project. If, for some reason, these records do not exist in the accounting data, the following method is used for assigning a project.

If `/UNKNOWN_PROJECT` is not present on the command line, the `SELECT` Utility checks the user's CIMS Authorization record to see if the `Default_Project` flag is enabled. If it is, the `SELECT` Utility uses the value of the `DefProject` field in the user's record if one exists. If `DefProject` has no value, the value in the user's record for either `Background_Project`, `Batch_Project`, `DBZ_Project`, `Inter_Project`, `Oracle.Project`, or `Print_Project`, depending on the jobtype of the process. If no value exists for these fields in the user's record, the username is used as a project name as a last resort.

If `/UNKNOWN_PROJECT=GROUP` is present on the command line, the `SELECT` Utility uses the group of the process from the Nightly Accounting File as a project name. The `GID Method` in the *Setup for Reporting in CIMS for NT/UNIX Parameter Administration* is used in conjunction with this value for the qualifier. The CIMS for UNIX `FORMAT` Utility derives the

group name in the Nightly Accounting File process records based on the value of `GID_Method` in the CIMS Parameter File. If `GID_Method` is set to `GROUP`, the `FORMAT` Utility will search the UNIX group file for the `GID` of the process to derive the group name. If `GID_Method` is set to `PROJECT`, `FORMAT` will first check the Project `GIDs` in the CIMS Authorization File for a `GID` that matches the Process `GID`. If a match is found, the project name will be used as the group name. If no match is found, `FORMAT` will search the UNIX group file.

If `/UNKNOWN_PROJECT=LAST_PROJECT` is present on the command line, the CIMS `SELECT` Utility uses the last project the user either logged on with or switched to prior to the start time of the process.

If no `LAST_PROJECT` exists, the CIMS `SELECT` Utility uses the method described when `/UNKNOWN_PROJECT` is not present on the command line for assigning a project to the process.

The `/ZERO` qualifier is used to zero out all the resource statistics previously stored in the CIMS Intermediate Statistics File from a prior run of the `Select` command. Any record that has already been established in the statistics file is not deleted, just zeroed out.

You can use this qualifier if you are using the `Select` command to process the accounting data for several nodes, and one node's processing does not complete correctly. This qualifier is used to zero out the bad statistics for only that node's records before reprocessing the node's data. This way you do not have to remove the statistics file and reprocess all nodes.

If you are creating monthly reports you should remove the Intermediate Statistics File for last month prior to processing for the current month instead of using the `/ZERO` qualifier. If you continue to use the `/ZERO` qualifier from month to month, records for users that are no longer accessing your systems are retained until the statistics file is removed.

The `/ADD_ACCOUNTS` qualifier can be used to add users and projects found in the accounting data to the CIMS Authorization File. By default, users and projects are not added by `SELECT`. If `/ADD_ACCOUNTS` is used to add users and projects, the new record's fields will be copied from the `DEFAULT` user and `DEFAULT` project records.

The `/STATISTICS` qualifier outputs resource statistics that show how much CPU time, elapsed time, I/O, and so forth, the `Select` command itself used. This qualifier is valid only for certain platforms.

The `/VERBOSE` qualifier causes `Select` to output more verbose messages during the execution of the `SELECT` command.

An example command for consolidating data to produce resource and audit reports from the CIMS for NT/UNIX CLI is as follows.

```
# A_select
A_SELECT> SELECT/AUDIT /SINCE=01-Jan-1997/-
_A_SELECT> BEFORE=01-Feb-1997/NODE=SPARKY
A_SELECT> EXIT
#
```

CIMS for NT/UNIX Setup Utility

The CIMS for NT/UNIX **Setup** Utility lets the CIMS Administrator display or modify the system-wide parameters and options in the CIMS Parameter File (`A_setup.sys`). These options and parameters indicate what CIMS for NT/UNIX features are enabled and how these features are implemented.

The `SETUP` Utility also displays and manages values contained in the following CIMS Mapping Files.

- The CIMS Shift File (`A_shift.sys`)
- The CIMS DB Instance Mapping File (`A_dbinst.sys`)
- The CIMS Image Mapping File (`A_imgmap.sys`)
- The CIMS Terminal Parameter File (`A_term_par.sys`)
- The CIMS Queue Mapping File (`A_queuemap.sys`)

Note • Parameters and options that control individual user or project accounts are maintained by the CIMS for NT/UNIX Authorize Utility.

CIMS for NT/UNIX Super Merge Utility

The CIMS for NT and UNIX SUPER MERGE Utility (`A_smerge`) merges the consolidated CIMS statistics files from one group of nodes with another group of nodes. This utility reduces the processing time of the data consolidation phase for large numbers of nodes. The CIMS SUPER MERGE Utility (`A_smerge`) merges two sets of CIMS reporting data, including the Merged Statistics file (`A_stat.dat`), from two accounting servers into one set of CIMS reporting data. The Super Merge Utility reduces the time needed to process system-wide accounting data by letting you consolidate data on separate accounting servers, each with multiple clients, and then super merging the resulting reporting data sets. This method is recommended for very large sites with more than 500 machines.

The Super Merge Utility combines the results of running the process Multiple Nodes Script (`proc_multi`) on two accounting servers into one super data directory (`ARSAP_SUPER`). Once the resulting super merged reporting data files have are to the final `ARSAP_DATA` directory, reports and/or graphs using `A_report` and `A_graph`, respectively, can generate reports from the super merged data.

A set of CIMS reporting data needed by super merged reporting (referred to as the reporting dataset throughout this description) consists of the following files:

- `A_imgmap.sys`
- `A_odb_base.sys`
- `A_odb_user.sys`
- `A_queuemap.sys`
- `A_shift.sys`
- `A_term_par.sys`
- `A_stat.dat`

- 3 Copy reporting data to super directories.
 - a For accounting server x on which A_smerge is to be performed, copy the reporting data files from `arsap/data` directory to the super directory.
 - b For the remaining nodes, copy the reporting data files from the accounting server's `arsap/data` directory to the appropriate `arsap/super` subdirectory.

Staying with the example, the reporting data for x is copied from `arsap/data` to `arsap/super`. The reporting data for y and z are copied from their `arsap/data` directories to `arsap/super/y` and `arsap/super/z`, respectively.

- 4 Set `ARSAP_SUPER` to the super directory.
- 5 Set `ARSAP_DATA` to one of the super subdirectories.
- 6 Run `A_smerge` with the `MERGE` command. The resulting files are located in `ARSAP_SUPER`:
 - `A_imgmap.super`
 - `A_odb_base.super`
 - `A_odb_user.super`
 - `A_queuemap.super`
 - `A_shift.sys`
 - `A_term_par.super`
 - `A_stat.super`

- 7** Move the super merged (.super) files into the appropriate reporting data file with the following commands:
 - `mv A_imgmap.super A_imgmap.sys`
 - `mv A_odb_base.super A_odb_base.sys`
 - `mv A_odb_user.super A_odb_user.sys`
 - `mv A_queuemap.super A_queuemap.sys`
 - `mv A_shift.super A_shift.sys`
 - `mv A_term_par.super A_term_par.sys`
 - `mv A_stat.super A_stat.dat`
- 8** Repeat steps 5, 6, and 7 until all accounting server subdirectories have been merged into the final super merged files, in particular, `A_stat.dat`.
- 9** Set `ARSAP_DATA` to `ARSAP_SUPER`. The `ARSAP_SUPER` directory contains the final super merged files. `A_report` and `A_graph` can now be run on the node where the final super merged reporting data is located.

CIMS for NT/UNIX Switch Utility

The CIMS for NT/UNIX **Switch** Utility lets project accounting users change projects without ending their current session. This utility lets CIMS for NT/UNIX properly track and allocate resources to defined projects. This utility writes a switch activity record to the CIMS for UNIX Activity File (`A_activity.sys`) or the NT Application Event log. The information is formatted and merged with other raw accounting data.

For UNIX X-window sessions, you can only switch the project of individual terminal windows.

CIMS for UNIX X-window Login Utility

The CIMS for UNIX **X-window Login** Utility lets users logging in from X terminals enter a project for their session. After prompting the user for a project, this utility writes a login record to the CIMS for UNIX Activity File (`A_activity.sys`) and executes the system-wide Xsession file. `A_activity.sys` is formatted and merged with other raw accounting data during the nightly data collection.

CIMS for OpenVMS Utilities

When you have a license for Reporting on OpenVMS, you can process your accounting data on the OpenVMS system that is licensed for this option. For more information, see the *CIMS for OpenVMS Reference Guide*.

Using `gen_cims_sum`, `NT_gen_cims_sum`, `gen_cims_detail`, and `NT_gen_cims_detail`

NAME

- `gen_cims_sum`—procedure for generating summary CIMS Resource Accounting records from consolidated CIMS for Unix Accounting data.

- `gen_cims_detail`—procedure for generating detail CIMS Resource Accounting records from consolidated CIMS for Unix Accounting data.
- `NT_gen_cims_sum`—procedure for generating summary CIMS Resource Accounting records from consolidated CIMS for NT Accounting data.
- `NT_gen_cims_detail`—procedure for generating detail CIMS Resource Accounting records from consolidated CIMS for NT Accounting data.

SYNTAX

```
> gen_cims_sum
> gen_cims_sum    all|<jobtype>    <arg1>    [<arg2> ... <argn>]

> gen_cims_detail
> gen_cims_detail    all|<jobtype>    <arg1>    [<arg2> ... <argn>]

> NT_gen_cims_sum
> NT_gen_cims_sum    all|<jobtype>    <arg1>    [<arg2> ... <argn>]

> NT_gen_cims_detail
> NT_gen_cims_detail    all|<jobtype>    <arg1>    [<arg2> ... <argn>]
```

DESCRIPTION

To generate CIMS Resource Accounting records from consolidated CIMS for Unix and CIMS for NT accounting data. User enters jobtype of summary records to be generated or all jobtypes. Jobtype is followed by use of 1 - n identifiers as arguments. These identifiers can be any of the report selection keywords or user supplied text. The arguments correspond to identifiers that will appear in the CIMS Resource Accounting records. Argument 1 will be entered as identifier 1, argument 2 as identifier 2 and so on. If the argument is a selection keyword, the

■ CIMS for NT/UNIX Utilities

Using gen_cims_sum, NT_gen_cims_sum, gen_cims_detail, and NT_gen_cims_detail

value associated with the keyword will be entered in the corresponding identifier field. If the argument is not a selection keyword, the argument will be used as text in the corresponding identifier field.

Valid jobtypes

- all
- background
- batch
- db2
- interactive
- oracle
- package
- print
- storage

CIMS-UNIX selection keywords

user	username
user_description	user description from Authorization file
user_gos	user GOS from UNIX password file
user_group1	user group from Authorization file
user_group2	user group from Authorization file
user_group3	user group from Authorization file
user_group4	user group from Authorization file
user_group5	user group from Authorization file
user_cost_center	user cost center from Authorization file
uid	user UNIX UID
gid	user UNIX GID
packagename	software package name
project	projectname
project_description	project description form Authorization file
project_group1	project group from Authorization file
project_group2	project group from Authorization file
project_group3	project group from Authorization file
project_group4	project group from Authorization file
project_group5	project group from Authorization file
project_cost_center	project cost center from Authorization file

■ CIMS for NT/UNIX Utilities

Using gen_cims_sum, NT_gen_cims_sum, gen_cims_detail, and NT_gen_cims_detail

queue	queuename
storage_group	GID group
terminal	terminal name
shift	shift name
or_user	Oracle username
or_base	Oracle instance name
db2_base	DB2 database name

At least one of the arguments must be a selection keyword.

User may indicate a blank identifier by using "" for the corresponding argument.

The output from running `gen_cims_sum` is written to the file, `cims_sum.csv` in the `ARSAP_DATA` directory. Use of certain selection keywords limits the selection to only records of the jobtype associated with the keyword. This is done to eliminate useless records with 0 stats from the output. They are:

or_user and or_base	selection limited to only Oracle jobtype.
db2_base	selection limited to DB2 jobtype.

Software Package records are generated exclusively by using the selection keyword `packagename` as one of the arguments. Software Package accounting statistics are supplemental to statistics contained in interactive and background jobtypes. Therefore, they are not included if `packagename` is not present on the command line.

When using either `gen_cims_sum` or `NT_gen_cims_sum` the resulting file of summary records, `gen_cims_sum.csv` is in `ARSAP_DATA` directory and contains data for the reporting period defined when performing the CIMS for Unix or CIMS for NT data consolidation phase.

Using gen_cims_sum, NT_gen_cims_sum, gen_cims_detail, and NT_gen_cims_detail

When using either `gen_cims_detail` or `NT_gen_cims_detail` the resulting file of detail, (audit) records, `gen_cims_detail.csv` is in `ARSAP_DATA` directory and contains data for the reporting period defined when performing the CIMS for UNIX or CIMS for NT data consolidation phase.

If any one of these procedures is called with no arguments, the following default arguments are supplied, all node user

USAGE

To generate summary records for all jobtypes, (except Software Package), on a UNIX platform with the following identifier fields,

identifier1	node name
identifier2	username
identifier3	a string "mystring"

```
> gen_cims_sum all node user mystring
```

To generate detail records for Software Package jobtype on an NT platform with `nodename`, `username`, and `packagename`

identifier1	node name
identifier2	username
identifier3	packagename

```
> NT_gen_cims_detail package node user packagename
```

■ CIMS for NT/UNIX Utilities

Using gen_cims_sum, NT_gen_cims_sum, gen_cims_detail, and NT_gen_cims_detail

The CIMS for NT/UNIX Files

CIMS for UNIX Configuration Files	5-3
CIMS for NT/UNIX Configuration Parameter Files	5-3
CIMS for NT/UNIX Node Parameter File	5-3
CIMS for NT/UNIX Storage Parameter File	5-4
CIMS for NT/UNIX Parameter File	5-4
CIMS for NT/UNIX Accounting Files	5-5
CIMS for UNIX Activity File	5-5
CIMS for NT/UNIX Authorization File	5-5
CIMS for UNIX Collapsible Terminal File	5-6
CIMS for NT/UNIX Interim Statistics Files	5-6
CIMS for NT/UNIX Holiday File	5-7
CIMS for NT/UNIX Image Mapping File	5-7
CIMS for NT/UNIX Intermediate Statistics File	5-8
CIMS for UNIX Login Projects File	5-8
CIMS for NT/UNIX Merged Statistics File	5-8

CIMS for NT/UNIX Oracle Accounting File 5-8

CIMS for NT/UNIX Oracle Basename File 5-8

CIMS for NT/UNIX Oracle Username File 5-9

CIMS for UNIX Print Parameter File 5-9

CIMS for UNIX Queue Mapping File 5-9

CIMS for NT/UNIX Rates File 5-9

CIMS for NT/UNIX Shift File5-10

CIMS for NT/UNIX Storage File5-11

CIMS for UNIX Terminal Parameter File5-11

CIMS for NT/UNIX Validation File5-11

CIMS for UNIX X-Application Resource Files5-11

CIMS for UNIX Report Description Files5-12

Select Section5-12

Sort Section5-18

Body Section5-19

Report Description File Considerations5-27

Report Description File Examples5-28

CIMS for UNIX Graph Description File5-31

CIMS for UNIX Configuration Files

The CIMS for UNIX Configuration File defines the `ARSAP_DATA` environment variable that is necessary for the license facility to locate the license information. The installation process creates this file. The CIMS for UNIX configuration file is `/etc/arsap.conf`. The CIMS for NT configuration file is located on the system drive in `\ETC\NT_arsap_conf.bat`.

WARNING • These files are crucial to the operation of CIMS for NT/UNIX. These files must NOT be moved, modified or deleted.

CIMS for NT/UNIX Configuration Parameter Files

The CIMS for NT/UNIX Configuration Parameter Files define various environment variables used by CIMS operational scripts. These files are located in the `ARSAP_DATA` directory. To modify your configuration, simply edit the variable definitions in these text files. These files have comments to assist you in changing the correct environment variables.

The CIMS for UNIX Configuration Parameter file is `A_config.par`. This file is created during installation according to your responses to configuration questions.

The CIMS for NT Configuration Parameter File is `NT_config_par.bat`. A default `NT_config_par.bat` is provided with the distribution and can be modified following installation.

CIMS for NT/UNIX Node Parameter File

The CIMS for NT/UNIX Node Parameter File (`A_node.par`) specifies the nodes that are being processed on an accounting server. Several CIMS for NT/UNIX scripts that typically run on an accounting server use this file. This `A_node.par` file is located in the `ARSAP_DATA` directory. This file is created during installation but only the node you are installing on is

recorded in this file. In order to get a complete `A_node.par` file, concatenate the resulting `A_node.par` files from all your client nodes into the `A_node.par` file on your accounting server.

CIMS for NT/UNIX Storage Parameter File

The CIMS for NT/UNIX Storage Parameter File (`A_storage.par`) contains the names of the UNIX filesystems or NT storage drives to be sampled by the CIMS for NT/UNIX Sampler utility. The `A_storage.par` file is located in the `ARSAP_DATA` directory. CIMS for UNIX creates this file during installation according to the local filesystems on your system. NFS mount points are excluded. A default `A_storage.par` file is delivered with CIMS for NT. You can modify this text file according to your storage reporting needs.

Under UNIX, migration and AFS (Andrew File System) filesystems need to be specified in the `A_storage.par` file. A migration filesystem must be designated by the letter M. An AFS filesystem must be designated by the letter A followed by the backup subdirectory. The following example shows a migration and an AFS filesystem entry in the `A_storage.par` file:

```
/mig1 M  
/afs A OldFiles
```

CIMS for NT/UNIX Parameter File

The CIMS for NT/UNIX Parameter File (`A_setup.sys`) contains system-wide flags and parameters that define your configuration. It is created and maintained by the CIMS SETUP Utility.

For first time installations, the Parameter File is delivered so that a minimal CIMS for NT/UNIX system operates without any tailoring.

Refer to the [CIMS for NT/UNIX User Guide](#) and the CIMS SETUP Utility on-line help to set the parameters for your configuration.

CIMS for NT/UNIX Accounting Files

CIMS for NT/UNIX Accounting Files (`acc_<tag>.dat`) contains formatted and chronologically sorted records obtained from the UNIX Acct File, the UNIX Wtmp File, the CIMS for UNIX Activity File, the NQS and CXbatch Accounting Logfile, individual Print Accounting Files listed in the CIMS for UNIX Print Parameter File, the CIMS for UNIX Oracle Accounting File, the CIMS for UNIX IBM DB2 Accounting File and CIMS for NT event logs. It is a mandatory file to the CIMS for NT/UNIX product but can be deleted after the data is processed.

This file can be moved or copied to the CIMS accounting directory on accounting server depending on options supplied to the CIMS for NT/UNIX Nightly scripts.

CIMS for UNIX Activity File

The CIMS for UNIX Activity File (`A_activity.sys`) contains CIMS for UNIX login/logout records, CIMS for UNIX Switch records and batch login and logout records. This file is used by the CIMS for UNIX Login, X-window Login and Switch Utilities to support the CIMS for UNIX project accounting process. This file is required for the CIMS for UNIX product to operate correctly and cannot be deleted.

CIMS for NT/UNIX Authorization File

The CIMS for NT/UNIX Authorization File (`A_uaf.sys`) contains flags, parameters, and values pertaining to USER and PROJECT accounts. It is created and maintained by the CIMS AUTHORIZE Utility.

For first time installations, the CIMS Authorization File is delivered so that a minimal CIMS for NT/UNIX system operates without any modifications. Refer to *Chapter 2, CIMS for NT/UNIX Accounting* of this book for more information about the various flags and parameters.

■ The CIMS for NT/UNIX Files

CIMS for UNIX Collapsible Terminal File

Two records in the CIMS Authorization File are of particular importance. They are the DEFAULT user and DEFAULT project records, that contain the parameter settings that are used for values that are not supplied when adding new users and projects to the CIMS Authorization File.

User accounts are automatically added to this file when the user logs into the system for the first time through the CIMS LOGIN Utility using the DEFAULT user record settings. You can also manually add user accounts.

Project accounts are not automatically added unless you disable all project validation methods, so you need to add project accounts before you enable Project Accounting for any kind of project validation.

Refer to the *CIMS Chargeback NT/UNIX User Guide* and the CIMS AUTHORIZE Utility on-line help.

CIMS for UNIX Collapsible Terminal File

The CIMS for UNIX Collapsible Terminal File (`A_terminals.par`) contains a list of terminals that are collapsed by the CIMS SELECT Utility when the system-wide `terminal_by_user` option is enabled. If this file is not present when you install CIMS for UNIX, it is created with these terminals:

ftp	x:	lta	xty	ptm
uucp	NT:	tty	pts	

This means all terminals starting with “ftp” (for example, “ftp1256”) are collapsed to simply “ftp”. This file is mandatory and cannot be deleted.

CIMS for NT/UNIX Interim Statistics Files

The CIMS for NT/UNIX Interim Statistics Files are a group of files containing formatted accounting records. All of these files reside in the `ARSAP_DATA` directory. These files are sorted together to create the CIMS Nightly Accounting File. This group of files consists of:

- CIMS for UNIX Login Statistics File (`A_lgstat.sys`)
- CIMS for UNIX Process Statistics File (`A_procstat.sys`)
- CIMS for UNIX Wtmp Statistics File (`A_wtmpstat.sys`)
- CIMS for NT Application Event Statistics File (`A_ntappstat.sys`)
- CIMS for NT Security Even Statistics File (`A_ntsecstat.sys`)

and, optionally,

- CIMS for UNIX Batch Statistics File (`A_batstat.sys`)
- CIMS for UNIX Print Statistics File. (`A_prtstat.sys`)
- CIMS for UNIX DB Statistics File (`A_dbstat.sys`).

Once the CIMS Nightly Accounting File has been created by the CIMS Nightly Script, these files can be removed. If you want these files to be automatically removed by the CIMS Nightly Script, set `CLEANUP_STATFILES=Y` in the CIMS Configuration Parameter File.

CIMS for NT/UNIX Holiday File

The CIMS for NT/UNIX Holiday File (`A_holiday.sys`) contains the dates of those days that you want to consider holidays. The options and parameters of this file are administered by the CIMS SETUP Utility. The file contains dates that are holidays in your company. Entering dates to this file is only required if you are going to charge different rates for holidays. This file is mandatory and cannot be deleted.

CIMS for NT/UNIX Image Mapping File

The CIMS for NT/UNIX Image Mapping File (`A_imgmap.sys`) contains the names of images that you want to track and packages they are associated with. The options and parameters of this file are administered by the CIMS SETUP Utility. This file is mandatory and cannot be deleted.

CIMS for NT/UNIX Intermediate Statistics File

The CIMS for NT/UNIX Intermediate Statistics File (`A_stat.sys`) is created by the CIMS SELECT Utility as a result of processing the CIMS Accounting File(s). This file is then input to the CIMS MERGE Utility, to produce the CIMS Merged Statistics File.

CIMS for UNIX Login Projects File

The CIMS for UNIX Login Projects File (`A_lgiproj.sys`) is created by the CIMS FORMAT Utility. It records the date/time, node name, user name, and project name for the initial session of each user detected in the most recently processed CIMS Activity File. It is an optional file for CIMS for UNIX.

CIMS for NT/UNIX Merged Statistics File

The CIMS for NT/UNIX Merged Statistics File (`A_stat.dat`) is created by the CIMS MERGE Utility as a result of processing the CIMS Intermediate Statistics File (`A_stat.sys`), the CIMS Storage File(s), and grouping information from the CIMS Authorization File, the UNIX Passwd File, and the UNIX Group File. This file is then used as input to the CIMS Report Utility.

CIMS for NT/UNIX Oracle Accounting File

The CIMS for NT/UNIX Oracle Accounting File (`A_dbacct.sys`) is a file created by the CIMS Oracle Accounting daemons. The file contains records of resource usage collected by the accounting daemon.

CIMS for NT/UNIX Oracle Basename File

The CIMS for NT/UNIX Oracle Basename File (`A_odb_base.sys`) is a file maintained by the CIMS SETUP and SELECT Utilities. The file contains a record for each Oracle instance being tracked by the CIMS Oracle Accounting daemons.

CIMS for NT/UNIX Oracle Username File

The CIMS for NT/UNIX Oracle Username File (`A_odb_user.sys`) is a file created by the CIMS SETUP and SELECT Utilities. The file contains a record for each Oracle username detected by the CIMS SELECT Utility while processing the CIMS Oracle Accounting File.

CIMS for UNIX Print Parameter File

The CIMS for UNIX Print Parameter File (`A_print.par`) contains a list of print accounting files associated with the various print queues. When print jobs and pages printed are to be accounted for, this is a required file for the CIMS for UNIX product to operate correctly and cannot be deleted.

CIMS for UNIX Queue Mapping File

The CIMS for UNIX Queue Mapping File (`A_queuemap.sys`) contains a list of batch and print queues. This file is maintained by the CIMS SETUP Utility. This is a mandatory file and cannot be deleted.

CIMS for NT/UNIX Rates File

The CIMS for NT/UNIX Rates File (`A_rates.sys`) contains rate tables of charge amounts that can be applied to user and project account resources. A rate table is a group of at least 8 rate records. At least one record exists for each jobtype: background, batch, interactive, package, print, storage, Oracle and DB2.

A record for a particular jobtype contains charge amounts for each of the resource statistics tracked for that jobtype. Additional records can be assigned to rate tables to define charge amounts that are applied for specific resource usage. For instance, you can define different charge amounts for interactive usage during different shifts or on a particular terminal. You can define different charge amounts for batch usage on different batch queues or print usage for jobs run on different printers.

