HONEYWELL

CP-6 SYSTEM MANAGER HANDBOOK

SOFTWARE

CP-6

SYSTEM MANAGER HANDBOOK

SUBJECT

Description of Techniques for Efficient Operation of an Installation

SOFTWARE SUPPORTED

Operating System C00.

ORDER NUMBER

CE60-00

March 1985

Honeywell

Preface

This handbook provides guidelines, procedures, and strategies for the system manager. This handbook is intended to be used in conjunction with the CP-6 System Support Reference manual (CE41) which gives encyclopedic detail on the system management processors.

The Los Angeles Development Center (L.A.D.C.) of Honeywell Information Systems Inc. has developed Computer Aided Publications (CAP). CAP is an advanced document processing system providing automatic table of contents, automatic indexing, format control, integrated text and graphics, and other features. This manual is a product of CP-6 CAP.

Readers of this document may report errors or suggest changes through a STAR on the CP-6 STARLOG system. Prompt response is made to any STAR against a CP-6 manual, and changes will be incorporated into subsequent releases and/or revisions of the manuals.

The information in this publication is believed to be accurate in all respects. Honeywell Information Systems cannot assume responsibility for any consequences resulting from unauthorized use thereof. The information contained herein is subject to change. New editions of this publication may be issued to incorporate such changes.

File No.: 1W13 CE60-00

Table of Contents

Tables:

Figures:

About This Manual

This handbook documents how the system manager and the system manager's staff use CP-6 system processors to set up and run a CP-6 system. The handbook is built as a modular document; each module or group of modules documenting one aspect of managing a CP-6 system. The current publication includes modules that document:

- Pre-installation planning considerations, including information on how to establish account conventions (Module 1-1).
- CP-6 security features (Module 2-1).
- Using the TIGR processor to define a hardware configuration and set software parameters (Module 3-1).
- Using the SUPER processor to create and maintain projects and logon accounts (Modules 4-1 through 4-3).
- Using the TIGR, NETCON, SUPER and PIGETTE processors to create and boot a CP-6 network (Module 5-1).
- Using the STATS processor to generate statistics that can be used in system tuning, and how to tune a CP-6 system (Modules 6-1 through 6-3).

Modules on additional subjects of concern in system management will be added to this document.

The tasks of the system management team are diverse and are generally performed by people with differing levels of awareness about the system. This document attempts to respond to that diversity.

Some modules assume a thorough knowledge of the system and are presented as annotated examples of how to do a particular system management task. Module 3-1, 'Device Configuration' and Module 5-1, 'Defining and Using a CP-6 Network' are examples of this type of module.

Other modules (for example, Module 4-2 and 4-3 on Project and User Authorization) are written in a more tutorial style for a less seasoned user.

Module 1-1
Pre-installation Planning

Need For Planning

If you are a newcomer to the world of large time-sharing systems, there are several new concepts that might be new to you. If you are an old hand, you will find that the CP-6 system has some unique ways of solving old problems. A re-thinking of the way you currently solve these problems is in order so that the full range of CP-6 capability is available to you, especially in the areas of security and performance.

This module will point out some of the decisions that must be made early which will affect how you will configure your system. Correct choices in the manner of use of CP-6 features will make the rest of the system management job much easier. The problems to be solved will be discussed in this module; the choices of methods available to solve them are discussed in later modules. The tools (i.e., system management processors) used by these methods are fully described in the System Support Reference manual (CE41).

Identifying and Categorizing Users

One of the first tasks involved in pre-installation planning is to identify and categorize the users of the system. The user-grouping scheme that is implemented by the system manager will have profound implications upon the performance of the CP-6 system; this module will attempt to explore these implications in detail, present some examples of grouping systems, and illustrate how these are implemented in a CP-6 system.

A CP-6 user may be an online user at a terminal, a batch job, a CP-6 ghost, or a transaction processing user. All users share common characteristics (such as the maximum CPU-time and resources useable by the user). These characteristics common characteristics; these are defined in the System Support Reference manual (CE41) under User Authorization.

Users may be grouped:

- By shared interest
- By existing organizational structure
- By common accounting requirements
- By arbitrary schemes

Some of these methods of categorizing users may overlap. For example, users may be grouped according to an existing organizational structure, but within several projects there may be common support functions which can be identified as such. This can be useful in accounting or in statistical analysis.

For example, Projects A, B, and C may all have a common function such as testing. Despite being attached to different projects, the testing function has similar requirements. By incorporating a common identifier (e.g., TEST) within the account designation for each testing subgroup (e.g., ATEST, BTEST, CTEST), the system manager can take advantage of the CP-6 feature, wildcarding. Substituting the wildcard character (?), as in the account specification ?TEST, identifies all accounts ending in TEST. Wildcarding allows the system manager to refer to a class of accounts rather than

identifying each account individually. More important than the convenience factor is that the system manager can use wildcarding to restrict file access to specific subgroups, or implement account procedures which affect only specific subgroups. (The heading 'Planning Account Designations' provides more detailed information on this topic.)

Interrelationships Between Groups

In defining the interrelationship between user groups, important considerations are:

- Security
- Availability
- Administration or control

Security

The following security questions should be taken into account:

1. Which groups should be permitted to access another's files, or be prevented from accessing them — for security reasons, or to avoid accidental damage?
2. Which groups should be restricted to the use of only a specific processor or group of processors?

System Availability

1. Which groups must be supported in degraded performance mode, and which groups are "expendable" given the necessity to make a choice?
2. Which groups should have their files stored in duplicate (i.e., DUALed via the EFT processor) and thus restored quickly in the event of disk hardware incapacity?
3. Which groups are autonomous in their file requirements?

Administration

In considering project administration as applied to grouping, the system manager must determine the scope of the project i.e., what CP-6 resources will be required, and what constraints are to be imposed.

Planning Account Designations

In a CP-6 system, each user runs under a user ID that consists of "account,name,password".

- account can be two or more things. It defines the user's file management account. It also defines the "access key" to be used to gain access to the files of others. It is the account field that appears on access lists attached to files, accounts, and packsets. When used in conjunction with CP-6 wildcarding, "account" is the cornerstone of CP-6 security and file management procedures. If the "account" matches (or fails to match) an account in a file access list, access is granted (or denied).
- name is only used by accounting routines as a way of separating users with the same account. The NAME identifies the individual user; however, there are grouping schemes in which the user NAME incorporates an additional designator which may serve to identify department or project. See grouping scheme 3 below.
- password is used only by the user.

Sample Grouping Schemes

At this point it might be a good idea to look at several schemes for grouping users, and discuss the advantages and disadvantages of each.

Scheme 1 - Using Random-number Accounts

When random numbers are assigned as accounts, there is, in effect, no grouping scheme. The user is identified by a name assigned to him and an account number. Each user within a project uses the same account number for accounting purposes, but this is a number which has been assigned with no classification or grouping scheme in mind. This has advantages for security reasons. However, it becomes laborious and time consuming to assign different projects access to common files, as CP-6 wildcarding can not be used to achieve this. The larger the number of accounts in this type of scheme, the more unwieldy the handling of accounts becomes, from a file access-granting standpoint. unavailable.

Scheme 2 - Account Grouping

A commonly used grouping scheme is one that assigns a two-letter designator for a department code, a three-number designator for a project code, a single letter to designate the user's job title, and a two-number programmer number.

By means of illustration, assume an installation has assigned an account as follows:

```
CD002A00
```

Where:
- CD — Represents the department code
- 002 — Represents the project code
- A — Represents the user's job title
- 00 — Represents the programmer number (Only project leaders receive a 00 number)

With this grouping scheme, the advantages of CP-6 wildcarding may be demonstrated: Through use of wildcarding, account access attributes may be set to limit the accessibility (READ, WRITE, NOLIST, etc.) of individual files to account groups. Assume the following 'MODIFY' command is issued for a specific file:

```
!MOD fid TO (ACC=(CD?,READ),ACC=(CD?00,WNEW),ACC=(?,NOLIST))
```

All accounts with the 'CD' prefix would have 'READ' access, those accounts with an 'CD' prefix and a '00' suffix would have 'WNEW' access (the ability to write new records), and all other accounts would not be able to 'LIST' the file, except the account within which the file resides.

This technique might be particularly useful when utilized within a software factory environment, where programmers are given separate accounts for each project to which they're assigned. Similarly, in an academic environment this might apply where students are given an account for each computer-utilizing course in which they're enrolled. This technique can be applied at the account level via PIG.

Scheme 3 - Account Grouping

Another grouping scheme uses an eight-letter alphabetic designator for account:

```
ZZZALPHA
```

The first three letters identify the individual user; the next five letters identify the user group. Everyone in the "ALPHA" group may be given access to all other ALPHA accounts (?ALPHA). Packset allocation could be by user groups.

In this scheme, since the account designator has no information concerning project or department number, this is added as a prefix to the name:

```
dddname
```

where ddd is the project, and NAME identifies the user.

Instead of project, ddd may designate an individual box number. For example, at a college site, student printouts are routed via a "box number". To accomplish this, the box numbers are made part of the logon:

```
ZZZALPHA,100JONES
```

where 100 is the box number.

Account Wildcarding

Wildcarding as applied to accounts gives the system manager the ability to "wildcard" ACCOUNT specifications in access lists. This allows simplified lists, enabling an entire group of accounts to be named by one keyword. Thus, ?HOST will match AHOST, AAHOST, BBBHOST, etc.

It is to the advantage of the system manager if he can divide his world into groups so that all the members of a group require the same type of access to the same type of files. It is more desirable to have all members of the administrative programming staff have ADMN in their account than it is to have each with his own name as the account. It is possible to have ?ADMN or ADMN? in an access list. It is not practical to list 75 accounts like TOM, DICK, HARRY, MARY, SALLY, SUE, etc.

Wildcarding comes in handy in SUPER also. User privileges maintained by SUPER can be changed for a group more easily if the group can be addressed by wildcarded account.

Grouping and Packset Allocation

Other reasons for grouping involve packset allocations. File management accounts are placed on packsets. These sets can be spread across physical devices or kept on one device. A packset may be the only packset on the device, or there may be others on the same physical device. User groups should be planned with packsets in mind. The CP-6 file backup processor, EFT, is optimized for backup and restore by entire packset. If users are logically grouped, file maintenance can take place with service interruption on a group by group basis. rather than the whole user base. A hardware failure on one disk drive need not prevent all groups from accessing their own packsets. Instead, if multiple removable packsets are available, a choice can be made as to which groups can continue to access their packsets, and which groups cannot.

Grouping and PIG/SUPER

Grouping will also make the Project Administration features of SUPER and PIG more readily available. Project Administration is a way of delegating the creation of file and user accounts to a less privileged user (e.g., group leader). This user can only create accounts within the subset of privileges and of file space passed down to him (e.g., by the system administrator). This is usually not easy to do unless the user base is already partitioned into logical groups.

Planning File Management Accounts

After a scheme has been established to identify and categorize users, the next step is to map out file management accounts into packsets. At this point it will be necessary to determine whether to define a packset per project, multiple packsets per project, or multiple projects per packset. It is necessary to:

1. Determine total storage requirements.

2. Decide on degree of over-allocation, if appropriate.

3. Identify access restrictions applied to other projects.

4. Decide on degree of backup required.

The next step is to allocate the packsets to physical storage, bearing in mind that a physical device failure will affect users of all packsets allocated to that device. From the point of view of performance, putting heavy activity packsets on the same physical device will lead to disk arm contention. Backup requirements enter into these decisions. In allocating DUALs, the system manager should bear in mind that they are most useful on removable devices.

Modules in this handbook will provide specific information as to how these pre-installation decisions are implemented in the CP-6 system.

For example, assume that a system includes three disk spindles. The following figure shows how file management accounts can be grouped as three separate logical packsets: DP#SYS, DP#ALPHA, DP#BETA. In this sample, the SYS packset is split across two physical packs; the BETA packset occupies a portion of one pack; the ALPHA packset is also split across two physical packs. For convenience, account designations could contain common identifiers; for instance, file management accounts on the ALPHA packset could end in ALPHA to permit use of the wildcard feature (i.e., referring to accounts on the ALPHA packset as ?ALPHA).

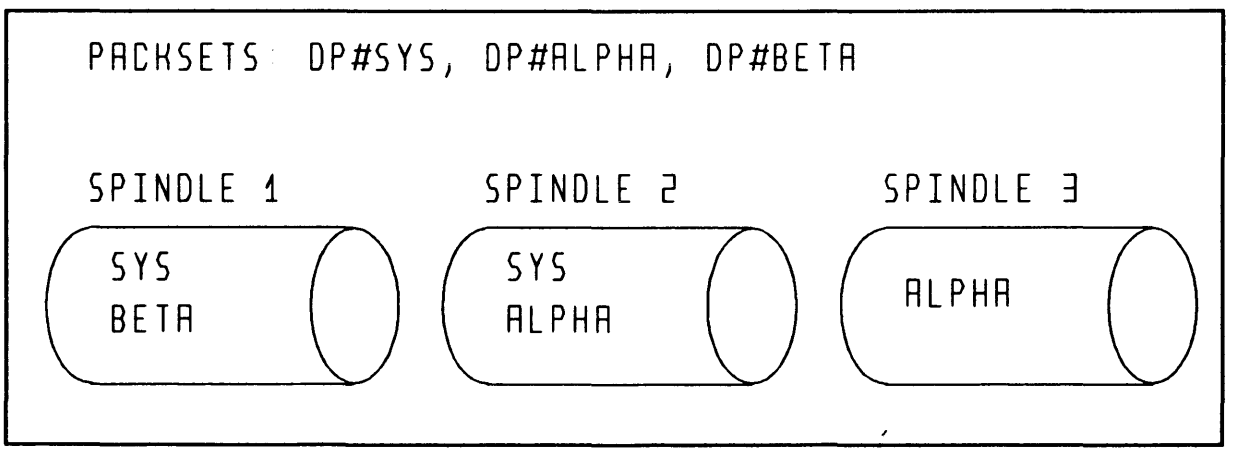


Figure 1. Sample Packset Grouping

Module 2-1

Security

This module describes the CP-6 system and file security features made available to CP-6 customers. Security ensures that only authorized users can access computer-stored information. It protects data against accidental or deliberate unauthorized disclosure, as well as unintentional or malicious modification and destruction. The security features described in this section include:

- Logon security, which controls user access to the system.
- File security, which provides a considerable array of security techniques including account protection, access controls, passwords and data encryption.
- Tape security, which features three levels of ANS tape volume protection: full, semi-protected, and unprotected.
- Privileges, which allow the system manager to control special access to powerful or disclosive system features.
- Processor privileges, which allow the system manager to assign sensitive privileges to processors rather than individual users.
- The security logging facility, which allows the system manager to create an audit trail of such system activities as file accesses, privilege use, privileged monitor service use and logons.

The CP-6 system has built-in protections against deliberate attempts to violate security. In the CP-6 system, no one can read a disk area before writing to it. If only part of a granule contains user data, only that part of the data that belongs to the user is made available. These system features prevent browsing through disk granules in a search for sensitive data.

Data security is a joint responsibility. The CP-6 system must — and does — provide the tools to secure data from unauthorized access; but, the system manager and user must secure the information by using those tools. Physical security is the responsibility of each installation. Each site must ensure that dial-up phone lines are protected, offices with hardwired lines are locked and that access to data processing rooms is restricted.

Logon Security

The first kind of security in a CP-6 system is LOGON authorization. A user must give a valid name, account and password to gain access to the system. The LOGON authorization is defined by the system manager using the SUPER processor. Although the password is optional when defining LOGON accounts, supplying a password for every account will increase overall system security. Users may change their own password at any time via a simple terminal command. Requiring the use of passwords and establishing policies to change them often is a strongly recommended practice, as very little privilege is needed to see other name/account combinations.

The LOGON authorizations are kept in two files named :USERS and :HLP in the :SYS account. These files are protected by the file access controls described later in this module. Only the system processors IBEX, LOGON, NETCON, PIGETTE, PRESCAN, SLUG, TPA and TPCP may access these files.

The passwords supplied to SUPER for LOGON accounts are irreversibly scrambled when they are stored with the logon information. Passwords given at LOGON time are scrambled in the same way, so it is the scrambled version of the passwords that are compared for equality during the LOGON process. If the user forgets the password, a new one must be assigned with SUPER, as even SUPER cannot reverse the scrambling process.

The user must specify the correct password for the account each time the user logs on to the system. On full duplex lines, echoing is turned off during the logon process to reduce the possibility of accidental disclosure. If the installation maintains a file of SUPER commands for the purpose of recreating accounts, this file must be protected carefully.

Logon information is retained in each user's :USERS record including the UTS of the last good logon, the number of bad logon attempts which have occurred during the current session, and the number of bad logon attempts which occurred between the last logoff and the most recent logon. Since the recording of bad logon counts is contingent upon the existence of a :USERS record, only bad logons which fail the password check will cause the count to be incremented. The most recent good logon time will always be recorded for timesharing logons. Batch logons will only be recorded if they are submitted from a device (e.g., a card reader) or if the logon specified on the !JOB card does not match the spawing user's logon.

When a CP-6 system is first booted and if no :USERS file exists, SUPER automatically builds three logon accounts, :SYS,LJS and :FED,SUPPORT and :SYSTAC,LADC. :SYS has all privileges and :FED has enough privileges to allow test and diagnostic programs to be run. The accounts are not passworded. It is strongly recommended that these accounts be deleted (:SYS) or passworded (:SYSTAC, :FED) as part of standard tape-boot procedure, probably as a part of the SUPER user authorization job.

