Honeywell

LEVEL 68

SOFTWARE

MULTICS ADMINISTRATORS' MANUAL— RESOURCE CONTROL

Roach

MULTICS ADMINISTRATORS' MANUAL—
RESOURCE CONTROL
ADDENDUM C

SUBJECT

Additions and Changes to the Manual

SPECIAL INSTRUCTIONS

Refer to the Preface for "Significant Changes".

This manual is one of five manuals that constitute the *Multics Administrators' Manual (MAM)*.

Order No.	*Title*
AK50	*System*
AK51	*Project*
AS68	*Registration and Accounting*
CC74	*Resource Control*
CC75	*Communications*

This is the third addendum to CC74, Revision 0, dated November 1978.

Insert the attached pages into the manual according to the collating instructions on the back of this cover. Throughout the manual, change bars in the margins indicate technical additions and asterisks denote deletions.

Note:

Insert this cover after the manual cover to indicate the updating of the document with Addendum C.

SOFTWARE SUPPORTED:

Multics Software Release 10.1

ORDER NUMBER

CC74-00C

February 1983

35990
1282
Printed in U.S.A.

Honeywell

COLLATING INSTRUCTIONS

To update the manual, remove old pages and insert new pages as follows:

Remove	Insert
iii, iv	iii, blank
3-3, 3-4	3-3, 3-4
4-5, 4-6	4-5, blank 4-5.1, 4-6
4-9, 4-10	4-9, 4-10

The information and specifications in this document are subject to change without notice. This document contains information about Honeywell products or services that may not be available outside the United States. Consult your Honeywell Marketing Representative.

© Honeywell Information Systems Inc., 1983

File No.: 1L13, 1U13

2/83

CC74C

SERIES 60 (LEVEL 68)

MULTICS ADMINISTRATORS' MANUAL — RESOURCE CONTROL

SUBJECT

Description of Commands and Procedures to Be Used with the Resource Control Package (RCP) Resource Management Facility

SPECIAL INSTRUCTIONS

This manual is one of five manuals that constitute the *Multics Administrators' Manual (MAM)*.

Project Registration and Accounting System	Order No. AK51 Order No. AS68 Order No. AK50
RCP	Order No. CC74
Communications	Order No. CC75

Note:

This manual is a preliminary edition provided to describe a special release of software that is not being distributed to all customers at this time. It will be extensively revised in the near future, and the user commands and subroutines moved to the MPM Commands and the MPM Subroutines.

SOFTWARE SUPPORTED

Multics Software Release 7.0R (MR7.0 augmented by Resource Management Special)

ORDER NUMBER

CC74, Rev. 0

November 1978

Honeywell

PREFACE

Multics system administration software controls the use of system resources and keeps records about how they are used. It supports rationing of resources, provides system security services, and produces usage reports and bills as required.

The administrative and resource control functions of the Multics system compose a sizeable subsystem. They are designed to be expanded or optionally bypassed and to allow flexibility for each installation.

There are four kinds of administrators who manage Multics system administration facilities: the project administrator, the registration and accounting administrator (referred to as the accounting administrator), the system security administrator, and the system administrator.

The reference manuals for Multics administrators are collectively referred to as the Multics Administrators' Manual (MAM). Throughout this document, references to the MAM are as follows:

Document	Referenced In Text As
Project (Order No. AK51)	MAM Project
Registration and Accounting (Order No. AS68)	MAM Accounting
System (Order No. AK50)	MAM System
Resource Control (Order No. CC74)	MAM RCP
Communications (Order No. CC75)	MAM Communications

The MAM Project is a guide to the operation of programs in the project-administration area. The information in this manual is of interest not only to project administrators but also to accounting administrators (who may function as project administrators) and to system administrators (who may function in any administrative capacity).

The MAM Accounting is a guide to the operation of Multics billing and accounting programs. It is necessary that both the accounting and system administrators know how to perform the Multics billing operations.

© Honeywell Information Systems Inc., 1979

The MAM Accounting is a guide to the operation of Multics billing and accounting programs. It is necessary that both the accounting and system administrators know how to perform the Multics billing operations.

The MAM System is a guide to the overall administration of the Multics system. This manual discusses the contents of administrative directories and data bases and special user identities (such as the daemons), describes installation parameters and system logs, explains the various tasks that are the responsibility of the system administrator, and includes the commands needed to carry out these responsibilities. Also, the functions of the system security administrator are explained in the MAM System.

The MAM Communications is a guide to the operation of the Multics Communication System (MCS). The manual includes information on terminal types, line types, and channel management.

Significant Changes in CC74-00C

In Section 3, information has been added to the Naming Rules for Attribution.

There have been some changes to the cv_rtmf command in Section 4. Also, the control argument, -severity, has been added to this command.

For purposes of clarity and ease of use, the MPM set has been reorganized. The six former MPM manuals, the Tools manual, and the RCP Users' Guide have been consolidated into a new set of three manuals.

Multics Programmer's Reference Manual (AG91)
: contains all the reference material from the former eight manuals.

Multics Commands and Active Functions (AG92)
: contains all the commands and active functions from the former eight manuals.

Multics Subroutines and Input/Output Modules (AG93)
: contains all the subroutines and I/O modules from the former eight manuals.

The following manuals are obsolete:

Name	Order No.
MPM Peripheral Input/Output	AX49
MPM Subsystem Writers' Guide	AK92
Programming Tools	AZ03
MPM Communications I/O	CC92
Resource Control Users'Guide	CT38

References to these manuals still exist on pages not published with this addendum. When this manual is revised, the references in the text to the old manuals will be changed to reflect the new organization.

CONTENTS

		Page
Section 1	Introduction	1-1
	Overview of the Resource Management Facility	1-1
	Reservation, Assignment, and Attachment	1-2
	Access Control	1-4
	Access Control Segments	1-4
	Access Class Ranges	1-4
	RCP Effective Access	1-5
	Manipulating RCP Effective Access	1-5
Section 2	Administrative Interfaces	2-1
	Administrative Data Bases and Interfaces	2-1
	Resource Type Description Table	2-1
	Registries	2-2
	Resource Management Activation and Auto Registration	2-3
Section 3	Resource Type Master File	3-1
	Syntax of the RTMF	3-1
	Resource Type Entries	3-1
	Fixed Resource Parameters	3-2
	Default Resource Parameters	3-2
	Canonicalization Routines	3-2.1
	Special Registration Parameters	3-3
	Resource Type Synonyms	3-3
	Naming Rules for Attributes	3-3
	Application of Defaults	3-4
	Source File Example	3-5
	Reserved Names	3-7
	Reserved Resource Names	3-7
	Reserved Attribute Names	3-8
Section 4	Administrative Commands	4-1
	clear_resource	4-4
	copy_registry	4-4.1
	cv_rtmf	4-5
	delete_registry	4-6
	deregister_resource, drr	4-6.1
	display_rtdt	4-7
	reconstruct_registry	4-8
	register_resource, rgr	4-9
	remove_registry	4-11
Appendix A	User Commands	A-1
Appendix B	Registry Checkpoint and Recovery	B-1
Appendix C	Administrator's Guide to Tapes at Sample Site	C-1
Index		i-1

TABLES

Table 1-1.	RCP Effective Access	1-5
Table 2-1.	Registry Data Manipulation	2-3

SECTION 1

INTRODUCTION

This manual contains the information necessary to perform the administrative duties required by the Resource Management Facility. Procedures are described to enable administrators to:

- construct the resource type master file (RTMF) (define the types of resources available on the system)
- register resources (introduce them to the system as available to authorized users)
- acquire resources for a specified user
- deregister resources (retract the availability of these resources)

Section 2 describes administrative procedures, the necessary data bases for resource management, and how to turn on resource management and auto registration. Section 3 describes the resource type master file (RTMF), syntax of entries therein, allowable parameters and defaults, a sample RTMF, and reserved names. Section 4 contains the descriptions of the administrative commands. Appendix B, "Registry Checkpoint and Recovery," describes the procedures that record registry information and the steps that reconstruct Resource Management registries. Appendix C describes how administrators handle tapes at a sample site.