■ The CIMS for NT/UNIX Files

CIMS for NT/UNIX Shift File

For first time installations, the CIMS Rates File is delivered with a default rate table, `DEFAULT_RATES` defined. The default rates table is used as a template when additional rate tables are added to the CIMS Rates File.

Each user and project account in the CIMS Authorization File contains a rate table field. This field defines the rate table that is used when charge amounts are applied to the resources used by the user or the project. Any rate table that exists in the CIMS Rates File can be assigned to a user or project account.

Prior to adding a rate record for a particular shift, terminal, package or database, the entity must already exist in the CIMS mapping file that defines that entity. For example, to add a rate record for a particular shift requires that the shift be defined in the CIMS Shift File. Any changes to the Shift File after defining a rate record for a particular shift could invalidate the rate records. For this reason, shifts should be established prior to defining rate tables.

Refer to the *CIMS Chargeback NT/UNIX User Guide* and CIMS RATE Utility on-line help.

CIMS for NT/UNIX Shift File

The CIMS for NT/UNIX Shift File (`A_shift.sys`) contains necessary shifts information. Several CIMS Utilities access this file to obtain the shift start time, shift end time, and the shift name of defined shifts.

For first time installations, the CIMS Shift File is created with four shifts defined: prime, non-prime, weekend, and holiday.

Other CIMS data files store offsets to the records contained in the CIMS Shift File. Any modifications to this file should be made before tailoring the other CIMS files in the `ARSAP_DATA` environment.

Refer to the *CIMS Chargeback NT/UNIX User Guide* and the CIMS SETUP Utility on-line help.

CIMS for NT/UNIX Storage File

The CIMS for NT/UNIX Storage File (`A_storage.sys`) contains the disk space information gathered by the CIMS for NT/UNIX Sampler Utility. This file is copied into a daily storage file (`str_YYYYMMDD.dat`, where YYYY is the year, MM is the month, DD is the day) by the CIMS for NT/UNIX Nightly script.

CIMS for UNIX Terminal Parameter File

The CIMS for UNIX Terminal Parameter File (`A_term_par.sys`) contains terminal names, groups, and factor values. This is a mandatory file and cannot be deleted.

CIMS for NT/UNIX Validation File

The CIMS for NT/UNIX Validation File (`A_validate.sys`) contains user and project name associations used to implement the user/project validation method. The file is maintained by the CIMS AUTHORIZE Utility. It is a mandatory file when the user/project validation method is used.

CIMS for UNIX X-Application Resource Files

Resource files have been provided for the CIMS for UNIX GUI and GRAPH Utilities. In order for this application to operate with desirable results, you should utilize these resource files.

These files are in the CIMS for UNIX examples directory and should be either moved or linked to your system X app-default directory. We recommend linking the files using the following commands:

```
cd /usr/lib/X11/app-defaults
ln -s /usr/arsap/examples/A_graph.rf A_graph
ln -s /usr/arsap/examples/A_gui.rf A_gui
```

CIMS for UNIX Report Description Files

The Report Description File (RDF) lets you define how each type of report is generated and what data is included in that report. You have control over what the report looks like and what data the report contains. The RDFs are: Resource Description File (rdf), Audit Description File (adf) and Capacity Description File (cdf).

An RDF consists of a select section, a sort section (rdf only) and a body section. The select section determines what is selected from the CIMS Merged Statistics File. The sort section defines ordering of the selected data. The body section defines the content and format of the generated report. Each section contains statements with one statement per line.

Select Section

The `SELECT` section determines the data selected from the CIMS Merged Statistics File. The select section contains select statements, specification statements and jobtype statements. The syntax for these statements is defined and must be followed exactly. This `SELECT` section must contain at least one select statement.

A `SELECT` statement consists of a select keyword and a select member list. The select keyword must be followed by a colon (:). The valid select keywords differ for each report type. All select keywords except categorization apply to resource (r) and audit (a) reports. The capacity select keyword is designated with a (c). The `SELECT` keywords are:

Select Keywords	Description
categorization (c only)	Categorization for report (see category table below)
db2_base	DB2 database name
db2_user	DB2 username (which is synonymous to user)

Select Keywords	Description
gid	GID (UNIX only)
node (c)	Nodes to be included
or_base	Oracle database name
or_user	Oracle user name
packagename	Software package name
project	Project name
project_cost_center	Project's cost center from CIMS for NT/UNIX Authorization File
project_description	Project's description field from CIMS for NT/UNIX Authorization File
project_group1	Project's group 1 field from CIMS for NT/UNIX Authorization File
project_group2	Project's group 2 field from CIMS for NT/UNIX Authorization File
project_group3	Project's group 3 field from CIMS for NT/UNIX Authorization File
project_group4	Project's group 4 field from CIMS for NT/UNIX Authorization File
project_group5	Project's group 5 field from CIMS for NT/UNIX Authorization File
queue	Batch or output queue name (UNIX only)
shift	Shift name
storage_group (r only)	Group name associated with storage on local platform

Select Keywords	Description
terminal	Terminal name (UNIX only)
u_gos	User's description field from UNIX passwd File
uid	User's UID (UNIX only)
user	Username (also used as DB2_user)
user_cost_center	User's cost center from CIMS for NT/UNIX Authorization File
user_description	User's description field from CIMS for NT/UNIX Authorization File
user_group1	User's group 1 field from CIMS for NT/UNIX Authorization File
user_group2	User's group 2 field from CIMS for NT/UNIX Authorization File
user_group3	User's group 3 field from CIMS for NT/UNIX Authorization File
user_group4	User's group 4 field from CIMS for NT/UNIX Authorization File
user_group5	User's group 5 field from CIMS for NT/UNIX Authorization File

For capacity reports, categorization should be one of:

hours	Report data by hour of the day. Hours are given in 24 hour format, for example, 23:00.
days	Report data by date, for example, 2-Feb-1997.
weekdays	Report data by day of the week, for example, Sunday.

weeks	Report data by week. Weeks start on Sunday, and only Sunday's date is reported, for example, 9-Mar-1997.
months	Report data by month, for example, March.
years	Report data by year, for example, 1998.
interval	Report data by the capacity interval defined by A_setup. This interval is expressed in minutes and can include any of the following time intervals: 1, 2, 3, 4, 5, 6, 10, 12, 15, 20, 30 or 60.

A select member list consists of one or more possible select members separated by commas, which can include wildcarded strings. The wildcard character is an asterisk (*). If spaces are used in the member list string, the string must be enclosed in quotes ("). A select member can be preceded by a minus sign (-), which omits the member. Any omissions must be the first entry in the member list. For example, the member list `user:-root,*` produces a report containing information for all users except root.

For summary resource reports (RDF) a line of data will be displayed for each unique combination of select keywords in the select section.

Example

```
user:*  
gid:*
```

This could produce two records for joe: one in which his GID is 10, and one in which his GID is 20.

A specification statement consists of a specification keyword and a specification member. The specification keyword must be followed by a colon (:). The specification keywords are:

delimiter A delimiter is any string of up to 12 printable characters. The string must be enclosed in quotes ("). This string delimits the output fields requested by the user in the body section. The starting columns specified in the body section are ignored (see *Body Section* on page 5-19). This mechanism is useful when you need to produce a flat file for input to another source. For example, the CIMS for NT/UNIX Graph Utility produces a pound sign (#) delimited file that is used to produce the graphical report.

rate_table A rate table is any string of up to 32 characters representing the rate table that is applied to the accounting statistics. You can designate either a specific rate table, a user-specific rate table, a project specific rate table, or the DEFAULT_RATES rate table (refer to *Rates Set-Up* on page 2-49, of the *CIMS Chargeback NT/UNIX User Guide*).

If `rate_table:user` is specified, the rate table indicated in the user's record in the CIMS Authorization File is used. If a user does not have a rate table assigned, the DEFAULT_RATES rate table is used.

If `rate_table:project` is specified, the rate table indicated in the project's record in the CIMS Authorization File is used. If the project does not have a rate table, the DEFAULT_RATES rate table is used.

If any other value for the rate table is specified, the CIMS for NT/UNIX Rates File is searched for a matching rate table. If the rate table is not found or the `rate_table` keyword is not used, the `DEFAULT_RATES` rate table from the CIMS Rates File is used.

A jobtype statement consists of a jobtype keyword. The jobtype keyword must be followed by a colon (:). The jobtype keywords are:

background	include all background job records
batch	include all batch job records
interactive	include all interactive job records
oracle	include all Oracle job records
db2	include all DB2 job records
package	include all software package records
print	include all print records
storage	include all storage records

If jobtype statement is not included, the default jobtypes are used (that is, interactive, background, and storage).

An example select section follows:

```
select
user:*
delimiter:", "
rate_table:user
batch:
interactive:
```

This example selects all the batch and interactive records for all users, and excludes the other types of records (background, package, print, and storage), output fields are delimited by a comma, and the user account rate table applies to charges.

Sort Section

The sort section controls how the data that has been selected is sorted and pertains only to a summary resource type (rdf).

If no sort section is present, the data is sorted in ascending order according to the select keyword series. If a sort section is present, data is sorted in the order specified by the sort order keyword for each sort statement according to the sort statement series. For either select keyword or sort statement series, the first entry indicates the primary sort field, the second indicates the secondary sort field, the third is tertiary and so on. You cannot use a keyword in the sort section unless you have also used it in the select section.

A sort statement consists of a sort keyword and a sort order keyword. The valid sort keywords are the same as the select keywords used in the select section. The sort keyword must be followed by a colon (:). The sort order keywords are:

ascending sort the keyword field in ascending order

descending sort the keyword field in descending order

An example sort section follows:

```
sort
user:ascending
project:descending
```

This example sorts the usernames first in ascending order, followed by project names in descending order.

Body Section

The body section controls what is printed out in the report and the format of output. Each body statement describes how data fields are to be placed on the output line. The body statement consists of four required fields and a fifth optional field (separated by a colon (:)) as follows:

Field	Description
1	body keyword
2	row number (future use, always 0)
3	starting column (columns start at 0, not 1)
4	field length
5	attribute

As a rule, unless a select keyword has been used in the select section, do not print it in the body section. An exception to this rule occurs when there is a one-to-one correspondence between the select keyword and body keyword. For example, if the `user_description` field has been used to store the user's full name and you have selected `user` (UNIX username), but you want to print the `user_description` field on the report.

The valid body keywords include all the select keywords listed in the select section above, a special `text` keyword, plus these statistical keywords:

Statistic	Explanation
<code>block_weeks (r only)</code>	storage block weeks
<code>cap_interval (c only)</code>	capacity time interval
<code>cio</code>	character I/O in units of 100,000

Statistic	Explanation
connect_time	connect (seat) time in hours
db2_commit_sql_stmts	IBM DB2 SQL commit statements
db2_connect	IBM DB2 connect time in hours
db2_deadlocks_rollback	IBM DB2 deadlocks
db2_direct_reads	IBM DB2 direct reads
db2_direct_writes	IBM DB2 direct writes
db2_int_deadlock_rollback	IBM DB2 internal rollback due to deadlocks
db2_lock_wait_time	IBM DB2 lock wait minutes
db2_logins	IBM DB2 logins
db2_pd_lreads	IBM DB2 buffer pool data logical reads
db2_pd_preads	IBM DB2 buffer pool data physical reads
db2_pd_writes	IBM DB2 buffer pool data writes
db2_pi_lreads	IBM DB2 buffer pool index logical reads
db2_pi_preads	IBM DB2 buffer pool index physical reads
db2_pi_writes	IBM DB2 buffer pool index writes
db2_rollback_sql_stmts	IBM DB2 SQL rollback statements attempted
db2_rows_deleted	IBM DB2 rows deleted
db2_rows_inserted	IBM DB2 rows inserted
db2_rows_selected	IBM DB2 rows selected

Statistic	Explanation
db2_rows_updated	IBM DB2 rows updated
db2_scpu	IBM DB2 system CPU time in minutes
db2_sort_overflows	IBM DB2 sort overflows
db2_total_sorts	IBM DB2 sorts
db2_ucpu	IBM DB2 user CPU time in minutes
db2_uow_log_space_used	IBM DB2 unit of work log space used in bytes
dio	disk block I/O in units of 1000
end_time (r,a)	reporting period end date
image_count	number of images executed
image_time	image time in hours
jobtype (a only)	session jobtype (see below)
logins	number of logins
memory	memory utilization
or_connect	Oracle connect time in hours
or_disk_sorts	Oracle disk sorts
or_logins	Oracle logins
or_messages_received	Oracle message received
or_messages_sent	Oracle message sent
or_pga_memory	Oracle PGA memory
or_physical_reads	Oracle physical reads

Statistic	Explanation
or_physical_writes	Oracle physical writes
or_rec_cpu	Oracle recursive CPU time in minutes
or_sess_cpu	Oracle session CPU time in minutes
or_uga_memory	Oracle UGA memory
or_user_commits	Oracle user commits
or_write_request	Oracle write requests
pages	pages printed
prt_jobs	number of print jobs
scpu	system CPU time in minutes
start_time (r,a)	reporting period start date
su_count	number of su commands
su_image_count	images run as su
su_time	su time in hours
tcpu	total CPU time in minutes
ucpu	user CPU time in minutes
win_time	window time in hours
c_block_weeks	block weeks charge
c_charge1_db2_pd_lreads	IBM DB2 buffer pool data logical reads charge
c_cio	character I/O charge

Statistic	Explanation
c_connect_time	connect time charge
c_db2_commit_sql_stmts	IBM DB2 SQL commit statement charge
c_db2_connect	IBM DB2 connect time charge
c_db2_deadlocks_rollbacks	IBM DB2 deadlocks charge
c_db2_direct_reads	IBM DB2 direct reads charge
c_db2_direct_writes	IBM DB2 direct writes charge
c_db2_int_deadlock_rollbacks	IBM DB2 internal rollback due to deadlocks charge
c_db2_lock_wait_time	IBM DB2 lock wait minutes charge
c_db2_logins	IBM DB2 logins charge
c_db2_pd_preads	IBM DB2 buffer pool data physical reads charge
c_db2_pd_writes	IBM DB2 buffer pool data writes charge
c_db2_pi_lreads	IBM DB2 buffer pool index logical reads charge
c_db2_pi_preads	IBM DB2 buffer pool index physical reads charge
c_db2_pi_writes	IBM DB2 buffer pool index writes charge
c_db2_rollback_sql_stmts	IBM DB2 SQL rollback statements attempted charge
c_db2_rows_deleted	IBM DB2 rows deleted charge
c_db2_rows_inserted	IBM DB2 rows inserted charge
c_db2_rows_selected	IBM DB2 rows selected charge

Statistic	Explanation
c_db2_rows_updated	IBM DB2 rows updated charge
c_db2_scpu	IBM DB2 system CPU time minutes charge
c_db2_sort_overflows	IBM DB2 sort overflows charge
c_db2_total_sorts	IBM DB2 sorts charge
c_db2_ucpu	IBM DB2 user CPU time minutes charge
c_db2_uow_log_space_used	IBM DB2 unit of work log space used charge
c_dio	disk block I/O charge
c_image_count	images executed charge
c_image_time	image time charge
c_logins	logins charge
c_memory	mem utilization charge
c_or_connect	Oracle connect time charge
c_or_disk_sorts	Oracle disk sorts charge
c_or_logins	Oracle logins charge
c_or_messages_received	Oracle message received charge
c_or_messages_sent	Oracle message sent charge
c_or_pga_memory	Oracle PGA memory charge
c_or_physical_reads	Oracle physical reads charge
c_or_physical_writes	Oracle physical writes charge

Statistic	Explanation
<code>c_or_rec_cpu</code>	Oracle recursive CPU charge
<code>c_or_sess_cpu</code>	Oracle session CPU charge
<code>c_or_uga_memory</code>	Oracle UGA memory charge
<code>c_or_user_commits</code>	Oracle user commits charge
<code>c_or_write_request</code>	Oracle write requests charge
<code>c_pages</code>	pages printed charge
<code>c_prt_jobs</code>	print jobs charge
<code>c_scpu</code>	system CPU charge
<code>c_su_count</code>	number of su's charge
<code>c_su_image_count</code>	su images charge
<code>c_su_time</code>	su time charge
<code>c_tcpu</code>	total CPU charge
<code>c_ucpu</code>	user CPU charge
<code>c_win_time</code>	window time charge

Charge body keywords have the form `c_<keyword>`. In this case the statistics for that `<keyword>` are multiplied by the rates (which can differ for different jobtypes), and the resulting number is printed with a monetary symbol. The monetary symbol must be factored into the length of the field you want to print. Charges do not pertain to capacity reports.

The behavior of the `start_time` and `end_time` is dependent on the report. In a resource report, they are the beginning and the end of the selection period, respectively. In an audit report, they are the beginning and the end of each individual session, respectively.

The special `text` keyword can be used to insert constant text into each line of the output report. The `text` keyword can be used like this:

```
text="string":0:10:16:left
```

This entry specifies to output the text *string* padded with spaces left-justified starting at column ten.

The attribute list consists of a list of attribute separated by commas. The valid attributes and their meaning include:

Attribute	Meaning
right	Right justify the field
left	Left justify the field
mon_sym	Include monetary symbols from the CIMS Parameter File. Present by default for charge type data. Valid with charge and statistical type data.
nomon_sym	No monetary symbols. Valid only with charge and statistical type data.
digit_sep	Include digit separator from CIMS Parameter File. Valid only with charge and statistical type data.
unix_d	Valid with time type only. Day Mon DD YYYY
unix_dt	Valid with time type only. Day Mon DD HH:MM:SS YYYY
vms_d	Valid with time type only. DD-MMM-YYYY
vms_dt	Valid with time type only. DD-HH-YYYY HH:MM:SS

Attribute	Meaning
sort_d	Valid with time type only. YYYYMMDD
sort_dt	Valid with time type only. YYYYMMDDHHMMSS

An example body section follows:

```
body
  user:0:10:15:right
  ucpu:0:30:15:digit_sep,right
  text="Start date = ":0:50:15:left
  start_time:0:65:11:vms_d
```

This example produces output with: fifteen (15) characters for the username, which is right-justified and starts at column ten (10); fifteen (15) characters for user CPU using digit separators, which is right-justified and starts at column thirty (30); fifteen (15) characters for user-defined text string, which is left-justified and starts at column fifty (50); and eleven (11) characters for **start_time** in VMS date format, which starts at column sixty-five (65).

Report Description File Considerations

You must pay careful consideration to what you ask for when generating the reports.

If you use a select keyword in the body section, you must add those keywords to the select section. For example, if the project is to be printed in the report, the keyword `project` should appear in the select section as well as the body section of the report description file. If you fail to add the project keyword to the select section, the project is not included in selected data. Therefore, the reported projects are taken on a first-encountered basis, which might result in erroneous project association.

You must be careful that double statistics are not generated. Image records are either from background or interactive sessions. When you generate a report and specify a background jobtype, you get statistics only from image records for background work. When you specify an interactive jobtype, you get statistics only from image records for interactive work. However, if you specify both jobtypes, you get statistics from both types of records.

As a consequence, if you have mapped image names to software packages, when a report is generated in which the package jobtype is specified, you get statistics only from those mapped image records, whether they are from background or interactive sessions.

If you generate a report in which you specify jobtypes background, interactive, and package. You get statistics for all images either as part of the background or interactive jobtypes, and the mapped image records again as a software package. Never specify the package jobtype if you have also specified either the background or interactive jobtype.

Report Description File Examples

The following is an example of a resource description file (*rdf*), *A_rpt_desc.sys*. This *rdf* is the default resource description file, which is located in the *ARSAP_EXAMPLES* directory:

```
select
    node:*
    user:*
    interactive:
    background:
    storage:
sort
    node:ascending
    user:ascending
body
    node:0:0:8
    user:0:9:9
    logins:0:19:5
```

```
connect_time:0:27:6
tcpu:0:34:6
block_weeks:0:41:12
```

The output report of this example is:

sparky	arsap	25.00	1.017	0.180	78538.141
sparky	dlord	27.00	25.147	0.606	33006.480
sparky	epolidan	10.00	45.849	0.856	38075.039
sparky	ernest	11.00	27.099	1.912	163452.422
sparky	george	204.0	34.138	107.96	557770.000
sparky	mchrste	13.00	8.231	3.247	14844.699
sparky	root	13507	1.698	175.83	747361.312
sparky	shannan	14.00	9.501	0.033	2322.460
underdog	arsap	15.00	0.028	0.041	98402.977
underdog	db2user	0.000	0.000	0.000	401132.031
underdog	epolidan	3.000	3.685	0.020	0.000
underdog	ernest	793.0	544.19	148.76	146.140
underdog	george	14.00	7.229	31.558	93642.367
underdog	mchrste	8.000	5.689	0.142	41.140
underdog	root	16324	0.760	26.968	667578.625

The following is an example of an audit description file (adf), `A_audit_desc.sys`. This adf is the default audit description file, which is located in the `ARSAP_EXAMPLES` directory:

```
select
  user:*
  interactive:
  background:
body
  user:0:0:8
  node:0:9:8
  start_time:0:18:16:unix_dt
  end_time:0:35:16:unix_dt
  connect_time:0:52:6
  win_time:0:59:6
  tcpu:0:66:6
```

The output report of this example is:

shannan	ralph	Mon Mar 10	08:52	Mon Mar 10	17:17	8.411	0.000	0.000
epolidan	ralph	Mon Mar 10	11:20	Mon Mar 10	17:29	6.148	0.000	0.000
dlord	ralph	Tue Mar 11	08:55	Tue Mar 11	18:00	9.071	0.000	0.000
epolidan	ralph	Tue Mar 11	09:32	Tue Mar 11	18:31	8.995	0.000	0.000
mchryste	ralph	Tue Mar 11	11:04	Tue Mar 11	18:33	7.491	0.000	0.000
ernest	ralph	Tue Mar 11	22:56	Wed Mar 12	04:21	5.403	0.000	0.317
george	ralph	Wed Mar 12	08:37	Wed Mar 12	13:57	0.000	5.340	0.023
ernest	ralph	Wed Mar 12	10:16	Wed Mar 12	17:07	6.845	0.000	0.032
epolidan	ralph	Wed Mar 12	09:11	Wed Mar 12	17:28	8.297	0.000	0.017
george	ralph	Wed Mar 12	08:36	Wed Mar 12	17:37	9.011	0.000	0.000
george	ralph	Wed Mar 12	09:46	Wed Mar 12	17:36	0.000	7.833	4.577
dlord	ralph	Thu Mar 13	09:00	Thu Mar 13	09:37	0.602	0.000	0.023
george	ralph	Thu Mar 13	08:52	Thu Mar 13	17:10	8.299	0.000	0.000
ernest	ruff	Tue Mar 11	19:27	Wed Mar 12	09:09	0.000	0.000	50.954
ernest	ruff	Tue Mar 11	08:53	Tue Mar 11	18:31	9.640	0.000	0.047
mchryste	ruff	Thu Mar 13	15:28	Thu Mar 13	17:57	2.483	0.000	0.027
epolidan	ruff	Fri Mar 14	16:30	Fri Mar 14	18:21	1.848	0.000	0.006
epolidan	ruff	Fri Mar 14	16:31	Fri Mar 14	18:21	1.837	0.000	0.013
ernest	ruff	Mon Mar 10	00:00	Wed Mar 12	17:07	65.129	0.000	0.008
george	ruff	Mon Mar 17	10:19	Mon Mar 17	17:12	6.886	0.000	27.459
george	ruff	Mon Mar 17	10:19	Mon Mar 17	17:21	0.151	0.000	4.075
ernest	ruff	Mon Mar 10	00:00	Wed Mar 12	17:07	55.784	0.000	0.687
ernest	ruff	Mon Mar 17	09:17	Mon Mar 17	17:32	0.000	8.244	0.017

The following is an example of a capacity description file (cdf), A_cap_desc.sys. This cdf is the default capacity description file, which is located in the ARSAP_EXAMPLES directory:

```
select
  node:*
  categorization: hours
  interactive:
  background:
body
  cap_interval:0:0:5
  tcpu:0:6:8
  connect_time:0:15:8
```

The output report of this example is:

00:00	14.866	25.000
01:00	14.950	25.000
02:00	23.233	23.004
03:00	15.600	23.000
04:00	15.139	22.353
05:00	14.860	22.000
06:00	14.863	22.000
07:00	14.863	22.000
08:00	16.207	24.376
09:00	25.934	36.496
10:00	28.844	41.717
11:00	29.658	46.708
12:00	29.540	47.666
13:00	28.341	47.106
14:00	11.157	47.466
15:00	11.392	51.072
16:00	11.530	50.717
17:00	12.252	35.601
18:00	7.643	20.993
19:00	61.039	17.244
20:00	56.868	17.383
21:00	12.815	18.136
22:00	11.048	18.321
23:00	11.032	18.178

CIMS for UNIX Graph Description File

The CIMS for UNIX Graph Utility creates and reads graph description files (GDF). A GDF contains graph contents, titles, and other formatting information. By creating a GDF, you can automatically generate the same type of graph each reporting period. The formatting keyword descriptions are described below. The formatting keywords are case sensitive.

An example of an CIMS for UNIX GDF that displays user CPU charges is:

```
Batch: TRUE
Background: TRUE
Depth: 10
```

Footer:
Header: Comparison of charges for user CPU and system CPU usage
Inclination: 10
Invert: FALSE
Interactive: TRUE
KeepZero: TRUE
Legend: TRUE
Member: *
Package: TRUE
PieMin: 0.0001
Print: TRUE
RateTable: user
Rotation: 10
Sort: Ascending
Storage: TRUE
Transpose: FALSE
Type: Bar
XGrid: 0.0
XTitle:
X: user
YGrid: 0.0
YTitle:
Y: c_ucpu
Y: c_scpu

Note • GRAPH ignores any errors in a CIMS UNIX GDF. However, you should not modify a GDF outside of CIMS for UNIX GRAPH Utility.