In addition to logon security established via SUPER, logons can be monitored as a security measure. Logons and attempted logons can be logged in the Security Log by setting a system parameter via the CONTROL processor (see Security Log Facility, below).

File Security

File security ensures that only authorized users can access computer-stored information. It protects data against accidental or deliberate unauthorized disclosure, as well as unintentional or malicious modification and destruction.

Storage medium security features differ depending on whether the medium is disk or tape files. If the medium is disk files, the packset owner controls who can create accounts on the packset and establishes default values for account permissions. If the medium is tape files, ANS label protection features are available. The level of tape volume protection used is controlled by an installation option. The system manager can specify that tape volumes are to be fully protected, semi-protected or unprotected. The volume owner may, at volume creation time, specify protection for labeled tape.

The following table summarizes CP-6 input/output and file management security features:

Table 1. Security Features

Media	Owner	Items Controlled	Defaults Controlled
Disk	Pack set owner (identified by name and account)	Determines whether automatic or explicit permission is necessary to create accounts on the pack set.	Account permissions
Tape	System manager	Determines level of ANS label protection.	

Table 1. Security Features (cont.)

Media	Owner	Items Controlled	Defaults Controlled
Account	Account owner (identified by name and account)	Grants explicit permissions for directory access and file creation. Determines whether files in the account will be owned by all users of the account or whether ownership will be controlled by name and account.	File permissions
File	File owner (identified by account or name and account)	Grants explicit file permissions: i.e., determines which accounts and/or processors can read, update, list, append and delete files, delete records, and change file attributes.	

Note that:

- The owner of an account and the owner of the file need not be the same users. While typically the account owner will also own all files in the account, particular application requirements may dictate other usage. For example, a class teacher may set up an account whose files are owned by the members of the class.

- The levels of security have a hierarchical relationship. A failure at one level will prevent progression to the next level. A user who wishes to access a file must first pass security at the account level, then at the individual file level. The user then gains access to the file's data, where the ability to read the data will depend on security at the data level.

- At each level, the user is granted access to more information. Only the user(s) who can pass account level security checks can know the names and the types of the files in the account. Only the user(s) who can pass security at the file level can obtain other catalog information about the file. If there is security at the data level, the file data will be meaningful only if the user can supply the required security information.

An understanding of file security features assumes a clear understanding of the difference between logon accounts and file accounts. Whereas the logon account is the account that the user supplies to gain entry to the system, a file account is the account name under which a file is cataloged. A file account need not have a corresponding logon account, and a logon account need not have a corresponding file account.

File security features five levels of protection arranged in a hierarchical manner. The first level of protection is to isolate secure files on a pack set which is mounted for the exclusive use of a single user so that other users are unable to access the files. The four other file security levels are:

1. Account level security, where security features are supplied to all files within the account.

2. File level security, where individual access protections are applied by file.

3. Special access level security, where the file access is permitted only via a specific program.

4. Password/data level security, where file passwords and data encryption are applied.

These four file security levels are the subject of the remainder of this section of the module.

Note that:

- Files are controlled at both the account level, where defaults for all files in the account are established by the system manager, and at the file level, where defaults may be overridden or extended by the creator of a file. At both levels, access control lists may be established to specify who may know that a file exists, who may read it, who may write new records and/or modify existing ones, who may delete records, who may delete or rename the file, and who may change file attributes. In addition, at the file level, the file's creator may specify the type of access (read, write, execute) as well as an access vehicle (a named run-unit) through which the file must be accessed.

- All files may have passwords and all files may be encrypted. Because some CP-6 processors do not recognize encrypted files, use of the latter capability for protecting files should be evaluated on the basis of the processing to be performed on those files.

- File access can be monitored through the Security Log by setting system parameters via the CONTROL processor (see "Security Log Facility" below).

Account Level Protection

Access control on file accounts are maintained by the Packset Initializer processor (PIG), which can be run by the system manager or a pack set manager designated by the system manager. Four classes of security can be specified:

1. Mode of the account. If the mode is NOT PROTECTED, the creator and any user with a logon account matching the file account can access any of the files in the account regardless of access controls on the file except password. If the mode is PROTECTED, all users except the creator (name and account) are subject to the access controls of the file.

2. Directory access. A list of logon accounts that may access the directory may be specified. If permission to access the directory is not given, the user is unaware that the directory and the files contained therein exist, and access is denied regardless of individual file access controls. This permission is always given for users whose logon account matches the file account.

3. File creation. A list of logon accounts that may create new files in this file account may be specified. This permission is always given for users whose logon account is the same as the file account.

4. Access defaults. A list of logon accounts and the default access controls each is to have to the files in this account may be specified. (See the Table "File Permissions" for a list of the access controls.) Access controls specified on individual files override these default access controls unless the account has the MERGEACCESS attribute, which causes them to be concatenated with the defaults.

All account controls are authorized and changed through commands to the Packset Initializer Processor. The account owner defines the qualified accounts and the permissions granted. Account permissions can be granted either globally (assigned to all or denied to all accounts) or they can be granted to specific accounts.

Access Controls

The individual user specifies the third level of security — access controls on individual files, usually at the time a file is created. These controls replace or enhance the default from the account level, but cannot override access to the directory. Note that each permission is separate: therefore, a user who is permitted to delete a record from a file may not be able to read it. The table "File Permissions" lists the file access controls a user can establish or modify.

Table 2. File Permissions

Permission	Access Authorized	Comments
AU	Permission to be the administrative user of the comgroup.	Allows a user to call M$OPEN specifying AU=YES. (Only one user at a time may act as AU of the comgroup; the AU may change the comgroup via the following service calls to the monitor: M$CGCTL, M$ACTIVATE, and M$DEACTIVATE; the user may obtain information via the service call M$CGINFO. A user who uses this monitor service shares the privilege with any user with an AURD access to the file.)
AURD	Permission to invoke restricted monitor services to a comgroup that examine but do not change the comgroup.	Allows a user who is not the administrative user to invoke any monitor service that is restricted to the administrative user as long as no attempt is made to change the comgroup.
DELF	Delete file, change file name, change file password, change file access control for both vehicle and account.	Note that the REATTR permission is used to control who can change all other file attributes.
DELR	Delete records.	Enables the M$DELREC monitor service.
NOLIST	Suppress inclusion of file directory information from catalog listings.	If assigned, any account with this attribute is not permitted to discover the existence of the file.
READ	Read, or position and read, a file.	Enables the M$PFIL, M$PRECORD, M$READ and M$REW monitor services.
REATTR	Change file attributes, except file name, file password, and access controls.	Note that the DELF permission controls who can change a file name, file password, and/or access controls.

Table 2. File Permissions (cont.)

Permission	Access Authorized	Comments
TCTL	Permission to issue terminal control monitor services to a comgroup.	Enables the monitor services described in Section 5 of the Monitor Services Reference Manual.
UPDATE	Replace existing records.	Enables the following monitor services: M$WRITE for keyed, indexed and IREL files; M$WRITE with REWRITE for consecutive files. Note that another permission, WNEW, controls who can add new records.
VEH	File permissions depend on the accessing vehicle.	See the subsection "Access Vehicles".
WNEW	Add new records to the file.	For keyed, indexed and IREL organizations, enables random insertion of new records; for consecutive files enables addition of new records at the end of the file. Note that a different permission, UPDATE, controls who can alter an existing record.

When a user who is not an owner attempts to access an existing file, a check is made to see if there is a list of permissions for this file associated with the user's account. If a list is found, the list determines what the user can do with the file.

If a user has account permission to access an existing file, but there is no file permission list associated with the account, the user defaults to the account default file permission. The system manager or account owner can change the account file permission default through the PIG processor.

Access Vehicles

One of the permissions that can be assigned by a file owner, vehicle permission, permits alternate file access to a specific set of processors. The file owner determines which processors are to have access to the file, what file permissions they are to have, and which accounts are to have access through the processor(s). Each processor is given only the file permissions authorized in the vehicle access list. If no file permissions are included for the processor in the vehicle access list, no permission is authorized. If access is attempted by a processor not in the list, the remaining account list permissions prevail.

Passwords and Encryption

Passwords and data encryption are the two final file security features. They are controlled by users.

The owner of a file can assign a password to a file or change an existing password. If a password is assigned to a file, permission to open the file is denied any user (including the creating user) who cannot supply the correct password. To further enhance security, only a scrambled version of the password is stored with the file.

Anyone who can write records in a disk file can request that data in the file be encrypted. Encryption is available for all file organizations except indexed files. Encryption may be specified separately for each record of a file.

Data can be encrypted and decrypted through both the EDIT and PCL processors. Data is encrypted by replacing each character with a computed substitute value. The characters are generated from an algorithm which uses as its base a starting number supplied by the user called a seed. Because the seed can be specified on each read and write, it can be different for each record. The seed is not recorded anywhere in the system, which secures it from unauthorized detection. However, this added measure of security places a responsibility on the user to remember the seed.

Note that some CP-6 processors do not recognize encrypted files. Use of encryption for protecting files must be evaluated on the basis of the processing to be performed on the file.

Wildcarding

Account references in access control lists can be abbreviated through wildcarding, i.e., inserting one or more wildcard characters (the question mark character) in an account name. Each question mark replaces any number of characters and specifies that the characters it replaces can be matched with any character; that is, that the characters are not to be included in the match check between the access control list for the file and the name of the account requesting access.

The effect of wildcarding is to allow the system manager to specify a range of accounts in access control lists. For example, the account specification

XXA?

specifies that all accounts whose first three characters are XXA are to be selected. In order to make use of wildcarding (and to keep access control lists short), it is recommended that account names be created so that access can be given or denied to groups of users by the use of wildcard characters. Remember, however, that to take advantage of wildcarding, all accounts must use the same structure. An access control account 'GEO?' intended to give access to geology accounts of the form 'GEO3712' will also give access to the account 'GEORGE'.

Refer to the section "Files, Devices and Comgroups" in the CP-6 Programmer Reference Manual (CE40) for more details on Wildcarding.

Tape Security

The CP-6 file management system offers labeled tape protection both at the volume and file levels. (Free and managed (device, unlabeled) tapes have no standard labels and therefore cannot be protected.) Volume protection is performed for the volume itself to determine if the volume can be written at all, and for the volume owner to determine if the volume can be read or written by a specific user.

Protecting the volume itself ensures that a labeled tape cannot be overwritten until a specified expiration date and that a labeled tape cannot be 'changed' into an unlabeled tape or into a labeled tape with a different serial number. The strictness of enforcement is specified by the system manager through TIGR or CONTROL commands, and falls into one of three protection modes:

Mode	Description
Fully protected	Unexpired labeled tape cannot be overwritten; expired labeled tape can be overwritten.
Semi-protected	Unexpired labeled tape can only be overwritten after an OVER keyin; expired labeled tape can be overwritten.
Unprotected	Both unexpired and expired labeled tape can be overwritten.

User read and write protection ensures that users other than the volume owner (account of user creating the volume) have no access, only read access, or any access to a particular volume. These access controls are established by the user at volume creation time.

Access to labeled tape files with a CP-6 specific organization is controlled by the same access control features described above for disk files, except that there are no account defaults. These access controls are established by the user at file creation time.

Privilege Security Features

Some features of the system are so powerful that their use must be controlled and restricted. Such special access is granted through privileges. Privileges can be granted to individual users or programs. In addition, system security is maintained by requiring system processors to have processor privileges in order to run. Such processors are called privileged processors. A third level of privilege security is created by requiring special user privileges, called user processor privileges, to use a privileged processor. This section describes these three aspects of privilege security.

User Privileges

Not all features of the system need or should be granted to all users. Some features are intended to be used only by those users who need to monitor the system or diagnose or fix it. These users need special access, which is granted through privileges. The system manager assigns special privileges to these users. Once these rights are granted, the user enables them selectively for any task that requires privileged capabilities.

The SUPER processor is used to grant privileges to users. Modules 4-1, 4-2, and 4-3 describe how to use SUPER to assign privileges. (Section 7 or the System Support Reference Manual contains a complete list of privileges that can be assigned.)

Users granted the following powerful privileges can bypass most — if not all — of the system's security. These privileges must not be authorized lightly.

Privilege	Description
CFEP,MFEP	Examine and modify the front end processor.
EXMM	Store into any page of memory.
EXPM	Start/stop performance monitor.
FMDIAG	Read or write any disk granule.
FMSEC	Bypass all file and tape management security checks.
GPP	Get physical memory pages.
IOQ,IOQW	Call on I/O devices directly.
JIT	Allow modification of the JIT (including privileges).
MSYS	Allows use of certain features of GOOSE and the use of M$RUE.
SYSCON	Partition hardware devices.
TND	Use test and diagnostic services.

The following privileges do not allow the user to modify the system, but allow the user access to information that usually should not be disclosed or that affect system performance in such a way that authorization of these privileges should be granted sparingly.

Privilege	Description
ASAVE	Automatically saves user's image if terminal connection is lost.
DISPJOB	Display status of the jobs of other users.
FMREAD	Bypass all file and tape management security checks for READ only access.
MAXM	Allocate memory beyond authorized user limit.
PM	Display performance statistics.
SPCLMM	Examine other users memory.
SYSLOG	Write in error log.

Processors may be created via LINK with certain privileges. To insure that these privileges can be effected, run units linked with privileges must be run from :SYS. When the processor is fetched from :SYS, those privileges are in effect regardless of whether or not the calling user has those privileges. Therefore, it is critical that the system manager know the processors that get moved into :SYS, including their access controls and privileges. Some processors check the user's authorization and use M$PPRIV to set or M$RPRIV to reset the privileges as needed. The users of these processors normally do not, themselves, need to set privileges. For example, EFT is LINKed with both powerful and disclosive privileges, but it checks the user authorization to determine whether the caller can do SAVEs or only MOVEs. Note, also, that to prevent malicious misuse of the privileges, the system manager must never grant write access to any file in :SYS.

Temporary Privilege Restriction

The system manager can restrict privileges usually available to users through the use of CONTROL processor SYSTEM parameters.

The parameter PRIVMASK determines a set of privileges which may normally be available to users but should not be set active for a while. PRIVMASK may be set to ALL, NONE, or a list of individual privileges. That is, if PRIVMASK=(FMSEC,EXMM), then no user will be allowed to set either privilege active until the parameter is altered to allow the privileges to be set. Note that:

- Only a user's active privileges are affected.
- PRC privileges (those which come with processors run from :SYS) will not be affected; that is, PRC privileges that would usually be set active are still allowed to be set active regardless of this parameter.

If users are on the system when this parameter is set, each user will retain currently active privileges until he or she again requests that a privilege be set. An X account tool, PRIVWARN may be used to globaly reset specified active user privileges if this is desired.

The parameter STEPPRIVMASK determines a set of active user privileges to be reset at job step termination. Like PRIVMASK, STEPPRIVMASK may be set to ALL, NONE, or a list of individual privileges. The use of this parameter does not restrict any user from again setting any authorized privileges. It is intended only to keep a specified set of privileges from being carried through multiple job steps unless they are explicitly set at each step.

Privileged Processors and User Processor Privileges

There are cases where privileges are so powerful or disclosive of information that it is preferable to grant the privileges to a program rather than to individuals.

The system manager can use SUPER to authorize users to utilize specified privileged processors without giving the required privileges to the user. Section 7 of the CP-6 System Support Reference Manual gives more details on processor privileges (PPRIVILEGES). This technique allows the system manager to change control parameters and protect highly dangerous privileges from direct user availability. The privileges that can be granted in this way are:

Privilege	Description
CNTRLC	Display and change system parameters (CONTROL).
CNTRLD	Display system parameters (CONTROL).
EFT	Run EFT.
EL	Run the Error Log Analyzer (ELAN).
LABEL	Write labels on tapes.
NETCON	Control the FEP.
PIGC	Change pack set status.
PIGD	Display pack set status.
PIGETTE	Create bootstrap diskettes for remote FEPs.
RATES	Run RATES.
REPLAY	Run REPLAY.
SPIDERC	Change shared processor status.
SPIDERD	Display shared processor status.
SUPER	Run SUPER.
SUPERAUTH	Authorize users.
SUPERD	List SUPER data.
SUPERFORM	Create and change forms.
SUPERWSN	Authorize and modify workstations.

(cont.)	
Privilege	Description
SYSCON	Control/display availability of hardware components
VOLINIT	Use VOLINIT.

Security Log Facility

The Security Log is a collection of files used to audit the use of certain sensitive system facilities. The name of the Security Log is :SECLOGyymmdd, where 'yymmdd' is the ANS format for date. These daily files are maintained in the :SYS account. They are indexed files keyed by primary key only. The fields included in the key are referenced in the EL$HDR macro described later in this section.

The following five types of records can be included in the Security Log files:

- System access (i.e., logon) records
- System exit (both logoff and recovery) records
- Monitor service records
- Privilege request records
- File access records

The system manager determines the types of records included and protects the Security Log through CONTROL processor SYSTEM parameters.

The system manager uses the following CONTROL processor SYSTEM parameters to tune the Security Log:

PROTECTSECLOG
LOGFILEGRANT and LOGFILEDENY
PRIVCHNGMASK and LOGPRIVCHNG
MONSERTBL and LOGMONSER
LOGSYSACCESS and LOGSYSEXIT

Use of these parameters to tune the Security Log is summarized here. Section 2, "CONTROL:HOST System Management" of the CP-6 System Support Reference Manual details the parameters.