OVERVIEW OF THE RESOURCE MANAGEMENT FACILITY

The resource control package (RCP) resource management facility is the part of the Multics operating system that manages the use of peripheral I/O devices (such as tape drives, printers, and punches) and physical volumes that can be mounted on these devices (such as tape reels, forms, and disk packs). These resources are managed by programs located in the administrative ring (ring 1), and run in the user's process.

The resource management facility handles registration and acquisition of resources, which includes deregistration and release.

RCP software reserves, assigns, and mounts resources; also demounts, unassigns, and cancels reservations.

The hierarchical level of these functions are:

```
1   register          }
    2   acquire       }   Resource Management

        3   reserve       }
            4   assign    }
                5   attach }
                5   detach }   Resource Control
            4   unassign   }
        3   cancel        }

    2   release       }
1   deregister        }   Resource Management
```

The function of RCP is to control the access to and usage of I/O devices. RCP executes in ring 1. Access to the various functions of RCP are controlled by the ring 1 gates that must be used to call RCP. One of the primary functions of RCP as a device manager is to control access to the I/O interfacer (IOI). In order to do this, no IOI gate entries are available to perform device attachments, detachments, and other privileged administrative functions. User ring programs, therefore, call RCP in order to request IOI to perform these functions.

An important feature of RCP is its ability to retain registration information for all resources that it controls. It does this by providing administrative interfaces for the registration of resources (see Section 2). Registration of a resource provides information such as what type of resource it is, what its name is, which attributes it possesses, and in what access class range the resource can be used. Once a resource is registered, users may acquire it; system administrators can also acquire it to a user (or to the system pool) at the time it is registered (see the register_resource command in Section 4). The act of acquisition makes a user the owner of the resource--liable for all changes to that resource and in control of discretionary access to the resource.

Another important feature of RCP is its ability to control access to the various resources that it manages (where a resource is either a device or a volume). It does this through the use of access control segments (ACSs). An ACS is a zero length segment whose ACL and ring brackets are used to define the discretionary and intraprocess access to a resource. At a site's discretion, additional features of RCP can be enabled to provide nondiscretionary access control for resources. If this is done, access is also controlled by the AIM access class range of a resource. (See "Access Control" below.)

The resource management functions performed by RCP are:

1. maintain resource information
2. control access to resources
3. reserve and cancel reservation of resources
4. assign and unassign devices
5. attach and detach devices
6. perform special device control functions

Reservation, Assignment, and Attachment

The functions reserve, assign and attach are organized into hierarchical levels. Defaults are provided at each level so that users not desiring to exercise features specific to a level do not have to concern themselves with that level.

```
1  reserve
   2  assign
      3  attach
      3  detach
   2  unassign
1  cancel
```

The first level involves the reservation of resources by processes. Tape drives, disk drives, tape volumes and disk volumes can be reserved. Reservations are process-specific and remain in effect until the process requests a cancellation. Reservation implies that a process temporarily has exclusive rights to a resource. This exclusive right means that no other process can use that resource for the duration of the reservation. Reservation does not necessarily imply that a resource is actually being used. Multiple resources can be reserved with one reservation.

Assignment, like reservation, is process-specific and lasts until unassignment or process termination. Any resource type can be assigned. An assignment also gives a process temporary exclusive rights to a device. Assignment does not necessarily mean that a device is currently being used. That is the function of the next level, attachment. Only one resource can be assigned per assignment.

A resource cannot be used until it is attached. When RCP is called to attach a resource, it initiates communication with the ring 0 subsystem that actually provides the use of the resource. Before the attachment is completed, RCP performs all initialization necessary to allow the attaching process to begin using the resource. For devices, this involves attaching the device via IOI and making sure that the device is ready and that any volume needed has been determined to be accessible, mounted, and authenticated.

The hierarchical relationship among reservation, assignment, and attachment implies that a higher-level function (e.g., reservation) can stand alone, while a lower-level function (e.g., attachment) can only be performed after all higher-level functions have been performed. RCP can perform the following device reservation, assignment, and attachment functions:

1. Reserving a resource. This means that no other process can use it during this period of time.
2. Explicitly assigning a reserved device. The device is assigned to a process but is not attached.
3. Attaching an explicitly assigned device.
4. Attaching an unassigned device. Since a device cannot be attached until it is assigned, RCP automatically reserves and assigns the device and then performs the attachment. The device is said to be implicitly assigned.
5. Detaching an implicitly assigned device. After the device is detached, RCP automatically unassigns the device.
6. Detaching an explicitly assigned device. The device is detached but is not unassigned.
7. Explicitly unassigning a device. If the device is attached, it is first detached and then unassigned.
8. Cancelling reservation of a resource.

The rules stated above imply that I/O modules do not have to be concerned with the assignment or unassignment of devices. They need to be concerned with only the attachment and detachment of a device. RCP, however, does allow the above rules to be overridden. When detaching a device an I/O module can tell RCP to retain the device assignment regardless of whether the device was explicitly or implicitly assigned.

When a process terminates, RCP automatically detaches and unassigns all devices currently assigned to that process and cancels any reservations for that process.

The reservation of resources and cancellation of reservations are done from command level via the reserve_resource and cancel_resource commands or by using the -resource control argument with the enter_abs_request command. The explicit assignment and unassignment of devices is done from command level via the assign_resource and unassign_resource commands. The listing of reservations, assignments, and attachments is done from command level via the list_resources command. These commands are described both in the MPM Commands and, with the exception of the enter_abs_request command, in the Resource Control Users' Guide.

ACCESS CONTROL

There are three types of access control on Multics: discretionary access control, which is regulated by access control lists (ACL); nondiscretionary access control, which is regulated by the access isolation mechanism (AIM); and intraprocess access control, which is regulated by the ring structure. (For detailed information on types of access, see the MPM Reference Guide.)

Access Control Segments

An important feature of RCP is its ability to control access to the various resources that it manages. It does this through the use of access control segments (ACSs). An ACS is a zero length segment whose ACL and ring brackets are used to define the discretionary access to a resource. RCP uses an ACS for each resource that it controls; however, an ACS can be shared by more than one resource. The name of an ACS consists of a name plus the suffix, acs (e.g., tape_01.acs). There are no restrictions on ACS names other than the required suffix. The user creates an ACS and generates/manipulates its ACL with the create, set_acl, and delete_acl commands and ring brackets with the set_ring_brackets command.

The pathname of the ACS for a resource is usually specified when it is acquired (see the register_resource command in Section 4 of this manual and the acquire_resource command in the Resource Control Users' Guide). The specified ACS can later be changed via the set_resource command (see the Resource Control Users' Guide). If the ACS has not been specified or does not exist, access is by default rew for the owner of the resource and null for all other users (see access modes in the glossary of the Resource Control Users' Guide).

RCP uses the ACS along with other nondiscretionary controls (AIM) to determine the RCP effective access to a resource.

Access Class Ranges

Access class ranges are used by RCP to specify that a process within a range of authorizations can use a particular resource.

An access class range is simply a pair of AIM access classes separated by a colon. The first value of the pair is the minimum access class and the second is the maximum access class. If only a single access class is specified when an access class range is expected, the minimum and maximum access class values are both the same (i.e., a range of one value). The second access class of the pair (the maximum) must be greater than or equal to the first (the minimum) according to the aim_check_ subroutine (see the MPM Subroutines).

There are some interesting results which occur when categories are used in an access class range. For example, a process with authorization of:

```
level2,category1
```

would not be able to use a resource whose access class range was:

```
level1,category1,category2:level3,category1,category2,category3
```

where level3 is greater than level2, which is greater than level1. This is due to the fact that the authorization of the process is isolated from the minimum of the access class range. In order to allow this process access to the resource in question, the range would have to exclude category2 or the user would have to have category2 authorization. In general, to include categories within an access class range, both the minimum and maximum must include the categories desired. If combinations of categories are desired, the minimum should list only required categories and the maximum should include all categories allowed. For example, the access class range:

```
level1,category1:level3,category1,category2,category3
```

allows read and write access to any level1, level2, or level3 process with category1 and any combination of category2 and category3.