Jobtype Formatting Keywords

Jobtype formatting keywords indicate which job types should be included in the report.

Keyword Syntax
Batch: <status>
Background: <status>
Interactive: <status>
Package: <status>
Print: <status>
Storage: <status>

The valid values for <status> are:

TRUE This jobtype is included.

FALSE This jobtype is not included.

Depth

The Depth formatting keyword specifies the depth of the chart. Increasing this value causes the vertical axis to appear wider.

Depth: <value>

where <value> is a number between 0 and 500, inclusive.

Footer

The Footer formatting keyword specifies the string that is displayed beneath the chart.

Footer: <value>

where <value> is the footer string. It can contain blanks and should *not* be enclosed in quotes.

Header

The Header formatting keyword specifies the string that is displayed above the chart.

Header: <value>

where <value> is the header string. It can contain blanks and should *not* be enclosed in quotes.

Inclination

The Inclination formatting keyword specifies the number of degrees to rotate the chart around the Z axis. The Z axis is not displayed on the chart. Increasing this value causes the graph to rotate up from the horizon.

Inclination: <value>

where <value> is a number between 0 to 45, inclusive.

Invert

The Invert formatting keyword indicates whether the data should be inverted on the axis.

Invert: <value>

where <value> is:

TRUE Data is displayed along the vertical axis.

FALSE Data is displayed along the horizontal axis.

KeepZero

The KeepZero formatting keyword indicates whether the records with zero statistics is displayed.

KeepZero: <value>

where <value> is:

TRUE Display all records matching selection criteria.

FALSE Display only records matching selection criteria that have non-zero statistic.

Legend

The Legend formatting keyword indicates whether a legend is displayed beneath the chart.

Legend: <value>

where <value> is:

TRUE Display the legend.

FALSE Do not display the legend.

Member

The Member formatting keyword specifies the selection criteria for the X axis variable that a record must satisfy to be included in the chart.

Member: <value>

where <value> is the selection string. It can contain wildcards and should *not* be enclosed in quotes.

PieMin

The PieMin formatting keyword indicates the minimum percentage that is displayed on a pie chart.

PieMin: <value>

where <value> is a number between 0 and 100, inclusive.

RateTable

The RateTable formatting keyword specifies the rates table that is applied to chargeable statistics on the chart.

RateTable: <value>

where <value> is user, project, or the name of a rate table defined in the CIMS for NT/UNIX Rates File.

Rotation

The Rotation formatting keyword specifies the number of degrees to rotate the chart around the Y axis.

Rotation: <value>

where <value> is a number between 0 to 45, inclusive.

Sort

The Sort formatting keyword specifies how the data is sorted. The sorting criteria is applied to the X axis variable.

Sort: <value>

where <value> is:

Ascending Sort data in ascending alphabetical order.

Descending Sort data in descending alphabetical order.

Transpose

The Transpose formatting keyword indicates whether the data is grouped by the X Axis variable or by the Y Axis variables.

Transpose: <value>

where <value> is:

TRUE Group data by Y Axis variable

FALSE Group data by X Axis variable

Type

The Type formatting keyword indicates how the data is represented.

Type: <value>

Where <value> is:

Bar	Display data in a bar chart
Area	Display data in an area graph
Pie	Display data in a pie chart
Plot	Display data in a plot graph
StackedBar	Display data in a stacked bar chart

Xgrid

The XGrid formatting keyword specifies the interval between grid lines along the X Axis.

XGrid: <n>

where <n> is a floating point number. The default is 0.0 that does not display grid lines.

XTitle

The XTitle formatting keyword specifies the string to be used as the label for the X Axis.

XTitle: <value>

where <value> is the string to use as the X Axis label. It can contain blanks and should *not* be enclosed in quotes.

X

The X formatting keyword specifies the value to be used as the X Axis variable. You must indicate exactly one X formatting keyword.

X: <item>

where <item> is a selection keyword (refer to [Select Section](#) on page 5-12).

YGrid

The YGrid formatting keyword specifies the interval between grid lines along the Y Axis.

YGrid: <n>

where <n> is a floating point number. The default is 0.0 that does not display grid lines.

YTitle

The YTitle formatting keyword specifies the string to be used as the label for the Y Axis.

YTitle: <value>

where <value> is the string to use as the Y Axis label. It can contain blanks and should *not* be enclosed in quotes.

Y

The Y formatting keyword specifies the value(s) to be used as the Y Axis variable(s). You must indicate at least one Y formatting keyword, each on a separate line.

Y: <item>

where <item> is a statistics keyword (refer to *Body Section* on page 5-19).

 **6**

CIMS for NT/UNIX Command Line Interface (CLI) and Commands

CIMS for NT/UNIX Command Line Interface (CLI)	6-4
Licensing	6-5
User Accounting	6-5
CIMS for NT/UNIX DEFAULT User Record	6-5
Enabling User Accounting	6-6
Adding User Accounts	6-6
User Shell	6-7
User Cost Center	6-7
User Groups	6-8
User Description	6-9
Project Accounting	6-9
CIMS for NT/UNIX DEFAULT Project Record	6-9
Enabling Project Accounting	6-10
Adding Project Accounts	6-11

User Default Project	6-11
User Default Jobtype Projects	6-12
Custom Project Prompting	6-12
Project Switching	6-14
Project GID	6-14
Project Default Directory	6-15
Project Cost Center	6-16
Project Groups	6-16
Project Description	6-17
Project Time Limit	6-18
Project Validation	6-18
Image Accounting	6-22
Enabling Proprietary Software	6-22
Defining Software Packages	6-23
Database Accounting	6-23
Reporting Parameters	6-25
Capacity Interval	6-26
Rates	6-27
Shifts	6-29
Holidays	6-30
Output Queue	6-31
GID Method	6-32
Report Header	6-32
Digit Separator	6-32
Memory Demand Algorithm	6-33
Monetary Symbols	6-34
Radix Point	6-34
Data Formatting	6-35
CIMS for UNIX Print Accounting Script	6-35
CIMS for NT/UNIX Runacct Script	6-36

Processing Multitude Nodes6-39

CIMS for NT/UNIX Command Line Interface (CLI)

CIMS provides two user interfaces: a Graphical User Interface (GUI) for UNIX and Command Line Interface (CLI) for both NT and UNIX. The CIMS for UNIX GUI provides a user-friendly interface to handle all CIMS for UNIX operational phases. The CIMS CLI provides a more direct interface with the CIMS utilities. This manual presents the use of the CIMS CLI for each of the resource management phases.

The CIMS CLI typically requires expert knowledge of the CIMS Utilities, however, useful online help is available for each utility. In order to use online help, the environment variable `ARSAP_HELP` must be defined to the "help" directory where you installed CIMS. Each CIMS Utility CLI provides a prompt with the appropriate utility name. For example, the prompt for the CIMS SETUP Utility is `A_SETUP>`.

The online help for a particular CIMS Utility is accessed by typing `help` or `?` at the utility prompt. The help pages are hierarchical. The initial page presents the top-level commands with subsequent pages describing the applicable qualifiers and associated arguments.

Each CIMS CLI consists of a command followed by qualifiers associated with the command. You do not have to enter the entire command and qualifiers to the CLI. The parsing mechanism lets partial information identify a unique portion of the command or qualifier. For example, the command and qualifier to display database instance settings in the CIMS SETUP Utility is "REPORT/DBINSTANCE". This can be abbreviated to "REP/DBINST".

The CIMS CLI lets you invoke CIMS utilities from scripts and background jobs. To accomplish your desired task you can invoke a CIMS utility from a script with the necessary command and qualifiers. The CIMS operational and administrative scripts utilize the CIMS CLI in this way.

Licensing

CIMS provides scripts to manage the license information through the CIMS CLI. These scripts are located in `CIMS_HOME/scripts/admin` directory. There are scripts to add and update the licenses that manage the entire process and build the appropriate CLI commands.

You can update or add more licenses at any time. You can manage the CIMS license PAKs through the CIMS for UNIX GUI or the CIMS CLI.

The CIMS CLI commands you can use in the CIMS License Utility (`G_license`) are:

- **ADD**—to add a new license
- **REMOVE**—to remove license(s)
- **REPORT**—to report on license(s)

User Accounting

CIMS for NT/UNIX DEFAULT User Record

Modify the DEFAULT user account through the CIMS CLI by performing a command similar to the one that follows:

```
# A_authorize
A_AUTHORIZE> DEFAULT/USER/DISABLE=PROJECT_ACCOUNTING
%AUTHORIZE-S-MODIFIED, 1 USER records successfully modified
A_AUTHORIZE> EXIT
#
```

The DEFAULT user record in the CIMS Authorization File is used as a template when adding new user records. Values for the fields in the DEFAULT user record are used as defaults when new user records are added. In the example above, the `PROJECT_ACCOUNTING` flag of the DEFAULT user record is set to "Disabled". Now any new user record added to the CIMS Authorization File will have the `PROJECT_ACCOUNTING` flag set to "Disabled".

Enabling User Accounting

Enable the system-wide user accounting option through the CIMS CLI as follows:

```
# A_setup
A_SETUP> DEFAULT/ENABLE=USER_ACCOUNTING
%SETUP-S-MODIFIED, 1 node record successfully modified
A_SETUP> EXIT
```

In the example above, the `USER_ACCOUNTING` flag in the CIMS Parameter File is set to "Enabled". This flag indicates if accounting is to be done based on the operating system username. This is usually the case and this flag is normally "Enabled" by default.

Adding User Accounts

Add a new user account through the CIMS CLI by performing a command similar to the one that follows:

```
# A_authorize
A_AUTHORIZE> ADD hjones
%AUTHORIZE-S-ADDED, USER record successfully added
A_AUTHORIZE> EXIT
#
```

Any user options and parameters not specified when adding a new user account are copied from the `DEFAULT` user account.

To modify an option or parameter in a user record that already exist:

```
# A_authorize
A_AUTHORIZE> MODIFY/SHELL="/bin/csh" hjones
%AUTHORIZE-I-MODIFIED, 1 USER record successfully modified
A_AUTHORIZE> EXIT
#
```

User Shell

Specify the user shell when adding a new user account through the CIMS CLI by performing a command similar to the one that follows:

```
# A_authorize
A_AUTHORIZE> ADD hjones/SHELL="/bin/csh"
%AUTHORIZE-S-ADDED, USER record successfully added
A_AUTHORIZE> EXIT
#
```

In this example, a record for user, `hjones` is added to the CIMS Authorization File. In addition to adding the record, the `SHELL` field for the user is set to `/bin/csh`. The `SHELL` field indicates the shell that should be exec'ed for the user when logging in through the CIMS LOGIN Utility, (`A_login`), on UNIX platforms. This field is present, but unused on NT platforms.

User Cost Center

Specify the user cost center when adding a new user account through the CIMS CLI by performing a command similar to the one that follows:

```
# A_authorize
A_AUTHORIZE> ADD hjones/COST_CENTER="DEVL"
%AUTHORIZE-S-ADDED, USER record successfully added
A_AUTHORIZE> EXIT
#
```

In this example, a record for user, `hjones` is added to the CIMS Authorization File. In addition to adding the record, the `COST_CENTER` field for the user is set to the string `DEVL`. When generating a report of `user_cost_centers`, usage consumed by `hjones` will be included in that associated with cost center `DEVL`. A `COST_CENTER` can be any string up to 32 characters long.

User Groups

Specify the user groups when adding a new user account through the CIMS CLI by performing a command similar to the one that follows:

```
# A_authorize
A_AUTHORIZE> ADD hjones/GRP1="Laurel"
%AUTHORIZE-S-ADDED, USER record successfully added
A_AUTHORIZE> EXIT
#
```

In this example, a record for user, `hjones` is added to the CIMS Authorization File. In addition to adding the record, the `GRP1` field for the user is set to the string `Laurel`. When generating a report of select keyword, (see *CIMS for UNIX Report Description Files* on page 5-12), `user_group1`, usage consumed by `hjones` will be included in that associated with the user group `Laurel`. There are five user group fields in each user record, `GRP1`, `GRP2`, `GRP3`, `GRP4`, and `GRP5`. The value contained in `GRP2` is used when reporting on select keyword `user_group2`, in `GRP3` when reporting on `user_group3` and so on. In this way, hierarchical groupings can be implemented. For instance, `GRP1` in each user record may contain the user's department, `GRP2` the user's division, `GRP3` the user's sub-division and so on. Any of the grouping fields can be up to 32 characters long.

User groupings are applied to usage generated for all jobtypes based on the operating system username. At some sites, usage for the ORACLE jobtype needs to be grouped based on the ORACLE username. The `DB_USER_GROUPING` feature provides a means of associating ORACLE usage with a group based on the ORACLE username. When you want to use the database user grouping feature, you need to enable the system-wide database user grouping option.

Enable the system-wide database user grouping through the CIMS CLI as follows:

```
#A_setup
A_SETUP>DEFAULT/ENABLE=DB_USER_GROUPING
%SETUP-S-MODIFIED, 1 node record successfully modified
A_SETUP>EXIT
```

User Description

Specify the user description when adding a new user account through the CIMS CLI by performing a command similar to the one that follows:

```
# A_authorize
A_AUTHORIZE> ADD hjones/DESCRIPTION="System Administrator"
%AUTHORIZE-S-ADDED, USER record successfully added
A_AUTHORIZE> EXIT
#
```

In this example, a record for user, `hjones` is added to the CIMS Authorization File. In addition to adding the record, the `DESCRIPTION` field for the user is set to the string "System Administrator". When generating a report including `select keyword`, (see [CIMS for UNIX Report Description Files](#) on page 5-12), `user_description`, "System Administrator" will be displayed. In addition to providing a description for the user, the `DESCRIPTION` field can be used as a grouping field as described in the previous example. The `DESCRIPTION` field can be up to 32 characters long.

Project Accounting

CIMS for NT/UNIX DEFAULT Project Record

Modify the DEFAULT project account through the CIMS CLI by performing a command similar to the one that follows:

```
# A_authorize
A_AUTHORIZE> DEFAULT/PROJECT/DISABLE=SWITCH_GID
%AUTHORIZE-S-MODIFIED, 1 PROJECT records successfully modified
A_AUTHORIZE> EXIT
#
```

The DEFAULT project record in the CIMS Authorization File is used as a template when adding new project records. Values for the fields in the DEFAULT user record are used as defaults when new project records are added. In the example above, the `SWITCH_GID` flag of the DEFAULT project record is set to "Disabled". Now any new project record added to the

Project Accounting

CIMS Authorization File will have the SWITCH_GID flag set to "Disabled". The SWITCH_GID flag is used to indicate if the user's effective GID should be changed to the project's GID when the user logs on to or switches to the project. In order to change the effective GID of the user, the SWITCH_GID flag in the user's record must also be set to "Enabled".

In this way, you can change the GID for specific users who access the project.

Enabling Project Accounting

Enable the system-wide project accounting option through the CIMS CLI as follows:

```
# A_setup
A_SETUP> DEFAULT/ENABLE=PROJECT_ACCOUNTING
%SETUP-S-MODIFIED, 1 node record successfully modified
A_SETUP> EXIT
#
```

Enable or disable the user account project accounting option when modifying a user account through the CIMS CLI by performing a command similar to the one that follows:

```
# A_authorizeA_AUTHORIZE> MODIFY/USER wdoe-
_A_AUTHORIZE> /ENABLE=PROJECT_ACCOUNTING
%AUTHORIZE-S-MODIFIED, 1 USER record successfully modified
A_AUTHORIZE> EXIT
#
```

Project accounting can be enabled for some users and not for others. In order for usage consumed by a particular user to be applied to a project, PROJECT_ACCOUNTING must be enabled system-wide in the CIMS Parameter File and in the user's record in the CIMS Authorization File.

Adding Project Accounts

Add a new project through the CIMS CLI by performing a command similar to the one that follows:

```
# A_authorize
A_AUTHORIZE> ADD/PROJECT sales -
_A_AUTHORIZE> /ENABLE=PROJECT_VALIDATION-
_A_AUTHORIZE> /PASSWORD="loophole"-
_A_AUTHORIZE> /VALIDATION_METHOD=PASSWORD
%AUTHORIZE-S-ADDED, PROJECT record successfully added
A_AUTHORIZE> EXIT
#
```

In the example above, a project record for project "sales" is added to the CIMS Authorization File. PROJECT_VALIDATION is enabled, VALIDATION_METHOD is PASSWORD and the project's PASSWORD is loophole. All other fields in this record will be copied from the DEFAULT project record. When implemented in this way, any user trying to login or switch to this project will be prompted for a password to which they will need to respond, loophole. (See Project Validation in the CIMS NT/UNIX User Guide).

User Default Project

Implement a user default project through the CIMS CLI by performing a command similar to the one that follows:

```
# A_authorize
A_AUTHORIZE> MODIFY/USER ysmith-
_A_AUTHORIZE> /DEFPROJECT=sales/ENABLE=-
_A_AUTHORIZE> (PROJECT_ACCOUNTING, -
_A_AUTHORIZE> DEFAULT_PROJECT)
%AUTHORIZE-S-MODIFIED, 1 USER records successfully modified
A_AUTHORIZE> EXIT
#
```

In this example, the user record for user ysmith is modified so that PROJECT_ACCOUNTING and DEFAULT_PROJECT are enabled and the DEFPROJECT field is set to sales. In this way, when user ysmith logs onto the system, their usage is automatically associated with the project sales

without the user being prompted to enter a project. During data consolidation, any usage consumed by user `ysmith` that has not been associated with a project will be assigned to the project `sales`.

User Default Jobtype Projects

Designate a user default jobtype project through the CIMS CLI by performing a command similar to the one that follows:

```
# A_authorize
A_AUTHORIZE> MODIFY/USER ysmith-
_A_AUTHORIZE> /INTER_PROJECT=interactive
%AUTHORIZE-S-MODIFIED, 1 USER records successfully modified
A_AUTHORIZE> EXIT
#
```

Default jobtype projects are used when data is consolidated prior to reporting. All usage by users that are enabled for project accounting must be associated with a project. If usage is found for which no project is available, a series of conditions are checked to determine the project that will be associated with the usage. If the user has `DEFAULT_PROJECT` enabled, the project contained in the user's `DEFPROJECT` field will be used. If `DEFAULT_PROJECT` is disabled, the value contained in the default jobtype project field will be used based on the jobtype of the usage. If it is interactive usage, the value in `INTER_PROJECT` will be used. If it is background usage the value in `BACKGROUND_PROJECT` will be used. If these fields are empty, the username will be used as a projectname as a last resort.

Custom Project Prompting

Set up this example through the CIMS CLI by performing commands similar to the ones that follow:

```
# A_authorize
A_AUTHORIZE> ADD george/PRTNUMBER=3 -
_A_AUTHORIZE> /ENABLE=(PROJECT_ACCOUNTING, -
_A_AUTHORIZE> SWITCH_PROJECT)
%AUTHORIZE-S-ADDED, 1 USER record successfully added
A_AUTHORIZE> MODIFY george/PRTDEFAULT=w104financ
```



```
%AUTHORIZE-S-MODIFIED, 1 USER record successfully modified
_A_AUTHORIZE> MODIFY george/PRT1=(START=1,END=4,-
_A_AUTHORIZE> PROMPT="Enter section name:",-
_A_AUTHORIZE> PRT_ACTION=USE_DEFAULT,-
_A_AUTHORIZE> SWITCH_ACTION=USE_DEFAULT)
%AUTHORIZE-S-MODIFIED, 1 USER record successfully modified
_A_AUTHORIZE> MODIFY george/PRT2=(START=5,END=10,-
_A_AUTHORIZE> PROMPT="Enter department name:",-
_A_AUTHORIZE> PRT_ACTION=VALUE_OPTIONAL,-
_A_AUTHORIZE> SWITCH_ACTION=VALUE_REQUIRED)
%AUTHORIZE-S-MODIFIED, 1 USER record successfully modified
_A_AUTHORIZE> MODIFY george/PRT3=(START=11,END=16,-
_A_AUTHORIZE> PROMPT="Enter task number:",-
_A_AUTHORIZE> PRT_ACTION=value_required,-
_A_AUTHORIZE> SWITCH_ACTION=VALUE_REQUIRED)
%AUTHORIZE-S-MODIFIED, 1 USER record successfully modified
_A_AUTHORIZE> EXIT
#
```

When the user does a project switch and enters the project name on the command line, the user must specify both the department name and the task number, the section name is supplied by default as indicated by SWITCH_ACTION=USE_DEFAULT for PRT1. If the user entered the following:

```
# exec A_switch financ765123
#
```

CIMS switches the user to project w104financ765123. On the other hand, when the user does this:

```
# exec A_switch
Enter department:
Enter task number: 765123
#
```

PRT1, "Enter section name:" is not displayed, but the first four characters of PRTDEFAULT, "w104", are present by default because SWITCH_ACTION=USE_DEFAULT. PRT2, "Enter department:" is displayed, but the user does not enter any value. Since PRT_ACTION for PRT2 is VALUE_OPTIONAL and the SWITCH_ACTION is VALUE_REQUIRED, the 5-10 characters of PRTDEFAULT, "financ" are used. For PRT3, "Enter task number:", the user is required to enter a value at the prompt because the

PRT_ACTION for PRT3 is set to VALUE_REQUIRED. The user responds with "765123" making the resulting project name w104financ765123. This project is now checked for normal project validation.

PRT[1-5] values can be used to build up to 5 prompts when the user logs in or switches to a new project. The value of PRTNUMBER defines how many prompts are used.

Project Switching

Enable project switching through the CIMS CLI by performing commands similar to the ones that follow:

```
# A_authorize
A_AUTHORIZE> MODIFY/USER bdunn-
_A_AUTHORIZE> /ENABLE=SWITCH_PROJECT-
_A_AUTHORIZE> /DISABLE=(SWITCH_GID,-
_A_AUTHORIZE> SWITCH_DIRECTORY)
%AUTHORIZE-S-MODIFIED, 1 USER records successfully modified
A_AUTHORIZE> MODIFY/PROJECT sales-
_A_AUTHORIZE> /ENABLE=SWITCH_PROJECT-
_A_AUTHORIZE> /DISABLE=(SWITCH_GID,-
_A_AUTHORIZE> SWITCH_DIRECTORY)
%AUTHORIZE-S-MODIFIED, 1 PROJECT records successfully modified
A_AUTHORIZE> EXIT
#
```

Project GID

Manage project GID through the CIMS CLI by performing commands similar to the ones that follow:

```
# A_authorize
A_AUTHORIZE> MODIFY/PROJECT sales/GID=15-
_A_AUTHORIZE> /ENABLE=SWITCH_GID-
_A_AUTHORIZE> /DISABLE=REVERT_GID
%AUTHORIZE-S-MODIFIED, 1 PROJECT records successfully modified
A_AUTHORIZE> MODIFY/USER bdunn-
_A_AUTHORIZE> /ENABLE=SWITCH_GID
```

```
%AUTHORIZE-S-MODIFIED, 1 USER records successfully modified
A_AUTHORIZE> EXIT
#
```

In this example, when user `bdunn` switches to project sales, his effective GID will be switched to 15. All Unix Process, (`pacct`) records generated by the user after the switch will contain the GID 15. All files created by the user will have the GID group of 15.

Project GIDs can be used to associate a project with background usage when `PROJECT_ACCOUNTING` is enabled in the CIMS Parameter File. The CIMS `FORMAT` Utility must look for group names using the value in the GID field of Unix Process, (`pacct`) Records. If `GID_method` is set to `PROJECT` in the CIMS Parameter File, the `FORMAT` Utility will first look for a matching GID in the `PROJECT` records in the CIMS Authorization File. If a matching GID is found, the `projectname` will be placed in the group field of the image record in the Nightly Accounting File, (`acc_<DATE>.dat`). If no matching Project GID is found, the Unix Group file is read for group names. When the Nightly Accounting Files are consolidated by the CIMS `SELECT` Utility, if `/UNKNOWN=GROUP` is included on the command line, the usage for background image records will be assigned the value in the image record's group field.

The default value of `GID_method` is `GROUP`, which indicates that the `FORMAT` Utility will only look in the Unix Group File for groupnames. The action of `/UNKNOWN=GROUP` in the `SELECT` Utility would then be to assign the Unix group name as the project for usage for which no project has been determined.

Project Default Directory

Manage project default directory through the CIMS CLI by performing commands similar to the ones that follows:

```
# A_authorize
A_AUTHORIZE> MODIFY/PROJECT sales/GID=15-
_A_AUTHORIZE> /DIRECTORY="/usr/users/sales"-
_A_AUTHORIZE> /ENABLE=SWITCH_DIRECTORY-
_A_AUTHORIZE> /DISABLE=REVERT_DIRECTORY
%AUTHORIZE-S-MODIFIED, 1 PROJECT records successfully modified
```

Project Accounting

```
A_AUTHORIZE> MODIFY/USER bdunn-
_A_AUTHORIZE> /ENABLE=SWITCH_DIRECTORY
%AUTHORIZE-S-MODIFIED, 1 USER records successfully modified
A_AUTHORIZE> EXIT
#
```

In this example, when user `bdunn` switches to project `sales`, his effective GID will be switched to 15. All Unix Process, (`pacct`) records generated by the user after the switch will contain the GID 15. All files created by the user will have the GID group of 15. In addition, the user's current working directory will be changed to `/usr/users/sales`.