Protecting the Security Log

The CONTROL processor SYSTEM parameter PROTECTSECLOG may be used to specify the number of days a security log file will be specially protected by file management since its creation. During this time frame, only the security logging processor will be allowed to perform any operation other than to read the file. Thereafter the files are maintained using standard file management capabilities. Since these files reside in :SYS, it is expected that :SYS will have adequate default file access controls to restrict perusal by users who do not have special privileges.

Logging Access or Attempted Access to Files

The CONTROL processor SYSTEM parameter LOGFILEGRANT determines whether certain granted file accesses will be logged for any file on the system. Since the decision to log granted file access is only available at the system level, these accesses will be logged only if a user has sufficient active privileges to cause normal security checking to be bypassed. LOGFILEGRANT may be set to YES or NO.

The parameter LOGFILEDENY determines whether denied accesses to any file on the system will be logged. LOGFILEDENY may be set to either YES or NO.

Logging User Privilege Changes

It is possible to cause privilege setting activities to be logged in the Security Log. The CONTROL processor SYSTEM parameter PRIVCHNGMASK is used to specify a set of privileges the use of which is to be logged. LOGPRIVCHNG controls how this set of privileges is to be interpreted for logging purposes. If set to NONE, PRIVCHNGMASK will be ignored and no logging will be done. Otherwise, it may be set to log only granted requests, only denied requests, or both.

Special Monitor Service Logging

Certain monitor services permit the usual security mechanisms to be bypassed. Therefore, the system manager may wish to have the use and/or attempted use of any of these monitor services logged in the Security Log. The CONTROL processor SYSTEM parameter MONSERTBL may be used to specify a set of special monitor services considered interesting enough to be logged. A list of the monitor services that may be specified can be found in Section 2 of this manual. LOGMONSER controls how these monitor services are to be interpreted for logging purposes. If set to NONE, MONSERTBL will be ignored and no logging will be done. Otherwise, LOGMONSER may be set to log only granted usage, only denied attempts, or both.

Logging System Access and Exit

The CONTROL processor SYSTEM parameter LOGSYSACCESS allows the system manager to determine which logons, if any, are to be logged. LOGSYSACCESS may be set to cause the following categories of logons to be logged: (1) none; (2) only logons which fail the password check; (3) logons which either fail the password check or look reasonable but do not actually exist; (4) logons which fail for any reason at all, including bad format; or (5) all logon attempts.

The parameter LOGSYSEXIT allows the system manager to specify whether system exits, either logoffs or exits due to recovery, are to be logged. It may be set to ALL or NONE.

Operational Security

A full consideration of operational security must include a discussion of the physical plant, the way in which computer center operator's are trained and other subjects that are beyond the scope of a Honeywell publication. Some aspects of operational security relate directly to appropriate use of the CP-6 system.

For purposes of this discussion, operational security is divided into the following categories:

- Physical security
- Security planning for data center operations

Physical Security

Many books have been written on the subject of physical security in the data processing environment. In those documents, the reader will be confronted with the need to address such physical security considerations as locking doors, shredding sensitive listings before disposing of them, disposition of carbons, and establishing secured areas.

Some tape security features are linked closely with physical security. In the earlier part of this section on "Tape Security", the reader was introduced to the three kinds of tape protection modes available via CP-6 file management. From a physical security aspect, the system manager will need to weigh advantages and disadvantages of two of those modes: namely, the fully protected and semi-protected modes, and will normally decide in favor of semi-protected mode. Using the fully protected mode provides full ANS protection on tapes which guarantees that a tape cannot be overwritten. But, fully protected tapes require operator intervention to mount, and such tapes cannot be extended. Their use may prove inconvenient and inefficient.

The system manager will also want to give deliberate consideration to establishing labeling techniques to guarantee that cross mounting of tapes and private packs cannot occur, thereby assuring that only appropriate users can mount tapes and private packs.

Security Planning for Data Center Operations

The system manager will want to establish policies that are explained to the computer center operators as part of their training to ensure that:

- Listing distribution is controlled
- Writing to tapes is controlled

In addition, operations planning should include some deliberate decisions about how operator consoles are to be defined.

Controlling Listing Distribution

Prior to operator training, a system needs to be established for controlling which listings can leave the data center and where they can go. It is recommended that in planning this security, the system manager consider use of the SUPER processor BANNERTEXT feature (described in the section "Device-Form Definition" in the CP-6 System Support Reference Manual (CE41)). Through this SUPER option, destination fields can be defined for inclusion on printout banners to define clearly who is to receive output. The system manager controls whether those fields can be changed or not. Deliberate planning of account naming conventions will help in the planning of this type of security.

Controlling Tape Writes

The system manager will want to develop and enforce policies on the use and removal of write rings to protect tape data from accidental overwriting. As part of physical security planning, the system manager can establish policies to ensure that write rings are removed from tape reels. It is especially important that PO (boot) tapes be protected from accidental destruction of data. One suggestion is to limit the number of write rings available. A short supply of write rings will help guarantee their removal from tape reels.

Defining Operator Consoles

At least one IOM-connected system console will probably need to be defined in any significant data center so that dramatic system intervention (i.e., ZAPs and DIEs) can be performed. As part of physical security planning, the system manager will want to make sure that system consoles are not made generally available, and will want to make a deliberate decision about whether and how to establish associated workstations.

System consoles and their associated workstations should only be established where a need is clearly defined for global control. In general, the system manager will want to evaluate the actual specific requirements the operator will have at the given location. In particular, the system manager will want to be deliberate in use of the DEVICE and UNPRIV ADMIN console attributes to control who can lock and unlock devices and who has administrative privileges.

Deliberate planning of workstation names and device names will simplify administrative control of the workstations. All workstations controlled by a console can be wildcarded with a single trailing ? (see "Wildcarding" above). The following considerations may be useful in planning operator console security:

- It is advisable to keep the console history log active to maintain a record of all privileged transactions and all significant keyins from a console. Since a system console can turn the console history log off, one security measure the system manager can take is to ensure that the GOOSE_EGG file contains the necessary commands to ensure that the console history log is turned on periodically so that no more than one hour's transactions will ever be lost. (See the section "GOOSE: Automatic Operations Control" in the CP-6 Systems Support Reference Manual (CE41)).

- The console history log is maintained in the :SYSTAC account. As part of security planning, the system manager may wish to back up the console history log to a more secure account to prevent tampering.

- An operator's console is intended as an operational tool. Such administrative functions as changing system configuration, changing rate tables, authorizing new users and creating new accounts require privileges not associated with the console, but, rather with the user. If it is desirable to perform privileged user functions from an operator's console then a timesharing logon account must be defined for the user of the console. Such users must be counselled as to discreet and suitable use of the logon account, and advised to change the password for the account periodically.

Program Security

The segmented structure of Honeywell DPS hardware guarantees the protection of users in the CP-6 system so that normally:

- a user cannot see another user

- a user cannot see monitor procedure or data.

This protection is enforced in both the DPS hardware and the CP-6 system architecture.

In the CP-6 system architecture programs are protected by framing data and using traps. Techniques have been established such that attempts to transfer data between domains requires that users must frame data within vectors and a hardware instruction, a Privileged Master Mode Entry (PMME) CLIMB, is performed so that the monitor is entered. All input/output statements are performed by the monitor in response to PMME instructions. Attempts to access data in any other way will cause the hardware to fault and CP-6 fault handling processing will take control. This procedure ensures that no user will cause damage to the system or to another user.

The CP-6 system includes ways for one user to access another user's data through special privileges which can be assigned to users via the SUPER processor. The SPCLMM privilege allows a user to perform a PMME to gain access to normally invisible areas of procedure and data, including the monitor. The EXMM privilege allows a user to write areas of memory otherwise not available to a user. The IOQ and IOQW privileges allow a user to circumvent the requirement to perform input/output statements via PMMEs.

Module 3-1

Device Configuration

After initial booting of a CP-6 system, using the defaults supplied on the PO tape, a minimal hardware configuration is operational. To make the complete hardware configuration operational, it is necessary to revise the TIGR deck on the PO tape to reflect the complete hardware configuration; the system can then be rebooted to make the complete hardware system available for use. For changes or additions to the hardware configuration, the same procedure is followed to make new or changed devices known to the CP-6 system.

This module describes how to define a basic hardware configuration and set critical software parameters using TIGR commands. It also describes how to create a bootable system tape with which to reboot the system. Sample jobs available with the system facilitate creating a bootable system tape, as illustrated.

Defining the Hardware Configuration via TIGR

The following example explains how to modify the TIGR commands supplied on the PO tape. The TIGR commands on the PO tape define a minimal hardware configuration via the AUTOCONFIG command. The system manager should substitute specific definitions of the actual configuration by modifying the file TIGR_REL.SUPPORT which is then used to create a new PO tape as discussed later.

The following example modifies the TIGR_REL.SUPPORT file to reflect a standard local configuration: two CPUs, one IO Multiplexer (IOM), a system console, a disk micro-programmed controller (MPC) and several disk drives (one drive is to be installed later), a tape micro-programmed controller, and a unit record controller with a line printer and card reader. Four FEPs are defined in the TIGR_REL file; only one FEP exists in the sample system.

```
!EDIT TIGR_REL.SUPPORT
EDIT C00 HERE
*TY 1-2
   1.000 !TIGR "L66 C00 RELEASE TIGR DECK
   2.000 "TIGR REL"

                The Honeywell release TIGR deck comes on the "tools" tape as the
                file TIGR_REL.SUPPORT.  The system manager has requested EDIT to
                display the first two lines of this file.

*IN1
   1.000 !TIGR "MY OWN TIGR DECK
*DE 2
*  1 records deleted

                The system manager has renamed the TIGR deck from "L66 C00
                RELEASE" to "MY OWN", the name of the new deck to be built, and
                has also deleted line 2, "TIGR REL".
```

(cont. next page)

```
*ty 3
   3.000 AUTOCONFIG

                  AUTOCONFIG, line 3.000, the only command in the TIGR deck to
                  represent the default local device configuration.  (In addition
                  four FEPs, on channels 33-36, are represented in the TIGR_REL
                  file.)

*IN 3
   3.000 CPU PORT#=2
*IP,.01
   3.010 CPU PORT#=3
   3.020 IOM PORT#=0
   3.030 CONSOLE NAME=SC01,IOM#=1,CHAN=30

                  The system manager informs TIGR that two CPU's are connected to
                  ports 2 and 3 of the eight-port system control unit, one IOM is
                  connected to port 0.  One console which must be named SC01 is
                  hooked to IOM #1 on channel 30.  It is recommended that the
                  console always be configured to channel 30.

*IN 3.040,.01
   3.040 DISK  ;
   3.050       MPC,MPCNAME=DC01,MODEL=MSP0600 ;
   3.060            LA  ;
   3.070                 IOM#=0,CHAN=8-11      ;
   3.080       DEV  ;
   3.090            NAME=DP01,MODEL=(MSU0451,MSF0007),DEV#=1,SYSTEM ;
   3.100            NAME=DP02,MODEL=(MSU0451,MSF0007),DEV#=2,        ;
   3.110            NAME=DP03,MODEL=(MSU0451,MSF0007),DEV#=3,        ;
   3.120            NAME=DP23,MODEL=(MSU0501,MSF0007),DEV#=23        ;
   3.130            NAME=DP25,MODEL=(MSU0501,MSF0007),DEV#=25,STATUS=DOWN

                  These lines, starting with 3.040, are the disk command.  Each disk
                  must be hooked to the system through an MPC (micro-programmed
                  peripheral controller) which in turn hooks to the IOM.  The MPC
                  must be assigned the disk controller name DC01.  MSP0600, the disk
                  model name, indicates the firmware (for "disk controller")
                  necessary to load into the MPC.

                  Firmware is written, supported, and supplied by Honeywell Field
                  Engineering.  (Firmware supports the proper interaction between
                  the MPC and the host software.)  The link adaptor (line 3.060),
                  used to hook an MPC to an IOM, is connected to IOM number 0,
                  channels 8 through 11 (line 3.070).  This portion of the command
                  indicates where TIGR is to download the firmware.  The devices are
                  specified next:  the system disk (3.090), two high-speed
                  non-system disks (lines 3.100 and 3.110), a large disk drive,
                  wired at a high address since smaller disk drives will be put in
                  at lower addresses (line 3.120).  To configure the TIGR deck so it
                  appears as if the disk is connected but only partitioned out of
                  use, STATUS=DOWN (line 3.130) is entered, this to assure easy
                  connection of a 501-type disk at a later time.


*IN 3.140,.010
   3.140 TAPE  ;
   3.150       MPC,MPCNAME=TC01,MODEL=MTP0610  ;
   3.160       LA  ;
   3.170            IOM#=0,CHAN=16-17  ;
   3.180       DEV ;
   3.190            NAME=MT01,MODEL=(MTU0630,MTF0636),DEV#=1     ;
   3.200            NAME=MT02,MODEL=(MTU0610,MTF0608),DEV#=2     ;
   3.210            NAME=MT03,MODEL=(MTU0610,MTF0608),DEV#=3

                  Tapes are connected through an MPC, as are disks after the tape
```

(cont. next page)

```
                command (line 3.140) has been issued.  The MPC name must be TC01
                to indicate the tape controller number.  Next, the tape MPC is
                hooked into a link adaptor in IOM number 0 (the only available
                IOM) with channels 16 and 17 (line 3.170).  Line 3.180 through
                3.210 indicate three tape drives and their respective device
                numbers and model numbers connected on the MPC.

*IN 3.220,.010
   3.220 UNIT  ;
   3.230     MPC,MPCNAME=UC01,MODEL=URP0600 ;
   3.240       NAME=LP01,MODEL=(PRU1200,PRB0600),IOM#=0,CHAN=24,OUT,SYMBIONT;
   3.250       NAME=CR01,MODEL=(CRU0501),IOM#=0,CHAN=26,IN,SYMBIONT
                   The MPC name for unit record devices is UC01, and the ordered
                   model is URP0600 (line 3.230).  Because each device goes through
                   its own channel, a link adaptor is not needed with unit record
                   commands.  One printer is specified (line 3.240) with name, model
                   number, connected through channel number 24 and specified as an
                   output symbiont device.  Next (line 3.250) a card reader is
                   specified on channel 26, the default channel.  The card reader is
                   further designated as an input device (IN) as well as a symbiont
                   device (SYMBIONT).
```

Defining Software Parameters via TIGR

The TIGR_REL.SUPPORT file also includes the TIGR command MON which specifies a number of software parameters. The system manager should examine these parameters to determine if they match the site's requirements.

The following example gives explanations and recommendations for a number of the software options available in the MON command. Of particular interest is the information given on the USERS option. Several changes to defaults have been made in this example; the MTDFLT option specifying the default tape density has been added to the option list; an SPROC option for library :SHARED_RPG has been added.

```
 1.000 !TIGR "MY OWN TIGR DECK
 3.000 CPU PORT#=2
 3.010 CPU PORT#=3
 3.020 IOM PORT#=0
 3.030 CONSOLE NAME=SC01,IOM#=1,CHAN=30
 3.040 DISK  ;
 3.050       MPC,MPCNAME=DC01,MODEL=MSP0600 ;
 3.060            LA  ;
 3.070                IOM#=0,CHAN=8-11      ;
 3.080       DEV  ;
 3.090            NAME=DP01,MODEL=(MSU0451,MSF0007),DEV#=1,SYSTEM ;
 3.100            NAME=DP02,MODEL=(MSU0451,MSF0007),DEV#=2,       ;
 3.110            NAME=DP03,MODEL=(MSU0451,MSF0007),DEV#=3,       ;
 3.120            NAME=DP23,MODEL=(MSU0501,MSF0007),DEV#=23       ;
 3.130            NAME=DP25,MODEL=(MSU0501,MSF0007),DEV#=25,STATUS=DOWN
 3.140 TAPE  ;
 3.150          MPC,MPCNAME=TC01,MODEL=MTP0610  ;
 3.160          LA  ;
 3.170               IOM#=0,CHAN=16-17  ;
 3.180          DEV ;
 3.190               NAME=MT01,MODEL=(MTU0630,MTF0636),DEV#=1    ;
```

(cont. next page)

```
 3.200                NAME=MT02,MODEL=(MTU0610,MTF0608),DEV#=2     ;
 3.210                NAME=MT03,MODEL=(MTU0610,MTF0608),DEV#=3
 3.220 UNIT   ;
 3.230           MPC,MPCNAME=UC01,MODEL=URP0600 ;
 3.240           NAME=LP01,MODEL=(PRU1200,PRB0600),IOM#=0,CHAN=24,OUT,SYMBIONT   ;
 3.250           NAME=CR01,MODEL=(CRU0501),IOM#=0,CHAN=26,IN,SYMBIONT
 4.000 FEP NAME=FEP1,IOM#=0,CHAN33
 8.000 MON ;
*
```

The MON command (CONTROL processor) sets certain operating parameters of the CP-6 system. The options that follow illustrate the setting of some of these parameters:

```
 9.000      SITE='JJ''S CP-6',;
```

SITE sets the site identification.

```
10.000      SAL='*** JJ''S CP-6 AT YOUR SERVICE',;
```

SAL sets the salutation or greeting that will welcome on line users when they log on the system. MON can also put BEL characters in the quote string if you wish.

```
11.000      IOCACHE=500,;
```

IOCACHE sets the number of memory pages reserved for IOCACHE to 500.

```
*IP30.1
10.100      MTDFLT=(1600),;
```

MTDFLT specifies the default tape density for device MT.

```
10.110
12.000      STEALPGS=(15,30),;
```

STEALPGS specifies the limit on pages available for stealing from the monitor, that is, removed from the monitor's list of available free pages and used by other system functions.

```
13.000      QUEUE=(90,90),;
```

QUEUE sets IOS — the number of IOS packets to be built (the maximum number of current and pending local I/O operations), and IOQ — the maximum number of current and pending local and remote I/O operations. In this case, both IOS and IOQ are 90.

```
14.000      USERS=75,;
15.000      DOLIST=50,;
16.000      DEVMAX=100,;
18.000      ENQ=(4,10),;
```

The maximum number of users has been set to 75 (line 14). However, as fifteen of these slots are always reserved for system ghosts, the maximum number of online users is 60. DOLIST=50 establishes the number of dolist blocks to be built; these will be used in processing no-wait I/O and other asynchronous operations. DEVMAX establishes the maximum number of devices that can be connected to the system at any one time (including local devices, FEP-connected devices, and terminals); in this case, 100. ENQ sets the range of numbers of pages assigned for ENQ/DEQ table blocks. In this case, the minimum has been set to 4; the maximum to ten.