RCP Effective Access

Viewed separately, each type of access control answers the same question, "What access does a particular process have for a particular item?" The access mode granted a process to a resource by discretionary access control (the ACL) is known as the raw access mode.

The way RCP determines effective access to a resource for a process differs from the regular Multics method of determining effective access as follows. First, the effective access to the ACS for the resource is determined as for any segment. If the ACS does not exist, the user appears to have read, execute, and write access if he is the owner of the resource, or null access if he is not the owner. Then, two further checks are made. First, the current authorization of the process is compared to the maximum access class of the resource. If write access is not allowed (as defined by the write_allowed_ subroutine) then write and execute access are denied and only read is allowed. Next, the current authorization of the process is compared to the minimum access class of the resource. If read access is not allowed (as defined by the read_allowed_ subroutine) then all access is denied. The resulting access is termed the RCP effective access to the resource. One final restriction enforced by RCP is that, in order to use a device, the RCP effective access must include both read and write to that device (a restriction not imposed on volumes).

For example, the following table illustrates some examples of RCP effective access. In the examples below, L1, L2, L3 and L4 represent sensitivity levels and c1, c2, c3, and c4 represent categories. (This discussion mostly concerns devices--volumes should never be given a multiclassed access class range.)

Table 1-1. RCP Effective Access

Effective Access to ACS	Current Process Authorization	Resource Access Class Range	RCP Effective Access
rew	L1	L1:L3	rew
re	L1	L1:L3	re
rew	L1	L2:L3	null
rew	L3	L2:L3	rew
rw	L4	L2:L3	r
re	L4	L2:L3	r
rw	L2,c1	L1:L4	r
rw	L2,c2	L1,c1:L4,c1,c2	null
rw	L2,c1,c3	L1,c1:L4,c1,c2	r
rw	L2,c1	L1,c1:L4,c1,c2	rw

A user must have write RCP effective access to the resource named to perform any modification on the status of the resource. In addition, the user must have execute effective access to the resource named to modify protected attributes. Only the accounting owner may modify the ACS path.

For more information on AIM, access classes, authorizations, and comparisons involving access classes and authorizations, see the MPM Reference Guide. The access class range mentioned above is specified by the -access_class control argument, which can be specified in the register_resource command (described in Section 4), and the acquire_resource and set_resource commands (described in the Resource Control Users' Guide).

Manipulating RCP Effective Access

Since the access control mechanisms described above operate together to determine the RCP effective access of a process, there are actions that the user, as well as an administrator, can perform to control this effective access.

First, the user creates an ACS via the create command. Then, the desired ACL for that segment is established using the set_acl command to add desired ACL entries, and the delete_acl command to delete entries. (The above three commands are described in the MPM Commands.) To further affect the ACS, the user may modify its ring brackets by using the set_ring_brackets command (described in the MPM Subsystem Writers' Guide). The system security administrator sets the AIM access class range of the resource itself at the time it is registered using the register_resource command, and can change it by using the set_resource command.

SECTION 2

ADMINISTRATIVE INTERFACES

ADMINISTRATIVE DATA BASES AND INTERFACES

Several data bases and administrative commands are required to manage resources via the RCP facility. If resource management is not activated, these features can be ignored. However, once resource management is enabled (see "Resource Management Activation and Auto Registration" below), the RCP administrator must manage the data bases and perform privileged actions for the user community.

The sequence of events that must occur to use a resource with resource management enabled is:

1. The RCP administrator registers the resource using the register_resource command (described in Section 4), making the resource known to the system.

2. The user acquires the resource using the acquire_resource command (described in the Resource Control Users' Guide), telling the system to make him owner and stating his willingness to pay for the resource (this can also be done by the administrator for the user).

3. Now the resource may be used by any user with appropriate access.

A variety of information is stored by the system as part of resource management. This information is under the control of the RCP administrator. This includes all of the information in the resource type description table (RTDT) and most information in the registries. The RTDT, which is generated by the RCP administrator, defines all of the resource types known by the system. Also defined in the RTDT are default values for the potential attributes, the potential access class range, and the charge type to be used in billing for resources of a given type. The registries contain information specified by the administrator at registration time or when a resource is acquired for a user.

Resource Type Description Table

The resource type description table (RTDT) is one of the data bases that is a central part of the operation of RCP; it is located in the directory, >system_control_1.

The RTDT is a binary table containing an entry for each resource type in the system. Therefore, it cannot be examined or modified with text editors. The display_rtdt command is used to print the contents of the whole RTDT or selected entries. When the system administrator wishes to add or delete resource types or to change the information about a resource type in its RTDT entry, he modifies the resource type master file (RTMF), compiles the RTMF into a new copy of the RTDT with the cv_rtmf command, and uses the install command (see the MAM System) to signal the answering service to modify the system's copy of the RTDT. By this means it is possible for the site management to change the site's specification of resource types without waiting for a system shutdown. The RTDT describes all of the resource types known to the system--both device types and volume types. For each resource type, it specifies the attributes that a resource of this type may possesses and the type of resource(s) that may be used with it. For example, an RTDT would say that tape_drive and tape_volume are resource types known to the system. It would also indicate that a tape_drive is a device and that a tape_volume is a volume. It might say that this type of device can be either 7 or 9 track and can be used at densities of 800 or 1600. It would also mention that a tape_drive is used in conjunction with a tape_volume.

All of the resource types currently defined in the RTDT can be listed using the list_resource_types command (see Resource Control Users' Guide).

Registries

The remaining data bases involved in resource management are the registries. These are keyed files that are used to contain registration information for specific resources. Registries are created by RCP and manipulated using the commands described later in this manual. A registry might, for example, specify that tape_drive tape_01 is owned by the system and its access control segment (ACS) pathname is >sc1>rcp>tape_drives.acs. It might also specify that the tape drive can be used by a process with any authorization from system_low through system_high, and other information such as what attributes it possesses and where it is located.

Resources are entered in a registry by the register_resource command. Once registered, a resource's information can be manipulated via the set_resource, acquire_resource, release_resource, and deregister_resource commands (see Section 4 and Resource Control Users' Guide). Use of the -priv control argument provides rew raw access to the resource regardless of the access to the ACS.

Table 2-1. Registry Data Manipulation

Data	Control Argument
ACS pathname	-acs_path
access class range	-access_class
current attributes	-attributes
comment	-comment
location	-location
owner	-owner
potential access	-potential_access_class
potential attributes	-potential_attributes
release lock	-release_lock
usage lock	-lock
user allocation	-alloc

If the potential access class range or potential attributes are not specified via the control arguments listed above, the default is supplied from the RTDT (see "Default Resource Parameters" in Section 3).

Resource Management Activation and Auto Registration

To allow for a more expedient transition from RCP without resource management to RCP with resource management, an RCP mode is provided that may be turned on or off via the ed_installation_parms command, using the auto_registration and rsc_mgmt_enabled keywords (see the MAM System). It is important to note that the use of resource management is not automatic--it must be turned on.

Once resource management and auto registration are enabled, resources can be automatically registered and acquired by users. This only occurs for tape or disk volumes that are not already registered and for which the operator gives approval by honoring the mount request. It is important to note that this is a "guaranteed" acquisition. That is, the first person for whom the operator mounts the volume is guaranteed the ownership of the volume. If auto registration is not on, only registered resources are usable on the system.

SECTION 3

RESOURCE TYPE MASTER FILE

The RTMF describes all resource types on the system. It is an ASCII file, which when compiled into a binary table, is installed by the answering service at the system administrator's request.

SYNTAX OF THE RTMF

The RTMF consists of a series of entries, one for each resource type that is known to the system. Each entry consists of a series of statements and substatements of the form:

<keyword>: <parameter>;

PL/I-style comments beginning with /* and ending with */ may appear anywhere within the RTMF. Similarly, blanks, tabs, and newlines not embedded within a keyword or parameter are ignored. However, to include blanks, tabs, newlines, colons, or semicolons in a parameter, it must be enclosed in quotes. If a parameter begins with a quote, all immediately following characters up to the next quote are taken as the parameter. It is not possible to embed quotes within a quoted string in the RTMF.