Project Cost Center

Manage project cost center through the CIMS CLI by performing a command similar to the one that follows:

```
# A_authorize
A_AUTHORIZE> MODIFY/PROJECT sales-
_A_AUTHORIZE> /COST_CENTER="AABBCC"
%AUTHORIZE-S-MODIFIED, 1 PROJECT records successfully modified
A_AUTHORIZE> EXIT
#
```

In this example, the project record for project `sales` in the CIMS Authorization File is modified so that its `COST_CENTER` field is set to `AABBCC`. When generating a report of `project_cost_center`, usage consumed by project `sales` will be included in that associated with cost center `AABBCC`. A `COST_CENTER` can be any string up to 32 characters long.

Project Groups

Manage project grouping fields through the CIMS CLI by performing a command similar to the one that follows:

```
# A_authorize
A_AUTHORIZE> MODIFY/PROJECT sales/GRP1=-
_A_AUTHORIZE> "DDEEFF"-
%AUTHORIZE-S-MODIFIED, 1 PROJECT records successfully modified
A_AUTHORIZE> EXIT
#
```

In this example, a project record in the CIMS Authorization File for project sales is modified. The GRP1 field for the project is set to the string DDEEFF. When generating a report of select keyword, (see *CIMS for UNIX Report Description Files* on page 5-12), project_group1, usage associated with project sales will be included in that associated with the project group DDEEFF. There are five project group fields in each project record, GRP1, GRP2, GRP3, GRP4, and GRP5. The value contained in GRP2 is used when reporting on select keyword project_group2, in GRP3 when reporting on project_group3 and so on. In this way, hierarchical groupings can be implemented. Any of the grouping fields can be up to 32 characters long.

Project Description

Manage project description through the CIMS CLI by performing a command similar to the one that follows:

```
# A_authorize
A_AUTHORIZE> MODIFY/PROJECT sales-
_A_AUTHORIZE> /DESCRIPTION="Sales Project Account"
%AUTHORIZE-S-MODIFIED, 1 PROJECT records successfully modified
A_AUTHORIZE> EXIT
#
```

In this example, the project record for project, sales is modified in the CIMS Authorization File. The DESCRIPTION field for the project is set to the string "Sales Project Account". When generating a report including body keyword, (see *CIMS for UNIX Report Description Files* on page 5-12), project_description, "Sales Project Account" will be displayed. In addition to providing a description for the project, the DESCRIPTION field can be used as a grouping field as described in the previous example. The DESCRIPTION field can be up to 32 characters long.

Project Time Limit

Manage project time limit through the CIMS CLI by performing a command similar to the one that follows:

```
# A_authorize
A_AUTHORIZE> MODIFY/PROJECT sales-
_A_AUTHORIZE> /START_TIME="4-FEB-1999"-
_A_AUTHORIZE> /END_TIME="4-FEB-2001"
%AUTHORIZE-S-MODIFIED, 1 PROJECT records successfully modified
A_AUTHORIZE> EXIT
#
```

The example above modifies the project record for project `sales` in the CIMS Authorization File. The active time for the project is set to start on 4-Feb-1999 and end on 4-Feb-2001. Any attempt to access this project through the CIMS LOGIN and SWITCH Utilities outside of this timeframe will fail. `START_TIME` and `END_TIME` define the timeframe for which a project is active.

Project Validation

Validation Retry

Establish a validation retry count through the CIMS CLI by performing a command similar to the one that follows:

```
# A_setup
A_SETUP> DEFAULT/RETRY_COUNT=5
%SETUP-S-MODIFIED, 1 node record successfully modified
A_SETUP> EXIT
```

In the example above, the `RETRY_COUNT` for project access is set to 5. This field is used by the CIMS LOGIN Utility to define the number of attempts a user can make to enter a valid project when logging into the system. By default `RETRY_COUNT` is set to 3.

Validation By User Account Authorization

Set up user account authorization through the CIMS CLI by performing commands similar to the ones that follow:

```
# A_authorize
A_AUTHORIZE> MODIFY/PROJECT sales-
_A_AUTHORIZE> /ENABLE=PROJECT_VALIDATION-
_A_AUTHORIZE> /VALIDATION_METHOD=-
_A_AUTHORIZE> (NOPASSWORD,USERPROJ)
%AUTHORIZE-S-MODIFIED, 1 PROJECT records successfully modified
A_AUTHORIZE> DEFAULT/PROJECT-
_A_AUTHORIZE> /ENABLE=PROJECT_VALIDATION
%AUTHORIZE-S-MODIFIED, 1 PROJECT records successfully modified
A_AUTHORIZE> ADD/VALIDATE bdunn sales
%AUTHORIZE-1-USRPROADD user/project combination successfully added.
A_AUTHORIZE> EXIT
#
```

This example modifies the project record for project `sales` so that only users that are pre-validated in the CIMS Validation File for project `sales` may access the project with the CIMS LOGIN and SWITCH Utilities. The DEFAULT project record is also enabled for PROJECT_VALIDATION so that if a user enters a project that does not exist in the CIMS Authorization File, the settings for the DEFAULT project record are used. Finally, user `bdunn` is pre-validated in the CIMS Validation File to access the project `sales`.

Validation by Password

Set up password validation through the CIMS CLI by performing commands similar to the ones that follow:

```
# A_authorize
A_AUTHORIZE> MODIFY/PROJECT sales-
_A_AUTHORIZE> /ENABLE=PROJECT_VALIDATION-
_A_AUTHORIZE> /VALIDATION_METHOD=-
_A_AUTHORIZE> (PASSWORD,NOUSERPROJ)-
_A_AUTHORIZE> /PASSWORD="loophole"
%AUTHORIZE-S-MODIFIED, 1 PROJECT records successfully modified
A_AUTHORIZE> DEFAULT/PROJECT/ENABLE=-
_A_AUTHORIZE> PROJECT_VALIDATION
```

```
%AUTHORIZE-S-MODIFIED, 1 PROJECT records successfully modified
A_AUTHORIZE> EXIT
#
```

This example modifies the project record for project `sales` so that a password must be entered by users who wish to access this project with the CIMS LOGIN and SWITCH Utilities. In addition, the DEFAULT project record is enabled for PROJECT_VALIDATION so that if a user enters a project that does not exist in the CIMS Authorization File, the settings for the DEFAULT project record are used. NOUSERPROJ is also set so that any user who knows the password may access the project. If USERPROJ were set, in addition to knowing the password, the user would need to be pre-validated for the project `sales` in the CIMS Validation File.

Custom Validation Method

Sites can develop custom routines that further validate users for access to projects. The custom validation routine can be a binary executable or a script. The only requirement is that it must return the value 0 on success. The path and name of the custom routine must be entered into the VALIDATION_ROUTINE field of the CIMS Parameter file. Any project that has PROJECT_VALIDATION enabled and VALIDATION_METHOD set to CUSTOM will call this routine during the validation process. The routine will be called with the following arguments;

```
argv[1] - username
argv[2] - project name
argv[3] - project password (if one is used)
```

If a project has other validation methods like USERPROJ and PASSWORD, those methods are validated prior to calling the custom routine. A return value of 0 is considered success and the user is allowed to access the project. Any other return value is considered a validation failure and access is denied.

Establish a custom validation method through the CIMS CLI by performing commands similar to the ones that follow:

```
# A_authorize
A_AUTHORIZE> MODIFY/PROJECT sales-
_A_AUTHORIZE> /VALIDATION_METHOD=CUSTOM
```



```
%AUTHORIZE-S-MODIFIED, 1 PROJECT records successfully modified
A_AUTHORIZE> EXIT
#
#A_setup
A_SETUP> DEFAULT/VALIDATION_ROUTINE=-
_A_SETUP> "/usr/local/my_val_routine"
%SETUP-S-MODIFIED, 1 node record successfully modified
A_SETUP> EXIT
#
```

In this example, the project record for project sales is modified so that the `VALIDATION_METHOD` is set to `CUSTOM`. The custom validation routine is set to `/usr/local/my_val_routine` in the CIMS Parameter File. If user `bdunn` attempts to access the project sales through the CIMS LOGIN Utility, `/usr/local/my_val_routine` will be called with `argv[1]` set to "bdunn" and `argv[2]` set to "sales". No value is present for `argv[3]` because the project sales has no password. Access to the project will be determined by the return value from `usr/local/my_val_routine`.

Validation by Project Existence

Establish a project by existence method through the CIMS CLI by performing commands similar to the ones that follow:

```
# A_authorize
A_AUTHORIZE> ADD/PROJECT sales-
_A_AUTHORIZE> /ENABLE=PROJECT_VALIDATION-
_A_AUTHORIZE> /VALIDATION_METHOD=(NOPASSWORD,-
_A_AUTHORIZE> NOUSERPROJ,NOCUSTOM)
%AUTHORIZE-S-ADDED, PROJECT record successfully added
A_AUTHORIZE> DEFAULT/PROJECT/ENABLE=-
_A_AUTHORIZE> PROJECT_VALIDATION
%AUTHORIZE-S-MODIFIED, 1 PROJECT records successfully modified
A_AUTHORIZE> EXIT
#
```

In this example, a project record for project sales is added to the CIMS Authorization File. `PROJECT_VALIDATION` is enabled and `VALIDATION_METHOD` is set to "none". With this configuration, any user attempting to access the project sales will be accepted.

No Validation

The no validation method is defined through the CIMS CLI as follows:

```
# A_authorize
A_AUTHORIZE> DEFAULT/PROJECT-
_A_AUTHORIZE> /DISABLE=PROJECT_VALIDATION
%AUTHORIZE-S-MODIFIED, 1 PROJECT records successfully modified
A_AUTHORIZE> EXIT
#
```

This example modifies the DEFAULT project record so that PROJECT_VALIDATION is disabled. In most cases, this configuration is not recommended. Any value the user enters as a project will be accepted and a new project record, that is a copy of the DEFAULT project record, is added to the CIMS Authorization File.

Image Accounting

Enabling Proprietary Software

Enable the system-wide proprietary software through the CIMS CLI as follows:

```
# A_setup
A_SETUP> DEFAULT/ENABLE=PROPRIETARY_SOFTWARE
%SETUP-S-MODIFIED, 1 node record successfully modified
A_SETUP> EXIT
#
```

This example enables the PROPRIETARY_SOFTWARE flag in the CIMS Parameter File. During data consolidation, this flag is checked to see if software packages defined in the CIMS Image Mapping File are to be tracked. If the flag is enabled, the CIMS SELECT Utility will check the image names in the individual accounting records to see if they are associated with a defined package. If they are, the resource usage in the individual record is credited to the packagename for software package accounting reports. The resource usage is also credited to the user and project for user and project accounting reports.

Defining Software Packages

Images are mapped using the CIMS CLI commands:

- ADD/IMAGE—to add new image/package combinations
- REMOVE/IMAGE—to remove existing image/package combinations
- REPORT/IMAGE—to generate a report on images
- REPORT/PACKAGE—to generate a report on packages

Add images through the CIMS for NT/UNIX CLI by performing a command similar to the one that follows:

```
# A_setup
A_SETUP> ADD/IMAGE ksh SHELLS
%SETUP-S-ADDED, image record successfully added
A_SETUP> ADD/IMAGE csh SHELLS
%SETUP-S-ADDED, image record successfully added
A_SETUP> exit
#
```

In the previous example, the imagenames `ksh` and `csh` are defined to the package `SHELLS`. Note that imagenames are restricted to the first 10 characters of the imagename. Wildcards are supported for imagenames.

Database Accounting

Manage database and instance parameters through these CIMS CLI commands:

- ADD/DBINST—add new database parameters
- MODIFY/DBINST—modify existing database parameters
- REMOVE/DBINST—remove database parameters
- REPORT/DBINST—report database parameters

Add database and instance parameters through the CIMS CLI by performing a command similar to the one that follows:

```
# A_setup
A_SETUP> ADD/DBINST DEVL/DBTYPE=Oracle-
_A_SETUP> /USERNAME=CIMS/PASSWORD=123/FREQUENCY=60
%SETUP-S-ADDED, instance record successfully added
A_SETUP> EXIT
```

Here a DB instance record has been added to the CIMS Database Instance File for ORACLE instance DEVL. The values in this record are used when starting CIMS ORACLE or DB2 Accounting Daemons. For each ORACLE instance or DB2 database, an accounting daemon is required to track usage within the instance or database. The value of USERNAME and PASSWORD are used by the daemon to attach to the instance or database. The FREQUENCY value is expressed in seconds and indicates how often the instance or database is sampled by the accounting daemon.

When accounting for ORACLE instances, the environment variable A_ORACLE_ACCT must be set to "Y" in the CIMS for Windows NT and UNIX Configuration Parameter File. Also the configuration parameter, timed_statistics must be set to true in the instance initialization file, init<instance_name>.ora.

When accounting for DB2 Databases, the environment variable A_DB2_ACCT must be set to "Y" in the CIMS for Windows NT and UNIX Configuration Parameter File.

CIMS provides several scripts for starting and stopping the daemon. They are located in the CIMS_HOME/scripts/oracle and CIMS_HOME/scripts/db2 directories. These scripts can be used in conjunction with the Database startup and shutdown procedures used at your site.

The CIMS ORACLE and DB2 Accounting Daemons write messages to log files in the CIMS_HOME/log directory.

Enable the system-wide database absolute shift and daily accounting options through the CIMS for CLI as follows:

```
#A_setup
A_SETUP>DEFAULT/ENABLE=(DB_ABSOLUTE_SHIFT, DB_ABSOLUTE_DAILY)
%SETUP-S-MODIFIED, 1 node record successfully modified
A_SETUP>EXIT
```

This example enables `DB_ABSOLUTE_SHIFT` and `DB_ABSOLUTE_DAILY` flags in the CIMS Parameter File. These flags are useful at sites where long running ORACLE or DB2 sessions occur. These are sessions that last for days at a time. When `DB_ABSOLUTE_SHIFT` is enabled the CIMS ORACLE or DB2 Accounting Daemon will write a partial accounting record for each active session at each shift boundary defined in the CIMS Shift File. When `DB_ABSOLUTE_DAILY` is enabled the CIMS ORACLE or DB2 Accounting Daemon will write a partial accounting record for each active session at midnight each night. These flags are not required for ORACLE and DB2 accounting and should only be used at sites where long running database sessions occur.

Reporting Parameters

Enabling Terminal-by-User

Enable or disable the system-wide `TERMINAL_BY_USER` option through the CIMS CLI as follows:

```
# A_setup
A_SETUP> DEFAULT/ENABLE=TERMINAL_BY_USER
%SETUP-S-MODIFIED, 1 node record successfully modified
A_SETUP> EXIT
```

`TERMINAL_BY_USER` should be enabled if you want to report the terminal name in interactive usage. This is important at sites that want to track "seat_time" in workstation environments. Seat_time would be the connect time of the user on the console terminal. For example, if a user logs onto a workstation in an X window environment and opens two xterms, (`ttyp1` and `ttyp2`), interactive connect_time would be tracked to three sessions, one for each xterm terminal and one to the terminal "console". In this case, after an hour, the user would have consumed

Reporting Parameters

three hours of interactive connect time. If you want to track the actual seat time for the session, you could report the user's connect time on terminal, "console".

When `TERMINAL_BY_USER` is enabled, the values in the CIMS Collapsible Terminal File are used to determine which terminal names should be collapsed when reporting usage on a particular terminal. In the example in the previous paragraph, you may not be interested in usage on individual pseudo terminals like `ttyp1`, but total usage on all pseudo terminals could be collapsed to "tty". Another example where the CIMS Collapsible Terminal File values are useful is ftp usage. When a user connects to a server with ftp, the terminal name is `ftp<number>` where `<number>` is random. The entry "ftp" in the CIMS Collapsible Terminal File combines all ftp sessions under the terminal name ftp. This way you could report the connect time and number of ftp connections by individual users.

Capacity Interval

Establish a capacity interval value through the CIMS CLI by performing a command similar to the one that follows:

```
# A_setup
A_SETUP> DEFAULT/CAPACITY_INTERVAL=30
%SETUP-S-MODIFIED, 1 node record successfully modified
A_SETUP> EXIT
#
```

`CAPACITY_INTERVAL` is expressed in minutes and defines the interval for capacity reporting. By default, `CAPACITY_INTERVAL` is set to 60 or hourly units. In the example above, usage will be reported in 30 minute units. In this way, you could determine the busiest half hour of the day. `CAPACITY_INTERVAL` must be some divisible portion of an hour. In other words 1, 2, 3, 4, 5, 6, 10, 15, 20, or 30 minutes. It should be noted that smaller values for `CAPACITY_INTERVAL` result in much larger CIMS Capacity Files for reporting. `CAPACITY_INTERVAL` should only be as small as the smallest time unit to be reported.

Rates

CIMS for NT/UNIX DEFAULT Rate Table

Modify the DEFAULT_RATES table through the CIMS CLI by performing a command similar to the one that follows:

```
# A_rates
A_RATES> DEFAULT/BACKGROUND/SCPU=1.0/UCPU=0.5
%RATES-I-MODIFIED, 1 record(s) successfully modified
A_RATES>
#
```

The DEFAULT_RATES table is used as a template when adding new rates tables to the CIMS Rates File. Users and projects are assigned a rate table in the user and project records of the CIMS Authorization File. Users and projects can be assigned one rate table for usage generated on one node and another rate table for usage generated on another node.

A rate table is made up of at least one rate record for each jobtype, interactive, background, batch, print, storage, oracle, db2, and package. Multiple rates records for individual jobtypes may exist in a rate table. For instance, interactive usage for shift "prime" may be charged a different rate than interactive usage for shift "non-prime". Oracle usage for instance D732 may be charged a different rate than oracle usage for instance D716. Once a rate table is created with all the rate records needed to define the rates needed at a particular site, the rate table can be assigned to users and projects for reporting.

Adding A Rate Table

Add a rate table through the CIMS CLI by performing a command similar to the one that follows:

```
# A_rates
A_RATES> ADD TRIPLE_RATES
%RATES-I-ADDED, 8 record(s) successfully added
A_RATES>
#
```

Reporting Parameters

In the example above, a new rate table, `TRIPLE_RATES`, is added to the CIMS Rates File. This new rate table is a copy of the `DEFAULT_RATES` table. Note that eight records are added, one for each jobtype. These eight rates records make up the rate table, `TRIPLE_RATES`. In the next example a new rate record is added to the rate table `TRIPLE_RATES` for interactive usage during the shift, "Prime".

Add a specific element to a rate table through the CIMS CLI by performing a command similar to the one that follows:

```
# A_rates
A_RATES> ADD TRIPLE_RATES /INTERACTIVE/SHIFT=Prime
%RATES-I-ADDED, 1 record(s) successfully added
A_RATES>
#
```

Modifying a Rate Table

Modify a rate table through the CIMS CLI by performing a command similar to the one that follows:

```
# A_rates
A_RATES> modify TRIPLE_RATES -
_A_RATES> /interactive/shift=Prime/connect_time=3.0
%RATES-I-MODIFIED, 1 record(s) successfully modified
A_RATES>
#
```

In this example, the interactive rate record for shift Prime in the rate table `TRIPLE_RATES` is modified so that the rate for `connect_time` is set to 3.00.

Remove A Rate Table

Remove a rate table through the CIMS CLI by performing a command similar to the one that follows:

```
# A_rates
A_RATES> remove TRIPLE_RATES
%RATES-I-REMOVED, 9 record(s) successfully removed
A_RATES>
#
```


In this example the nine rate records that make up the rate table TRIPLE_RATES are removed from the CIMS Rates File. If users or projects have been assigned this rate table and a charge report is generated after removing the rate table, the rates in the DEFAULT_RATES table will be used.

Zeroing A Rate Table

Zero a rate table through the CIMS CLI by modifying the rate table and setting all rates to zero.

```
# A_rates
A_RATES> zero ZERO_RATES
%RATES-I-ZEROED, 8 record(s) successfully zeroed
A_RATES>
```

This example set the rates to 0.00 in all rate records that make up the rate table ZERO_RATES.

Assigning A Rate Table

Assign a rate table to a user or project account through the CIMS CLI by performing a command similar to the one that follows:

```
#
# A_authorize
A_AUTHORIZE> MODIFY jdoe -
_A_AUTHORIZE> /NODE=SPARKY/RATE_TABLE=SPARKY_RATES
%AUTHORIZE-S-ADDED, USER record successfully added
#
```

This example assigns the rate table SPARKY_RATES to usage consumed by user jdoe on node SPARKY.

Shifts

Add shifts through the CIMS CLI by performing commands similar to those that follow:

```
# A_setup
A_SETUP> ADD/SHIFT prime /DAYS=(MONDAY,-
_A_SETUP> TUESDAY,WEDNESDAY,THURSDAY,FRIDAY) -
_A_SETUP> /START_TIME=9:00
```

Reporting Parameters

```
%SETUP-S-SHADDED, 5 shift record(s) successfully added
A_SETUP> ADD/SHIFT non-prime /DAYS=(MONDAY,-
_A_SETUP> TUESDAY,WEDNESDAY,THURSDAY,FRIDAY) -
_A_SETUP> /START_TIME=18:00
%SETUP-S-SHADDED, 5 shift record(s) successfully added
A_SETUP> ADD/SHIFT WEEKEND/DAYS=(SATURDAY,SUNDAY) -
_A_SETUP> /START_TIME=00:00
%SETUP-S-SHADDED, 2 shift record(s) successfully added
A_SETUP> ADD/SHIFT ho1iday/DAYS=HOLIDAY/START_TIME=00:00
%SETUP-S-SHADDED, 1 shift record(s) successfully added
A_SETUP> EXIT
#
```

In this example, shift, prime, begins on Monday through Friday at 9:00 AM. Note that no end time is designated when the shift is added to the CIMS Shift File. The end time of a shift is defined by the start time of the next shift. For instance, here shift prime would start at 9:00 on Monday and end at 18:00 when shift non-prime begins. Shift non-prime would then end on Tuesday at 9:00, when shift prime begins again. Also in this example, a shift "holiday" is added which starts at 00:00 on all days defined in the CIMS Holiday File.

Holidays

The CIMS CLI commands to manage Holiday set-up are:

- ADD/HOLIDAY—to add new holidays
- COMPRESS—to compress the Holiday File
- MODIFY/HOLIDAY—to modify an existing holiday's description
- REMOVE/HOLIDAY—to remove an existing holiday
- REPORT/HOLIDAY—to report on the holidays

Add holidays through the CIMS for NT/UNIX CLI by performing a command similar to the one that follows:

```
# A_setup
A_SETUP> ADD/HOLIDAY 1-JAN-1997
%SETUP-S-ADDED, holiday record successfully added
```

```
A_SETUP> ADD/HOLIDAY 4-JUL-2000/DESC="Independence Day"  
%SETUP-S-ADDED, holiday record successfully added  
A_SETUP> ADD/HOLIDAY 31-MAY-2000/DESC="Memorial Day"  
%SETUP-S-ADDED, holiday record successfully added  
A_SETUP> EXIT  
#
```

Modify a holiday through the CIMS CLI by performing a command similar to the one that follows:

```
# A_setup  
A_SETUP> MOD/HOLIDAY 1-JAN-2000/DESC="New Year's Day"  
%SETUP-S-MODIFIED, 1 holiday record successfully modified  
A_SETUP> EXIT  
#
```

Defining holidays is only required if different rates are going to be charged for resource usage consumed on holidays. If not, there is no need to define holidays.

Output Queue

The CIMS CLI commands to manage records in the CIMS Queue Mapping File are:

- ADD/QUEUE—to add new batch or print queues
- MODIFY/QUEUE—to modify an existing queue's description
- REMOVE/QUEUE—to remove an existing queue
- REPORT/QUEUE—to report on the queues

Add a record for a queue to the CIMS Queue Mapping File through the CIMS CLI by performing a command similar to the one that follows:

```
# A_setup  
A_SETUP> ADD/QUEUE/PRINT 1ta0  
%SETUP-S-ADDED, queue record successfully added  
A_SETUP> EXIT  
#
```

Individual records for print and batch queues are only required if different rates are going to be charged for different print and batch queue usage.

GID Method

Establish the GID Method through the CIMS CLI by performing a command similar to the one that follows:

```
# A_setup
A_SETUP> DEFAULT/GID_METHOD="project"
%SETUP-S-MODIFIED, 1 node record successfully modified
A_SETUP> EXIT
#
```

See the notes on *Project GID* on page 6-14.

Report Header

Establish the report header through the CIMS CLI by performing a command similar to the one that follows:

```
# A_setup
A_SETUP> DEFAULT/HEADER="CIMS Lab"
%SETUP-S-MODIFIED, 1 node record successfully modified
A_SETUP> EXIT
#
```

The value contained in the `HEADER` field of the CIMS Parameter File is used by the report scripts in `CIMS_HOME/scripts/reports`. These scripts generate textual reports that contain column headings and totals.

Digit Separator

Establish a digit separator value through the CIMS CLI by performing a command similar to the one that follows:

```
# A_setup
A_SETUP> DEFAULT/DIGIT_SEPARATOR="."
%SETUP-S-MODIFIED, 1 node record successfully modified
A_SETUP> EXIT
#
```

In the United States, the typical digit separator is ',' and the radix point is '.'. Thus a number might be written as 12,864.32. In other countries, the digit separator might be '.' and the radix point ',' so that the same number is written 12.864,32. DIGIT_SEPARATOR and RADIX_POINT are used to set the values that are used in your environment.