(cont. next page)

```
19.000      CFU=(1,20),;
21.000      SPSPACE=20,;
22.000      SPAUTOSPACE=50,;

                  CFU sets the range of numbers of pages assigned for the current
                  file usage (CFU) table blocks used by file management.  SPSPACE
                  designates the number (20) of shared processor slots to be
                  reserved for dynamic addition of shared processors.  SPSPACE is
                  also used by the SPROCS option, below.  SPAUTOSPACE designates the
                  number (50) of automatically shared processor slots to be reserved
                  for autosharing of processors.

23.000      SPROC=(LOGON,CP),;
24.000      SPROC=(IBEX,CP),;
25.000      SPROC=(DELTA,DB),;
26.000      SPROC=(IDS,AS),;
27.000      SPROC=(:SHARED_COBOL,LI),;
28.000      SPROC=(:SHARED_SYSTEM,LI),;
29.000      SPROC=(:SHARED_COMMON,LI),;

                  SPROC (lines 23.000 - 29.000) is used to set the status of shared
                  processors; SPROC=(IBEX,CP) indicates Command Processor (CP)
                  status (in this case, for the IBEX processor). AS indicates
                  Alternate Shared Library status (for IDS); DB indicates debugger
                  status (for DELTA); LI indicates Library status.

*   EOF hit after 49.000
*IP50
50.000      SPROC=(:SHARED_RPG,LI)

                  The system manager has inserted an LI (Library) status for RPG.
```

Creating a Bootable PO Tape via DEF

To facilitate creating a new bootable PO tape that matches a site's own requirements, two jobs are provided in the .SUPPORT account: $XDEF_MINI and $XDEF_FULL. The jobs, intended to be run in batch mode, perform these functions:

- $XDEF_MINI re-DEFs the "mini" portion of the PO tape, that is, just the portion of the PO tape that is bootable and contains the TIGR commands and patches for the monitor and system processors and language processors.
- $XDEF_FULL re-DEFs the entire PO tape set, including the bootable portion and the processor reels. A system manager may need to re-DEF the entire PO tape set, if, for example, a maintenance release of a processor has been received.

Using $XDEF_MINI and $XDEF_FULL

Several points must be considered when using the $XDEF_MINI or $XDEF_FULL jobs:

- The density defaults, resource requirements and limits may need to be adjusted to what is appropriate for the system manager's site.

- DEF's MINI_ID command allows the system manager to create a unique bootable tape that will identify itself at boot time (through AARDVARK I/O to the console) and after booting through the use of !WHAT.X. It is recommended that the mini ID be specified as the patch revision level (which is unique over a 10-year span) to aid in remote debugging efforts.

- Certain files in accounts .SUPPORT and .:C00PRC are used by the $XDEF_MINI and $XDEF_FULL jobs, as detailed in the JCL.

In addition, prior to DEFing a full PO tape set, the following steps must be taken:

```
!PIG
CR DP#SYS.:SYSGEN G=10000,WR=me
MADADD DP#SYS.:SYSGEN
END
!PRIV ALL     ...otherwise you miss the IDS products
!PCL
CA LT#CP6P02[#CP6P03...] TO .:SYSGEN
CA newprocessor.otheraccount OVER .:SYSGEN
      .
      .
      .
CA :FEP_?.linkaccount OVER .:SYSGEN
COPY M:MON.:SYS OVER M:MON.:C00PRC
END
```

Sample $XDEF_MINI Job

The following extended example illustrates modifying and running a $XDEF_MINI job. This example shows two significant changes to the $XDEF_MINI job which affect DEF's PATCH command:

- An up-to-date patch file is named in DEF's PATCH command.

- A TIGR deck reflecting the site's specific hardware configuration is merged into the file named in DEF's PATCH command. (This is the TIGR_REL.SUPPORT file modified in the preceding examples.)

```
!C  $XDEF_MINI.SUPPORT TO $XDEF_TEST.MYACCT
 ..COPYing
!E $XDEF_TEST
EDIT C00 HERE
*TY1-4
   1.000 !DEFAULT SITE-ID='THE C',SITE-NAME='THE C','TAPE1'='CP6P01',;
   1.500 !PATCHWEEK=000
   2.000 !JOB PRIO=7,RERUN
   3.000 !RES MT=1,MEM=100,TIME=10
   4.000 !LIMIT FPOOLS=30
```

(cont. next page)

```
                    This job creates a single-reel bootable PO tape, to be used in
                    conjunction with existing reels CP6P02, CP6P03, etc...  The
                    primary reason for making such a tape is to place new patches on
                    the bootable portion of the PO tape for application to the monitor
                    and system processors (found on CP6P02, etc).

*RR1-4
    1.000 !DEFAULT SITE-ID='JJ''S CP-6',SITE-NAME='JJ''S CP-6',;
    1.500 !'TAPE1='CP6P01',PATCHWEEK=224
    2.000 !JOB PRIO=7,RERUN
    3.000 !RES MT=1,MEM=100,TIME=10
    4.000 !LIMIT FPOOLS=30

                    The site ID has been changed, site name has been changed, and a
                    change has also been made to PATCHWEEK=.

   12.000 !M MOUNT #TAPE1 -- RING IN FOR SITE-NAME PO TAPE
   13.000 !DEFAULT DEN=1600
   14.000 !SET M$PO FT#TAPE1,ORG=FREE
   15.000 !DEF
   16.000 PO C00,NOFILES
   17.000 MINI_ID 'PATCHWEEK'
   18.000 DENSITY DEN
   19.000 NOLIST
   20.000 SITEID 'SITE-ID'
   21.000 SITENAME 'SITE-NAME'
   22.000 FIRMWARE FIRMB3.:C00PRC
   23.000 BOOT AARDVARK.:C00PRC
   24.000 MON M:MON.:C00PRC
   25.000 MHJIT MHJIT.:C00PRC
   26.000 XDLT XDELTA.:C00PRC
   27.000 XDLTLS XDELTALS.:C00PRC
   28.000 GHOST1 GHOST1.:C00PRC
   29.000 G1HJIT G1HJIT.:C00PRC
   30.000 PATCH :C00PATCH.SUPPORT
   31.000 INSTALL $XINSTALL.SUPPORT
   32.000 END
*   EOF hit after 32.000
*EX :C00PATCH.SUPPORT
*FT0-9999,/!TIGR/
  450.000 !TIGR "L66
*   EOF hit after 3337.000

                    The system manager examines :C00PATCH.SUPPORT, and determines that
                    !TIGR "L66 starts on line 450.000.

!C :C00PATCH.SUPPORT TO :C00PATCH_TEST.MYACCT
 ..COPYing

                    :C00PATCH.SUPPORT is copied to a file in the system manager's own
                    account.

!E :C00PATCH_TEST.MYACCT
EDIT C00 HERE
*TY450
  450.000 !TIGR "L66
*TN23
  451.000 "TIGR REL"
  452.000 AUTOCONFIG
  453.000 FEP NAME=FEP1,IOM#=0,CHAN=33
  454.000 FEP NAME=FEP2,IOM#=0,CHAN=34
  455.000 FEP NAME=FEP3,IOM#=0,CHAN=35
  456.000 FEP NAME=FEP4,IOM#=0,CHAN=36
  457.000 MON ;
  458.000     SITE='CP-6',;
  459.000     SAL='*** CP-6 AT YOUR SERVICE',;
```

(cont. next page)

```
  460.000      IOCACHE=500,;
  461.000      STEALPGS=(15,30),;
  462.000      QUEUE=(90,90),;
  463.000      USERS=200,;
  464.000      DOLIST=50,;
  465.000      DEVMAX=300,;
  466.000      PATCH=600,;
  467.000      ENQ=(4,10),;
  468.000      CFU=(1,20),;
  469.000      SPSPACE=20,;
  470.000      SPAUTOSPACE=100,;
  471.000      SPROC=(LOGON,CP),;
  472.000      SPROC=(IBEX,CP),;
  473.000      SPROC=(DELTA,DB),;
*DE
*    23 records deleted
```

23 records are deleted (the first part of the issued TIGR deck).

```
*TN10
  474.000      SPROC=(IDS,AS),;
  475.000      SPROC=(:SHARED_COBOL,LI),;
  476.000      SPROC=(:SHARED_SYSTEM,LI),;
  477.000      SPROC=(:SHARED_COMMON,LI)
  478.000 !RUM "C00
  479.000 RUM :SHARED_COB,UTS=05/04/84 16:11:40.47
  480.000 RUM :SHARED_COBOL,UTS=04/20/84 10:49:27.44
  481.000 RUM :SHARED_COMMON,UTS=04/19/84 14:28:49.10
  484.000 RUM :SHARED_RPG,UTS=05/05/82 15:49:56.30
  483.000 RUM :SHARED_SYSTEM,UTS=04/13/84 16:25:32.02
*DE474-477
*    4 records deleted
```

The last four records of the issued TIGR deck which designate SPROC options are deleted.

```
*TP10
  441.000 " STUFF SCRATCH AREA WITH LOWER CASE IF QUOTED/OCTAL
              LEXEME #12253
  442.000 M F+.1226 TRA @ (TRA F+.1253) "DEG C00 05/25/84 2734 #12253
  443.000 M @ LDP1 PTR                 "DEG C00 05/25/84 28-34 #12253
  444.000 M @ .000100101500 "MRL        DEG C00 05/25/84 29-34 #12253
  445.000 M @ .200010000005 " SOURCE      DEG C00 05/25/84 30-34 #12253
  446.000 M @ .100000000005 " DEST       DEG C00 05/25/84 31-34 #12253
  447.000 M @ $RI                       "DEG C00 05/25/84 32-34 #12253
  448.000 KILL DEF PTR                  "DEG C00 05/25/84 33-34 #12253
  449.000 KILL DEF F                    "DEG C00 05/25/84 34-34 #12253
  450.000 !TIGR "L66
*DE450
*    1 records deleted
```

Line 450.000, which invokes TIGR, is deleted.

```
*MERGE TIGR_TEST.MYACCT,0-999 INTO :C00PATCH_TEST.MYACCT,450,.01
*    EDIT stopped
*    MERGE started
*    EOF hit after 50.000
*    Done at 450.500
*    51 records moved
*    MERGE done
```

The system manager merges the TIGR file into the system manager owned version of the :C00PATCH file.

(cont. next page)

```
*TY
  450.000 !TIGR "MY OWN TIGR DECK
  450.010 CPU PORT#=2
  450.020 CPU PORT#=3
  450.030 IOM PORT#=0
  450.040 CONSOLE NAME=SC01,IOM#=1,CHAN=30
  450.050 DISK  ;
  450.060       MPC,MPCNAME=DC01,MODEL=MSP0600 ;
  450.070            LA  ;
  450.080                 IOM#=0,CHAN=8-11      ;
  450.090       DEV  ;
  450.100            NAME=DP01,MODEL=(MSU0451,MSF0007),DEV#=1,SYSTEM ;
  450.110            NAME=DP02,MODEL=(MSU0451,MSF0007),DEV#=2        ;
  450.120            NAME=DP03,MODEL=(MSU0451,MSF0007),DEV#=3        ;
  450.130            NAME=DP23,MODEL=(MSU0501,MSF0015),DEV#=23       ;
  450.140            NAME=DP25,MODEL=(MSU0501,MSF0015),DEV#=25,STATUS=DOWN
  450.150 TAPE  ;
  450.160       MPC,MPCNAME=TC01,MODEL=MTP0610  ;
  450.170       LA  ;
  450.180            IOM#=0,CHAN=16-17  ;
  450.190       DEV ;
  450.200            NAME=MT01,MODEL=(MTU0630,MTF0636),DEV#=1     ;
  450.210            NAME=MT02,MODEL=(MTU0610,MTF0608),DEV#=2     ;
  450.220            NAME=MT03,MODEL=(MTU0610,MTF0608),DEV#=3
  450.230 UNIT  ;
  450.240     MPC,MPCNAME=UC01,MODEL=URP0600 ;
  450.250       NAME=LP01,MODEL=(PRU1200,PRB0600),IOM#=0,CHAN=24,OUT,SYMBIONT;
  450.260       NAME=CR01,MODEL=(CRU0501),IOM#=0,CHAN=26,IN,SYMBIONT
  450.270 FEP NAME=FEP1,IOM#=0,CHAN=33
  450.280 MON ;
  450.290       SITE='JJ''S CP-6',;
  450.300       SAL='*** JJ''S CP-6 AT YOUR SERVICE',;
  450.310       MTDFLT=(1600),;
  450.320       IOCACHE=500,;
  450.330       STEALPGS=(15,30),;
  450.340       QUEUE=(90,90),;
  450.350       USERS=75,;
  450.360       DOLIST=50,;
  450.370       DEVMAX=100,;
  450.380       PATCH=600,;
  450.390       ENQ=(4,10),;
  450.400       CFU=(1,20),;
  450.410       SPSPACE=20,;
  450.420       SPAUTOSPACE=50,;
  450.430       SPROC=(LOGON,CP),;
  450.440       SPROC=(IBEX,CP),;
  450.450       SPROC=(DELTA,DB),;
  450.460       SPROC=(IDS,AS),;
  450.470       SPROC=(:SHARED_COBOL,LI),;
  450.480       SPROC=(:SHARED_SYSTEM,LI),;
  450.490       SPROC=(:SHARED_COMMON,LI),;
  450.500       SPROC=(:SHARED_RPG,LI)
*END
```

The merged records are displayed.

```
!E $XDEF_TEST
EDIT C00 HERE
*AP
   33.000
*TP23
   10.000 !" monitor and system processors (found on CP6P02, etc)
   11.000 !"
   12.000 !M MOUNT #TAPE1 — RING IN FOR SITE-NAME PO TAPE
   13.000 !DEFAULT DEN=1600
   14.000 !SET M$PO FT#TAPE1,ORG=FREE
```

(cont. next page)

```
   15.000 !DEF
   16.000 PO C00,NOFILES
   17.000 MINI_ID 'PATCHWEEK'
   18.000 DENSITY DEN
   19.000 NOLIST
   20.000 SITEID 'SITE-ID'
   21.000 SITENAME 'SITE-NAME'
   22.000 FIRMWARE FIRMA3.:C00PRC
   23.000 BOOT AARDVARK.:C00PRC
   24.000 MON M:MON.:C00PRC
   25.000 MHJIT MHJIT.:C00PRC
   26.000 XDLT XDELTA.:C00PRC
   27.000 XDLTLS XDELTALS.:C00PRC
   28.000 GHOST1 GHOST1.:C00PRC
   29.000 G1HJIT G1HJIT.:C00PRC
   30.000 PATCH :C00PATCH.SUPPORT
   31.000 INSTALL $XINSTALL.SUPPORT
   32.000 END
*RR30
   30.000 PATCH :C00PATCH_TEST.MYACCT

                  The system manager changes line 30.000 to :C00PATCH_TEST.

*
*END
!BATCH $XDEF_TEST

                  As a final step, the system manager batches the modified file to
                  create a new PO tape.
```

Module 4-1

Introduction to Project and User Authorization

The system manager is responsible for enabling, monitoring and maintaining use of the CP-6 system. The many users of the CP-6 system must be controlled and the ultimate responsibility for that control lies with the system manager. The system manager needs to have a plan for controlling who will use the system and how.

In a typical installation, management of system use is facilitated by dividing use of the CP-6 system into a hierarchy of projects, each project consisting of a defined set of users. The system manager can administer projects directly, or the system manager can delegate responsibility to manage projects to one or more project administrators. If project administrators are used, the system manager defines who they are and what they can do.

This module describes the process which authorizes projects and logon users on the CP-6 system. Module 4-2 contains details and examples of project authorization. Module 4-3 contains details and examples of user authorization. Modules 4-2 and 4-3 are self-contained introductions to project and user authorization, respectively. Those modules can be distributed to appropriate staff as stand-alone tutorials. As such, an expanded version of some of the material presented here is repeated in those modules. Also, some material not presented here appears in both of those modules.

The tasks of these authorization processes involve the use of the SUPER and Packset Initializer Ghost (PIG) processors.

The system manager controls access to the CP-6 system in two different ways:

1. Through logon account authorization, which enables users to logon to and use the CP-6 system.

2. Through file management account authorization, which enables creation and maintenance of permanent files on the CP-6 system.