The last statement in the RTMF must be the statement:

end;

Resource Type Entries

The RTMF consists of one or more complete resource type entries. Each entry must include the name of the resource type being described, fixed parameters for that resource type, and default parameters to be applied at registration time in the absence of explicit parameters. In addition, a resource type entry may contain one or more sets of special registration parameters.

Each entry must begin with one of the following statements:

Volume: <volume_type_id>;
Device: <device_type_id>;

All statements following, until the next occurrence of a Volume or Device keyword, apply to this entry.

Fixed Resource Parameters

The following statements describe fixed parameters that apply to all resources of the given resource type, and may not be changed at registration time:

Limit: <number>;
Limit: open;
defines the maximum number of this type of resource that any user is allowed to assign at one time. If open is specified, no limit is enforced. This limit applies only to resources for which the system is the accounting owner. The limit does not apply to users using privileged RCP facilities. If this statement is not supplied, open is assumed.

Attribute_domain: {<attribute_list>};
specifies all of the allowable attributes that any resource of this type may possess. The <attribute_list> is of the form:

<name1>,<name2>,...,<nameN>

The attribute list is allowed to be empty. In entries for a volume type, any attribute may be followed by an asterisk. This specifies that any volume for which this attribute is currently active can be mounted only on a device which also possesses this attribute. See "Naming Rules for Attributes" below for more about the consequences of naming attributes.

Canonicalizer: <virtual_entry>;
specifies a program which is to be used to perform standardization of resource names as typed by the user, before they are presented to RCP Resource Management. If this statement is omitted, no canonicalization is performed. See "Canonicalization Routines", below.

Manual_clear: <yes_or_no>;
controls the operation of the resource data security features of automatic acquisition and release. If <yes_or_no> is the string "yes", volumes of this type are locked (when released by an accounting owner) in a way that does not allow another user to acquire them until the operator certifies that the volume has been cleared of all residual information. If this statement is omitted, "no" is assumed.

Default Resource Parameters

The following statements describe default parameters that apply during registration of all resources if no values for the parameters are explicitly provided in the registration command:

potential_attributes: <attribute_list>;
specifies the default potential attributes with which a resource is registered if no potential attributes are explicitly provided. The syntax of the attribute list is as given above for the Attributes keyword, except that asterisks on attributes are unnecessary.

access_range: <aim_range>;
specifies the default potential access class range with which a resource is registered if no potential access class is explicitly provided. If <aim_range> contains delimiters such as commas, colons, or spaces, it must be enclosed in quotes.

attributes: <attribute_list>;
specifies the default initial attributes with which a resource is registered if no attributes are explicitly provided. These attributes are also used to provide any defaults for attributes specifications at the time a resource is used (i.e., track=9 if no track specification is present).

Canonicalization Routines

Each resource type entry in the RTMF may specify an associated canonicalization routine. This routine is responsible for canonicalizing resource names as typed by the user into standard forms (for example, stripped of leading zeroes). Standard canonicalization routines exist and are supplied for certain resources (see the table below). Sites may choose to replace or augment the standard algorithms with those of their own choosing (for example, to enforce a more restrictive set of rules for tape volume names).

The canonicalization routine for a resource type is specified as a virtual entry in the "Canonicalizer" statement in an RTMF. (See the documentation for cv_entry_ in the MPM Subsystem Writers' Guide for a more detailed description of the syntax of virtual entries.) The canonicalization routine specified should be installed in a system library and must be executable from ring 1, as it will be used by RCP Management in that ring.

The calling sequence for all canonicalization routines supplied by the site must conform to the following:

```
declare <virtual_entry> entry (char(*), char(*), pointer, fixed!bin(35));

call <virtual_entry> (input_name, output_name, info_ptr, code);
```

where:

1. input_name (Input)
   is the resource name to be canonicalized.

2. output_name (Output)
   is the canonicalized representation of input_name.

3. info_ptr (Input)
   is currently unused and is null.

4. code (Output)
   is a standard error code.

Standard Canonicalization Routines

Resource Type	Routine Name
tape_vol	canon_resource_name_$tape_vol
tape_drive	None
punch	None
reader	None
console	None
printer	None
disk_vol	None
disk_drive	None
Special	None

Special Registration Parameters

The system administrator may specify, by name, a special class of any resource, and establish defaults for this class that are different from the normal defaults. A statement of the form:

```
type:  <type_name>;
```

introduces a special class of the resource named <type_name>. All default resource parameters following, until the next occurrence of a type, Volume, or Device keyword apply to this special class.

Resource Type Synonyms

The system administrator may specify that a certain resource type is to be known by more than one name. Each additional name must be defined via a Volume or Device keyword, as above, followed by the statement:

```
Like:  <volume_type_id>;
   or
Like:  <device_type_id>;
```

where <volume_type_id> or <device_type_id> specifies the resource type for which this entry is to be a synonym. For example, if the resource type "tape" is to be made synonymous with resource type "tape_vol", the RTMF should contain an entry of the form:

```
Volume:  tape;
Like:  tape_vol;
```

No other keywords should appear in a synonym definition entry.

Naming Rules for Attributes

Attributes provide a description of a volume or device that assists the resource management facility in the proper matching of volumes with compatible devices. To produce correct combinations, attribute names must comply with the set of rules described below.

Attributes may be grouped or ungrouped. Grouped attributes specify a set of properties applicable to a device or volume such that only one attribute of that set can be currently active at any given time. For example, a reel of tape may have potential attributes that allow it to be recorded at densities of 556, 800, or 1600; however, at any given time, the data on it is in only one of those densities. Grouped attributes have names of the form:

```
<identifier>=<value>
```

For example, the attributes mentioned above are named "den=556", "den=800", and "den=1600". This notation allows RCP to recognize that any request to make one of these attributes the current attribute of a device or volume also implies that all other attributes in that grouping must be made inactive.

When adding or changing an attribute in a string of attributes, all attributes in the string must be respecified or else existing attributes are nullified by the change. Also, any attribute string must contain a value for each grouped attribute. For example, if the attribute domain includes "track=..., model=..., and den=...," the device you are setting the attributes for (or registering) must contain values for each grouped attribute.

Ungrouped attributes have simple names, such as "trainok" (to specify that this device accepts a removable print train) or "building_12" (to specify that this device or volume is located in building 12).

APPLICATION OF DEFAULTS

When the system administrator registers a resource, that resource may be registered using the defaults for the registration parameters that are specified in the RTDT. Alternately, he may explicitly specify parameters for which defaults may also be specified in the table, such as attributes and AIM classes (see the register_resource command in Section 4). If any such parameter is explicitly specified, the corresponding default for that parameter is overridden.

When the resource is registered, any default parameters defined for that resource type are applied in the absence of a corresponding explicitly specified parameter.

If the resource is registered with the "-type <subtype_name>" control argument, any default parameter defined for the special class named <subtype_name> is applied in the absence of a corresponding explicitly specified parameter. In the case of duplicate resource type and special class parameters, the special class default parameters override the general resource type parameters. In addition, any default parameters specified for that resource other than those defaults in the special class are applied.

If no special classes of a resource are defined, and the defaults for the resource are not all present, it is always necessary for the missing parameters to be explicitly specified for every registration request for a resource of this type. If special classes of a resource are defined, then defaults within the definition of special classes can be used either to replace corresponding defaults specified for the resource in general, or to supplement for missing defaults that are not specified for the resource in general. In the latter case, the system administrator cannot perform a simple default registration of the resource, but must either specify the missing items explicitly in the command line, or use the "-type <subtype_name>" control argument to take advantage of the additional defaults provided in a special class.