Memory Demand Algorithm

Establish the memory demand algorithm through the CIMS CLI by performing a command similar to the one that follows:

```
# A_setup
A_SETUP> DEFAULT/MEMORY_DEMAND_ALGORITHM=0
%SETUP-S-MODIFIED, 1 node record successfully modified
A_SETUP> EXIT
#
```

Memory resources can be very large numbers. MEMORY_DEMAND_ALGORITHM is used to establish the unit that will be used when reporting memory resources. Valid values for MEMORY_DEMAND_ALGORITHM are 0 - 5 which are interpreted as follows:

- 0 - Memory statistics reported unchanged
- 1 - Memory statistics reported in units of 10
- 2 - Memory statistics reported in units of 100
- 3 - Memory statistics reported in units of 1000
- 4 - Memory statistics reported in units of 10000
- 5 - Memory statistics reported in units of 100000

Reporting Parameters

The following resources are reported with MEMORY_DEMAND_ALGORITHM applied.

```
UNIX      memory utilization
DB2       uow_log_space_used
ORACLE    pga_memory
ORACLE    uga_memory
```

Monetary Symbols

Establish the monetary prefix and monetary suffix values through the CIMS CLI by performing commands similar to those that follow:

```
# A_setup
A_SETUP> DEFAULT/MONPREFIX="$"
%SETUP-S-MODIFIED, 1 node record successfully modified
A_SETUP> EXIT
#
# A_setup
A_SETUP> DEFAULT/MONSUFFIX=""
%SETUP-S-MODIFIED, 1 node record successfully modified
A_SETUP> EXIT
#
```

MONPREFIX and MONSUFFIX are used to define the prefix and suffix used in your environment when displaying monetary values. In the United States, the MONPREFIX would typically be set to "\$" and MONSUFFIX set to ". Each value can be up to 3 characters long.

Radix Point

Establish the radix point value through the CIMS CLI by performing a command similar to the one that follows:

```
# A_setup
A_SETUP> DEFAULT/RADIX_POINT=","
%SETUP-S-MODIFIED, 1 node record successfully modified
A_SETUP> EXIT
#
```

See notes for *Digit Separator* on page 6-33.

Data Formatting

CIMS for UNIX Print Accounting Script

The CIMS for UNIX Print Accounting script (`prtacct`) controls the UNIX print accounting files. `prtacct` copies the files designated in the CIMS for UNIX Print Parameter File (`A_print.par`), invokes the CIMS for UNIX Format Utility (`A_format`), and then re initializes them.

The `FORMAT` Utility processes the UNIX print accounting data in the UNIX print accounting files specified in `A_print.par`. The `FORMAT` Utility writes the formatted data into the CIMS for UNIX Print Statistics File (`A_prtstat.sys`).

The following is the CIMS CLI command for executing the `FORMAT` Utility on the print accounting data listed in the CIMS for UNIX Print Parameter File.

```
A_format
A_FORMAT> FORMAT/TYPE=PRINT "/usr/arsap/data/A_print.par"
%FRMT-I-PROCFILE, processing file-"usr/arsap/data/A_print.par"
A_FORMAT> EXIT
#
```

`prtacct` is only performed when `PROC_PRINT` is set to `Y` in the CIMS for UNIX Configuration Parameter File.

CIMS for NT/UNIX Runacct Script

The CIMS Runacct script runs the CIMS FORMAT Utility against the remaining raw accounting files in the ARSAP_HISTORY directory. The raw accounting files and their corresponding intermediate statistics file are as follows:

UNIX Wtmp File	A_wtmpstat.sys
UNIX Acct File	A_procstat.sys
NT security log	A_ntsecstat.sys
NT application log	A_ntappstat.sys
CIMS for UNIX Activity File	A_lgistat.sys/A_lgiproj.sys
CXbatch Accounting File	A_batstat.sys (optional)
NQS Batch Accounting File	A_batstat.sys (optional)
CIMS for NT and UNIX Oracle Accounting File	A_dbstat.sys (optional)
CIMS for NT and UNIX IBM DB2 Accounting File	A_dbstat.sys (optional)

The following is a sample of the CIMS CLI commands for the CIMS FORMAT Utility (A_format) to format accounting data from within the CIMS Runacct script (runacct):

```
# A_format
FORMAT> FORMAT/TYPE=ACTIVITY/ZERO "/usr/arsap/history/activity_<date>"
%FRMT-I-PROCFILE, processing file-"usr/arsap/history/activity_<date>"
FORMAT> FORMAT/TYPE=ACCT/ROLL/ZERO "/usr/arsap/history/pacct_<date>"
%FRMT-I-PROCFILE, processing file-"usr/arsap/history/pacct_<date>"
FORMAT> FORMAT/TYPE=WTMP/ZERO "/usr/arsap/history/wtmp_<date>"
%FRMT-I-PROCFILE, processing file-"usr/arsap/history/wtmp_<date>"
FORMAT> EXIT
#
```


The /ZERO qualifier re initializes the CIMS intermediate accounting file. This qualifier assures that only the current day's statistics are contained in the file.

The CIMS FORMAT Utility has an additional qualifier when processing the UNIX image accounting data in the UNIX Process Accounting File, `pacct`. This qualifier is /ROLL. The /ROLL qualifier combines similar image records from background jobs into a rolled accounting record. When the same image is run by a user several times in background (no controlling terminal), these image records are rolled into one roll-up record that indicates the number of times the image was executed. This qualifier can reduce the size of the resulting CIMS Nightly accounting file when numerous background jobs are performed. You do not want to use the /ROLL qualifier when you need to utilize the /UNKNOWN_PROJECT qualifier with the CIMS SELECT Utility (refer to *CIMS for NT/UNIX Select Utility* on page 4-7). You also do not want to use the /ROLL qualifier on AIX 3.2.x when CIMS Project Accounting is enabled.

The type of data formatted determines the jobtype of the accounting record. The UNIX `Wtmp` file records produce CIMS for UNIX login and logout records that determine start and end times of UNIX interactive sessions. The UNIX `Acct` file records produce CIMS for UNIX image records that can be interactive or background jobtypes. The CIMS for UNIX activity records produce CIMS for UNIX project login, switch, and logout as well as `CXbatch` and `NQS` login records that are associated with the appropriate jobtypes. The `CXbatch` records produce `CXbatch` logout records that determine batch jobtypes. The `NQS` records produce `NQS` logout records that determine batch jobtypes. The Oracle records produce Oracle login and switch records that specify Oracle jobtypes. The `DB2` records produce `DB2` logout records that specify `DB2` jobtypes. Once projects are entered into the accounting file, CIMS for NT/UNIX does not provide any means to alter the accounting file.

Data Formatting

The NT security log records produce CIMS for NT login, logout, image start and image end records that are interactive jobtypes. The NT application log records produce CIMS for NT project login, switch, and logout records. The CIMS for NT Format Utility (`NT_format`) maintains a CIMS for NT Image List File (`A_imglist.sys`) file for images that have not completed yet.

After the formatting is completed, the CIMS for NT/UNIX Runacct script produces the CIMS for NT/UNIX Nightly Accounting File by chronologically sorting the intermediate statistics data files. This script uses the first sixteen (16) characters of the accounting record to sort. These characters contain the date and record type, which are unique, to assure proper sorting. Under UNIX, the UNIX sort command that is used is: `sort +0.0 -0.16`. Under NT, a CIMS for NT Sort Utility is used.

Processing Multitude Nodes

You consolidate the data with the CIMS for NT/UNIX CLI using the `proc_multi` script. Note that **Clump Support** must be enabled in the CIMS Parameter File to use this script. (See *Chapter 4, CIMS for NT/UNIX Data Consolidation* of the *CIMS for NT/UNIX User Guide*.)

This script follows these steps:

- 1 Reads `A_node.par` to get the first node to be processed.
- 2 Links the CIMS for NT/UNIX Nightly accounting and storage files for that node into the `ARSAP_DATA` directory.
- 3 Runs the CIMS for NT/UNIX Select Utility.
- 4 Runs the CIMS for NT/UNIX Merge Utility.
- 5 Removes the CIMS for NT/UNIX Nightly accounting and storage files from the `ARSAP_DATA` directory.
- 6 Moves the CIMS for NT/UNIX Merged Statistics file (`A_stat.dat`) to the CIMS for NT/UNIX Intermediate Statistics file (`A_stat.sys`).
- 7 Reads `A_node.par` for the next node to be processed
- 8 Repeats steps 2-7 until all nodes have been consolidated.

You run the CIMS for NT/UNIX Process Multiple Nodes script with the applicable command line arguments from the CIMS for NT/UNIX CLI as follows:

```
# $ARSAP_ETC/proc_multi Begin_date End_date New_acct Audit_flag Capacity_flag Range_flag Err_action
```

Begin_date Beginning of the A_select processing range in the format dd-mmm-yyyy. The keywords **yesterday**, **today**, and **tomorrow** are also valid.

End_date End of the A_select processing range. Please note this is the day *after* the last date you want included in the statistics. The format for this date is dd-mmm-yyyy. The keywords **yesterday**, **today**, and **tomorrow** are also valid

New_acct **new**—Start with a totally new statistics file

zero—Zero out current statistics file

blank—Append to current statistics file

Audit_flag **audit**—process for audit reporting in addition to resource reporting

blank—do not process for audit reporting

Capacity_flag **cap**—process for capacity reporting in addition to resource reporting

blank—do not process for capacity reporting

Range_flag **range**—turns on range file limits; links in Nightly accounting and storage files for only the dates within range specified by `begin_date` and `end_date`, plus a few days back (defaults to 3 days, controlled by `range_back` variable in `proc_multi` script). This option is recommended for more accurate reporting.

For example, when you want to process for the period 1-jan-2000 through 7-jan-2000, with the `range` option used, the script links in the files for 1-jan-2000 through 8-jan-2000, plus 29, 30, and 31-dec-1999.

blank—uses *all* `acc_<date>.dat` and `str_<date>.dat` files located in the `accounting/<nodename>` directory.

Err_action **continue**—continue processing the rest of the nodes through `A_select` and `A_merge` when processing fails on an individual node.

blank—halts all processing when an individual node fails in `A_select` or `A_merge`.



7

CIMS for NT/UNIX Advanced Features

Advanced Features	7-2
Advanced CIMS for NT/UNIX Operational Scripts	7-2
Data Window	7-4
CIMS for NT/UNIX Disk Space Management	7-5
Out-of-Sequence Management	7-7
Performance Management	7-7
CIMS for UNIX GID Method Implementation	7-8

Advanced Features

The standard operations of CIMS for NT/UNIX have been covered in the previous sections. Some additional CIMS for NT/UNIX scripts are available to handle some advanced operational needs of your environment. In addition, you might need to address some operational considerations to let CIMS for NT/UNIX meet your needs. These operational considerations involve an extended data reporting cycle, disk space management, out-of-sequence management, performance management, and GID method implementation. You need to establish a plan to ensure you are utilizing CIMS for NT/UNIX to fulfill your needs. The approaches in this chapter are guides and should be followed to obtain proper results.

Advanced CIMS for NT/UNIX Operational Scripts

The additional scripts you might need for your environment are:

CIMS for NT/UNIX Get Accounting script

CIMS for UNIX Check Pct Acct File script

CIMS for NT/UNIX Check Nightly script

CIMS for NT/UNIX Get Accounting script

The CIMS for NT/UNIX Get Accounting script retrieves the CIMS for NT/UNIX accounting and storage files from the CIMS for NT/UNIX accounting (`ARSAP_ACCDAT`) directory on each specified `ACCOUNTING_CLIENT` node. This script is useful when you do not want root to have unrestricted permissions for file transfers such as `ftp` and `rcp` and for NFS mount points. The transfer method and location of the CIMS for NT/UNIX accounting and storage files on the remote node must be specified in the CIMS for NT/UNIX Node Parameter File on the `ACCOUNTING_SERVER`. You run this script on the `ACCOUNTING_SERVER` only and you can run it as a cron job under a different user (such as a CIMS Administration User).

CIMS for UNIX Check Pacct File Script

The CIMS for UNIX Check Pacct File script checks the size of the UNIX Pacct file. You schedule this script to perform periodic checks through the UNIX clock daemon (cron). The UNIX system imposes a limit on the size of the pacct. When the usage of the root filesystem reaches 98%, the UNIX kernel will turn Process Accounting off without notification. No information is recorded until the usage of the root file system is below system threshold.

This script helps to assure that no UNIX accounting data is lost because it maintains the size of the pacct below a user specified limit. When the pacct file exceeds your designated limit, this script turns the current pacct over into a holding area in the ARSAP_HISTORY directory and re initializes the live pacct. Multiple paccts can be produced, depending on the activity on your system.

The CIMS for UNIX Runacct script processes the multiple paccts to incorporate all the UNIX accounting data into the CIMS for NT/UNIX accounting file.

The CIMS for UNIX Check Pacct Script can be run every hour from cron with the following entry in the root crontab file.

Example

```
30 * * * * /usr/arsap/etc/check_pacct
```

In this example, check_pacct is run every hour on the half-hour.

CIMS for NT/UNIX Check Nightly script

The CIMS for NT/UNIX Check Nightly script checks the CIMS for NT/UNIX accounting directory on the ACCOUNTING_SERVER node. When you are gathering CIMS for NT/UNIX accounting and storage files in a common repository, the CIMS for NT/UNIX accounting directory on the ACCOUNTING_SERVER node consists of a subdirectory for each ACCOUNTING_CLIENT node. This script searches each accounting subdirectory to determine the existence of the current day's CIMS for NT/UNIX accounting and storage files. This script logs any discrepancies in an exception log for the current day.

This script should be run once a day on the `ACCOUNTING_SERVER`. The following is an example entry for the root crontab file.

Example

```
0 6 * * * /usr/arsap/etc/check_nightly
```

Here, `check_nightly` is run every morning at 6:00 AM.

Data Window

Your reporting cycle might be daily, weekly, monthly, yearly, or a mix. When you have an ideal situation, the users log in and log out on a daily basis and do not have long-running jobs that exceed the daily accounting boundaries. When these conditions do not hold, you need to manage the data to assure these cases are recorded properly in the CIMS for NT/UNIX statistics files. You need to extend your data window to encompass the CIMS for NT/UNIX data that is processed during data consolidation. The data window is two-fold. You need to extend your date range backwards to cover multi-day logins. You also need to extend your date range forward to encapsulate long-running jobs.

The date range used for data consolidation defines the reporting period (refer to [Chapter 4, Data Consolidation Overview](#) contained in the [CIMS Chargeback NT/UNIX User Guide](#)). The reports consist of consolidated information drawn from the accounting data contained in the CIMS for NT/UNIX Nightly Accounting Files. By default, the CIMS Process Multiple Node Script will include the accounting files within the specified date range. However, additional nightly accounting files might be needed to include pertinent information for more accurate reports. This full range of accounting files, that encompasses the files in the date range, as well as those dated before and after the date range, define the data window. The breadth of the data window depends on your operational demands, but are of particular importance when project accounting is enabled, when tracking `connect_time` and when tracking `image_time`.

The accounting data records that make up the data window before the date range consist of CIMS login and switch records. These records contain the CIMS for NT/UNIX project that the user logged in under or switched to during a session. If the login or switch occurred prior to the since date of the date range, the Data Window range is used to include those days when consolidating data for the date range. For instance, if you are consolidating data for since 3-Nov-1999 and before 10-Nov-1999, you may want to include login and switch records from nightly accounting files earlier than 3-Nov-1999.

In addition to records that occurred prior to the date range, accounting records from after the date range may be needed to include image records of long-running jobs.

After long-running jobs are complete and the CIMS for NT/UNIX accounting files that contain these image records are created, you can then process the data in accordance with the appropriate reporting period. The resources used by these jobs are prorated to the reporting period. Therefore, the next reporting period will contain only the portion of the resource utilization consumed in that reporting period.

As an example, if you want to consolidate the accounting information for the month of November 1999, the reporting period would consist of usage that occurred between the `SINCE_DATE=1-Nov-1999` and `BEFORE_DATE=1-Dec-1999`. If you want to define a Data Window of 7 days prior and 7 days after the reporting period, you would set the environment variables `RANGE_BACK=7` and `RANGE_AHEAD=7` in the CIMS for NT/UNIX Configuration Parameter File and include the argument "range" when running the CIMS Process Multiple Nodes Script.

CIMS for NT/UNIX Disk Space Management

The disk space needed for the accounting files is always a concern. The activity on your system governs the size of the accounting files and thus the amount of space needed. The best approach is to develop a plan to manage the accounting files that includes cleaning up and/or archiving extraneous accounting files.

Cleanup History Option

The ARSAP_HISTORY directory contains historical raw accounting files. These files include the raw pacct, wtmp, A_dbacct.sys, A_db2acct.sys, A_activity, NQS Accounting and UNIX Print Accounting files. Once the information in these files has been formatted into the CIMS Nightly Accounting File, they can be safely removed. It is recommended that the raw files be saved until the nightly accounting files have been consolidated at the end of the reporting period. In this way, if you need to re-create a nightly accounting file, you can do so using the Redo Nightly Script.

The CIMS for NT/UNIX Nightly script performs historical data cleanup. The CIMS for NT/UNIX Configuration Parameter file contains the CLEANUP_HISTORY and CLEANUP_AGE environment variables. Set CLEANUP_HISTORY to **Y** to perform cleanup and set CLEANUP_AGE to the number of days of historical data you want to keep in the ARSAP_HISTORY directory. The CIMS for NT/UNIX Nightly script will automatically remove the historical data beyond this age.

Archiving Data

Archiving the historical data is another option for disk space management. You can use this option when you are transferring your CIMS for NT/UNIX accounting and storage files to a designated ACCOUNTING_SERVER. The ACCOUNTING_SERVER is the central repository for ACCOUNTING_CLIENT data. This option is useful when the ACCOUNTING_CLIENTs have limited disk space and the ACCOUNTING_SERVER has enough disk space to handle all client data. In addition, you might want to perform regular tape backups on the ACCOUNTING_SERVER. Therefore, you can periodically remove the client data without concern about unrecoverable data.

The CIMS for NT/UNIX Nightly Send script performs data archiving. The CIMS for NT/UNIX Configuration Parameter file contains the CREATE_BACKUP environment variable. You set CREATE_BACKUP to **Y**. This script tape archives (tar) the raw accounting files, the CIMS for NT/UNIX accounting files and the CIMS for NT/UNIX Storage files. This tarred file is transferred to the ARSAP_SERVER with individual copies of the CIMS for NT/UNIX Nightly Accounting and Storage Files.

Out-of-Sequence Management

The CIMS for NT/UNIX Select Utility processes the CIMS for NT/UNIX accounting files in chronological order. This utility generates out-of-sequence errors when records in one CIMS for NT/UNIX accounting file are dated after records in a subsequent CIMS for NT/UNIX accounting file. This utility automatically handles minor and major out-of-sequence errors.

Minor out-of-sequence errors are defined by a tolerance of up to three (3) seconds. These errors are indicated by warning messages. The offending time is adjusted and processing continues.

Major out-of-sequence errors also are ignored according to the `/IGNORE_BEFORE` and `/IGNORE_AFTER` qualifiers. These errors are indicated by ignore warning messages. These qualifiers are present by default with a before date of current year - 1, and the after date of current year + 1. These errors are typically attributed to a corrupt system accounting file.

When out-of-sequence errors fall between these two categories, you have two choices to correct this situation. You can edit the CIMS for NT/UNIX accounting files and manually delete the offending records. The second corrective measure is to sort the CIMS for NT/UNIX accounting files into a single accounting file.

Performance Management

CIMS for UNIX performance degradation typically occurs when CIMS for UNIX is installed in an NFS environment. CIMS for UNIX utilizes the NFS environment to allow easier maintenance for the CIMS for UNIX administrator. However, performance problems can occur in two situations: project login and data consolidation. These known areas of concern are attributed to high volume network traffic that might occur in an NFS environment.

Slow Project Login

The CIMS for UNIX Authorization and Validation files are used when users log in to a project. When these files are located in an NFS mount point, CIMS for UNIX must go over the network to access these files. When there is a high amount of network traffic, the users might experience a lag in the project login procedure.

Slow Data Consolidation

You should perform data consolidation on a local mount point of the node. The CIMS for UNIX Select and Merge utilities constantly reference data files in the `ARSAP_DATA` directory. When the `ARSAP_DATA` directory is on an NFS mount point, the data consolidation is contending with network traffic.

When data consolidation is needed for many nodes, you should most likely use the CIMS for UNIX Super Merge Utility. This utility is explained in *CIMS for NT/UNIX Super Merge Utility* on page 4-12.

CIMS for UNIX GID Method Implementation

There are two choices for GID Method: `GROUP` or `PROJECT`. The GID Method value is a CIMS for NT/UNIX system-wide parameter and defaults to `GROUP`. The CIMS for NT/UNIX Format Utility assigns user name and group names to image records based on the UID and GID in each record in the UNIX `Pacct` file. When the GID Method is `GROUP`, CIMS for UNIX retrieves the group name from the `/etc/group` file. When the GID Method is `PROJECT`, CIMS for UNIX searches for the GID in the project records in the CIMS for UNIX Authorization File and assigns the project name to the group name.

The project GID Method is used in conjunction with the `/UNKNOWN_PROJECT=GROUP` option in the CIMS for NT/UNIX Select Utility. You use this method when unique terminal names are not tied to project login records with the image records that represent the activity of that session (such as when it was a background job with no terminal).

CIMS for NT/UNIX Messages

Introduction to CIMS for NT/UNIX Messages	8-3
Format of CIMS for NT/UNIX Messages	8-4
Example Error Message	8-6
Returned Error Status	8-7
CIMS for NT/UNIX RTL Messages	8-7
GEJAC RTL Messages	8-16
CIMS for UNIX DATABASE RTL Messages	8-18
CIMS for UNIX LOGIN Utility Messages	8-22
CIMS for UNIX X-Window LOGIN Utility Messages	8-25
CIMS for UNIX SWITCH Utility Messages	8-27
CIMS for NT/UNIX SAMPLER Utility Messages	8-30
CIMS for NT/UNIX CONVERT Utility Messages	8-32
CIMS for UNIX Oracle Accounting Daemon Messages	8-35
CIMS for UNIX Oracle Project LOGIN Utility Messages	8-36
CIMS for UNIX Oracle Project SWITCH Utility Messages	8-37

CIMS for UNIX IBM DB2 Accounting Daemon Messages	8-38
CIMS for UNIX GUI Messages	8-40
Diagnostics	8-40
Error Messages	8-47
CIMS for UNIX GRAPH Utility Messages	8-48
Diagnostics	8-48
Error Messages	8-52
CIMS for NT LOGIN Utility Messages	8-53

Introduction to CIMS for NT/UNIX Messages

CIMS for NT/UNIX utilizes a centralized message facility. The CIMS for NT/UNIX messages are produced for various levels of severity: success, informational, warning, error and fatal. Each CIMS for NT/UNIX Utility defines messages that pertain to its operation. In addition, CIMS for NT/UNIX utilizes run-time libraries (rtls) that are common routines and these routines produce messages. Some CIMS for NT/UNIX messages occur due to a system error so the message contains the system error, too.

The CIMS for NT/UNIX utilities can be grouped into four categories: interactive, non-interactive, and graphical user interface (GUI) and Windows applications. The interactive utilities interface with you through the CIMS for NT/UNIX Command Line Interface (CLI). The non-interactive utilities do not provide a CLI because they are shell routines or daemons. The GUI and Windows applications interface with you but provide their messages via diagnostic windows.

This message manual presents the messages generated by the CIMS for NT/UNIX rtls, the non-interactive utilities, and the GUI and Windows applications. The messages for the interactive utilities are described in that routines on-line help facility.

■ CIMS for NT/UNIX Messages

Format of CIMS for NT/UNIX Messages

The non-interactive utilities are:

- CIMS for NT/UNIX CONVERT
- CIMS for NT/UNIX SAMPLER
- CIMS for UNIX LOGIN
- CIMS for UNIX X-window LOGIN
- CIMS for UNIX SWITCH
- CIMS for UNIX Oracle Accounting Daemon
- CIMS for UNIX Oracle Project LOGIN
- CIMS for UNIX Oracle Project SWITCH
- CIMS for UNIX IBM DB2 Accounting Daemon

The GUI and Windows applications are:

- CIMS for UNIX GUI
- CIMS for UNIX GRAPH
- CIMS for NT LOGIN

Format of CIMS for NT/UNIX Messages

The messages displayed by CIMS for NT/UNIX have the following format:

```
%FACILITY-L-IDENT, text  
[errno = unix-errno, errno-text] (unix only)
```

The *FACILITY* is the CIMS for NT/UNIX component name. The facility names are:

ARSAP	CIMS for NT/UNIX Run-Time Library
AUTHORIZE	CIMS for NT/UNIX AUTHORIZE
DBADB2	CIMS for UNIX DB2
DBAO	CIMS for UNIX Oracle Accounting Daemon
DBPLO	CIMS for UNIX Oracle Project LOGIN
DBPSO	CIMS for UNIX Oracle Project SWITCH
DBRTL	CIMS for UNIX DATABASE Run-Time Library
GEJAC	GEJAC Run-Time Library
FRMT	CIMS for NT/UNIX Format
LOGIN	CIMS for UNIX LOGIN
MERGE	CIMS for NT/UNIX MERGE
RATES	CIMS for NT/UNIX RATES
RPT	CIMS for NT/UNIX REPORT
SAMPLER	CIMS for NT/UNIX SAMPLER
SELECT	CIMS for NT/UNIX SELECT
SETUP	CIMS for NT/UNIX SETUP
SMERGE	CIMS for NT/UNIX SMERGE
SWITCH	CIMS for UNIX SWITCH
VALID	CIMS for NT/UNIX AUTHORIZE

The *L* is a severity level indicator with one of the following values:

- S Success informs you that the component performed your request successfully.
- I Information informs you that the component performed your request successfully and provides some additional meaningful information.
- W Warning informs you that the component could have performed some, but not all, of its function and you may need to verify the results.
- E Error informs you that the component's output is incorrect but the component can continue execution.
- F Fatal informs you that the component cannot continue execution because the error is so severe.