Note that these authorizations are separate. Therefore, a user can be authorized to log onto and use the CP-6 system by means of a logon account, but not to maintain any files in the logon account. (Likewise, a file management account can be created to contain files only — no user can then log onto that file management account.)

The SUPER processor is used for logon account authorization. SUPER is used to create, modify, list and remove (delete) projects and logon users. For each project and each user, the authorization includes a name, the system resources that will be used, and any special privileges.

The PIG processor is used for file management authorization. PIG is used to establish project packsets and user file management account(s). The PIG processor can be accessed from the SUPER processor directly so that some file management authorization control functions can be accomplished from the SUPER processor.

The modules in this section describe the logon authorization processes for projects and users and describe how to set up file management accounts from SUPER. These modules assume that the system environment in which the projects and users are created has already been established. For example, these modules describe establishing file management accounts in the context of an already initialized packset. The reader of this manual will be interested in module 1-1, which describes some general environment planning considerations, including grouping users, and the reader will want to refer to the CP-6 System Support Reference Manual (CE41) for details on all the SUPER commands and options introduced in this module and in module 4-2 and 4-3.

These modules present user authorization as occurring in a single record in an attempt to simplify the conceptual presentation. In fact, the user authorization information is contained in two system files: the :USERS and :HLP files.

Authorization Process

Authorization is the process of creating authorization records that identify projects and logon users and that specify the capabilities and limits that will be assigned to them. A similar process is used to create projects and to create logon users. Separate authorization records are built for each project and each user. Authorization records are created and maintained via SUPER commands. The fields in an authorization record are changed via SUPER options and suboptions. (Module 4–2 describes how project authorization records are built and maintained. Module 4–3 describes how user authorization records are built and maintained.)

Each CP–6 system is installed with five authorization records already in it:

Authorization Record	Purpose
DEFAULT	The system default user authorization record.
DEFAULTP	The system default user authorization record for projects.
:SYS,LJS	An initial logon user record for the system manager.
:FED,SUPPORT	A logon user record for Honeywell field engineering support staff.
:SYSTAC,LADC	A logon user record for Honeywell CP–6 software support staff.

Default records are provided with each CP–6 system to simplify the task of authorization. In a project or user authorization, each record field not explicitly assigned a value through the corresponding SUPER option or suboption is set to the value of the corresponding field in the default authorization record. One default record, DEFAULTP, is used to simplify the task of assigning attributes to users assigned to a project. The other default record, DEFAULT, is used to simplify the task of creating logon users who are not assigned to a project.

As part of CP–6 system initialization, the system manager will use the :SYS,LJS record as the logon user record until the system manager builds an authorization record. Typically, at this time also, the system manager will consider whether to modify the contents of the system–supplied DEFAULT and DEFAULTP record. At least the HSET field in the DEFAULT and DEFAULTP authorization records should be modified. If they are not, the system packset (#SYS) will be assumed as the default packset. Once the system manager builds the own user authorization record and once the field support personnel have completed testing of the system, the :SYS,LJS authorization record must be deleted from the system. Then, the process of building project and user authorization records begins.

Default Records

The DEFAULTP, DEFAULT and the user authorization records created through SUPER all contain an identification and the following classes of authorizations:

Authorization Class	Description
Budget	Establish dollar usage limits and accounting methods.
FEP	Establish miscellaneous, privilege and processor privilege authorizations for front end processors (FEPs).
Miscellaneous	Establish resource utilization and other limits, and tailor the CP-6 environment.
Privileges	Grant capabilities not available without special authorization.
Processor Privileges	Grant capabilities to use system processors.
Services	Establish service limits.

Authorization Record Contents

This section describes an authorization record by listing the contents of the system default authorization record (used to assign default values if there is no project default authorized record). Since all authorization records contain the fields in the system default record, this listing introduces the reader to the principal contents of any authorization record. This discussion also identifies the SUPER options and suboptions used to modify the contents of these fields. However, no attempt is made to describe each of the fields. Modules 4-2 and 4-3 describe some of the fields; all of the fields in an authorization record are described in the section "SUPER: System Administration" in the CP-6 System Support Reference Manual (CE41). However, this discussion does indicate the SUPER options and suboptions used to modify the fields introduced in this section.

Each field in the record is assigned a name. Values are listed with the associated field name. If the field contains a null value, the field name is listed with a blank value. Some fields have multiple entries for the different host and FEP modes. The host modes are:

B: Batch
G: Ghost
O: On-line
T: Transaction Processing

The FEP modes are:

U: User
C: Comgroup
H: Handler
G: Ghost

The first lines in the record listing list the default budget and accounting information, as follows:

```
ACHARGES   MCHARGES  BKACCESS  BUDLIM  ICHARGE  PCHARGE  BLINDACCOUNTING
 $0.00      NONE       YES       NO      YES      YES       NO

     These fields are changed via the BUDGET authorization
     suboptions.
```

The first field on the next line of the record listing identifies the project administrator and the next field on that line identifies the default home packset (HSET). The rest of the fields on that line and on the next several lines of the record are environment specific information including:

- the (default) terminal profile (PROFILE).
- whether a password is required to initially logon (PASSWORD).
- the identification of the (default) workstation, which usually describes where printed output will go (WSN).

```
PROJECT ADMIN          HSET     NATIVEL  PASSWORD  PROFILE      WSN
 NONE                   SYS               NO        TTY         LOCAL

OUTPUTPRIO  STEPACCNT  *S_ACCOUNTING  EXPIRE MAX  EXPIRE DEF  BATNUM
 7            NO          NO           NEVER       NEVER       -1

     PROJECT ADMIN can be changed via the SUPER command MODIFY
     PROJECT.  The other fields are changed using SUPER options
     that are the same as the field names except for EXPIRE MAX
     and EXPIRE DEF which are both changed through the EXPIRE
     option.
```

The following record fields describe the nine fields on print out banners.

```
BANNER1 ALTERABLE BANNER2 ALTERABLE BANNER3 ALTERABLE BANNER4 ALTERABLE
 YES               YES               YES               YES

BANNER5 ALTERABLE BANNER6 ALTERABLE BANNER7 ALTERABLE BANNER8 ALTERABLE
 YES               YES               YES               YES

BANNER9 ALTERABLE
 YES
                        By system default, banner fields can all
BANNER1                 be altered and no banner field contains
                        any text.
BANNER2
 o
 o
 o
BANNER9

     These fields are changed through the SUPER option BANNERTEXTn.
```

The following fields define the (default) command processor commands that are to be executed at logon.

```
  ALTERABLE SETUP
B: YES                  By system default, SETUP commands can be
G: YES                  altered.
O: YES
T: YES

  SETUP
B:                      By system default, no SETUP commands are
G:                      executed at logon time.
O:
T:

     These fields are changed through the SUPER option SETUP.
```

The following fields define additional environment and resource attributes.

```
   BILLING  MEM MAX  MEM DEF  TIME MAX  TIME DEF  QUAN      PRIOB
B:  1        511       64       9999      10        0         0
G:  1        511       256      9999      9999      0         0
O:  1        511       128      9999      9999      0         0
T:  1        511       256      9999      9999      0         0

  CPROC                  LAST CPROC
B: IBEX
G: IBEX
O: IBEX
T: IBEX
```

These fields are changed using SUPER options that are the same as the field names except for MEM MAX and MEM DEF which are both changed through the MEMORY option, and TIME MAX and TIME DEF which are both changed through the TIME option.

The following fields contain default service limits.

```
                                                    MAX LO  DEF LO  MAX PO
                                                     99999   1000    99999
                                                     99999   99999   99999
                                                     99999   99999   99999
                                                     99999   99999   99999

  DEF PO MAX DO DEF DO  MAX TDIS DEF TDIS  MAX PDIS DEF PDIS  MAX FPOOLS
B: 100    99999  50       99999   2000      99999    99999     31
G: 99999  99999  99999    99999   2000      99999    99999     31
O: 99999  99999  99999    99999   9999      99999    99999     31
T: 99999  99999  99999    99999   2000      99999    99999     31

  DEF FPOOLS MAX PRIO  DEF PRIO
B: 10         7         7
G: 10         7         7
O: 10         7         7
T: 10         7         7
```

The values in these fields can be changed via suboptions specified after the SUPER option SERVICES has been specified. (See the SUPER User Authorization Options and Suboptions table in Module 4-3.)

The following fields contain additional environment fields:

```
  ACCESS
B: YES
G: NO
O: YES
T: NO
F: NO

  KEY
   NONE

      The values in each of these fields can be changed via the SUPER
      options ACCESS and KEY, respectively.  The F: field enables
      front-end access.
```

The following fields contain the set of (default) privileges:

```
       PRIVILEGES

   ASAVE  DISPJOB  CFEP  EXMM  EXPM  FMDIAG  FMREAD  FMSEC  GPP  IOQ  IOQW
B: NO     NO       NO    NO    NO    NO      NO      NO     NO   NO   NO
G: NO     NO       NO    NO    NO    NO      NO      NO     NO   NO   NO
O: NO     NO       NO    NO    NO    NO      NO      NO     NO   NO   NO
T: NO     NO       NO    NO    NO    NO      NO      NO     NO   NO   NO

    JIT  MAXM  MFEP  MSYS  PM   SPCLMM  SYSCON  SYSLOG  TND
B:  NO   NO    NO    NO    NO   NO      NO      NO      NO
G:  NO   NO    NO    NO    NO   NO      NO      NO      NO
O:  NO   NO    NO    NO    NO   NO      NO      NO      NO
T:  NO   NO    NO    NO    NO   NO      NO      NO      NO

     The values in these fields can be changed via suboptions
     specified after the SUPER option PRIVILEGES has been
     specified.  (See the SUPER User Authorization Options and
     Suboptions Table in Module 4-3.)
```

The following fields contain the set of default processor privileges:

```
        PROCESSOR PRIVILEGES

   CNTRLC CNTRLD  EFT EL  NETCON   LABEL  PADMIN  PIGC PIGD  RATES REPLAY
B: NO     NO      NO  NO  NO       NO     NO      NO   NO    NO    NO
G: NO     NO      NO  NO  NO       NO     NO      NO   NO    NO    NO
O: NO     NO      NO  NO  NO       NO     NO      NO   NO    NO    NO
T: NO     NO      NO  NO  NO       NO     NO      NO   NO    NO    NO

   SPIDERC SPIDERD  SUPER SUPERA  SUPERD  SUPERF  SUPERW  VOLINIT  SYSCON
B: NO      NO       NO    NO     NO      NO      NO      NO       NO
G: NO      NO       NO    NO     NO      NO      NO      NO       NO
O: NO      NO       NO    NO     NO      NO      NO      NO       NO
T: NO      NO       NO    NO     NO      NO      NO      NO       NO

   PIGETTE
B:  NO
G:  NO
O:  NO
T:  NO

     The values in these fields can be changed via suboptions
     specified after the SUPER option PPRIVILEGES has been
     specified.  (See the SUPER User Authorization Options and
     Suboptions Table in Module 4-3.)
```

The following fields contain the set of (default) allocatable system peripherals:

```
     RESOURCES

    MT    DP
B:  4     4
G:  4     4
O:  4     4
T:  4     4

     The values in these fields can be changed via suboptions
     specified after the SUPER option RESOURCES has been specified.
     The SUPER option RESOURCES is used to allocate peripherals
     (tape drives, line printers, card punches and so on).
     (See the SUPER User Authorization Options and Suboptions
     Table in Module 4-3.)
```

The following field contains the default file management account information:

FACCOUNT

FXP-00027-0 Packset (packsetid) not currently mounted.

File management account information is assigned and maintained via the CP-6 PIG processor. PIG can be accessed through the SUPER option FACCOUNT so that matching logon account and file management account information can be processed as part of the same activity. However, PIG will still need to be entered to make the appropriate entry in the Master Account Directory.

The following fields contain FEP resource, privilege and processor privilege information. Note that while front end (FE) processor privileges (FE PROCESSOR PRIVILEGES) can be set in SUPER, they are not supported currently in the system.

```
  FE-MFPRG  FE-MAX-ACCT-MEM  FE-DBACCN
   255       9999

  FE-MINTS  FE-MAX-TIME  FE-MAX-MEM  FE-BILLING
U: 0         9999         128         1
C: 0         9999         128         1
H: 0         9999         128         1
G: 0         9999         128         1

    FE-PRIVILEGES

  EXMM  EXPM  FMREAD  FMSEC  GPP  MAXM  MSYS  SPCLMM  SYSLOG  TND  INTCON
U: NO    NO    NO      NO     NO   NO    NO    NO      NO      NO   NO
C: NO    NO    NO      NO     NO   NO    NO    NO      NO      NO   NO
H: NO    NO    NO      NO     NO   NO    NO    NO      NO      NO   NO
G: NO    NO    NO      NO     NO   NO    NO    NO      NO      NO   NO

  CQ   SNAP  SCREECH
U: NO   NO    NO
C: NO   NO    NO
H: NO   NO    NO
G: NO   NO    NO

       FE PROCESSOR PRIVILEGES

  CNTRLC CNTRLD  EFT  EL   NETCON  LABEL PADMIN  PIGC PIGD RATES  REPLAY
U: NO    NO      NO   NO   NO      NO    NO      NO   NO   NO     NO
C: NO    NO      NO   NO   NO      NO    NO      NO   NO   NO     NO
H: NO    NO      NO   NO   NO      NO    NO      NO   NO   NO     NO
G: NO    NO      NO   NO   NO      NO    NO      NO   NO   NO     NO

  SPIDERC SPIDERD  SUPER SUPERA  SUPERD  SUPERF  SUPERW  VOLINIT  SYSCON
U: NO     NO       NO    NO      NO      NO      NO      NO       NO
C: NO     NO       NO    NO      NO      NO      NO      NO       NO
H: NO     NO       NO    NO      NO      NO      NO      NO       NO
G: NO     NO       NO    NO      NO      NO      NO      NO       NO

  PIGETTE
U: NO
C: NO
H: NO
G: NO

     The values in these fields can be changed via the appropriate
     SUPER FEP options and suboptions (see the SUPER User Authorization
     Options and Suboptions Table in Module 4-3.)
```

Using SUPER

SUPER can be run as a batch or online job. Normally, SUPER is run as an online job. It is invoked by the command !SUPER. When SUPER is invoked, command mode is initiated and SUPER commands are issued. The online prompt in command mode is CMD*.

When SUPER commands are issued, they cause a sublevel mode, the option mode, to be entered. The online prompt in option mode is OPT*.

When some SUPER command options are issued, they cause a sub-sublevel mode, the command suboption mode to be entered. The online prompt to enter command suboptions is SUB*. The following example illustrates online prompting.

```
!SUPER

*** CP-6 SUPER C00***

CMD*CREATE ABC001,473JONES
OPT*HSET=USER
OPT*PASSWORD=001ABC
OPT*PROFILE=VIP7205
OPT*WSN=REMOTE
OPT*BUDGET        <--  Introduces suboption mode.
SUB*MCHARGES=200
SUB*BKACCESS=NO
SUB*              <--  Suboption mode terminated by a null line.
OPT*FACCOUNT      <--  Introduces suboption mode.
SUB*GR=50
SUB*              <--  Suboption mode terminated by a null line.
OPT*PRIV          <--  Introduces suboption mode.
SUB*ASAVE O,B,G
SUB*              <--  Suboption mode terminated by a null line.
OPT*              <--  Option mode terminated by a null line.
CMD*END           <--  SUPER terminated by the END command.
!
```

The next prompting level (down) is always initiated automatically. The previous prompting level (up) is returned to by responding to a prompt with END or a null line (i.e., either an immediate RETURN or a blank line). END must be specified in response to a command level prompt to terminate SUPER processing.

In most cases, multiple options and suboptions may be specified on the same line. If this is done, entries are separated by a semicolon.

If SUPER is run in the batch mode, the user must anticipate level changes and exit properly from each level before entering a higher level option or command.

Module 4-2
Project Administration

Projects are defined to provide a structure within which logon users are defined. Each project is a controlling element, arranged in a simple hierarchy; each project is subject to the resource limitations of the project above it, and may further restrict the resources available to the project below it. As these resources are consumed, they are charged against each project all the way up the hierarchy.

The following figure illustrates the hierarchical structure of project administration.

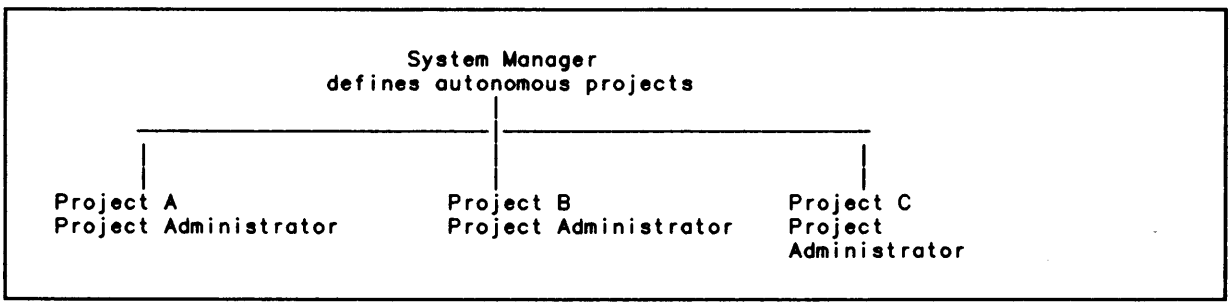


Figure 2. Project Structure

Within each project, logon users are authorized, as illustrated in the following figure.