SOURCE FILE EXAMPLE

```
/* Sample RCP Device and Volume Management Table */

/* Tapes and tape drives */

Volume:                 tape_vol;
Attribute_domain:       track=9*,track=7*,den=200,den=556,den=800,
                          den=1600*,den=6250*;
Limit:                  open;
Manual_clear:           no;

potential_attributes:
                        track=9,track=7,den=200,den=556,den=1600;
attributes:             track=9,den=1600;
access_range:           "system_low : system_high";

/* --------------- */

Device:                 tape_drive;
Attribute_domain:       track=7,track=9,model=400,model=500,model=600,
                          den=200,den=556,den=800,den=1600;
Limit:                  2;
Manual_clear:           no;

attributes:             track=9;
access_range:           "system_low : system_high";

type:                   tape7;
     potential_attributes:
                             track=7,model=500,den=200,den=556,den=800;
     attributes:             track=7;

type:                   tape9;
     potential_attributes:
                             track=9,model=500,den=200,den=556,den=800,
                               den=1600;
     attributes:             track=9;

/* --------------- */

Device:                 tape;
Like:                   tape_drive;

/* Punch */

Device:                 punch;
Attribute_domain:       ;
Limit:                  1;
Manual_clear:           no;

potential_attributes: ;
access_range:           "system_low : system_high";
```

```
/* Printers, print trains, and special forms */

Device:              printer;
Attribute_domain:    model=1200,model=301,model=1600,lpm=1200,lpm=1150,
                       trainok;
Limit:               1;
Manual_clear:        no;

potential_attributes:
                     model=1200,lpm=1200;
attributes:          model=1200;
access_range:        "system_low : system_high";

/* --------------- */

Volume:              print_train;
Attribute_domain:    uppercase,trainok*;
Limit:               2;
Manual_clear:        no;

potential_attributes: ;
access_range:        "system_low : system_high";

/* --------------- */

Volume:              form;
Attribute_domain:    labels;
Limit:               2;
Manual_clear:        no;

potential_attributes: ;
access_range:        "system_low : system_high";

/* Disks and disk drives */

Volume:              disk_vol;
Attribute_domain:    model=400*,model=451*,model=190*,use=io*,use=pv*;
Limit:               open;
Manual_clear:        no;

potential_attributes:
                     model=451,use=pv;
access_range:        "system_low : system_high";

/* --------------- */
```

```
Device:             disk_drive;
Attribute_domain:   model=400,model=451,model=190,model=500,use=io,
                      use=pv;
Limit:              open;
Manual_clear:       no;

potential_attributes:
                    model=451,use=pv;
access_range:       "system_low : system_high";

/* --------------- */

Device:             disk;
Like:               disk_drive;

/* --------------- */

end;
```

RESERVED NAMES

RCP uses the information in the RTDT to decide what classes of resources are known to the system, how they are to be handled, and what important attributes they possess. In the initial implementation, sites may use this flexibility to augment the standard complement of attributes for certain resources. For example, a site with tape drives in more than one location may register these drives with an additional simple attribute, thereby allowing users to request assignment of a tape drive in the remote location. Additionally, the tape reels in the remote location may be tagged with a matching attribute, marked in the RTDT as requiring that attribute of its tape drive.

Although this mechanism is very flexible, the necessity of having certain standard and reserved resource type names and attribute names cannot be avoided. Standard software (e.g., tape and disk I/O modules) needs to refer to a domain of resources by standard names, as well as certain attributes of the resources. Since these strings must be the same at all sites, certain resource types and certain resource attributes must be contained in all RTMFs. The cv_rtmf command checks for their existence and refuses to process an RTMF that lacks them. This list of required resource type names and attributes is also found in the include file, rcp_mandatories.incl.pl1.

RCP does not allow the name "scratch" to be used in registering a resource. A scratch tape is one of the unmarked tapes in an unreserved pool that is used for "scratch"--that is, no information is saved on it from session to session. After every use, it is demounted and returned to the system pool.

Reserved Resource Names

The following resources are mandatory and must appear in all RTMFs:

```
Device:   disk_drive
Device:   tape_drive
Volume:   tape_vol
Volume:   disk_vol
```

The following attributes are mandatory for the devices named, and must appear in all RTMFs:

For the disk_drive device:

model=400	model=451	model=181
model=191	model=500	model=402

For the tape_drive device:

track=7	track=9
den=200	den=556
den=800	den=1600
model=400	model=500
model=600	model=610

SECTION 4

ADMINISTRATIVE COMMANDS

COMMAND DESCRIPTION FORMAT

This section contains descriptions of the commands used by the various system administrators to register and deregister resources, convert the RTMF to an RTDT, and display the RTDT. Each description contains the name of the command (including the abbreviated form, if any), discusses the purpose of the command, and shows the correct usage. Notes and examples are included when deemed necessary for clarity. The discussion below briefly describes the content of the various divisions of the command descriptions.

Name

The "Name" heading lists the full command name and its abbreviated form. The name is usually followed by a discussion of the purpose and function of the command and the expected results from the invocation.

Usage

This part of the command description first shows a single line that demonstrates the proper format to use when invoking the command and then explains each element in the line. The following conventions apply in the usage line.

1. Optional arguments are enclosed in braces (e.g., {path}, {User_ids}). All other arguments are required.

2. Control arguments are identified in the usage line with a leading hyphen (e.g., {-control_args}) simply as a reminder that all control arguments must be preceded by a hyphen in the actual invocation of the command.

3. To indicate that a command accepts more than one of a specific argument, an "s" is added to the argument name (e.g., paths, {paths}, {-control_args}).

NOTE: Keep in mind the difference between a plural argument name that is enclosed in braces (i.e., optional) and one that is not (i.e., required). If the plural argument is enclosed in braces, clearly no argument of that type need be given. However, if there are no braces, at least one argument of that type must be given. Thus "paths" in a usage line could also be written as:

path1 {path2 ... pathn}

The convention of using "paths" rather than the above is merely a method of saving space.

4. Different arguments that must be given in pairs are numbered (e.g., xxx1 yyy1 {... xxxn yyyn}).

5. To indicate that the same generic argument must be given in pairs, the arguments are given letters and numbers (e.g., pathA1 pathB1 {... pathAn pathBn}).

6. To indicate one of a group of the same arguments, an "i" is added to the argument name (e.g., pathi, User_idi).

To illustrate these conventions, consider the following usage line:

```
command {paths} {-control_args}
```

The lines below are just a few examples of valid invocations of this command:

```
command
command path path
command path -control_arg
command -control_arg -control_arg
command path path path -control_arg -control_arg -control_arg
```

In many cases, the control arguments take values. For simplicity, common values are indicated as follows:

STR	any character string; individual command descriptions indicate any restrictions (e.g., must be chosen from specified list; must be either the string on or the string off).
N	number; individual command descriptions indicate whether it is octal or decimal and any other restrictions (e.g., cannot be greater than 4).
DT	date-time character string in a form acceptable to the convert_date_to_binary_ subroutine described in the MPM Subroutines.
path	pathname of an entry; unless otherwise indicated, it may be either a relative or an absolute pathname.

The lines below are samples of control arguments that take values:

```
-access_name STR, -an STR
-ring N, -rg N
-date DT, -dt DT
-home_dir path, -hd path
```

Notes

Comments or clarifications that relate to the command as a whole are given under the "Notes" heading. Also, where applicable, the required access modes, the default condition (invoking the command without any arguments), and any special case information are included.

Examples

The examples show different valid invocations of the command. An exclamation mark (!) is printed at the beginning of each user-typed line. This is done only to distinguish user-typed lines from system-typed lines. The results of each example command line are either shown or explained.

Other Headings

Additional headings are used in some descriptions, particularly the more lengthy ones, to introduce specific subject matter. These additional headings may appear in place of, or in addition to, the notes.

clear_resource

Name: clear_resource

The clear_resource command specifies that a resource has been manually cleared and should be returned to the free pool.

Usage

clear_resource type STR1 ... STRn

where:

1. type
   is the resource type defined in the RTDT.

2. STRi
   is the unique identifying name of the particular resource being cleared. If STR looks like a control argument (i.e., if it is preceded by a hyphen), then it must be preceded by -name or -nm.

Notes

For more information on clearing a resource, see "Fixed Resource Parameters" in Section 3.

Access Restrictions

The use of this command requires execute access to the rcp_sys_ gate.

copy_registry

Name: copy_registry

The copy_registry command is used by the system administrator to make checkpoint copies of RCP Resource Management registries. These copies can be used as a basis for the reconstruction of registries destroyed by catastrophic system failure.

Usage

```
copy_registry from_path {to_path} {-control_arg}
```

where:

1. from_path
   is the pathname of the registry to be copied. The star convention is accepted. If the suffix rcpr is not given, it is assumed.