The *IDENT* is an abbreviation of the message text. The message descriptions in this manual are alphabetized by this abbreviation.

The *text* is an explanation of the message.

The *unix-errno* is the value of `errno` returned by UNIX.

The *errno-text* is the text associated with the UNIX `errno`.

Example Error Message

An example of a CIMS for UNIX message is:

```
%ARSAP-F-OPENFAIL, error opening file - A_uaf.sys
UNIX errno = 2, No such file or directory
```

where ARSAP is the *FACILITY* name, F is the severity level, OPENFAIL is the *IDENT*, "error opening file-A_uaf.sys" is the *text*, 2 is the *unix-errno*, and "No such file or directory" is the *errno-text*.

Returned Error Status

When a UNIX shell interpreter executes a command or application program, the exit status of the process is returned in a shell variable (that is, `$?` for `sh` or `ksh` and `$status` for `csh`). The severity level of the last signaled message is returned to the shell by the CIMS for UNIX Utilities. You can use the shell status variable to determine the severity level of the last error message.

Values returned by the severity levels

S	Success	1
I	Info	3
W	Warning	0
E	Error	2
F	Fatal	4

CIMS for NT/UNIX RTL Messages

The CIMS for NT/UNIX RTLs generate the following messages:

ARSAPVER, CIMS for NT/UNIX product version V04.0

Explanation: A CIMS for NT/UNIX Utility displayed the CIMS for NT/UNIX product version in response to the utility's `IDENT` command.

User Action: None.

BUFTOSMALL, buffer too small for input

Explanation: A buffer used to retrieve input was too small for an user's input.

User Action: Specify a smaller response when possible, otherwise report the problem to the CIMS Lab.

CMDAMBIG, ambiguous command verb <string>

Explanation: A command was entered in response to a utility's prompt, and the verb indicated by <string> was ambiguous.

User Action: Enter enough characters of the command verb so as to make it unambiguous.

CMDUNKNOWN, unknown command verb

Explanation: A command was entered in response to a utility's prompt, but the command is not a valid command in that utility.

User Action: Enter a valid command.

CONFLICT, illegal combination of command elements—<string>,<string>

Explanation: A command was entered in response to a utility's prompt, and the elements indicated are conflicting.

User Action: Enter the command without conflicting elements.

CREATE, error creating file—<string>

Explanation: An error was encountered during the creation of the specified file.

User Action: This message is accompanied by a system call error. Resolve the error by examining the cause of the system call error. Check the owner and group permissions on the directory where the file is to be created.

ESTABLISH, error establishing facility

Explanation: An internal software error has occurred.

User Action: Report this to the CIMS Lab.

EXPIRED, this copy of CIMS for NT/UNIX has expired as of <date>

Explanation: The CIMS for NT/UNIX software has expired as of the indicated date.

User Action: If you have a newer CIMS for NT/UNIX license PAK, install it. Otherwise, report this to the CIMS Lab.

FDOPENERR, error returned from fdopen

Explanation: An error has occurred during an fdopen operation on a file.

User Action: This message is accompanied by a system call error. Resolve the error by examining the cause of the system call error. Check the owner and group permissions on the file.

FILNOTCONV, file not converted—<string>version found—<vers>, version expected—<vers>

Explanation: The file indicated by the <string> has not been converted to the latest version.

User Action: You must use the CIMS for NT/UNIX CONVERT Utility to convert the file so it can be accessed properly by this utility.

INTCONERR, internal inconsistency error

Explanation: An internal software error has occurred.

User Action: Report this to the CIMS Lab.

INVPASSWD, invalid password

Explanation: An invalid password was entered.

User Action: Use only characters that are valid for passwords.

INVPROJNAM, invalid project name

Explanation: An invalid project name was entered.

User Action: Use only characters that are valid for project names.

INVRECTYPE, invalid record type detected

Explanation: An invalid record type was detected in the CIMS for NT/UNIX Authorization File.

User Action: Report this to the CIMS Lab.

INVTIME, invalid time

Explanation: A CIMS for NT/UNIX routine detected an invalid date/time.

User Action: This error occurred entering a time on the command line. The CIMS CLI accepts dates in the format DD-MMM-YYYY.

IOERROR, I/O error from studio routine—<string>

Explanation: An error was returned by one of the standard UNIX I/O routines.

User Action: Examine the protection of the file indicated in the message to determine if access is prohibited. Check the owner and group permissions of the file.

KEYAMBIG, ambiguous keyword—<string>

Explanation: A command was entered in response to a utility's prompt, and the keyword indicated by <string> was ambiguous.

User Action: Enter enough characters in the keyword to make it unique.

KEYUNKNOWN, unknown keyword—<string>

Explanation: A command was entered in response to a utility's prompt, and the keyword indicated is unknown.

User Action: Enter a valid keyword.

KEYWRDREQ, keyword value is required—<string>

Explanation: A qualifier has been entered on the command line that requires a keyword.

User Action: Check on-line help for the valid keywords for this qualifier and enter the command again with an appropriate keyword.

KWDVALREQ, keyword requires a value—<string>

Explanation: A keyword has been entered on the command line that requires a value, but none was supplied.

User Action: Enter the command with appropriate value for the indicated keyword.

LOCKERR, error locking file—<file-name>

Explanation: An error has occurred during a lock operation on a file.

User Action: This message is accompanied by a system call error. Resolve the error by examining the cause of the system call error.

MATCHQUOT, matching quote not found—<string>

Explanation: A command was entered in response to a utility's prompt, and a matching quote was not found.

User Action: Enter the command correctly with a matching quote.

MAXPARAM, too many parameters—reenter with fewer parameters

Explanation: A command was entered in response to a utility's prompt containing too many parameters.

User Action: Enter the command with the correct number of parameters.

MEMALLOC, error allocating memory

Explanation: An error was encountered while allocating memory.

User Action: Determine the necessary resource quota and expand the quota.

NODEFPROJ, error reading DEFAULT project

Explanation: An error has occurred reading the DEFAULT project record from the CIMS for UNIX Authorization File.

User Action: Verify that the CIMS for NT/UNIX Authorization File is accessible and no file corruption has occurred.

NODEFUSR, error reading DEFAULT user

Explanation: An error has occurred reading the DEFAULT user record from the CIMS for NT/UNIX Authorization File.

User Action: Verify that the CIMS for NT/UNIX Authorization File is accessible and no file corruption has occurred.

NOLIST, qualifier or keyword does not take a list of values—<string>

Explanation: A command qualifier or a keyword that does not take a list of values was specified with a value list.

User Action: Enter the command again correctly specifying the qualifier or keyword.

NOPAREN, value improperly delimited—supply parenthesis—\`<string>`\

Explanation: A command value supplied as part of a parenthesized value list is missing a parenthesis.

User Action: Enter the command with the missing parenthesis.

NOPWIDENT, error getting passwd entry

Explanation: CIMS for UNIX could not retrieve an entry from the UNIX Passwd File.

User Action: This message is accompanied by a system call error. Resolve the error by examining the cause of the system call error.

NOREPKWD, error repeating keyword `<key>` for qualifier `<qual>`

Explanation: You repeated a keyword on the command line.

User Action: Enter the specified keyword only once.

NOREQUAL, error repeating qualifier `<qual>`

Explanation: You repeated the indicated qualifier on the command line.

User Action: Enter the specified qualifier only once.

NOVALIDENT, no entries in the CIMS for NT/UNIX Validation File for USER `<string>`

Explanation: The user requested that a list of authorized projects be displayed and no entries were found in the CIMS for NT/UNIX Validation File for that user.

User Action: None.

NOVALUE, qualifier does not take a value—`<string>`

Explanation: A command qualifier that does not take a value was specified with a value.

User Action: Enter the command and qualifier with no value.

NULVAL, missing qualifier value—respecify

Explanation: A command qualifier that requires a qualifier value was specified, but no value was found.

User Action: Enter the command correctly specifying the qualifier and its value.

NUMERIC, invalid numeric value—supply an integer—\

Explanation: The command value must be an integer.

User Action: Enter the command correctly specifying the value as an integer.

OPENERR, error opening file—\

Explanation: An error has occurred during an open operation on a file.

User Action: This message is accompanied by a system call error. Resolve the error by examining the cause of the system call error.

PARMDEL, invalid parameter delimiter—check use of special characters \

Explanation: A command was entered incorrectly. The incorrect input is indicated by the <string>.

User Action: Enter the command using correct syntax.

POSERR, error positioning file—\

Explanation: An error has occurred during a seek operation on a file.

User Action: This message is accompanied by a system call error. Resolve the error by examining the cause of the system call error.

PUTENV, error expanding environment

Explanation: An error was encountered while expanding the environment.

User Action: Determine the necessary resource quota and expand the quota.

QUALAMBIG, ambiguous command qualifier-<string>

Explanation: A command qualifier was entered that was ambiguous.

User Action: Enter enough characters in the qualifier to make it unique.

QUALUNKNOWN, unrecognized command qualifier-<string>

Explanation: A command qualifier was entered that is unknown.

User Action: Enter the command correctly specifying all qualifiers.

QUALVALREQ, qualifier value required-<string>

Explanation: A command qualifier was specified that requires a value, but no value was specified.

User Action: Enter the command correctly specifying the qualifier and its value.

READENV, error reading <environ-variable> from environment

Explanation: CIMS for NT/UNIX was not able to read the indicated environment variable.

User Action: Ensure that the environment variable indicated has been placed in the process environment.

READERR, error reading file-<string>

Explanation: An error has occurred during a read operation on a file.

User Action: This message is accompanied by a system call error. Resolve the error by examining the cause of the system call error.

RENAMERR, error renaming file from <name-1> to <name-2>

Explanation: The Utility could not rename the name-1 file to name-2.

User Action: The system call error should indicate why the file could not be renamed. Correct the error and try the command again.

SCANFERR, scanf error

Explanation: An error has occurred during a scanf operation on a file.

User Action: This message is accompanied by a system call error. Resolve the error by examining the cause of the system call error. When you cannot resolve the error, report this to the CIMS Lab.

STINXERR, error generating stat index tree

Explanation: An invalid index value into one of the CIMS for NT/UNIX mapping files has been detected.

User Action: This error usually occurs if a new copy of a CIMS for NT/UNIX mapping file has been created after the data consolidation phase. You will need to run data consolidation again.

TRUNERR, error truncating file-<string>

Explanation: An error has occurred during a truncate operation on a file.

User Action: This message is accompanied by a system call error. Resolve the error by examining the cause of the system call error.

TTYERR, error modifying terminal

Explanation: An error has occurred while performing an operation on a terminal.

User Action: This message is accompanied by a system call error. Resolve the error by examining the cause of the system call error.

UNLOCKERR, error unlocking file-<string>

Explanation: An error has occurred during an unlock operation on a file.

User Action: This message is accompanied by a system call error. Resolve the error by examining the cause of the system call error.

WRITEERR, error writing to file-<string>

Explanation: An error has occurred while performing a write operation on a file.

User Action: This message is accompanied by a system call error. Resolve the error by examining the cause of the system call error.

GEJAC RTL Messages

The CIMS for NT/UNIX GEJAC RTLs generate the following messages:

CHKREQ, checksum qualifier is required

Explanation: The /CHECKSUM qualifier is required on the ADD command.

User Action: Enter the command with the /CHECKSUM qualifier.

INVCHECKSUM, checksum does not validate

Explanation: You have entered one or more items from the license PAK incorrectly.

User Action: Reenter the information from the license PAK. The license PAK fields are case sensitive and must be entered exactly as they appear in the CIMS License PAK.

NOLICINFO, no license information found for this machine

Explanation: You have no license information entered for this machine.

User Action: Assuming you have a CIMS License PAK for this machine, on UNIX, run the `A_add_license` script in the CIMS for UNIX Scripts Directory and enter the license information. On NT, run the `NT_add_license` script in the CIMS for NT ETC directory and enter the license PAK information.

OPENERR, error opening file-<name>

Explanation: An error has occurred during an open operation on a file.

User Action: This message is accompanied by a system call error. Resolve the error by examining the cause of the system call error.

READERR, error reading file-<name>

Explanation: An error has occurred during a read operation on a file.

User Action: This message is accompanied by a system call error. Resolve the error by examining the cause of the system call error.

SWHASEXP, software has expired

Explanation: CIMS for NT/UNIX has expired on this machine.

User Action: When you have a CIMS for NT/UNIX update, install it. Else, report this to the CIMS Lab.

UNLICMACH, unlicensed machine

Explanation: You are not licensed to run CIMS for NT/UNIX on this machine.

User Action: Run CIMS for NT/UNIX on a licensed machine or contact the CIMS Lab for information about licensing the machine.

UNLICOPT, unlicensed option

Explanation: You are not licensed to run this Utility on this machine.

User Action: Run the Utility on a licensed machine or contact the CIMS Lab for information about licensing the option required to run the Utility.

UNLICVER, unlicensed version

Explanation: You are not licensed to run this version of CIMS for NT/UNIX on this machine.

User Action: Run a version of CIMS for NT/UNIX that is licensed on this machine or contact the CIMS Lab for information about licensing this version of CIMS for NT/UNIX.

CIMS for UNIX DATABASE RTL Messages

The CIMS for UNIX Database RTLs generate the following messages:

AUTHFAIL, project authorization failure

Explanation: A project authorization failure has been detected that is caused by an illegal project just entered, an illegal default project, an illegal password, or a user/project validation failure.

User Action: Check to see when the project name and password have been correctly entered and when the user is authorized to access this project.

COMERR, error committing transaction to Oracle Database

Explanation: An error has been returned committing an update to the ARSAP_PROJECT table. This error can only occur in environments using the CIMS for NT/UNIX Project Accounting feature.

User Action: Verify that the ARSAP_PROJECT table exists and is accessible to the ORACLE user used by the ORACLE Accounting Daemon to connect to the ORACLE instance.

CONNERR, error connecting to Oracle Database

Explanation: An error has occurred in connecting to the Oracle Instance.

User Action: Verify that the ORACLE username and password used by the CIMS for NT/UNIX ORACLE Accounting Daemon are valid. These values are entered into the CIMS for NT/UNIX Database Instance Parameter File, (A_dbinst.sys) using the SETUP Utility.

CURSCLOSE, error closing Oracle Database cursor data area

Explanation: An error has occurred in closing the Oracle Database cursor data area.

User Action: This error occurred while shutting down the CIMS ORACLE Accounting Daemon. The daemon will continue to shutdown. Note that cursors are not used in ORACLE 8i.

CURSOROPEN, error opening Oracle Database cursor data area

Explanation: An error has occurred opening the Oracle Database cursor data area.

User Action: Cursors are not used in ORACLE 8i. In earlier versions, this error indicates a problem in allocation space for the cursor data area within ORACLE.

INTERR, internal error accessing Oracle Database

Explanation: An unexpected error condition occurred when calling an one of the OCI routines, odescr(), odefin(), oexfet(), or ofen().

User Action: Contact CIMS Lab Technical Support.

INVSQLE, invalid SQL statement

Explanation: An invalid SQL statement was passed to a CIMS Database RTL subroutine.

User Action: The SQL statements used by the CIMS ORACLE Accounting Daemon are internal to the Utility and this message should never occur. If you see this message contact CIMS Lab Technical Support.

LOGOFFERR, error logging off from Oracle Database

Explanation: An error has occurred logging off from the Oracle Database.

User Action: An error was returned from the ORACLE OCI `ologof()` while shutting down the CIMS ORACLE Accounting Daemon. Shutdown will continue.

LOGONERR, error logging on to Oracle Database

Explanation: An error has occurred logging on to the Oracle Database.

User Action: Verify that the ORACLE username and password used by the CIMS for NT/UNIX ORACLE Accounting Daemon are valid. These values are entered into the CIMS for NT/UNIX Database Instance Parameter File, (`A_dbinst.sys`) using the SETUP Utility.

PARAMERR, error fetching operating parameters

Explanation: An error has occurred fetching operating parameters.

User Action: Verify the values in the entry for this instance in the CIMS Database Instance File. Values can be view using the `REPORT/DBINSTANCE` command in the CIMS SETUP Utility. This error indicates that one of the fields in the Database Instance Record is empty. Try to remove and then re-enter the with the `REMOVE/DBINSTANCE` and `ADD/DBINSTANCE` commands.

PARSERR, error parsing SQL statement

Explanation: An error has occurred in parsing the SQL statement.

User Action: This error indicates a failure in the ORACLE OCI routine `oparse()`. The error will most often be reported to the CIMS ORACLE Accounting Daemon Log File during startup of the daemon when Project Accounting is enabled in the CIMS Parameter File, but the `ARSAP_PROJECT` table has not been created in the ORACLE Instance. If

you are not doing Project Accounting for ORACLE usage, this error can be ignored. If this error occurs after startup, the Accounting Daemon will shutdown. Try to re-start the daemon.

PROJECTEXP, project has expired

Explanation: User is attempting to enter a project account that has expired.

User Action: Extend the expiration date using the CIMS AUTHORIZE Utility.

READERR, error reading row from Oracle Database

Explanation: Oracle tried to read the indicated row, but could not.

User Action: This error indicates the CIMS ORACLE Accounting Daemon could not read one of the CIMS data files. Verify that the owner of the daemon process has access to the CIMS ARSAP_DATA files.

CIMS for UNIX LOGIN Utility Messages

The CIMS for UNIX LOGIN Utility generates the following messages:

AUTHFAIL, Authorization failure

Explanation: LOGIN detected a project authorization failure. The failure is caused by an illegal project entered by the user, an illegal default project, an illegal password, or a user/project validation failure.

User Action: Stop trying to access a project for which you are not authorized or correctly enter the project name and password.

CHDIR, error changing directory to <directory>

Explanation: LOGIN was not able to change the current directory of the process before returning control to the user.

User Action: Contact the CIMS for UNIX Administrator to have him verify that the directory indicated exists and that the user has access to it.

EXECDEF, standard shell being executed

Explanation: LOGIN was not able to exec the shell specified in the user's account in the CIMS for UNIX Authorization File and is now attempting to exec the standard shell /bin/sh.

User Action: Resolve the problem indicated by the previous error message.

EXECERR, EXEC error of shell \

Explanation: LOGIN was unable to exec the indicated shell.

User Action: Examine the indicated shell to determine why LOGIN could not exec it.

FATERRSYS, fatal error overridden, "root" allowed to login

Explanation: LOGIN has detected a fatal error for the root user account or the project account used by root user, but has allowed the login since the username is root.

User Action: Fix the problem that caused LOGIN to display the fatal error or disable CIMS for UNIX until the problem is rectified (letting users access the system).

INVDEFPRO, invalid default project

Explanation: The user account has been assigned a default project that is not a valid project account.

User Action: Enter a valid default project account in the CIMS for UNIX Authorization File for the user account. Either the current default project contains invalid project characters or the user account is not authorized to use the default project account assigned to him.

NEXUID, UID <uid> not found in password file

Explanation: An entry was not found in the UNIX Passwd File corresponding to the UID of the user executing the LOGIN Utility.

User Action: Verify that the UNIX Passwd File contains an entry that correctly specifies the UID for the user executing the LOGIN Utility.

NOENTRIES, no entries in CIMS for NT/UNIX Validation File for this user

Explanation: LOGIN could not find any entries in the CIMS for UNIX Validation File when displaying the projects a user is authorized to access.

User Action: None.

PROJCTEXP, project expired

Explanation: The PROJECT account that you attempted to switch to has expired.

User Action: Stop trying to use the expired project or have the CIMS for UNIX Administrator extend the expiration date.

PUTENV, error expanding environment

Explanation: LOGIN was not able to obtain enough space to expand environment.

User Action: Increase the resource limit that prohibited expanding the environment.

RETCOUEXC, Retry count exceeded

Explanation: The project prompt retry count has been exceeded. You have three tries to enter a valid project name.

User Action: Stop trying to access a project for which you are not authorized or correctly enter the project name and password.

SETEUID, error setting effective UID

Explanation: LOGIN was not able to set the effective UID of the process before returning control to the user.

User Action: Contact the CIMS for UNIX Administrator to verify that the LOGIN Utility is properly installed. The CIMS for UNIX LOGIN Utility, `A_login` must be owned by the root user and the file permissions of the executable must be 4555.

SETGID, error setting GID to <gid>

Explanation: LOGIN was not able to set the GID of the process before returning control to the user.

User Action: Contact the CIMS for UNIX Administrator verify that the LOGIN Utility is properly installed. The CIMS for UNIX LOGIN Utility, `A_login` must be owned by the root user and the file permissions of the executable must be 4555.

SETGROUPS, error initializing group access list for <user>

Explanation: LOGIN was not able to initialize the group access list.

User Action: Contact the CIMS for UNIX Administrator to verify that the LOGIN Utility is properly installed. The CIMS for UNIX LOGIN Utility, `A_login` must be owned by the root user and the file permissions of the executable must be 4555.

CIMS for UNIX X-Window LOGIN Utility Messages

The CIMS for UNIX X-Window LOGIN Utility does not display error messages like other CIMS for UNIX Utilities. Instead, a value is returned that can be evaluated in the Xsession script so that the appropriate action for your site can be taken.

Values returned by the CIMS for UNIX X-Window LOGIN Utility and their meanings follow:

0	user exited without entering project
1	system error
2	successful login execution
3	successful logout execution
20	error setting process UID
30	error returned from CIMS for UNIX license validation
51	error setting CIMS for UNIX environment variables
52	error getting passwd record for user
53	error opening CIMS for UNIX Authorization File
54	error getting CIMS for UNIX user record from Authorization File

55	error getting CIMS for UNIX setup record from the CIMS for UNIX Parameter File
56	project name too long, limit 96 characters
57	invalid project name entered
58	error getting CIMS for UNIX project record from Authorization File
59	no project exists by project name entered
60	error opening CIMS for UNIX Validation File
61	user not validated for project
62	error writing project record to CIMS for UNIX Authorization File
63	invalid user/project combination
64	error getting ARSAP_DATA environment variable
65	error allocating memory
66	error opening CIMS for UNIX Activity File
67	error positioning CIMS for UNIX Activity File descriptor
68	error writing to CIMS for UNIX Activity File
69	project outside valid start and end time
70	error status returned by CUSTOM validation routine
71	error returned from setgid
72	error returned from chdir
75	unknown reason for callback

80

invalid arguments to A_login_xm

CIMS for UNIX SWITCH Utility Messages

The CIMS for UNIX SWITCH Utility generates the following messages:

AUTHFAIL, Authorization failure

Explanation: SWITCH detected a project authorization failure and did not change to the new project. The failure is caused by an illegal project entered by the user, an illegal default project, an illegal password, or a user/project validation failure.

User Action: Stop trying to switch to a project for which you are not authorized to access or correctly enter the project name and password.

CHDIR, error changing directory to <directory>

Explanation: SWITCH was not able to change the current directory of the process before returning control to the user.

User Action: Contact the CIMS for UNIX Administrator to have him verify that the directory indicated exists and that the user has access to it.

EXECDEF, standard shell being executed

Explanation: SWITCH was not able to exec the shell specified by the "SHELL" environment variable and is now attempting to exec the standard shell "/bin/sh".

User Action: Determine the value of the user's SHELL environment variable and correct the problem with exec'ing this shell.

EXECERR, EXEC error of shell \<shell>\

Explanation: SWITCH was unable to exec the indicated shell.

User Action: Examine the indicated shell to determine why the shell could not be exec'ed.

NEXUID, UID <uid> not found in password file

Explanation: An entry was not found in the UNIX Passwd File corresponding to the UID of the user executing the SWITCH Utility.

User Action: Verify that the UNIX Passwd File contains an entry that correctly specifies the UID for the user executing the SWITCH Utility.

NOENTRIES, no entries in CIMS for UNIX Validation File for this user

Explanation: SWITCH could find any entries in the CIMS for UNIX Validation File when displaying the projects a user is authorized to access.

User Action: None.

NOSWITCH, cannot switch projects because of previous error

Explanation: An error occurred previous to this one that prevented SWITCH from changing projects.

User Action: Resolve problem indicated by the previous error message.

PROACCFN, Project accounting disabled on this node

Explanation: Project Accounting is disabled.

User Action: When you want to use CIMS for UNIX Project Accounting, you must enable it using the CIMS for UNIX SETUP Utility. See the *CIMS Chargeback NT/UNIX User Guide* for more information on project accounting.

PROACCFU, Project accounting disabled for this user

Explanation: The CIMS for UNIX Project Accounting feature is disabled for your username account.

User Action: When you want to use the Project Accounting feature of CIMS for UNIX, the CIMS for UNIX Administrator must enable it for your username account.

PROJCTEXP, project expired

Explanation: The PROJECT account that you attempted to switch to has expired.

User Action: Stop trying to use the expired project or have the CIMS for UNIX Administrator extend the expiration date.

PROSWIDFU, Project switching disabled for this user

Explanation: The CIMS for UNIX Project Switching feature is disabled for your username account.

User Action: When you want to use the Project Switching feature of CIMS for UNIX, the CIMS for UNIX Administrator must enable it for your username account.

PUTENV, error expanding environment

Explanation: SWITCH was not able to obtain enough space to expand environment.

User Action: Increase the resource limit that prohibited expanding the environment.

RETCOUEXC, Retry count exceeded

Explanation: The project prompt retry count has been exceeded. When you entered the project on the same line as the SWITCH command, then you have one try. When SWITCH prompted you for the project, then you get three tries.

User Action: Stop trying to switch to a project for which you are not authorized to access or correctly enter the project name and password.

SETEUID, error setting effective UID

■ CIMS for NT/UNIX Messages

CIMS for NT/UNIX SAMPLER Utility Messages

Explanation: SWITCH was not able to set the effective UID of the process before returning control to the user.