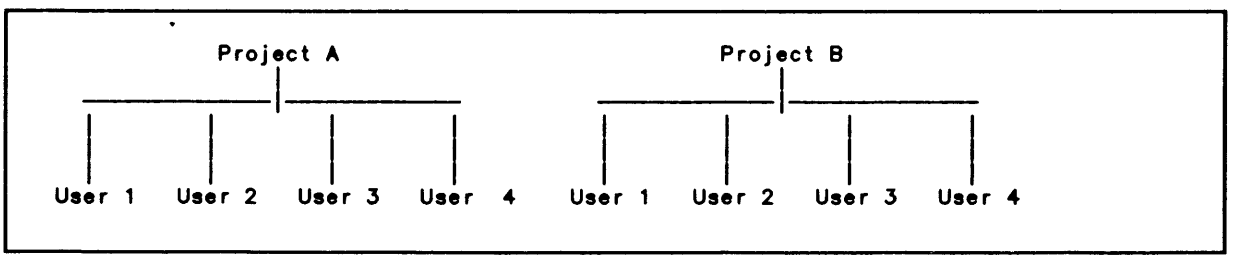


Figure 3. Project/User Structure

CP-6 users may then logon and use the CP-6 operating system. No user may be a member of more than one immediate project. In this case, each project is independent of the others.

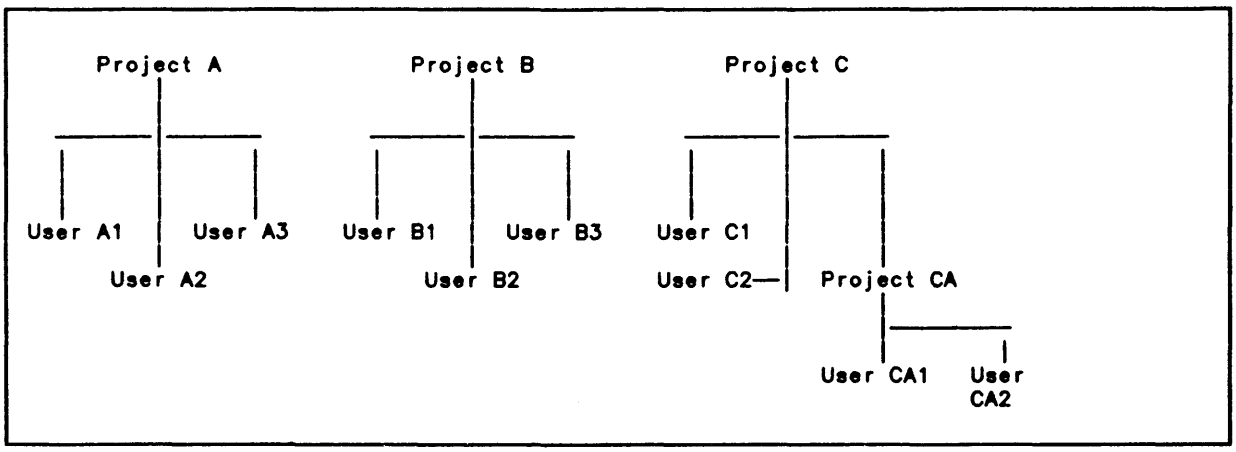


Figure 4. Multilevel Project Structure

Initially, the system manager must define projects with project administrators who can create subprojects. (If project administrators with project creation authorization are not created, then the system manager must define all projects in the system.) Each project is defined by:

- Assigning the project a logon id,
- Defining the maximum number of members allowed in the project,
- Defining the packset that will hold the project's disk storage,
- Defining the project default record.
- Defining the project administrator, if appropriate. If no project administrator is created, the system manager is the project administrator.

In practice, each project administrator is an accounting center. At definition, each project administrator is allocated a set of resources the major items of which are budget dollars and disk space.

The project administrator does not have a free hand. The system manager imposes the following constraints when creating the project:

- The number of logon IDs per project.
- The number of file management accounts per project.
- A maximum set of privileges, services, and resources that may be given to project members. At the system manager's discretion, the project administrator may further restrict this set or be forced to give each member exactly the same attributes specified for the project administrator.
- The total amount of disk space that may be used by all project members.

In addition, the system manager can preestablish a naming convention for logon IDs and file management accounts

Creating Projects

Once SUPER has been invoked, creation of a project is initiated by entering the following SUPER command:

```
CREATE PROJECT account,name
```

The values entered for account and name are the logon id that will be used for administration of this project (if a project administrator is created, the logon id of the project administrator). The account value consists of up to 8 alphanumeric characters and the name value consists of 1–12 alphanumeric characters and the symbols $ and :.

This command initiates the project authorization mode. The online prompt to enter project level options is PROJ*. Note that in project administration there are four levels of data entry and four prompts:

Data Entry Level	Prompt
The SUPER command level	CMD*
The project level	PROJ*
The option level	OPT*
The suboption level	SUB*

The following table summarizes all data entry levels except the command level and lists the options that can be specified at each level.

Table 3. SUPER Options

Project Level	Option Level	Suboption Level
ACCOUNTS		
ACHARGES		
ADMINISTRATOR	See the next table.	
DEFAULT	The same options as are available for ADMINISTRATOR are available for DEFAULT except that FACCOUNT and its suboptions are not available for DEFAULT.	Only the BUDGET suboptions are available. They are the same as for ADMINISTRATOR.
MCHARGES		
PACKSET	ACCOUNTS	
	ATTRIBUTES	The same sub–options as are available for the ADMINISTRATOR FACCOUNT option are available for the ATTRIBUTES option.
	GRANULES	
	SKELETON	

Table 3. SUPER Options (cont.)		
Project Level	Option Level	Suboption Level
PROJECTS		
REMOVE PACKSET		

The following table lists the ADMINISTRATOR options and suboptions. These user authorization options and are described further in Module 4-3.

Table 4. ADMINISTRATOR Options and Suboptions		
Option	Suboption	
ACCESS		
BANNERTEXT		
BATNUM		Note: The BATNUM option can be set in SUPER but is currently not supported in the system.
BILLING		
BUDGET	ACHARGES BKACCESS BLINDACCOUNTING BUDLIM ICHARGE MCHARGES PCHARGE	
CPROC		
EXPIRE		
FACCOUNT	[NO]ACUP [NO]BACKUP CGMEM [NO]CHECKWRITE [NO]DATACHECKWRITE DEFAULTBACKUP GRANLIM [NO]MERGEACCESS [NO]NEWFDS [NO]SHELFTIME OWNER [NOT]PROTECTED [NO]PURGE [NO]STOW File Access options: DELR, DELFILE, EXECUTE, FITMOD, NOLIST, READ,	

Table 4. ADMINISTRATOR Options and Suboptions (cont.)

Option	Suboption	
	REATTR, SCRATCH UPDATE, WNEW, WRITE Account access: CREATE, NONE	
FEBILLING		
FEDBACCN		
FEMACCTMEM		
FEMFPRG		
FEMINTS		
FEMMEMORY		
FEMTIME		
FEPPRIVILEGE	CNTRLC CNTRLD EFT EL LABEL NETCON PADMIN PIGC PIGD PIGETTE RATES REPLAY SPIDERC SPIDERD SUPER SUPERAUTH SUPERD SUPERFORM SUPERWSN SYSCON VOLINIT	Note: The FEPPRIVILEGE suboptions may be set in SUPER but currently are not supported in the system.
FEPRIVILEGE	CQ EXMM EXPM FMREAD FMSEC GPP INTCON MAXM MSYS SCREECH SNAP SPCLMM SYSLOG TND	
FEPSEUDO	One or more FEP pseudo resources.	

Table 4. ADMINISTRATOR Options and Suboptions (cont.)

Option	Suboption
FERESOURCES	One or more FEP resources.
HSET	
KEY	
LAST CPROC	
MEMORY	
NATIVEL	
OUTPUTPRIO	
PASSWORD	
PPRIVILEGE	CNTRLC CNTRLD EFT EL LABEL NETCON PADMIN PIGC PIGD PIGETTE RATES REPLAY SPIDERC SPIDERD SUPER SUPERAUTH SUPERD SUPERFORM SUPERWSN SYSCON VOLINIT
PRIOB	
PRIVILEGE	ASAVE CFEP DISPJOB EXMM EXPM FMDIAG FMREAD FMSEC GPP IOQ IOQW JIT MAXM MFEP MSYS

Table 4. ADMINISTRATOR Options and Suboptions (cont.)

Option	Suboption
	PM
	SPCLMM
	SYSCON
	SYSLOG
	TND
PROFILE	
PSEUDO	One or more pseudo resources.
QUAN	
RESOURCES	One or more resources.
*S_ACCOUNTING	
SERVICES	DO
	FPOOLS
	LO
	MAXJOBPRIO
	PDIS
	PO
	TDIS
SETUP	
STEPACCNT	
TIME	
WSN	

The following steps are required to create a project:

1. The project is named (i.e., assigned a logon ID) via the SUPER command CREATE PROJECT.

2. The number of logon accounts are specified (via the project level ACCOUNTS option).

3. The project authorization record is created (via the project level ADMINISTRATOR option). This record establishes the limits of the projects administrator which will normally set the limits the project administrator can establish for subprojects.

4. The packset that will be used for the project's disk storage is defined (via the project level PACKSET option). It is essential that the packset be specified; otherwise, the default is #SYS.

5. The project default record is created (via the project level DEFAULT option). This record contains the defaults that will be assigned to users created in this project.

The following figure illustrates how a project is created using the SUPER command CREATE PROJECT, SUPER options, and SUPER suboptions.

```
CMD*CREATE PROJECT ABC100,455WAI     <— Names the project.
PROJ*ACCOUNTS=4                      <— Specifies the total number of
                                        logon accounts for this project
                                        and any subprojects.
PROJ*MCHARGES=1000
PROJ*PROJECTS=2                      <— Specifies the number of
                                        subprojects that may be
                                        created.
PROJ*ADMINISTRATOR                   <— Initiates suboption mode to
                                        define the project administrator's
                                        logon user authorization.
OPT*HSET=USER
OPT*PASSWORD=TERESA
OPT*PROFILE=VIP7801                  <— The PROFILE VIP7801 is supplied
                                        standardly with the CP-6 system.
OPT*SETUP O='!XEQ SETUP'
OPT*SETUP B='!XEQ MFILE',U
OPT*WSN=REMOTE                       <— REMOTE must have been defined
                                        as a workstation.
OPT*BUDGET
SUB*ICHARGE=YES
SUB*MCHARGES=200
SUB*BKACCESS=NO
SUB*
OPT*FACCOUNT                         <— Sets a limit for the project
SUB*GRANLIM=10000                       administrator's file management
SUB*                                    account.
ABC100         0 OF 10000   Read=?, DEFAULT BACKUP, NO ACUP
OPT*PRIV
SUB*ASAVE O,B,G,T
SUB*DISPJOB O,B,G
SUB*
OPT*PPRIV
SUB*CNTRLD O,B
SUB*
OPT*PSEUDO
SUB*P6 O=6,B=6
SUB*
OPT*RESOURCES
SUB*MT O=1,B=2
SUB*
OPT*SERVICES
SUB*MAX TDIS O=4444
SUB*
OPT*
PROJ*DEFAULT                         <— Creates a project logon user
                                        default record.
OPT*PRIV
SUB*ASAVE O,B,G
SUB*
OPT*PPRIV
SUB*PIGD O,B
SUB*
OPT*PROFILE=VIP7801
OPT*SETUP O='!XEQ SETUP'
OPT*HSET=USER
OPT*
PROJ*PACKSET=DP#USER                 <— Defines the project packset and
```

(cont. next page)

```
OPT*AC=3                                    packset attributes.
OPT*GR=10000
OPT*SK=ABC?
OPT*ATTRIBUTES
SUB*NOLIST=ZZZ?
SUB*
OPT*
PROJ*END
```

Note how the various data entry levels are used. The following options can be entered at the project level:

Option	Description
ACCOUNTS	Specifies the maximum number of logon accounts that can be authorized in the project.
ACHARGES	Specifies the current accumulated charges for a project.
ADMINISTRATOR	Defines the logon account for the project administrator. INITIATES OPTION MODE (see below).
DEFAULT	Defines the default record (for the project users). INITIATES OPTION MODE (see below).
MCHARGES	Specifies the maximum dollar amount that the project can accumulate.
PACKSET	Specifies a packset for the project's file management account. INITIATES OPTION MODE (see below).
PROJECTS	Specifies whether a project can have subprojects by specifying the maximum number of subprojects that can be included in the project.
REMOVE PACKSET	Removes a packset from a project.

Three of these options — ADMINISTRATOR, DEFAULT and PACKSET — initiate option mode. In online mode, entering any of these project level prompts causes the prompt string to change from PROJ* to OPT*.

Option mode is terminated by entering:

- one of the options that initiate suboption mode (see below).
- a blank line in response to the prompt (which returns the user to project level option mode).

Administrator Option

ADMINISTRATOR is specified to enable use of the SUPER authorization options to establish a unique logon account for the project administrator. The options that can be specified are listed in the table and are the same as the authorization options described in the module "User Authorization".

Two of these options — BUDGET and FACCOUNT —initiate suboption mode. BUDGET is specified in response to the OPT* prompt to enable use of the BUDGET suboptions to define project administrator budgetary limits. FACCOUNT is specified in response to the OPT* prompt to enable use of PIG processor packset and account attribute options to define project administrator file account attributes. The FACCOUNT option should be used at least to set the home packset (via the HSET suboption) for the project. Otherwise, the default is #SYS.

Note that PIG processor commands cannot be entered as FACCOUNT suboptions. Specifying FACCOUNT performs an implicit PIG CREATE or MODIFY command. However, using FACCOUNT does not make an entry in the Master Account Directory. The PIG processor must be used to make that entry.

Fields in the project administrator's authorization record not explicitly set default to the corresponding fields in the DEFAULTP record.

Default Options

DEFAULT is specified to enable use of the SUPER authorization options to create a default record for users in the project. The default record is set up to simplify the user authorization task. Most values for users are assigned implicitly from this default record.

Note that the authorization option FACCOUNT and its suboptions are not available through the DEFAULT option, and that one of these options — the BUDGET option — initiates suboption mode. BUDGET is specified to enable use of the BUDGET suboptions to define default record budgetary limits.

Packset Options

PACKSET is specified to define the home packset for users in the project and any other packsets available to the project. If not specified, the default is #SYS. The PIG processor packset and account attribute options are specified as suboptions. Note that specifying ATTRIBUTES initiates suboption mode. ATTRIBUTES is specified so that the system manager can use PIG processor packset and account attribute options when defining the project administrator's packset utilization.

Neither PIG processor commands nor PIG processor suboptions can be entered as ATTRIBUTES suboptions. Specifying ATTRIBUTES performs an implicit PIG CREATE or MODIFY command. Only PIG options may be specified.

Listing Projects

The SUPER command LIST PROJECT is used to list projects associated with a project administrator or the system manager and to list information about a specific project. To list projects associated with a project administrator or the system manager, the LIST PROJECT command is entered as follows:

CMD*LIST PROJECT

and a project list similar to the following is listed:

```
ABC101,455LARKIN        ABC102,455SMITH
ABC103,455WELSER

 .. 3 projects listed.
```

The LIST command:

CMD*LIST PROJECT DEFAULT

lists the system authorization default record for users created in a project.

To list information about a specific project, the LIST PROJECT command is entered as follows:

CMD*LIST PROJECT account,name

A list option is then selected to display the information. The project list options are:

Project List Option	Information Listed
AC[COUNTS]	The maximum number of accounts that can be created in the project and the number of accounts that have been created in the project.
ACH[ARGES]	The accumulated charges for the project.
AD[MINISTRATION]	The project administrator's logon record.
A[LL]	The ACCOUNTS, ACHARGES, MCCHARGES, PACKSET and PROJECT information.
DE[FAULT]	The project default record.
MC[HARGES]	The maximum charges that the account may accumulate.

(cont.)	
Project List Option	Information Listed
PA[CKSET]	The project packset information, including account, granule, and attribute information.
PR[OJECT]	The maximum number of subprojects that can be created in the project and the number of subprojects that have been created in the project.

The following examples illustrate the information that is listed through LIST PROJECT options. (The project record listed is the one created in the earlier example.) In these examples, note that:

- All commands to list project information are formatted:

  LIST PROJECT logon-id

  in response to the CMD* prompt.

- The display to be listed is specified in response to the PROJ* or OPT* prompts, as appropriate. That is, the same prompting hierarchy is descended to create, modify or list elements of a project. Further, with the exception of the first example below, no display will appear until the user enters an END command or RETURN in response to a PROJ* prompt.