2. to_path
   is the pathname of the copy to be created. The equals convention is accepted. If the suffix rcpr is not given, it is assumed. If to_path is not supplied, the copy will be placed in the working directory and will have the same name as the original. (See "Notes" below.)

3. control_arg
   can be -reset to specify that the contents of the registry journal are to be discarded after the copy operation has been successfully completed. (See "Notes" below.)

Notes

It is strongly recommended that the RCP Administrator NOT copy registries into >sc1>rcp (for reconstruction purposes or otherwise) except under special session.

copy_registry

The registry journal contains a record of all operations performed against all registries since the time its contents were last reset via the use of the -reset control argument described above. Since a successful reconstruction operation depends on the journal containing a record of all operations performed since the copies of the registries were created, it is important that the -reset control argument only be specified for invocations which result in the copying of all registries. The copying of any number of registries and the resetting of the journal within one invocation of the copy_registry command is performed as an indivisible operation, which guarantees that no operations can be performed against any of the registries involved until the copying operation is complete and the journal has been reset. Since this cannot be guaranteed between multiple invocations of the copy_registry command, the -reset control argument should never be used without copying all active registries.

When -reset is specified, the journal is reset only if the copy operations are completed successfully.

Copies of system registries are automatically made each night by the system accounting facility (crank) using this command.

Access Restrictions

This command requires access to the rcp_admin_ gate.

Example

To make checkpoint copies in the current working directory of all RCP Resource Management entries and discard journal entries made since the last checkpoint, type:

```
copy_registry ** -reset
```

cv_rtmf

Name: cv_rtmf

The cv_rtmf command converts an ASCII resource type master file (RTMF) into a binary resource type description table (RTDT). The binary table is installed using the install command (see the MAM System). If the user has made any errors in the RTMF, this command prints error messages while performing the conversion.

Usage

```
cv_rtmf rtmf_path {-control_args}
```

where:

1. rtmf_path
   is the pathname of the resource type master file. If path does not have a suffix of rtmf, one is assumed. However, the suffix rtmf must be the last component of the name of the source segment.

2. control_args
   can be chosen from the following.

   -brief, -bf
   prints short form of error messages

   -long, -lg
   prints long form of error messages

   -severity N, -sv N
   causes error messages whose severity is less than N (where N is 0, 1, 2, 3, or 4) not to be written to the user_output switch. If this control argument is not specified, a severity level of 0 is assumed (i.e., all error messages are written to the user_output switch).

Notes

If no control arguments are given, an error message is printed in long form the first time it occurs, and in short form thereafter.

The converted resource type master file is given a name corresponding to the entryname of the source segment, with the rtmf suffix replaced by rtdt. It is placed in the working directory.

cv_rtmf

The syntax of an RTMF is described in Section 3.

The cv_rtmf command associates the following severity values to be used by the severity active function:

Value	Meaning
0	No compilation yet or no error.
1	Warning.
2	Correctable error.
3	Fatal error.
4	Unrecoverable error.
5	Could not find source.

delete_registry

Name: delete_registry

The delete_registry command is used by the system administrator to delete old unused RCP Resource Management registries. These registries must have been previously removed from service via the remove_registry command.

Usage

delete_registry paths

where:

1. path
   is the pathname of an old registry to be deleted. The star convention is accepted. If the suffix old is not given, it is assumed.

Access Restrictions

This command requires access to the rcp_admin_ gate.

deregister_resource

Name: deregister_resource, drr

The deregister_resource command makes a particular resource unknown to the system. The deregistration process informs the system that the resource is no longer available for use.

Usage

deregister_resource type STR1 ... STRn

where:

1. type
   is a resource type defined in the RTDT.

2. STRi
   is the unique identifying name of the particular resource being deregistered. If STR looks like a control argument (i.e., if it is preceded by a hyphen), then it must be preceded by -name or -nm.

Notes

To be deregistered, the resource must be in the free state. A resource owned by a user (or belonging to the system pool) must be released (see the release_resource command in the Resource Control Users' Guide) before it may be deregistered.

If multiple resource names are specified to the deregister_resource command and an error occurs in the deregistration of any of these resources, none of the resources are deregistered.

Access Restrictions

The use of this command requires execute access to the rcp_admin_ gate.

display_rtdt

Name: display_rtdt

The display_rtdt command displays a resource type description table (RTDT).

Usage

display_rtdt {type1 ... typen} {-control_args}

where:

1. typei
   is the name of a resource type whose RTDT entry is to be displayed. If type is not given, all RTDT entries are printed.

2. control_args
   may be selected from the following:

   -no_header, -nhe
   does not print the RTDT header comment.

   -pathname path, -pn path
   displays the contents of the RTDT specified by path. If this control argument is not specified, the RTDT residing in >system_control_1 is displayed.

reconstruct_registry

Name: reconstruct_registry

The reconstruct_registry command is used to recover a current copy of RCP Resource Management registries after a catastrophic system failure causing the loss of one or more registries. It assumes that the registry to be reconstructed is a consistent earlier copy of the registry desired, and that the RCP Resource Management journal contains a record of all operations performed on the registry since the time represented by the earlier copy.

Usage

reconstruct_registry registry_names {-control_args}

where:

1. registry_names
   are the entrynames of the registries to be reconstructed. The star convention is accepted. If the suffix .rcpr is not given, it is assumed.

2. control_arg
   can be -pathname path (-pn path) to specify the directory in which the registries reside. If this control argument is not specified, the registries are sought in >sc1>rcp.

Notes

An explanation of the creation and maintenance of checkpointed registry copies can be found in the documentation of the copy_registry command.

The prescribed sequence of operations is to delete the damaged registries; copy the desired checkpointed registries into place; and invoke the reconstruct_registry command to update the registries. The command locates the RCP Resource Management journal relative to the directory in which the registries to be updated reside.

If an online checkpoint copy of a system registry is not available, a copy of the registry may be retrieved from a system backup tape. In this case, the file retrieved must be from a time that is more recent than the last time the RCP Resource Management journal was reset (see the documentation of the copy_registry command).

The reconstruction of system registries must only be performed from the Initializer, in the "standard" environment, before the answering service is activated.

Access Restrictions

This command requires access to the rcp_sys_ gate.

register_resource

Name: register_resource, rgr

The register_resource command is used by the registration administrator to make a particular resource known to the system. The registration process informs the system that the resource is available for users who are authorized to access it.

Usage

register_resource type STR1 ... STRn {-control_args}

where:

1. type
   is a resource type defined in the RTDT.

2. STRi
   is the unique identifying name of the particular resource being registered. If STR looks like a control argument (i.e., if it is preceded by a hyphen), then it must be preceded by -name or -nm. (The string "scratch" is not permitted--see "Reserved Names" in Section 3.)

3. control_args
   can be chosen from the following:

   -access_class accr, -acc accr
   sets the initial AIM access class parameters, where accr is an access class range. Users at any authorization within the access class range inclusive are allowed to read and write to the resource (provided they also meet other access requirements). For a detailed description see "Access Class Ranges" in Section 1.

   -acs_path path
   specifies the pathname of the access control segment (ACS) for this resource. The ACS is not created by this command, but must be created by the administrator, and the desired access control list set (see "Notes" below). If this control argument is not given, the accounting owner of the resource is given rew access by default. If path is a null string, the existing ACS, if any, is disassociated from the resource.

   -alloc STR
   sets the allocation state of the resource to free or allocated, where STR must be either the string on or the string off. If this control argument is not given, the allocation state is free. (The allocation state flag is a convenience to the user and is largely ignored by resource management.)
   on sets the allocation state to allocated
   off sets the allocation state to free

   -attributes STR, -attr STR
   specifies the initial values for the attributes of this resource. If this control argument is not given, the default attributes defined in the RTDT for this resource type are used (see "Notes" below).

register_resource

-comment STR, -com STR
specifies the initial value of the comment string for this resource.

-location STR, -loc STR
specifies a descriptive location for the resource, to aid the operator in locating it when it is stored in a special place (e.g., a vault, a different room, etc.).