User Action: Contact the CIMS for UNIX Administrator to have him verify that the SWITCH Utility is properly installed. The CIMS for UNIX SWITCH Utility, (A_switch) must be owned by the root user and the file permissions of the executable must be 4555.

SETGID, error setting GID to <gid>

Explanation: SWITCH was not able to set the GID of the process before returning control to the user.

User Action: Contact the CIMS for UNIX Administrator to have him verify that the SWITCH Utility is properly installed. The CIMS for UNIX SWITCH Utility, (A_switch) must be owned by the root user and the file permissions of the executable must be 4555.

SETGROUPS, error initializing group access list for user <user>

Explanation: SWITCH was not able to initialize the group access list.

User Action: Contact the CIMS for UNIX Administrator to have him verify that the SWITCH Utility is properly installed. The CIMS for UNIX SWITCH Utility, (A_switch) must be owned by the root user and the file permissions of the executable must be 4555.

CIMS for NT/UNIX SAMPLER Utility Messages

The CIMS for NT/UNIX SAMPLER Utility generates the following messages:

CHDIRFAIL, cannot change to directory <dir>

Explanation: SAMPLER has encountered an error while attempting to move to the indicated directory.

User Action: The errno status was indicated in the error message. Review the error indicated by errno to resolve the problem.

NOENTRY, no entry in A_storage.par file

Explanation: There are no directory tree entries in the CIMS for UNIX Storage Parameter File.

User Action: Enter at least one directory tree to sample in the Storage Parameter File.

OPENFAIL, error opening file—<file>

Explanation: SAMPLER has detected an error when opening the CIMS for NTI UNIX Storage Parameter file or the indicated directory with the UNIX opendir system call.

User Action: The UNIX errno status was indicated in the error message. Review the error indicated by errno to resolve the problem.

PARAMINV, invalid Storage Parameter File specification—“<spec>”

Explanation: An entry in the CIMS for NTI UNIX Storage Parameter File is not a directory.

User Action: Enter only valid directory names in the Storage Parameter File.

POSERR, error positioning file—<file>

Explanation: SAMPLER has detected an error when positioning a file descriptor.

User Action: The errno status was indicated in the error message. Review the error indicated by errno to resolve the problem.

READENV, error reading <environ-variable> from environment

Explanation: SAMPLER was not able to read the indicated environment variable from the environment.

User Action: Ensure that the environment variable indicated has been placed in the process' environment.

SAMPLDIR, beginning to sample directory <dir>

Explanation: SAMPLER is beginning to sample the indicated directory.

User Action: None.

SMPLRDONE, A_sampler completed

Explanation: SAMPLER has completed processing.

User Action: None.

STATFAIL, error getting file statistics—“<file>”

Explanation: SAMPLER tried to stat the indicated file and failed.

User Action: The errno status was indicated in the error message.
Review the error indicated by errno to resolve the problem.

STATFSERR, failing to stat file system for <file-system>

Explanation: SAMPLER tried to get the number of blocks available in the filesystem using the statfs call, but failed.

User Action: The errno status was indicated in the error message.
Review the error indicated by errno to resolve the problem.

CIMS for NT/UNIX CONVERT Utility Messages

The CIMS for NT/UNIX CONVERT Utility generates the following messages:

CNVTDONE, A_convert completed

Explanation: A_convert has completed processing.

User Action: None.

CREATERR, error creating file—<string>

Explanation: An error was encountered during the creation of the specified file.

User Action: This message is accompanied by a system call error.
Resolve the error by examining the cause of the system call error.

FILCON, file converted—<string>version found—<vers>, version expected—<vers>

Explanation: The file indicated by the string has been converted to the latest version.

User Action: None.

FILUTD, "<filename> is up to date.",

Explanation: The file indicated is up to date.

User Action: None

INVFILVER, "invalid file version detected—<filename>;version <version number>",

Explanation: The file indicated is an invalid version.

User Action: The indicated file may be corrupted. New versions of most CIMS for NT/UNIX files can be created by renaming the old file and running the CIMS Utility that normally maintains the file. For instance, if this message was produced while trying to convert the CIMS for NT/UNIX AUTHORIZATION File, rename A_uaf.sys to A_uaf.org. Now run the CIMS for NT/UNIX AUTHORIZATION Utility and you will be prompted to create a new A_uaf.sys file.

INVRATJT; invalid rate record jobtype detected

Explanation: The rate record jobtype detected is invalid.

User Action: The CIMS for NT/UNIX Rates file is corrupt. You will need to create a new file. Rename the Rates File to A_rates.corrupt. Run the CIMS RATES Utility, A_rates. You will be prompted to create a new Rates file.

MEMERR, error allocating memory

Explanation: An error was encountered while allocating memory.

User Action: Determine the necessary resource quota and expand the quota.

MLTDEFUSER, creating new user record; <filename>; for multiple DEFAULT user record detected

Explanation: A new user record was created as the stated username because multiple DEFAULT user records were detected.

User Action: None.

MLTDEFPROJ, creating new project record; <filename>; for multiple DEFAULT project record detected

Explanation: A new project record was created as the stated filename because multiple DEFAULT project records were detected.

User Action: None.

NOPWDENT, error getting passwd entry

Explanation: CIMS for UNIX could not read the user password entry

User Action: This message is accompanied by a system call error. Resolve the error by examining the cause of the system call error.

PRODVER, converting CIMS for NT/UNIX files to <version number>

Explanation: CIMS for NT/UNIX is displaying the current product version.

User Action: None.

RDVERR, error reading file version—<filename>

Explanation: An error occurred in reading the version of the file.

User Action: The indicated file may be corrupted. New versions of most CIMS for NT/UNIX files can be created by renaming the old file and running the CIMS Utility that normally maintains the file. For instance, if this message was produced while trying to convert the CIMS for NT/UNIX AUTHORIZATION File, rename A_uaf.sys to A_uaf.org. Now run the CIMS for NT/UNIX AUTHORIZATION Utility and you will be prompted to create a new A_uaf.sys file.

RENAMERR, error renaming <filename> to <filename>

Explanation: An error occurred during the execution of the COMPRESS command.

User Action: Correct the system error displayed and attempt the command again.

CIMS for UNIX Oracle Accounting Daemon Messages

The CIMS for UNIX Oracle Accounting daemon generates the following messages:

CLOSERR, error closing file

Explanation: An error has occurred during a close operation on a file.

User Action: This message is accompanied by a system call error. Resolve the error by examining the cause of the system call error.

INTRNLERR, internal table or mechanical failure

Explanation: An error occurred getting the column of an ORACLE Instance table.

User Action: This error is preceded by another message in the CIMS ORACLE Accounting Daemon Log File. Check the earlier message and correct that condition.

INVARG, invalid command line argument specified

Explanation: The command line contains an invalid argument.

User Action: An invalid argument was used when starting the CIMS ORACLE Accounting Daemon. Use the startup scripts provided in the ARSAP_SCRIPTS/oracle directory.

INVSID, unable to obtain a valid SID

Explanation: The CIMS ORACLE Accounting Daemon did not have a valid ORACLE_SID value on startup.

User Action: Use the startup scripts provided in the ARSAP_SCRIPTS/oracle directory.

OPENERR, error returned from open

Explanation: An error has occurred during an open operation on a file.

User Action: This message is accompanied by a system call error. Resolve the error by examining the cause of the system call error.

STATNAMERR, error in formatting A_dbstatname.sys

Explanation: The format of the ARSAP_DATA/A_dbstatname.sys file is incorrect.

User Action: A copy of the A_dbstatname.sys file is located in ARSAP_SCRIPTS/oracle directory. Copy this file to the ARSAP_DATA directory and re-start the CIMS ORACLE Accounting Daemon. If you are running ORACLE 8.1.5 or higher, use ARSAP_SCRIPTS/oracle/A_dbstatname.sys.815 file and copy to ARSAP_DATA directory with the name A_dbstatname.sys.

WRITEHEAD, error updating CIMS for NT/UNIX DB Accounting file header

Explanation: An error has occurred in updating the CIMS for NT/UNIX DB Accounting file header.

User Action: Verify that the owner of the CIMS ORACLE Accounting Daemon process has access to the CIMS Database Accounting File.

CIMS for UNIX Oracle Project LOGIN Utility Messages

The CIMS for UNIX Oracle Project LOGIN Utility generates the following messages:

CLOSERR, error closing file

Explanation: An error has occurred during a close operation.

User Action: This message is accompanied by a system call error. Resolve the error by examining the cause of the system call error.

NOPROJ, error getting CIMS for NT/UNIX current project

Explanation: An error has occurred reading the project record from the Authorization File.

User Action: Verify that the Authorization File is accessible and no file corruption has occurred.

OPENERR, error returned from open

Explanation: An error has occurred during an open operation on a file.

User Action: This message is accompanied by a system call error. Resolve the error by examining the cause of the system call error.

UPDERR, error updating project table

Explanation: An error has occurred in updating the project table.

User Action: Verify that the ARSAP_PROJECT table exists for the ORACLE Instance and is accessible.

USAGE, usage: A_dbplo oracle_utility [-proj] [arg1 arg2...]

Explanation: An example of correct command line syntax.

User Action: Re-enter the command with valid arguments.

CIMS for UNIX Oracle Project SWITCH Utility Messages

The CIMS for UNIX Oracle Project SWITCH Utility generates the following messages:

CLOSERR, error closing file

Explanation: An error has occurred during a close operation.

User Action: This message is accompanied by a system call error. Resolve the error by examining the cause of the system call error.

NOPROJ, error getting project

Explanation: An error has occurred reading the project record from the Authorization File.

User Action: Verify that the Authorization File is accessible and no file corruption has occurred.

OPENERR, error returned from open

Explanation: An error has occurred during an open operation on a file.

User Action: This message is accompanied by a system call error. Resolve the error by examining the cause of the system call error.

UPDERR, error updating project table

Explanation: An error has occurred in updating the project table.

User Action: Verify that the ARSAP_PROJECT table exists for the ORACLE Instance and is accessible.

USAGE, usage: A_dbpso project

Explanation: An example of correct command line syntax.

User Action: Re-enter the command with valid arguments.

CIMS for UNIX IBM DB2 Accounting Daemon Messages

The CIMS for UNIX IBM DB2 Accounting daemon generates the following messages:

ABNORMAL, DB2 API <API name>;abnormal completion code

Explanation: Unsuccessful completion due to DB2 API call failure.

User Action: Correct the problem reported by the DB2 API and restart the CIMS DB2 Accounting Daemon.

CLOSEACCT, error closing CIMS for UNIX IBM DB2 Accounting file; <filename>

Explanation: An error has occurred in closing DB2 accounting file <filename>.

User Action: The error occurred while shutting down the CIMS DB2 Accounting Daemon. Shutdown will continue.

DATAERR, error reading snapshot data

Explanation: Unexpected data/format was captured.

User Action: Report this error to CIMS Lab Technical Support.

INVARG, invalid command line argument <DB2 name>

Explanation: When starting the CIMS DB2 Accounting Daemon, an invalid argument was detected on the command line.

User Action:

OPENLOG, error opening CIMS for UNIX IBM DB2 log file

Explanation: An error has occurred in an open action on the DB2 log file.

User Action: Use the startup script provided in the CIMS ARSAP_SCRIPTS/db2 directory.

OPENACCT, error opening CIMS for UNIX IBM DB2 Accounting file;<filename>

Explanation: An error has occurred in an open action on the DB2 accounting file.

User Action: Check the ownership and group permissions on the CIMS DB2 Accounting Log File as well as the ARSAP_LOG directory.

RENAMERR, error renaming old CIMS for UNIX IBM DB2 Accounting file

Explanation: An error has occurred during execution of the COMPRESS command.

User Action: Correct the system error displayed and re-start the CIMS DB2 Accounting Daemon.

SNAPSHOT, error obtaining snapshot from DB2 database;<DB2 name>

Explanation: An error has occurred in obtaining snapshot data from a DB2 database.

User Action: Verify that DB2 Monitor parameters are set and that the Monitor table is accessible.

WRITEHEAD, error updating CIMS for UNIX IBM DB2 Accounting file

Explanation: An error has occurred in writing to the DB2 Accounting file.

User Action: Verify that the owner of the CIMS DB2 Accounting Daemon process has access to the CIMS Database Accounting File.

CIMS for UNIX GUI Messages

The CIMS for UNIX GUI generates both diagnostic and error messages.

Diagnosics

If CIMS for UNIX GUI encounters a problem when trying to perform an action, it displays diagnostic messages in a Dialog Message Box. These messages are:

Error: <type> name is required for <action>.

Explanation: The <action> indicated requires a <type> name.

User Action: Specify a <type> name and retry. Where <type> is either User or Project and <action> is either Modify or Remove.

Error: A From: <type> account is required for <action>.

Explanation: The <action> indicated requires a <type> account to be specified in the From: text box. Where <type> is either User or Project and <action> is either Rename or Copy.

User Action: Specify a <type> account in the From: text box and retry.

Error: A To: <type> account is required for <action>.

Explanation: The <action> indicated requires that a <type> account in the To: text box. Where <type> is either User or Project and <action> is either Copy or Rename.

User Action: Specify a <type> account in the To: text box and retry.

Error: Before Date is required.

Explanation: CIMS for UNIX Select requires a before date.

User Action: Specify a before date in the form DD-MMM-YYYY, for example, 1-Nov-1999.

Error: Checksum is required.

Explanation: A checksum is required for the license.

User Action: Enter the checksum from the CIMS for UNIX License PAK.

Error: Holiday Date is required for <action>.

Explanation: The <action> indicated requires a Holiday Date. Where <action> is one of Add, Modify, or Remove a holiday.

User Action: Specify a Holiday Date in the DD-MMM-YYYY format and retry.

Error: Insufficient permission. Unable to read <file>.

Explanation: A_gui could not read <file>.

User Action: Verify that the filename was typed correctly and that you have read permission to the file.

Error: Insufficient permission. Unable to exec <utility>.

Explanation: A_gui cannot execute <utility>.

User Action: Verify that the ARSAP_BIN environment variable is defined to the arsap/bin directory and that you have execute privilege on <utility>.

Error: Node Name is required.

Explanation: A node name is required to enter the license.

User Action: Enter a node name.

Error: Package <package> not found in Image Map File

Explanation: The requested software package does not exist in the CIMS for UNIX Image Map file.

User Action: None.

Error: Project is required.

Explanation: The project <project> to be copied does not exist in the CIMS for UNIX Authorization File.

User Action: Enter a project account.

Error: Queue <queue> not found in Queue Map File

Explanation: The requested queue does not exist in the CIMS for UNIX Queue Map file.

User Action: None.

Error: Rate Table name is required for <action>.

Explanation: The <action> indicated requires a rate table name where <action> is one of Copy, Rename, or Report.

User Action: Specify a rate table name and retry.

Error: Rate Table record not found

Explanation: The requested rate table record does not exist in the CIMS for UNIX Rates file.

User Action: Enter a rate table that currently exists or create a new rate table.

Error: Shift <shift> not found in Shift File

Explanation: The requested shift does not exist in the CIMS for UNIX Shift file.

User Action: None.

Error: Since Date is required.

Explanation: CIMS for UNIX Select requires a since date.

User Action: Specify a since date in the form DD-MMM-YYYY, for example 1-Nov-1999.

Error: Terminal <terminal> not found in Terminal Parameter File

Explanation: The requested terminal does not exist in the CIMS for UNIX Terminal Parameter file.

User Action: None.

Error: The date <date> is invalid. Please use DD-MMM-YYYY Format.

Explanation: <date> is not in a format recognizable by A_gui.

User Action: Enter the date in DD-MMM-YYYY format, for example, 1-Nov-1999.

Error: The Holiday <name> was not found. rdeq=<n>

Explanation: <name> was not found in the CIMS for UNIX Holiday file. The rdeq indicates the status of the read command.

User Action: None.

Error: To select <type>, you must specify a queue

Explanation: <type> rates require queue names.

User Action: Specify a <type> queue. Where <type> is either Print or Batch.

Error: To select <type>, you must specify a shift.

Explanation: <type> rates require shifts. Where <type> is one of the following jobtypes; Interactive, Background, Batch, Package, DB2, or ORACLE.

User Action: Specify a shift name.

Error: To select Package, you must specify a Package Name.

Explanation: Software Package rates require package names.

User Action: Specify a software package name. Which is defined in the CIMS for UNIX Image Mapping File.

Error: Unable to exec() after fork().

Explanation: A_gui could not attach the child process created by fork().

User Action: Correct the condition indicated by the error written to standard output.

Error: Unable to find User <user>.

Explanation: A_gui could not find user <user> in the CIMS for UNIX Authorization File.

User Action: None.

Error: Unable to fork() process.

Explanation: A_gui could not create a child process.

User Action: Correct the condition indicated by the error written to standard output.

Error: Unable to open <filename>. Open failed with <n>.

Explanation: A_gui tried to open the <filename> file and failed with status <n>.

User Action: Correct the condition indicated by the error written to standard output.

Error: Unable to read <filename>. Read failed with <n>.

Explanation: A_gui tried to read the <filename> file and failed with status <n>.

User Action: Correct the condition indicated by the error written to standard output. Check the ownership and permissions on the file.

Error: Unable to set ARSAP_BIN. status=<n>.

Explanation: A_gui tried to set the ARSAP_BIN environment variable and failed with status <n>.

User Action: Correct the condition indicated by the error written to standard output.

Error: Unable to set ARSAP_DATA. status=<n>

Explanation: A_gui tried to set the ARSAP_DATA environment variable and failed with status <n>.

User Action: Correct the condition indicated by the error written to standard output.

Error: User name is required.

Explanation: A user name is required for the add.

User Action: Enter a valid username.

Error: You must specify a rate table to <action>.

Explanation: The <action> indicated requires a rate table name. Where <action> is one of Add, Remove, Modify, or Zero.

User Action: Specify a rate table name and retry.

Info: A_sampler is now running in background.

Explanation: A disk storage inventory was requested. The CIMS for UNIX A_SAMPLER Utility is now running.

Success: Printing completed.

Explanation: A_gui completed printing the CIMS for NT/UNIX System Log.

User Action: None.

Success: The sort was successful

Explanation: A_gui successfully sorted the formatted UNIX accounting files.

User Action: None.

User Action: None.

Warning: Nothing to Print!

Explanation: The CIMS for UNIX System Log is empty. There is nothing for A_gui to print.

User Action: None.

Warning: Size MisMatch! Length of Prompt Default Value <prtn> is <n> characters, should be <m>.

Explanation: The prompt default value for prompt <prtn> is <n> characters long. The start and end positions indicate the value should be <m> characters.

User Action: Verify the prompt default value and start and end positions for prompt <prtn>.

Warning: Unable to find Project <project> Node <node>

Explanation: A_gui could not find a project/node record for project <project> and node <node> in the CIMS for UNIX Authorization File.

User Action: None.

Warning: Unable to find User <user> Node <node>

Explanation: A_gui could not find a user/node record for user <user> and node <node> in the CIMS for UNIX Authorization File.

User Action: None.

Error Messages

For certain errors CIMS for UNIX GUI is not able to display information in the Message Box. When one of these errors occurs, GUI displays error messages to standard output. These messages are:

BADCONF, /etc/arsap.conf contains invalid directory

Explanation: While trying to set the CIMS for UNIX environment variables, A_gui detected an invalid directory path in the /etc/arsap.conf File.

User Action: Determine where CIMS for UNIX was installed on the system, and correct the directory path in the /etc/arsap.conf file.

NORMAL, success error code

Explanation: Successful completion of operation.

User Action: None.

OPENCONF, open error on /etc/arsap.conf

Explanation: While trying to set the CIMS for UNIX environment variables, A_gui was unable to open the /etc/arsap.conf file for reading.

User Action: Determine where CIMS for UNIX was installed on the system, and create the /etc/arsap.conf file.

PROGIDENT, A_gui program version Vxx.yy

Explanation: A_gui displays the version number of the CIMS for UNIX GUI Utility that you are running.

User Action: Report this version number to the CIMS Lab should you encounter problems associated with A_gui.

SCANCONF, scanf error on /etc/arsap.conf

Explanation: While trying to set the CIMS for UNIX environment variables, A_gui was unable to read the /etc/arsap.conf file.

User Action: Determine why the `/etc/arsap.conf` file could not be read, and correct.

CIMS for UNIX GRAPH Utility Messages

The CIMS for UNIX GRAPH Utility generates both diagnostic and error messages.

Diagnostics

If CIMS for UNIX GRAPH Utility encounters a problem when trying to perform an action, it displays messages in the Message Box. These messages are:

Error: A filename is required at this time.

Explanation: GRAPH requires a filename for the action you requested.

User Action: Specify a filename and retry.

Error: A_report failed.

Explanation: GRAPH is unable to generate a chart because the CIMS for UNIX REPORT Utility failed.

User Action: Correct the error written to standard output by the CIMS for UNIX REPORT Utility.

Error: Depth must be a decimal number in the range of 0 and 500.

Explanation: The value you entered for depth is outside the acceptable range of values.

User Action: Enter a number between 0 and 500, inclusive.

Error: Inclination must be a decimal number in the range of 0 and 45.

Explanation: The value you entered for inclination is outside the acceptable range of values.

User Action: Enter a number between 0 and 45, inclusive.

Error: Insufficient permission. Unable to exec <file>

Explanation: You do not have execute permission to <file>.

User Action: Verify that the environment variable ARSAP_BIN is defined to the CIMS for UNIX Bin directory was installed. Check the ownership and permissions on <file>.

Error: No data found in report.

Explanation: GRAPH could not produce a chart because there is no data in the the CIMS for UNIX REPORT Utility report.

User Action: None.

Error: No data kept from report

Explanation: GRAPH could not produce a chart because all statistics in the report generated by the CIMS for UNIX REPORT Utility are zero, and the Keep Zero toggle button is off.

User Action: Turn on the **Keep Zero** toggle button, or generate a different report.

Error: Pie % Minimum must be a decimal number in the range of 0 and 100.

Explanation: The value you entered for Minimum Pie Percentage is outside the acceptable range of values.

User Action: Enter a number between 0 and 100, inclusive.

Error: Print failed during call to xpr.

Explanation: GRAPH was unable to print the chart because of an error in xpr.

User Action: Correct the error written to standard error and retry.

Error: Rotation must be a decimal number in the range of 0 and 45.

Explanation: The value you entered for rotation is outside the acceptable range of values.

User Action: Enter a number between 0 and 45, inclusive.

Error: Unable to find executable "xpr" in your environment \$PATH.

Explanation: GRAPH tried to execute xpr to print the chart, but could not find the executable.

User Action: Verify that the PATH environment variable contains the path to "xpr".

Error: Unable to load selected file.

Explanation: In response to the open command, GRAPH could not load the specified graph description file.

User Action: Correct the error written to standard output.

Error: Unable to open <file>.

Explanation: GRAPH was not able to open the file <file>.

User Action: Correct the error written to standard output.

Error: Unable to open rdf.

Explanation: GRAPH was not able to open the CIMS for UNIX Report Description File in the /tmp directory.

User Action: Correct the error written to standard output.

Error: Unable to save to <file>.

Explanation: GRAPH was not able to save the CIMS for UNIX Graph Description File in <file>.

User Action: Correct the error written to standard output.

Error: Unable to set ARSAP_BIN status=<n>.

Explanation: GRAPH was not able to set the ARSAP_BIN environment variable. It failed with status <n>.

User Action: Correct the error written to standard output.

Error: Unable to set ARSAP_DATA status=<n>.

Explanation: GRAPH was not able to set the ARSAP_DATA environment variable. It failed with status <n>.

User Action: Correct the error written to standard output.

Error: While generating xwd file, fopen() failed.

Explanation: GRAPH was attempting to create an xwd file in response to a print command when open failed.

User Action: Correct the error written to standard output.

Error: You must select an X axis variable.

Explanation: You requested that GRAPH generate a chart, but did not select an item for the X axis variable. GRAPH requires exactly one X axis variable.

User Action: Select an item from the X axis list to be the X axis variable and retry.

Error: You must select one or more Y axis variables.

Explanation: You requested that GRAPH generate a chart, but did not select an item for the Y axis variable. GRAPH requires at least one Y axis variable.

User Action: Select item(s) from the Y axis list to be the Y axis variable(s) and retry.

Success: <file> Saved.

Explanation: GRAPH has saved the graph description file in <file>.

User Action: None.

Success: Print complete.

Explanation: GRAPH has printed the chart.

User Action: None.

Success: XWD Printing completed.

Explanation: GRAPH has printed the chart in xwd format.

User Action: None.

Warning: This print format is not yet supported.

Explanation: You attempted to print in a format that has not been implemented.

User Action: Print in another format.

Error Messages

For certain errors, CIMS for UNIX GRAPH Utility is not able to display information in the Message Box. When one of these errors occurs, GRAPH displays error messages to standard output. These messages are:

BADCONF, /etc/arsap.conf contains invalid directory

Explanation: While trying to set the CIMS for UNIX environment variables, GRAPH detected an invalid directory path in the /etc/arsap.conf file.

User Action: Determine where CIMS for UNIX was installed on the system and correct the directory path in the /etc/arsap.conf file.

FSETENV, error setting environment variable <var>.

Explanation: While trying to set the CIMS for UNIX environment variable <var>, GRAPH encountered an error.

User Action: Correct the error written to standard error.

NORMAL, success error code.

Explanation: Successful completion of operation.

User Action: None.

PROGIDENT, A_graph program version Vxx.yy.

Explanation: GRAPH displays the version number of the CIMS for UNIX GRAPH Utility that you are running.