The following example uses the ALL option to list the information available through the ACCOUNTS, ACHARGES, MCHARGES, PACKSET and PROJECTS options:

```
CMD*L PROJ ABC100,455WAI
PROJ*ALL


ABC100,455WAI

   ACCOUNTS MAX  ACCOUNTS ACCUM  CHARGES MAX  CHARGES ACCUM  PROJECTS MAX
    10            5              $10000.00    $4000.00       2

   PROJECTS ACCUM
    0

        PACKSET DP#USER

   ACCOUNTS MAX  ACCOUNTS ACCUM  GRANULES MAX  GRANULES ACCUM  SKELETON
    10            5              10000         1000            ABC?

   ATTRIBUTES
    NOLIST=ZZZ?
.. 1 projects listed.
```

These fields are changed via the following options and suboptions:

Project Record Field	Project Level Option
ACCOUNTS MAX	ACCOUNTS
ACCOUNTS ACCUM	Accumulated by system.
CHARGES MAX	MCHARGES
CHARGES ACCUM	Normally, accumulated by the system. Can be set by the ACHARGES option.
PROJECTS MAX	PROJECTS
PROJECTS ACCUM	Accumulated by system.
PACKSET	PACKSET
ACCOUNTS MAX	These fields are all set as OPTION level options specified after the PACKSET option has been selected.
ACCOUNTS ACCUM	
GRANULES MAX	
GRANULES ACCUM	
SKELETON	
ATTRIBUTES	Initiates suboption mode. See the SUPER option and the ADMINISTRATOR Options and Suboptions tables.

The following are examples of using individual options instead of the ALL option to list the general project information:

```
CMD*LIST PROJ ABC100,455WAI
PROJ*AC
PROJ*

 ABC100,455WAI

   ACCOUNTS MAX  ACCOUNTS ACCUM
    10            5
 .. 1 projects listed.

CMD*L PROJ ABC100,455WAI
PROJ*PA
PROJ*
 ABC100,455WAI

       PACKSET DP#USER

  ACCOUNTS MAX  ACCOUNTS ACCUM  GRANULES MAX  GRANULES ACCUM  SKELETON
   10            5               10000         1000            ABC?

  ATTRIBUTES
   NOLIST=ZZZ?
 .. 1 projects listed.

CMD*L PROJ ABC100,455WAI
PROJ*PR
PROJ*

 ABC100,455WAI


   PROJECTS MAX  PROJECTS ACCUM
    2             0
 .. 1 projects listed.
```

The following example uses the ADMINISTRATION option to list the project administrator's logon id record:

```
CMD*L PROJ ABC100,455WAI
PROJ*ADMIN
OPT*ALL
PROJ*


ABC100,455WAI

      PROJECT ADMINISTRATOR
```

(cont. next page)

```
 ACHARGES     MCHARGES     BKACCESS  BUDLIM  ICHARGE  PCHARGE  BLINDACCOUNTING
  $3375.97     $5000.00     NO        NO      YES      YES      NO

 PROJECT ADMIN          HSET     NATIVEL  PASSWORD  PROFILE     WSN
  NONE                   USER              YES       VIP7801     REMOTE

 OUTPUTPRIO  STEPACCNT  *S_ACCOUNTING  EXPIRE MAX  EXPIRE DEF  BATNUM
  7           NO         NO             NEVER       NEVER       -1

 BANNER1 ALTERABLE  BANNER2 ALTERABLE  BANNER3 ALTERABLE  BANNER4 ALTERABLE
  YES                YES                YES                YES

 BANNER5 ALTERABLE  BANNER6 ALTERABLE  BANNER7 ALTERABLE  BANNER8 ALTERABLE
  YES                YES                YES                YES

 BANNER9 ALTERABLE
  YES

 BANNER1


 BANNER2

   o
   o
   o

 BANNER9


   ALTERABLE SETUP
B: NO
G: YES
O: YES
T: YES

   SETUP
B: !XEQ MFILE
G:
O: !XEQ SETUP
T:

   BILLING  MEM MAX  MEM DEF  TIME MAX  TIME DEF  QUAN      PRIOB
B: 1        511      64       9999      10        0         0
G: 1        511      256      9999      9999      0         0
O: 1        511      128      9999      9999      0         0
T: 1        511      256      9999      9999      0         0

   CPROC                LAST CPROC            MAX LO  DEF LO  MAX PO
B: IBEX                                       99999   1000    99999
G: IBEX                                       99999   99999   99999
O: IBEX                                       99999   99999   99999
T: IBEX                                       99999   99999   99999

   DEF PO  MAX DO  DEF DO  MAX TDIS  DEF TDIS  MAX PDIS  DEF PDIS  MAX FPOOLS
B: 100     99999   50      99999     2000      99999     99999     31
G: 99999   99999   99999   99999     2000      99999     99999     31
O: 99999   99999   99999   4444      4444      99999     99999     31
T: 99999   99999   99999   99999     2000      99999     99999     31
```

(cont. next page)

```
  DEF FPOOLS  MAX PRIO  DEF PRIO
B: 10          7         7
G: 10          7         7
O: 10          7         7
T: 10          7         7

  ACCESS
B: YES
G: NO
O: YES
T: NO
F: NO

  KEY
   NONE



      PRIVILEGES

  ASAVE  DISPJOB  CFEP  EXMM  EXPM  FMDIAG  FMREAD  FMSEC  GPP  IOQ  IOQW
B: YES     YES     NO    NO    NO    NO      NO      NO     NO   NO   NO
G: YES     YES     NO    NO    NO    NO      NO      NO     NO   NO   NO
O: YES     YES     NO    NO    NO    NO      NO      NO     NO   NO   NO
T: YES     NO      NO    NO    NO    NO      NO      NO     NO   NO   NO

  JIT  MAXM  MFEP  MSYS  PM  SPCLMM  SYSCON  SYSLOG  TND
B: NO   NO    NO    NO   NO   NO      NO      NO      NO
G: NO   NO    NO    NO   NO   NO      NO      NO      NO
O: NO   NO    NO    NO   NO   NO      NO      NO      NO
T: NO   NO    NO    NO   NO   NO      NO      NO      NO

      PROCESSOR PRIVILEGES

  CNTRLC CNTRLD  EFT  EL  NETCON  LABEL  PADMIN  PIGC  PIGD  RATES  REPLAY
B: NO      YES    NO   NO  NO      NO     YES     NO    NO    NO     NO
G: NO      NO     NO   NO  NO      NO     YES     NO    NO    NO     NO
O: NO      YES    NO   NO  NO      NO     YES     NO    NO    NO     NO
T: NO      NO     NO   NO  NO      NO     NO      NO    NO    NO     NO

  SPIDERC SPIDERD  SUPER  SUPERA  SUPERD  SUPERF  SUPERW  VOLINIT  SYSCON
B: NO       NO       NO     NO      NO      NO      NO      NO       NO
G: NO       NO       NO     NO      NO      NO      NO      NO       NO
O: NO       NO       NO     NO      NO      NO      NO      NO       NO
T: NO       NO       NO     NO      NO      NO      NO      NO       NO

  PIGETTE
B: NO
G: NO
O: NO
T: NO

    RESOURCES

  MT   DP
B: 6    4
G: 4    4
O: 6    4
T: 4    4
```

(cont. next page)

```
      FACCOUNT

ACCOUNT #GRANULES USED    ReadDelrecWnewUpdateScratchNolistFitmodExecCreate
ABC100      75 of 10000   Read=?,Nolist=ZZZ?

  FE-MFPRG FE-MAX-ACCT-MEM  FE-DBACCN
   0         9999

  FE-MINTS FE-MAX-TIME  FE-MAX-MEM  FE-BILLING
U: 0         9999          128          1
C: 0         9999          128          1
H: 0         9999          128          1
G: 0         9999          128          1

    FE-PRIVILEGES

  EXMM EXPM  FMREAD  FMSEC  GPP  MAXM  MSYS  SPCLMM  SYSLOG  TND  INTCON
U: NO   NO     NO      NO     NO   NO    NO    NO      NO      NO   NO
C: NO   NO     NO      NO     NO   NO    NO    NO      NO      NO   NO
H: NO   NO     NO      NO     NO   NO    NO    NO      NO      NO   NO
G: NO   NO     NO      NO     NO   NO    NO    NO      NO      NO   NO


  CQ  SNAP  SCREECH
U: NO  NO     NO
C: NO  NO     NO
H: NO  NO     NO
G: NO  NO     NO

      FE PROCESSOR PRIVILEGES

  CNTRLC CNTRLD  EFT  EL   NETCON   LABEL  PADMIN  PIGC  PIGD  RATES  REPLAY
U: NO      NO     NO   NO   NO       NO     YES     NO    NO    NO     NO
C: NO      NO     NO   NO   NO       NO     YES     NO    NO    NO     NO
H: NO      NO     NO   NO   NO       NO     YES     NO    NO    NO     NO
G: NO      NO     NO   NO   NO       NO     YES     NO    NO    NO     NO

  SPIDERC SPIDERD  SUPER  SUPERA  SUPERD  SUPERF  SUPERW  VOLINIT  SYSCON
U: NO       NO       NO     NO      NO      NO      NO      NO       NO
C: NO       NO       NO     NO      NO      NO      NO      NO       NO
H: NO       NO       NO     NO      NO      NO      NO      NO       NO
G: NO       NO       NO     NO      NO      NO      NO      NO       NO

  PIGETTE
U: NO
C: NO
H: NO
G: NO


.. 1 projects listed.
```

Note that:

- the project administrator's logon id record contains the exact same fields as any user authorization record. These fields are broken down by class in module 4-1 and some of them are described in module 4-3. They are all described in detail in the CP-6 Systems Support Reference Manual (CE41).

- because the project list option ALL was specified, the complete authorization record for the project administrator has been listed.

Several fields can be selected for listing by entering the names of the fields prior to entering a null line or 'END'. For example:

```
CMD*LIST PROJ ABC100,455WAI
PROJ*ADMIN
OPT*HSET;SETUP                          <-- Multiple options must be
                                            separated with a semi-colon.
OPT*
PROJ*


 ABC100,455WAI

        PROJECT ADMINISTRATOR

   HSET
    USER

   SETUP
 B:!XEQ MFILE
 G:
 O:!XEQ SETUP
 T:


 .. 1 projects listed.
```

Options specified must come from the same group. That is, the ALL options (ACCOUNTS,ACHARGES, MCHARGES, PACKSET and PROJECTS can be specified together, any combination of ADMINISTRATION options can be specified together, and any combination of DEFAULT options can be specified together. For example:

```
CMD*LIST PROJ ABC100,455WAI
PROJ*ACH;MCH;PRO
PROJ*
 ABC100,455WAI

   CHARGES MAX  CHARGES ACCUM  PROJECTS MAX  PROJECTS ACCUM
    $10,000.00   $4000.00       2              0
 .. 1 projects listed.

CMD*L PROJ ABC100,455WAI
PROJ*DEFAULT
OPT*PRIV;PPRIV
OPT*
PROJ*

 ABC100,455WAI

        DEFAULT RECORD FOR PROJECT

       PRIVILEGES

   ASAVE DISPJOB CFEP EXMM EXPM FMDIAG FMREAD FMSEC  GPP  IOQ  IOQW
 B: NO    NO      NO   NO   NO   NO     NO     NO     NO   NO   NO
 G: NO    NO      NO   NO   NO   NO     NO     NO     NO   NO   NO
 O: NO    NO      NO   NO   NO   NO     NO     NO     NO   NO   NO
 T: NO    NO      NO   NO   NO   NO     NO     NO     NO   NO   NO

   JIT  MAXM  MFEP  MSYS  PM   SPCLMM  SYSCON  SYSLOG  TND
 B: NO  NO    NO    NO    NO   NO      NO      NO      NO
 G: NO  NO    NO    NO    NO   NO      NO      NO      NO
 O: NO  NO    NO    NO    NO   NO      NO      NO      NO
 T: NO  NO    NO    NO    NO   NO      NO      NO      NO

        PROCESSOR PRIVILEGES

   CNTRLC CNTRLD EFT EL NETCON  LABEL PADMIN PIGC PIGD RATES REPLAY
 B: NO     NO     NO  NO NO     NO    YES    NO   NO   NO    NO
 G: NO     NO     NO  NO NO     NO    YES    NO   NO   NO    NO
 O: NO     NO     NO  NO NO     NO    YES    NO   NO   NO    NO
 T: NO     NO     NO  NO NO     NO    YES    NO   NO   NO    NO

   SPIDERC SPIDERD SUPER SUPERA SUPERD SUPERF SUPERW VOLINIT SYSCON
 B: NO     NO      NO    NO     NO     NO     NO     NO      NO
 G: NO     NO      NO    NO     NO     NO     NO     NO      NO
 O: NO     NO      NO    NO     NO     NO     NO     NO      NO
 T: NO     NO      NO    NO     NO     NO     NO     NO      NO


   PIGETTE
 B: NO
 G: NO
 O: NO
 T: NO

 .. 1 projects listed.
```

(cont. next page)

However, the following command mixes project display groups and will not be honored:

```
CMD*LIST PROJ ABC100,455WAI
PROJ*ACHARGES;ASAVE;MCHARGES
```

The specification:

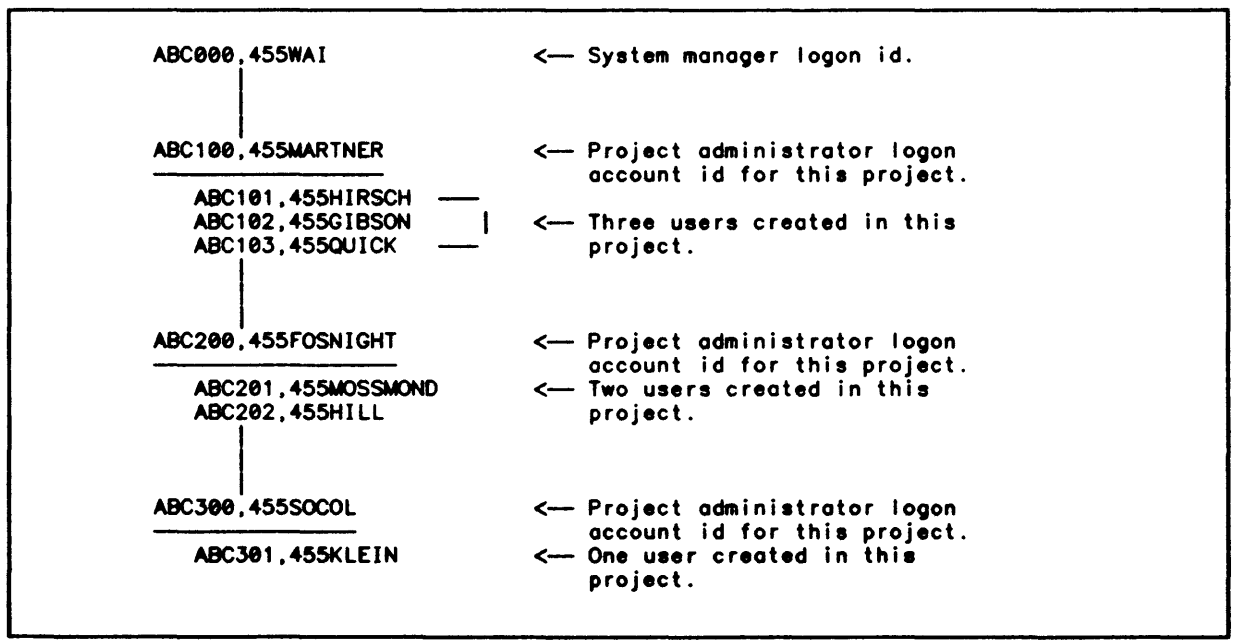
```
CMD*L PROJ ABC100,455SMITH
PROJ*DEFAULT
OPT*ALL
```

results in listing of the project default record. The project default record is an authorization record: it has the same fields as the default record (listed in module 4-1) and the project administrator's logon id record (listed above). The project default record contains the defaults that will be used for users defined within the project.

Administering Projects

SUPER features a number of commands that support the project authorization process.

Assume the following system manager, project, and user hierarchy has been defined:



The MAKEME command is used to move from project to project (and from project to system manager) within the hierarchy as follows:

- MAKEME project_logon_account_id is used to move through the project hierarchy.
- MAKEME RESET is used to move from anywhere in the hierarchy to the system manager level of the hierarchy. (This capability is not normally made available to project administrators.)

The SUPER processor responds to a number of commands that enable manipulation of the projects and users defined in a project hierarchy by allowing already defined projects and users to be moved within the hierarchy as follows:

- the TIE command attaches a logon user to a project; the TIE PROJECT command attaches a project as a subproject.
- the UNTIE command and UNTIE PROJECT command detach a user in a project and project in the hierarchy (respectively).

Note that in all these cases:

- all projects and users identified in these commands must already exist.
- the operation can only be performed at a lower level in the hierarchy than the current level. Thus:

  ABC300,455SOCOL cannot use any of the commands described above.

  ABC200,455FOSNIGHT can use the MAKEME command to become ABC300, 455SOCOL and to return to ABC200,455FOSNIGHT, and can use the TIE and MODIFY command only with the subordinate project (ABC300, 455SOCOL).

  ABC100,455MARTNER can use the MAKEME command to become either of the subordinate projects (ABC200,455FOSNIGHT or ABC300,455SOCOL), and can use the TIE and MODIFY commands with either of the subordinate projects, and so on.

The MODIFY command can also be used to turn a user into a project and a project into a user. In this case, too, the restrictions just described apply.

The MODIFY PROJECT command is used to modify the project administrator or project default record, and the REMOVE PROJECT command is used to remove a project.