-lock STR
locks or unlocks the resource, preventing or allowing use of that resource, where STR must be either the string on or the string off. If this control argument is not specified the lock is off.
on prevents any use of the resource
off allows use of the resource

-owner STR, -ow STR
specifies that this resource, as part of the registration process, is to be acquired on behalf of the user specified by STR. If STR is the string "system", then the resource is acquired to the system pool. If STR is of the form Person_id.Project_id (where neither Person_id nor Project_id may be a star), then the user specified has all the rights of ownership to the resource as if he had acquired it personally, except that if -release_lock on is specified, the owner may not release (give up ownership of) the resource voluntarily. If this control argument is not given, the resource is entered by default into the free pool.

-potential_attributes STR, -pattr STR
specifies the potential attributes to be assigned to this resource. If this control argument is not given, the default potential attributes defined in the RTDT for this resource type are used (see "Notes" below).

-potential_access_class accr, -pacc accr
sets the potential AIM access class parameters, where accr is the access class range. Users at any authorization within the access class range inclusive are allowed to acquire the resource. If the control argument is not given, the default potential access class defined in the RTDT for this resources type is used.

-release_lock STR, -rll STR
specifies whether this resource may be released by the owner, or may only be released by a privileged process. The STR argument must be either the string on or the string off. It is primarily useful to implement special arrangements between a site and a user whereby the user agrees to pay a fixed amount for the privilege of administrative power over a resource for an agreed-upon length of time. If this control argument is not specified, the resource may be released by the owner (does not require special privilege).
on resource may only be released by privileged process
off resource may be released by owner

-type subtype_name, -tp subtype_name
specifies that defaults for this resource are to be taken from the description of the resource subtype as defined in the RTDT (see "Application of Defaults" in Section 3).

register_resource

Notes

If multiple resources are specified to the register_resource command and an error occurs in the registration of any of these resources, none of the resources specified is registered.

The use of this command for controlling RCP effective access is explained under "Access Control Segments" in Section 1.

If no -owner is specified, the resource is placed in the free pool.

For a description of the syntax of attribute strings, see "Naming Rules for Attributes" in Section 3.

The use of the -access_class, -acs_path, -attributes, or -comment control argument requires that the -owner control argument be specified.

Access Restrictions

The use of this command requires execute access to the rcp_admin_ gate.

Certain specifications of AIM access class parameters (e.g., an access class lower than the user's current authorization) are rejected unless the user has the AIM rcp privilege.

remove_registry

Name: remove_registry

The remove_registry command is used by the system administrator to remove RCP Resource Management registries from service. This command should only be used in exceptional circumstances. (See "Notes" below.)

Usage

remove_registry paths

where:

1. path
   is the pathname of a registry to be removed from service. The star convention is accepted. If the suffix rcpr is not given, it is assumed.

Notes

When a registry is removed, its suffix is changed from rcpr to old.

The activity of removing registries is normally reserved to the Initializer process, which will automatically remove a registry when a new RTDT is installed that no longer contains an entry for the resource type associated with that registry. In general, manual removal of registries is only necessary in the process of recovery from a catastrophic system failure and reload, where the existing registries and the existing RTDT may be out of agreement. Manual removal of registries at other times can result in unrecoverable errors by RCP Resource Management.

Access Restrictions

This command requires access to the rcp_sys_ gate.

APPENDIX A

USER COMMANDS

This appendix has been moved to the RCP Users' Guide, Order No. CT38 and the MPM Subsystem Writers' Guide, Order No. AK92.

APPENDIX B

REGISTRY CHECKPOINT AND RECOVERY

When a resource is registered, its registration information is recorded in a registry. As the resource is acquired, released, and deregistered, the information in that resource's registry is altered.

This section describes the procedures used to record registry information so that if a registry is damaged, a sequence of steps can be performed to reconstruct a complete and current copy of the Resource Management registries.

Since the registry is a multisegment file, normal backup procedures are not sufficient; journalization and checkpointing procedures protect against loss of data through damaged registries. When a crash damages part of a registry, the damaged registry can be reconstructed from the checkpoint copy and the journal.

CHECKPOINT AND JOURNAL

Part of Resource Management is registry management. Every time that a change is made in a registry, an entry describing that change is made in a journal. The Resource Management journal contains a record of all operations performed on each registry since the time represented by the last checkpoint. The checkpoint copy is a complete copy of all the registries. The system accounting facility (see the SAM) automatically makes a checkpoint copy of the system registries every night using the copy_registry command (see Section 4). When the checkpoint copy is made, the journal entries are deleted since they are no longer needed. (The -reset control argument to the copy_registry command empties the registry journal of entries.)

RECOVERY

In the event of a system failure causing the loss of one or more registries, the system administrator can recover a current copy of the registries by using the reconstruct_registry command (see Section 4). The registry is reconstructed by merging the latest checkpoint copy with the journal entries.

The sequence of operations necessary to reconstruct damaged registries is:

1. Delete the damaged registries,
2. Copy the desired checkpoint registries into place, and
3. Invoke the reconstruct_registry command to update the registries.

1. Prior to deleting the damaged registries, the system administrator must remove the registries from service with the remove_registry command (see Section 4). All active registries have a suffix of ".rcpr". When registries are removed from service, the remove_registry command changes their suffix to ".old". When this has been accomplished, the delete_registry command is used to delete the unused registries.

2. The desired checkpoint registries are copied into place with the copy_registry command (see Section 4).

3. The reconstruct_registry command is then invoked to recover the registries. The journal is located and merged with the checkpoint copy, thus creating a current copy of the registries.

UNUSUAL RECOVERY

If an online checkpoint copy of a system registry is not available, a copy of the registry can be retrieved from a system backup tape. The tape must contain a complete dump of the registries, not an incremental or catchup dump. In this case, the copy retrieved must be from a more recent date than the last time the journal was reset.

APPENDIX C

ADMINISTRATOR'S GUIDE TO TAPES AT SAMPLE SITE

This section describes procedures used at a sample site by administrators in their handling of tapes. This can be viewed as an imaginary publication issued at a certain site, which includes certain capabilities that are site-specific. The nonstandard, site-specific items have been marked: (POLICY OF SITE).

USER TAPE POOL

A pool of tapes is kept available to be acquired by users via the acquire_resource command. (The names of the tapes given here are site-specific.) The names of these tapes have one of three prefixes, depending on the length of the tape. Tapes AU0001 through AU9999 are 600-foot tapes. Tapes BU0001 through BU9999 are 1200-foot tapes. Tapes CU0001 through CU9999 are 2400-foot tapes (POLICY OF SITE).

To add tapes to the pool, use the register_resource command:

```
register_resource tape_vol {tapes} -potential_attributes
     den=1600,den=6250,track=9,len=LEN
```

For example, to add the tapes AU0001 through AU0005:

```
rgr tape_vol au0001 au0002 au0003 au0004 au0005 -pattr
     den=1600,den=6250,track=9,len=600
```

To list all of the free (unacquired) tapes in the free pool, use the list_resources command:

```
list_resources -acq -type tape_vol -user free
```

To remove tapes from the pool, use the deregister_resource command:

```
deregister_resource tape_vol {tapes}
```

Only tapes that are currently free can be deregistered.

OTHER TAPES

Not all tapes are assigned via the acquire_resource command. Some tapes are assigned to users at the time the tape is registered (e.g., tapes to be used by the Backup system and tapes that users import from other computer systems) (POLICY OF SITE). By turning on the release lock, the administrator makes sure that these tapes may not released into the free pool by the owner using the release_resource command (POLICY OF SITE). These tapes must be released by an administrator.