User Action: Report this version number to the CIMS Lab should you encounter problems associated with GRAPH.

CIMS for NT LOGIN Utility Messages

The CIMS for NT LOGIN Utility generates only diagnostic messages. These messages are displayed in a Windows Message Box. These messages are:

AdjustTokenPrivileges enable failed

Explanation: Unable to acquire the necessary privileges for system shutdown.

Could not create ARSAP_CURRPROJ

Explanation: Unable to setup the ARSAP_CURRPROJ value in the registry

Could not open ARSAP_CURRPROJ registry key

Explanation: Unable to open the ARSAP_CURRPROJ value in the registry.

Could not set ARSAP_CURRPROJ

Explanation: Could not set the ARSAP_CURRPROJ environment variable for the local environment.

ERROR: Could not set ARSAP_DATA environment variable

Explanation: Could not set the ARSAP_DATA environment variable for the local environment.

ERROR: Creating Program Manager process

Explanation: Unable to start the Program Manager desktop—this event immediately follows ARSAP project authentication.

ERROR: getting ARSAP Setup record

Explanation: Could not fetch the ARSAP Setup record for this node from A_setup.sys in the currently defined ARSAP_DATA directory

User Action: Verify that the CIMS Parameter File, A_setup.sys is accessible using the NT_SETUP Utility in the ARSAP_BIN directory.

ERROR: getting ARSAP User record

Explanation: Could not fetch the user's record from A_uaf.sys in the currently defined ARSAP_DATA directory

User Action: Verify that the CIMS Authorization File, A_uaf.sys is accessible and that a record for the indicated user exists.

ERROR: No license information found for this machine

Explanation: A valid CIMS for NT/UNIX product license PAK for this machine could not be found.

User Action: Check the values in the CIMS license PAK installed on this platform using the NT_LICENSE Utility in the ARSAP_BIN directory. Verify that the license was entered for the correct machine name and has not expired.

ERROR: nt_login process creation

Explanation: This message comes from nt_login_scr, when it is trying to create the nt_login process during a batch login.

ERROR: opening ARSAP Authorization file

Explanation: Could not open A_uaf.sys in the currently defined ARSAP_DATA directory

User Action: Verify that the CIMS Authorization File, A_uaf.sys is accessible by running the NT_AUTHORIZE Utility in the ARSAP_BIN directory.

Error Initiating System Shutdown

Explanation: User is unable to shutdown the Windows NT operating system.

Exit Windows Error

Explanation: User elected to exit the Windows NT desktop, but is unable to do so.

Invalid Project or Password

Explanation: User's CIMS for NT/UNIX Project/Password combination does not authenticate.

User Action: Check the method of validation that is applicable system wide, to the user, and to the project that was entered.

OpenProcessToken failed

Explanation: Could not connect to the current user's process to obtain info, such as privileges, environment, and so forth.

Progman Process Create Error

Explanation: Unable to create a process for the Progman desktop.

Project has expired

Explanation: User's CIMS for NT/UNIX Project has expired.

User Action: Check the project settings with the NT_authorize Utility.

■ **CIMS for NT/UNIX Messages**

CIMS for NT LOGIN Utility Messages



The CIMS for NT/UNIX Statistics

UNIX	A-2
Oracle	A-6
DB2	A-9

UNIX

The following is a list of UNIX statistics processed by CIMS for UNIX:

block_weeks

A space/time measurement to indicate the amount of utilization of the disk storage system. Block week represents the use of a disk block for a one week (7 day) period.

The basic algorithm for calculating block weeks is this: given a starting time (date & time), and ending time, and the number of blocks used (blocks_used) by the user or project, the number of block weeks is computed as follows:

```
time_difference = ending_time - starting_time (days)
weeks = time_difference/7.0
storage_block_weeks = blocks_used * weeks
```

The time_difference is dependent on the /SINCE and /BEFORE qualifiers used with A_select. Therefore, the results are based on your selected period and the greater the selected period the better the reported block weeks.

The blocks_used is determined by the A_sampler Utility. Disk block units vary between 1k bytes or 512 bytes, depending on the operating system and the command. Typically, under SYSV disk usage (du) is reported in 512-byte blocks and under SunOS du reports in 1k bytes. The A_sampler Utility reports in 512-byte blocks that provides a consistent metric regardless of the operating system.

For example,

```
if A_select is run for a one-day period:
    storage_block_weeks = blocks_used * 1.0/7.0
if A_select is run for a two-day period:
    storage_block_weeks = blocks_used * 2.0/7.0
```


Suppose a user has 70 blocks on the first day of the month and that disk usage never changes during the month. At the end of the first week, the user has 70 block weeks or 10 blocks on a daily basis. At the end of the second week, the user has 140 block weeks. At the end of the four-week month, the user has 280 block weeks.

cio

Character I/O (*cio*) are the number of characters transferred (terminal I/O) by read/write. Character I/O might not be available for some BSD-based systems. The *cio* statistic is maintained per process. CIMS for UNIX reports *cio* in units of 100,000. The number generated by CIMS for UNIX can be reported for a user, project, or package, depending on the report description file.

cio is not reported for DomainOS.

connect_time

Connect time indicates the amount of time the user is connected to a console login session. The *connect_time* statistic consists of *cpu*, *scpu*, *ucpu*, and *tcpu* and is in units of hours. Connect time is the actual *seat* time or elapsed wall-clock time for a user. This means connect time does not amass when using window applications. Therefore, not giving inflated login time. Window time (*win_time*) specifies the window resource use.

dio

Disk I/O (*dio*) are the number of block reads/writes. The *dio* statistic is maintained per process. CIMS for UNIX reports *dio* in units of 1000. The number generated by CIMS for UNIX can be reported for a user, project, or package, depending on the report description file.

dio is not reported for AIX 3.2.5 because the operating system does not supply this information. However, for some reason, AIX 3.2.5 does seem to give *dio* for the following commands: *dosread* and *doswrite*.

image_time

Image time is the amount of time the image executed. CIMS for UNIX reports `image_time` in units of hours based on the package.

image_count

Image count is the number of times a particular image was run. CIMS for UNIX reports `image_count` based on the package.

logins

Login is the number of sessions that a person has connected to a system. A login is defined as:

- Interactive logins
- Project switches
- CXbatch jobs (convex only)
- NQS batch jobs

memory

Memory indicates the approximate amount of virtual memory. Memory utilization is based on the resident portion of the memory segments. The value and algorithm of this statistic varies from one platform to another. In general, virtual memory utilization is an approximation of the mean process size. Therefore, the amount of memory is determined for a process then adjusted according to the CPU time used by the process.

For an example, under HP-UX the process memory is determined by the following algorithm:

```
ac_mem = (data size) + (text size) +  
         (number of in-core processes sharing text) +  
         sum of ((shared memory segment size) /  
         (number of in-core processes attached to segment))
```

The virtual memory is determined by:

`ac_mem/(ac_stime+ac_utime)`

The units for memory utilization vary from platform to platform (for example, HP-UX:

1k block, SunOS: 1k bytes, IRIX: 4k page).

scpu

The percentage of time the CPU spends in the System state. This includes time spent executing system calls, administrative functions on the program's behalf, device driver code and any kernel code executed by the system. System CPU time can be controlled to some degree by the programmer, particularly in regard to I/O handling that uses extensive amounts of system services. The `ucpu` statistic is maintained per process. CIMS for UNIX reports `ucpu` in units of minutes.

su (switch user) Statistics

A user can become super-user or another user by using the `su` command. When a switch occurs, CIMS for UNIX tracks the statistics based on the user who performed the `su` command.

su_count

`su_count` is the number of times a user performed the `su` command.

su_image_count

`su_image_count` is the number of images (processes or daemons) a user invoked as a different user.

su_time

`su_time` is the time a user was switched as another user(s). The `su_time` statistic is in units of hours.

ucpu

System CPU is the amount of time the CPU spends running a program in User state. This includes time spent executing library functions, but excludes time spent executing system calls. User CPU time is under the programmer's control with the exception of library functions. The `scpu` statistic is maintained per process. CIMS for UNIX reports `scpu` in units of minutes.

win_time

Window time (`win_time`) indicates the amount of time the user is using a window application; `win_time` is the elapsed time for all window sessions during an interactive login session. The `win_time` statistic is in units of hours. Window time is specified for each window application.

Oracle

The following is the list of Oracle statistics gathered by CIMS for UNIX:

Oracle logins

Number of times a user connects to Oracle via an Oracle application (for example, SQL*Plus, `sqldba`, `sqlldr`)

Oracle session CPU

The amount of CPU utilized for this login session

Oracle connect time

The amount of time (hours) a user is connected to Oracle

Oracle UGA memory

The amount of memory used for the User Global Area (UGA), UGA is used in multi-threaded server mode, the amount of memory that is specific to each user session. For example, a user who simultaneously logs into Oracle three times has three active UGAs.

Oracle PGA memory

The amount of memory used for the Program Global Area or Process Global Area (PGA), PGA is a non-shareable and process specific memory region; one PGA is allocated for each process, when not in multi-threaded server mode, PGA contains process specific info such as stack, program variables, and so on, when running in multi-threaded server mode, only the stack info is stored in PGA and the other process information is stored in System Global Area (SGA).

The SGA is a shareable memory area for the Database instance. All processes connected to Oracle (User and Oracle background) have access to the SGA. This is probably the most sensitive area of an instance. Info stored within the SGA is:

- database buffer cache

- redo log buffer

- shred pool

- request and response queues (multi-threaded server)

- data dictionary cache

- other instance related info

Oracle recursive CPU

The amount of CPU used updating internal tables during Oracle recursive processing, Oracle performs recursive calls for the following operations:

- dynamic extension of segments
- misses on the data dictionary cache
- firing of database triggers
- execution of DDL (Data Definition Language) statements
- execution of SQL statements within stored procedures, functions, packages, and anonymous PL/SQL blocks
- enforcement of referential integrity constraints

Oracle user commits

The number of commits performed by the user.

Oracle physical reads

The number of reads from database files resulting in access to data files on disk

Oracle physical writes

The number of writes to database files resulting in access to data files on disk, in a multi-user environment, this statistic is always 0, except for the Oracle daemon DBWR

Oracle write requests

The number of multi-block writes performed

Oracle disk sorts

The amount of memory utilized to perform an external (disk) sort due to a temporary segment allocated to perform an in-memory sort, this does not include memory sorts

Oracle messages sent

The number of messages sent to perform database updates in a client-server environment, reflects traffic between server and client processes

Oracle messages received

The number of messages received to update database the number of messages received to perform database updates in a client-server environment; reflects traffic between server and client processes

DB2

The following is the list of IBM DB2 statistics gathered by CIMS for UNIX:

CPU Usage

user CPU time—the total user CPU time used by the database manager process

system CPU time—the total system CPU time used by the database manager process

Buffer Pool Activity

buffer pool data logical reads—the number of logical read requests for data pages that have gone through the buffer pool

buffer pool data physical reads—the number of read requests that required I/O to get data pages into the buffer pool

buffer pool data writes—the number of times a buffer pool data page was physically written to disk

buffer pool index logical reads—the number of logical read requests for index pages that have gone through the buffer pool

buffer pool index physical reads—the number of physical read requests to get index pages into the buffer pool

buffer pool index writes—the number of times a buffer pool index page was physically written to disk

I/O Activity

direct reads—the number of read operations that do not use the buffer pool*

direct writes—the number of write operations that do not use the buffer pool*

* not available for DB2 PE

Table Activity

rows deleted—the number of row deletion operations attempted

rows inserted—the number of row insertion operations attempted

rows updated—the number of row update operations attempted

rows selected—the number of rows selected and returned to the application

Unit of Work Status

unit of work log space used—the amount of log space (in bytes) used in the current unit of work of the monitored application

SQL Statement Activity

commit statements attempted—the total number of SQL commit statements that have been attempted

rollback statements attempted—the total number of SQL rollback statements that have been attempted

internal rollbacks due to deadlock—the total number of rollbacks initiated by the database manager due to a deadlock; the rollback is performed on the current unit of work in an application selected by the database manager to resolve the deadlock

Locks and Deadlocks

time waited on locks—the total elapsed time waited for a lock

deadlocks detected—the total number of deadlocks that have occurred

Sort Work

total sorts—the total number of sorts that have been executed

sort overflows—the total number of sorts that ran out of sort heap and might have required disk space for temporary storage



CIMS for UNIX GUI Files

Resource File	B-2
System Wide	B-2
Environment Variable	B-2
Local Copy	B-3
Colors	B-3
Fonts	B-6
Datatypes	B-7
Line Pattern	B-7
Fill Pattern	B-8
Color	B-8
Line Width	B-8
Point Style	B-8
Point Color	B-8
Point Size	B-9
X Resource Files	B-9

Resource File

A resource file is provided for both CIMS for UNIX Graph and GUI that lets you customize the appearance of these interfaces. These resource files are found in the `ARSAP_EXAMPLES` directory as `A_graph.rf` and `A_gui.rf`.

There are three ways to use these resource files in order of precedence:

System Wide

To have a resource file used system wide, copy the example resource file to the X11 app-defaults directory. This is usually located in `/usr/lib/X11/app-defaults`, but is installation dependent.

```
cp $ARSAP_HOME/examples/A_graph.rf /usr/lib/X11/app-defaults/A_graph
cp $ARSAP_HOME/examples/A_gui.rf /usr/lib/X11/app-defaults/A_gui
```

Environment Variable

Instead you can copy the example resource file to `A_graph` and define `XAPPLRESDIR` to the directory path. Resource files found in the directory pointed to by `XAPPLRESDIR` override resource files in the X11 `apps_default` directory.

```
cp $ARSAP_HOME/examples/A_graph.rf $ARSAP_HOME
  /examples/A_graph
setenv XAPPLRESDIR $ARSAP_HOME/examples
cp $ARSAP_HOME/examples/A_gui.rf $ARSAP_HOME/examples
  /A_gui
setenv XAPPLRESDIR $ARSAP_HOME/examples
```

Local Copy

When you want to use different resource values than those defined system wide, you can copy the CIMS for UNIX resource files to your home directory. Resource files found in your home directory override any resource files in the directory defined in `XAPPLRESDIR` and the X11 `apps-default` directory.

```
cp $ARSAP_HOME/examples/A_graph.rf ~/A_graph
cp $ARSAP_HOME/examples/A_gui.rf ~/A_gui
```

Colors

Colors are specified as either a color name or an RGB Device specification. To find the color names defined on your system, use the X11 `showrgb` Utility.

RGB Device specifications have two formats: `rgb:rrggbb` and `#rrggbb`. Where `rr` indicates the intensity of red in the color, `gg` indicates the intensity of green in the color, and `bb` indicates the intensity of blue in the color. The second format, with the pound sign(`#`), is an older specification provided for backwards compatibility. It might be discontinued at any time.

The eight primary colors are represented as:

black	rgb:0/0/0	#000000
red	rgb:ff/0/0	#ff0000
green	rgb:0/ff/0	#00ff00
blue	rgb:0/0/ff	#0000ff
yellow	rgb:ff/ff/0	#ffff00
magenta	rgb:ff/0/ff	#ff00ff
cyan	rgb:0/ff/ff	#00ffff
white	rgb:ff/ff/ff	#ffffff

CIMS for UNIX Graph and GUI use color resources for the following standard Motif widgets:

Application Shell	XmLabel	XmSeparator
XmArrowButton	XmLabelGadget	XmText
XmArrowButtonGadget	XmList	XmTextField
XmCascadeButton	XmMessageBox	XmToggleButton
XmDialogShell	XmPushButton	XmToggleButtonGadget
XmFileSelectionBox	XmPushButtonGadget	XmTransientShell
XmForm	XmRowColumn	
XmFrame	XmScrolledWindow	

Graph also includes the following color resources:

*.menu1.background	Background color for menu at the top of screen. All three resources must be set to the same color.
*.menu1_pFile.background	
*.menu1_top_b1.background	
*.menu1_pFile.borderColor	Color for edge around menu.
*.xrtBackgroundColor	Background color for frame around chart.
*.xrtFooterBackgroundColor	Background color for chart footer.
*.xrtFooterForegroundColor	Text color for chart footer.
*.xrtGraphBackgroundColor	Background color of screen.
*.xrtGraphForegroundColor	Default color of text.
*.xrtHeaderBackgroundColor	Background color for chart header.
*.xrtHeaderForegroundColor	Text color for chart header.
*.xrtLegendBackgroundColor	Background color for legend.
*.xrtLegendForegroundColor	Text color for legend.
*.BarChart_XrtGraph.background	
*.BarChart_XrtGraph.borderColor	
*.BarChart_XrtGraph.foreground	
*.GejacLogoBmp.background	

Fonts

CIMS for UNIX GUI and Graph use the default system font. You can select a different font by setting one of the font resources. To determine the fonts that are available, contact your system administrator, or you can run `xlsfont`. On some systems, you can use the utility, `xfontsel`, to view the fonts available.

CIMS for UNIX GUI and Graph use font resources for the following standard Motif widgets:

Application Shell

XmDialogShell

XmFileSelectionBox

XmForm

XmMessageBox

XmTransientShell

Additionally, CIMS for UNIX GUI sets the font resource

*.Main_header.fontlist and sets the font for welcome message.

Additional GRAPH sets the following font resources:

*.BarChart_XrtGraph.xrtAxisFont Sets the font for the x and y axis.

*.BarChart_XrtGraph.xrtFooterFont Sets the font for the footer.

*.BarChart_XrtGraph.xrtHeaderFont Sets the font for the header.

*.BarChart_XrtGraph.xrtLegendFont Sets the font for the legend.

Datastyles

CIMS for UNIX Graphlets you set the physical appearance of the data on the chart. This is done with a resource called data styles. Graph uses two data styles, `xrtDataStyles` and `xrtOtherDataStyle`. `xrtDataStyles` resource is one or more data styles enclosed in parenthesis, each defined data style represents one data set. The `xrtOtherDataStyle` is a single data style.

A data style is a parenthesis-enclosed list of strings that specify a value for each physical characteristic of a data set. The fields of a data style are: line pattern, fill pattern, color, line width, point style, point color and point size. The line continuation character is the back slash ("`\`").

The data styles specified in `A_graph.rdf` are:

```
■ *.BarChart_XrtGraph.xrtDataStyles: \
((LpatSolid FpatSolid \"red\" 1 PointDot \"red\" 7 )\
(LpatSolid FpatHorizStripe \"yellow\" 1 PointBox \"yellow\" 7 )\
(LpatSolid Fpat25percent \"#00a3ab\" 1 PointTri \"#00a3ab\" 7 )\
(LpatSolid FpatVertStripe \"red\" 1 PointDiamond \"red\" 7 )\
(LpatSolid FpatCrossHatched \"blue\" 1 PointStar \"blue\" 7 ) \
(LpatSolid Fpat50Percent \"green\" 1 PointCross \"green\" 7 ))
■ *.BarChart_XrtGraph.xrtOtherDataStyle: \
(LpatSolid Fpat45Stripe \"cyan\" \1 PointCircle \"cyan\" 7 )
```

Line Pattern

A line pattern indicates the graphical representation of lines used in plot graphs. There are several predefined patterns: `LpatNone`, `LpatSolid`, `LpatLongDash`, `LpatDotted`, `LpatShortDash`, `LpatLslDash`, and `LpatDashDot`.

When you have a monochrome display, or are printing, you might want to experiment with line patterns.

Fill Pattern

A fill pattern is used in bar, stacked bar, and pie charts and area graphs. The valid fill patterns are: FpatNone, FpatSolid, Fpat25Percent, Fpat50Percent, Fpat75Percent, FpatHorizStripe, FpatVertStripe, Fpat45Stripe, Fpat135Stripe, FpatDiagHatched, and FpatCrossHatched.

When you have a monochrome display, or are printing the chart, you might want to experiment with fill patterns.

Color

The data style colors are the same as for Xapplication resources (refer to *Colors* on page B-3).

Line Width

Line width represents the width of a line in plot graphs. The number must be greater than or equal to 1.

Point Style

Point Style specifies the symbol used to represent a point on a plot graph. The valid point styles are: PointNone, PointDot, PointBox, PointTri, PointDiamond, PointStar, PointVertLine, PointHorizLine, PointCross, PointCircle, and PointSquare.

When you have a monochrome display or are printing, you might want to experiment with point styles.

Point Color

The data style of Point colors are the same as for Xapplication resources (see *Colors* on page B-3).

Point Size

Point Size represents the relative size of a point in a plot graph. The number must be greater than or equal to 0. A point size of zero results in no point being drawn.

X Resource Files

When you do not have Motif 1.2.5 installed or you are getting key translation errors with `A_gui` or `A_graph`, you should do the following:

- 1 Locate the X-window key symbol mapping database file (`XKeysymDB`). This file is usually found in either `/usr/lib/X11`, `/usr/openwin/lib/X11` or `/usr/vue/X11` directories depending on which X-window system you are running.
- 2 In `$ARSAP_HOME/examples`, append `XKeysymDB` to `A_KeysymDB`. When you are on a SunOS system, use `A_KeysymDB.sun`; when you are on a Solaris system, use `A_KeysymDB.solaris`.

For a SunOS machine,

```
cat A_KeysymDB.sun /usr/openwin/lib/X11/XKeysymDB >> A_XKeysymDB
```

For a Solaris machine,

```
cat A_KeysymDB.solaris /usr/openwin/lib/X11/XKeysymDB >> A_XKeysymDB
```

- 3 Set the `XKEYSYMDB` environment variable to point to the `A_XKeysymDB` file in `$ARSAP_HOME/examples`.

When you are using `csh`,

```
setenv XKEYSYMDB /usr/arsap/examples/A_XKeysymDB
```

When you are using `sh` or `ksh`,

```
XKEYSYMDB=/usr/arsap/examples/A_XKeysymDB; export XKEYSYMDB
```

You should put this in your `.login` or `.profile` files to have this set when you use `A_gui` or `A_graph`.



Index

A

- Advanced Features 7-2
- Application Log 2-8
- Archiving Data 7-6
- ARSAP
 - Utilities 4-3
- AUTHORIZE Utility 4-3

B

- Batch Accounting 2-11
- Body Section 5-19

C

- CIMS for NT/UNIX
 - Accounting File 5-5
 - Get Accounting script 7-2
 - Holiday File 5-7
 - Image Mapping File 5-7
 - Intermediate Statistics File 5-8
 - Merge Utility 4-5
 - Runacct Script 6-36
 - Select Utility 4-7
 - Validation File 5-11

CIMS for UNIX

- Activity File 5-5
- Check Nightly script 7-3
- Check UNIX Acct File script 7-3
- Collapsible Terminal File 5-6
- GID Method Implementation 7-8
- Login Projects File 5-8
- Login Utility 2-20
- Merged Statistics File 5-8
- Oracle Accounting File 5-8
- Oracle Basename File 5-8
- Oracle Login 2-39
- Oracle Project Switch 2-40
- Oracle Username File 5-9
- Print Accounting Script 6-35
- Print Parameter File 5-9
- Project Activity File 2-6
- Queue Mapping File 5-9
- Statistics A-2
- Storage File 2-7, 5-11
- Super Merge Utility 4-12
- Switch Utility 2-31
- System Accounting 2-5



Terminal Parameter File 5-11
X-window Login Utility 2-22
CIMS Lab, contacting x
Cleanup History Option 7-6
Colors B-3
contacting, CIMS Lab x
Convex Batch Accounting File 2-12

D

Database System Accounting 2-9
Datatypes B-7
DB2 A-9

F

Files

acct 2-4
CIMS for NT/UNIX Accounting 5-5
CIMS for UNIX Activity 5-5
CIMS for UNIX Oracle Accounting 5-8
CIMS for UNIX Oracle Basename 5-8
CIMS for UNIX Oracle Username 5-9
CIMS for UNIX Project Activity 2-6
pacct 2-4
UNIX Acct 2-4
UNIX Print Accounting 2-5
UNIX Wtmp 2-3
UNIX Wtmpx 2-3
wtmp 2-3
wtmpx 2-3

Fonts B-6
Format Utility 4-3
FS_Report Utility 4-3

G

Graph

Depth 5-33
Footer 5-33

Header 5-33
Inclination 5-34
Invert 5-34
Jobtype Formatting Keywords 5-32
KeepZero 5-34
Legend 5-35
Member 5-35
PieMin 5-35
RateTable 5-35
Rotation 5-36
Sort 5-36
Transpose 5-36
Type 5-36
X 5-37
Xgrid 5-37
XTitle 5-37
Y 5-38
YGrid 5-38
YTitle 5-38
Graph Description File 5-31
Graph Report Utility 4-4
GUI Utility 4-4

H

help, contacting the CIMS Lab x

L

LOGIN Utility 4-4

M

MERGE Utility 4-4

O

Oracle A-6
Out-of-Sequence Management 7-7
Overview 1-1

P

Performance Management 7-7
Project Accounting for Batch Processing
2-36
Project Accounting for NT 2-32
Project Accounting for UNIX 2-18

R

RATES Utility 4-6
Report Description File Considerations
5-27
Report Description File Examples 5-28
Report Description Files 5-12
REPORT Utility 4-6
Reports
 Body Section
 Body Keywords 5-19

S

SAMPLER Utility 4-6
Select Section 5-12
SELECT Utility 4-7
SETUP Utility 4-11
Slow Data Consolidation 7-8
Slow Project Login 7-8
Sort Section 5-18
SUPER_MERGE Utility 4-12
SWITCH Utility 4-16
System Accounting 2-2

T

technical support, contacting the CIMS
 Lab x

U

UNIX
 Acct File 2-4

Print Accounting File 2-5

Wtmp File 2-3

Wtmpx File 2-3

UNIX System Accounting 2-2

X

X-window Utility 4-16