To assign these tapes at the time they are registered, use the register_resource command with the -owner control argument:

```
register_resource tape_vol {tapes} -potential_attributes
     den=1600,den=6250,track=9,len=LEN -owner Person.Project -release_lock
     on
```

For instance, to register 2400-foot tapes V00001 through V00004 to be used by the volume dumper (and only at 6250 bpi):

```
rgr tape_vol v00001 v00002 v00003 v00004 -release_lock on -pattr
     den=6250,track=9,len=2400 -owner Volume_Dumper.Daemon
```

To register a 1200-foot foreign tape (assigned the name EX0001) for a user (Jones.CVall):

```
rgr tape_vol ex0001 -pattr den=6250,den=1600,track=9,len=1200 -owner
     Jones.CVall -release_lock on
```

To release and deregister such tapes, use the release_resource command with the -priv argument and then the deregister_resource command:

```
rlr tape_vol TAPENAME -priv;drr tape_vol TAPENAME
```

When a tape owned by the computer center leaves the machine room temporarily, its usage lock can be turned on. This prevents the user from trying to mount the tape and from releasing the tape. In addition, the location field may be set to indicate that the tape is not in the machine room. For example:

```
setr tape_vol TAPENAME -lock on -location offsite -priv
```

When the tape is returned, the usage lock and location fields should be reset. For example:

```
setr tape_vol TAPENAME -lock off -location "" -priv
```

OTHER OPERATIONS

To list all tapes that have been acquired by users, use the list_resources command as shown here:

```
list_resources -acq -type tape_vol -user *.*
```

To get detailed information about a particular tape (either a free tape or a tape that has been acquired by a user), use the resource_status command:

```
resource_status tape_vol TAPENAME -all
```

To get detailed information about a group of tapes, the resource_status command can be used in conjunction with the list_resources active function:

```
resource_status tape_vol [list_resources -acq -type tape_vol -user *.*]
```

INDEX

A

access class ranges 1-4

access control 1-4
  resources 1-4
  segment (ACS) 1-4

access control segment (ACS) 1-2, 1-4
  access class ranges 1-4
  RCP effective access 1-4

ACS
  see access control segment

administrative commands 4-1

administrative interfaces 2-1
  auto registration 2-3
  registries 2-2
    registry data manipulation 2-3
  resource management activation 2-3
  resource type description table (RTDT) 2-1

assignment 1-3

C

clear_resource 4-4

copy_registry 4-4.1

cv_rtmf 4-5

D

delete_registry 4-6

deregister_resource, drr 4-6.1

display_rtdt 4-7

R

reconstruct_registry 4-8

register_resource, rgr 4-9

remove_registry 4-11

reservation 1-2

resource 1-3

resource control package (RCP) 1-2
  access control
    resources 1-2, 1-4
      access control segment (ACS) 1-2
  assigning devices 1-2
  attaching devices 1-2
  cancelling resources 1-2

resource control package (RCP) (cont)
  control functions
    devices 1-2
  detaching devices 1-2
  reserving resources 1-2
  resource information 1-2
  unassigning devices 1-2

resource type master file
  application of defaults 3-4
  reserved names 3-7
    reserved attribute names 3-8
    reserved resource names 3-7
  source file example 3-5
  syntax of the RTMF 3-1
    canonicalization routines 3-2.1
    default resource parameters 3-2
    fixed resource parameters 3-2
    naming rules for attributes 3-3
    resource type entries 3-1
    resource type synonyms 3-3
    special registration parameters 3-3

RTMF
  see resource type master file

U

user commands
  see Resource Control Users' Guide

CUT ALONG LINE

HONEYWELL INFORMATION SYSTEMS
Technical Publications Remarks Form

TITLE	MULTICS ADMINISTRATORS' MANUAL - RESOURCE CONTROL ADDENDUM C	**ORDER NO.**	CC74-00C
		DATED	FEBRUARY 1983

ERRORS IN PUBLICATION

SUGGESTIONS FOR IMPROVEMENT TO PUBLICATION

Your comments will be investigated by appropriate technical personnel and action will be taken as required. Receipt of all forms will be acknowledged; however, if you require a detailed reply, check here. ☐

FROM: NAME _____ **DATE** _____

TITLE _____

COMPANY _____

ADDRESS _____

_____

PLEASE FOLD AND TAPE–
NOTE: U. S. Postal Service will not deliver stapled forms

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 39531 WALTHAM, MA02154

POSTAGE WILL BE PAID BY ADDRESSEE

HONEYWELL INFORMATION SYSTEMS
200 SMITH STREET
WALTHAM, MA 02154

ATTN: PUBLICATIONS, MS486

FOLD ALONG LINE

FOLD ALONG LINE

CUT ALONG LINE

Honeywell

HONEYWELL INFORMATION SYSTEMS
Technical Publications Remarks Form

CUT ALONG LINE

TITLE: SERIES 60 (LEVEL 68) MULTICS ADMINISTRATORS' MANUAL – RESOURCE CONTROL ADDENDUM B

ORDER NO. CC74-00B

DATED JUNE 1981

ERRORS IN PUBLICATION

SUGGESTIONS FOR IMPROVEMENT TO PUBLICATION

Your comments will be investigated by appropriate technical personnel and action will be taken as required. Receipt of all forms will be acknowledged; however, if you require a detailed reply, check here. ☐

FROM: NAME _____ DATE _____

TITLE _____

COMPANY _____

ADDRESS _____

_____

PLEASE FOLD AND TAPE—
NOTE: U. S. Postal Service will not deliver stapled forms

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 39531 WALTHAM, MA02154

POSTAGE WILL BE PAID BY ADDRESSEE

HONEYWELL INFORMATION SYSTEMS
200 SMITH STREET
WALTHAM, MA 02154

ATTN: PUBLICATIONS, MS486

Honeywell

CUT ALONG L

FOLD ALONG LINE

FOLD ALONG LINE

CUT ALONG LINE

HONEYWELL INFORMATION SYSTEMS

Technical Publications Remarks Form

TITLE: SERIES 60 (LEVEL 68) MULTICS ADMINISTRATORS' MANUAL – RCP PRELIMINARY EDITION

ORDER NO. CC74, REV. 0

DATED NOVEMBER 1978

ERRORS IN PUBLICATION

SUGGESTIONS FOR IMPROVEMENT TO PUBLICATION

Your comments will be promptly investigated by appropriate technical personnel and action will be taken as required. If you require a written reply, check here and furnish complete mailing address below. ☐

FROM: NAME _____ **DATE** _____

TITLE _____

COMPANY _____

ADDRESS _____

_____

CUT ALONG LINE

PLEASE FOLD AND TAPE –
NOTE: U. S. Postal Service will not deliver stapled forms

FOLD ALONG LINE

FIRST CLASS
PERMIT NO. 39531
WALTHAM, MA
02154

Business Reply Mail
Postage Stamp Not Necessary if Mailed in the United States

Postage Will Be Paid By:

HONEYWELL INFORMATION SYSTEMS
200 SMITH STREET
WALTHAM, MA 02154

ATTENTION: PUBLICATIONS, MS 486

FOLD ALONG LINE

Honeywell

HONEYWELL INFORMATION SYSTEMS
Technical Publications Remarks Form

CUT ALONG LINE

TITLE: SERIES 60 (LEVEL 68) MULTICS ADMINISTRATORS' MANUAL – RESOURCE CONTROL ADDENDUM A

ORDER NO. CC74-00A

DATED NOVEMBER 1979

ERRORS IN PUBLICATION

SUGGESTIONS FOR IMPROVEMENT TO PUBLICATION

Your comments will be investigated by appropriate technical personnel and action will be taken as required. Receipt of all forms will be acknowledged; however, if you require a detailed reply, check here. ☐

FROM: NAME _____ DATE _____

TITLE _____

COMPANY _____

ADDRESS _____

_____

PLEASE FOLD AND TAPE–
NOTE: U. S. Postal Service will not deliver stapled forms

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 39531 WALTHAM, MA02154

POSTAGE WILL BE PAID BY ADDRESSEE

HONEYWELL INFORMATION SYSTEMS
200 SMITH STREET
WALTHAM, MA 02154

ATTN: PUBLICATIONS, MS486

Honeywell

CUT ALONG LINE

FOLD ALONG LINE

FOLD ALONG LINE

Honeywell

Honeywell Information Systems
In the U.S.A.: 200 Smith Street, MS 486, Waltham, Massachusetts 02154
In Canada: 2025 Sheppard Avenue East, Willowdale, Ontario M2J 1W5
In Mexico: Avenida Nuevo Leon 250, Mexico 11, D.F.

22372, 5C1278, Printed in U.S.A.

CC74, Rev. 0