The following example illustrates use of some of these commands.

```
!SUPER                                  <— The SUPER processor is invoked.

*** CP-6 SUPER C00 ***

CMD*LIST PROJ                           <— The LIST PROJECT command with
PROJ*                                       no further specification is
                                            issued to list all projects (at
                                            the system manager level) or sub-
                                            projects to the current project.


ABC100,455MARTNER       ABC200,455FOSNIGHT       ABC300,SOCOL
```

(cont. next page)

```
 ..3 projects listed.

CMD*MAKEME ABC100,455MARTNER          <— The MAKEME command is issued to
                                          access the first level project.
  Project: ABC100,455MARTNER          <— A verification message listing
                                          the project logon id is
                                          displayed.
CMD*LIST                              <— The LIST command with no further
OPT*                                      specification is issued to list
                                          all user logon ids in the
                                          project.

ABC101,455HIRSCH          ABC102,455GIBSON          ABC103,455QUICK

 ..3 users listed.

CMD*MAKEME ABC200,455FOSNIGHT         <— The MAKEME command is issued to
                                          access the second level project.
  Project: ABC200,455FOSNIGHT

CMD*LIST                              <— The LIST command with no further
OPT*                                      specification is issued to list
                                          all user logon ids in the
                                          project.  Note that users in a
                                          project can only be listed after
                                          the owning project logon id has
                                          been established as the current
                                          project.

ABC201,455MOSSMOND        ABC202,455HILL

 ..2 users listed.

CMD*TIE ABC201,455MOSSMOND TO ABC100,455MARTNER
                                          The TIE command is used to move
                                          a user from project ABC200,
                                          455FOSNIGHT to project ABC100,
                                          455MARTNER. Note that, in this
                                          case, a USER is moved to a
                                          PROJECT.  The following messages
                                          are produced to indicate the move
                                          is successful:
 .. User "ABC201,455MOSSMOND" untied from project "ABC200,455FOSNIGHT".
 .. User "ABC201,455MOSSMOND" tied to project "ABC100,455MARTNER".

CMD*MODIFY ABC202,455HILL TO PROJ     <— The MODIFY command is used to
                                          change a user to a project logon
                                          id.  The following messages
                                          confirm the change:
 ..User "ABC202,455HILL" untied from project "ABC200,455FOSNIGHT".
 ..User "ABC202,455HILL" modified to project "ABC202,455HILL".

CMD*REMOVE PROJ ABC200,455FOSNIGHT    <— The REMOVE PROJECT command is
                                          used to remove a project from the
                                          project hierarchy.  The following
                                          messages confirm the deletion,
                                          indicating that no users have
                                          been deleted since all users in
```

(cont. next page)

```
                                        the project have already been
                                        moved:
  Project "ABC200,455FOSNIGHT" removed. 0 users removed.
 .. 1 projects removed.
CMD*END

***  NO Errors  ***                <-- A count of errors and warning
***  NO Warnings ***                   messages terminates the SUPER
                                       session.  Note that if an error
                                       occurs or warning diagnostic is
                                       issued, the message is listed
                                       at the time the command, option
                                       or suboption is processed.
```

Notes on Using SUPER

The following notes will alert the project administrator to some features of SUPER that might not be immediately apparent or that need special emphasis:

1. When creating a project:

   - Always create the project authorization record before the project default record or project packset association is established.

   - To enable file account management to function correctly, always declare the home packset (HSET) for the project administrator and for the default record. The home packset will normally be the same for both, and will normally be the same as the packset declared via the PACKSET option.

   - To enable packset management, always use the PACKSET option to establish the packset association.

   - The system default for the SUPER PRIVILEGE ASAVE, which enables users to reconnect to a saved image following a terminal disconnect, is NO. Frequently, the ASAVE privilege will be set to YES for users. The project administrator may wish to simplify assignment of a YES value to the ASAVE field in a user record by changing the field value from NO to YES in the project default record.

   - Be aware that using the FACCOUNT option will invoke the PIG processor to Create or Modify a file management account; it will not automatically make an entry in the Master Account Directory. Subsequent to creating project accounts, the person creating the accounts will want to be sure to invoke the PIG processor to enter the accounts in the Master Account Directory.

2. For project authorization, the abbreviation PA can be specified in response to the prompts PROJ* or SUB*, and means different things. At the option level, in response to the prompt PROJ*, PA stands for PACKSET. However, when the ADMINISTRATOR or DEFAULT options have been specified, PA is the abbreviation for PASSWORD and is entered in response to the prompt SUB*. For user authorization, PA is always entered at the option level, in response to the prompt OPT*, and is always the abbreviation for PASSWORD.

3. For project authorization, values for accounting control fields are entered at the option level (via the ACHARGES and MCHARGES option). For user authorization, values for accounting control fields are entered at the suboption level (first the BUDGET option is specified, and then the suboptions ACHARGES and MCHARGES, as well as others, may be specified).

4. The PACKSET suboption SKELETON is used to establish account naming conventions. The SKELETON suboption imposes a formatting requirement on the account creator. For example:

```
PACKSET=DP#USER
GRANULES=500
SKELETON=ABC?
```

establishes the packset USER as the project packset, establishes its size as 500 granules and establishes that the names of all accounts must begin with ABC.

Module 4-3

User Authorization

A user may be:

- A human being at a timesharing (online) terminal.
- A series of commands running in the batch stream.
- Special processes called ghosts.
- Transaction Processing Users (TPUs) running as part of a TP instance.
- Front-end programs (FPRGs) and handlers running in Front-End Processors (FEPs).

Note that in the CP-6 system these users are distinguished by the way they are defined in SUPER.

In SUPER, these different types of users are seen as different modes. Therefore, the process of authorizing a user is the same regardless of the type of user. The process of authorizing these users is the same, also, regardless of whether or not a project hierarchy is used.

Authorization Elements

The precise steps to authorize a user via SUPER will vary according to the policies at a given installation. However, regardless of any installation specific requirements, the following steps must be considered when authorizing a user:

- Give the user a user logon ID that will identify the user to the system.
- Establish budget limits for the user.
- Establish the system resources available to the user.
- Define the user's service limits.
- Define the user's physical resource limits.
- If appropriate, define the number of pseudo resources that may be authorized by the user.
- Set up an environment tailored to the specific user.
- Identify the file management packset that will contain the user's files. (The file management account(s) are created by entering PIG via the SUPER option FACCOUNT.)

Establishing Budget Limits

Basically, each user is given a dollar budget and then charged for everything that is done. When the last of the budget is used, the user may be denied access to the system. There are two steps in the process:

1. The rate schedule established through the RATES processor (see Module 10) is assigned to the user with the SUPER option BILLING.

2. A budget is established for the user by setting a maximum value on the charges the user may accumulate. The SUPER option BUDGET and its suboptions are used to indicate whether or not the user is to be denied access to the system if the budget is exhausted, and whether or not the user is to be linked (for budget purposes) to another logon id. In addition, the user's accumulated charges may be modified via the BUDGET option. The BUDLIM option can be used to indicate that the user is to be charged at the end of every job step instead of at the end of a job. A list of charges for system resources is established by the RATES processor (see Module 10).

Establishing System Resource Limits and Defaults

The scope of each user's demand on the finite set of system resources must be limited. There are two basic ways of setting limitations: the first is to limit access to the resource itself; the second is to assign dollar values to the resources and limit the budget (in dollars) available to the user. A combination of the two methods is usually used.

The SUPER option MEMORY is used to establish maximum and default memory values. That is, the MEMORY option is used to limit the size of each user's memory and assign a default memory allocation when logging on to the system.

The SUPER option TIME is used to establish maximum and default limits for time for batch jobs only (measured in CPU time, not in wall-clock time).

Defining Service Limits and Defaults

Service limits include such things as line printer paper, punched cards, and temporary disk space. These limits apply on a per job basis; there is no global maximum covering multiple jobs submitted by the user. The SUPER option SERVICES and its suboptions establish the maximum and default values for these items.

Defining Physical Resource Limits

Physical resource limits define the largest request that will be honored for a user for physical devices (such as tapes, disk packs, and line printers). The SUPER option RESOURCE and its suboptions are used to define these limits. The limits established for physical resources define the largest request that will be honored for the user. The limits set here are for resources defined at boot-time (through the TIGR check) or through the SUPER processor DEFINE DEVICE command (for FEP-connected resources). Physical resources must be specifically requested by the user (via the IBEX command !RESOURCE (in batch mode) or !ORES (in online mode)).

Defining Pseudo Resources

Up to eight pseudo resources can be authorized for a user. The SUPER option PSEUDO and its suboptions establish the maximum number of the named pseudo resources the user may acquire. As with physical resources, defaults are not permitted.

Tailoring the Environment to the User

This step is more of a convenience to the user than a restriction of access to the CP-6 system. Some of the more commonly used options available to set up an environment tailored to the specific user include:

- PROFILE, which establishes the name of the default terminal profile to be used when this user logs on a timesharing terminal.
- WSN, which establishes the name of the default destination of printed output.
- PRIVILEGE, which enables selected users to access CP-6 capabilities not intended or necessary for most users. The privileges are ASAVE, CFEP, DISPJOB, EXMM, EXPM, FMDIAG, FMREAD, FMSEC, GPP, IOQ, IOQW, JIT, MAXM, MFEP, MSYS, PM, SPCLMM, SYSCON, SYSLOG, and TND. They are all described in the CP-6 System Support Reference Manual (CE41) in the Section "SUPER: System Administration".)
- PPRIVILEGE, which enables a user to have access to processors that should not normally be available to the general user of the system. (These processors are ANLZ, CONTROL, EFT, ELAN, LABEL, NETCON, PIG, PIGETTE, SPIDER, SUPER, VOLINIT, RATES and REPLAY.)
- CPROC, which permits the specification of a command program to be associated with the user when the user logs on (the default is IBEX).
- SETUP, used with CPROC, to define a single command or command file to be executed by the associated command program immediately after logging on. The command(s) defined via SETUP will be the first one(s) executed every time the user logs on.
- LAST CPROC, which permits the specification of a command program to be associated with the user when the user logs off.

In most cases, the system manager or project administrator will consider and set up these limitations and capabilities only once. The decisions made are stored in a default record (described below) which is used to assign most values to newly authorized users.

User Authorization Record

For each user, a user authorization record is created that contains the authorization information. Every user authorization record contains the same fields for establishing these characteristics of the user.

Each authorization record consists of over 140 fields of information. Each field is assigned a name and a value. When a user authorization record is listed, the field name appears above the field value. For example:

```
ACHARGES    <— Field name
 $0.00      <— Field value
```

Many of these fields (especially those fields associated with privileges and budgetary limits) take YES or NO values that determine whether a user has or does not have the authority to use a system capability. For example:

```
ACHARGES   MCHARGES   BKACCESS   BUDLIM   ICHARGE   PCHARGE   BLINDACCOUNTING
$3617.22   NONE       YES        NO       YES       YES       NO
```

Other fields (especially those fields associated with service and resource limits) take decimal values. For example:

```
   BILLING   MEM MAX   MEM DEF   TIME MAX   TIME DEF   QUAN      PRIOB
B: 1         511       64        9999       10         0         0
G: 1         511       256       9999       9999       0         0
O: 1         511       128       9999       9999       0         0
T: 1         511       256       9999       9999       0         0

   CPROC                    LAST CPROC                MAX LO   DEF LO   MAX PO
B: IBEX                                               99999    1000     99999
G: IBEX                                               99999    99999    99999
O: IBEX                                               99999    99999    99999
T: IBEX                                               99999    99999    99999
```

Note that:

- All authorization fields are present in the record even if the value is null (see LAST CPROC above).
- Where appropriate, the fields are divided into modes: B(atch), G(host), O(nline) and T(P).
- Where appropriate, the fields take text values other than YES or NO (e.g., CPROC, above).

The following figure is a listing of a user authorization record.

```
ACHARGES  MCHARGES  BKACCESS  BUDLIM  ICHARGE  PCHARGE  BLINDACCOUNTING
 $3617.00  NONE      YES       NO      YES      YES      NO

     These fields are set and changed via the BUDGET authorization
     suboptions.

PROJECT ADMIN          HSET    NATIVEL  PASSWORD  PROFILE     WSN
 LNHOO1,100101          USER             YES       VIP7205     LOCAL

OUTPUTPRIO  STEPACCNT  *S_ACCOUNTING  EXPIRE MAX  EXPIRE DEF  BATNUM
 7           NO         NO             NEVER       NEVER       -1

     PROJECT ADMIN is set when the user authorization record is
     created.  It can be changed via the SUPER command MODIFY
     PROJECT.  The other fields are set and changed using SUPER options
     that are the same as the field names except for EXPIRE MAX
     and EXPIRE DEF both of which are set and changed through the
     EXPIRE option.

BANNER1 ALTERABLE BANNER2 ALTERABLE BANNER3 ALTERABLE BANNER4 ALTERABLE
 YES               YES               YES               YES

BANNER5 ALTERABLE BANNER6 ALTERABLE BANNER7 ALTERABLE BANNER8 ALTERABLE
 YES               YES               YES               YES

BANNER9 ALTERABLE
 YES                    These record fields describe the nine user
                        fields on print out banners:  whether those
BANNER1                 fields can be altered and the text they
                        contain.
BANNER2
 o
 o
 o
BANNER9

     These fields are set and changed through the SUPER option
     BANNERTEXTn.

  ALTERABLE SETUP
B: YES                  These fields define the command processor
G: YES                  commands that are to be executed at logon.
O: YES
T: YES

  SETUP
B: !XEQ BFILE
G:
O: !XEQ $SETUP
T:

     These fields are set and changed through the SUPER option
     SETUP.
```

(cont. next page)

```
  BILLING   MEM MAX   MEM DEF   TIME MAX   TIME DEF   QUAN      PRIOB
B: 1          511        64       9999         10       0          0
G: 1          511       256       9999       9999       0          0
O: 1          511       128       9999       9999       0          0
T: 1          511       256       9999       9999       0          0

  CPROC                   LAST CPROC
B: IBEX
G: IBEX
O: IBEX
T: IBEX
```

These fields define additional environment and resource attributes. They are set and changed using SUPER options that are the same as the field names except for MEM MAX and MEM DEF which are both changed through the MEMORY option, and TIME MAX and TIME DEF which are both changed through the TIME option.

These fields contain service limits.

```
                                                  MAX LO   DEF LO   MAX PO
                                                   99999    1000     99999
                                                   99999   99999     99999
                                                   99999   99999     99999
                                                   99999   99999     99999

  DEF PO MAX DO DEF DO  MAX TDIS DEF TDIS  MAX PDIS DEF PDIS  MAX FPOOLS
B: 100    99999  50       99999    2000      99999    99999      31
G: 99999  99999  99999    99999    2000      99999    99999      31
O: 99999  99999  99999    99999    9999      99999    99999      31
T: 99999  99999  99999    99999    2000      99999    99999      31

  DEF FPOOLS MAX PRIO  DEF PRIO
B: 10          7          7
G: 10          7          7
O: 10          7          7
T: 10          7          7
```

The values in these fields are set and changed via suboptions specified after the SUPER option SERVICES has been specified.

```
  ACCESS
B: YES
G: NO
O: YES
T: NO
F: NO

   KEY
    NONE
```

These fields contain additional environment fields. The values in each of these fields can be set and changed via the SUPER options ACCESS and KEY, respectively.

PRIVILEGES

```
  ASAVE   DISPJOB   CFEP   EXMM   EXPM   FMDIAG   FMREAD   FMSEC   GPP   IOQ   IOQW
B: YES      YES      NO     NO     NO      NO       NO       NO     NO    NO    NO
G: YES      YES      NO     NO     NO      NO       NO       NO     NO    NO    NO
O: YES      YES      NO     NO     NO      NO       NO       NO     NO    NO    NO
T: YES      YES      NO     NO     NO      NO       NO       NO     NO    NO    NO
```

(cont. next page)

```
    JIT  MAXM  MFEP  MSYS  PM   SPCLMM  SYSCON  SYSLOG  TND
B:  NO   NO    NO    NO    NO   NO      NO      NO      NO
G:  NO   NO    NO    NO    NO   NO      NO      NO      NO
O:  NO   NO    NO    NO    NO   NO      NO      NO      NO
T:  NO   NO    NO    NO    NO   NO      NO      NO      NO
```

These fields contain the user's privileges. The values in these fields are set and changed via suboptions specified after the SUPER option PRIVILEGES has been specified.

PROCESSOR PRIVILEGES

```
   CNTRLC CNTRLD  EFT EL  NETCON   LABEL  PADMIN  PIGC PIGD  RATES REPLAY
B: NO     NO      NO  NO  NO       NO     NO      NO   NO    NO    NO
G: NO     NO      NO  NO  NO       NO     NO      NO   NO    NO    NO
O: NO     NO      NO  NO  NO       NO     NO      NO   NO    NO    NO
T: NO     NO      NO  NO  NO       NO     NO      NO   NO    NO    NO

   SPIDERC SPIDERD  SUPER SUPERA  SUPERD  SUPERF  SUPERW  VOLINIT  SYSCON
B: NO      NO       NO    NO      NO      YES     YES     NO       NO
G: NO      NO       NO    NO      NO      YES     YES     NO       NO
O: NO      NO       NO    NO      NO      YES     YES     NO       NO
T: NO      NO       NO    NO      NO      YES     YES     NO       NO

   PIGETTE
B: NO
G: NO
O: NO
T: NO
```

These fields contain the user's processor privileges. The values in these fields are set and changed via suboptions specified after the SUPER option PPRIVILEGES has been specified.

PSEUDO RESOURCES

```
   P6       STAR
B: 1        1
G: 0        0
O: 1        0
T: 16416    0
```

These fields contain the user's pseudo resources. The values in these fields are set and changed via suboptions specified after the SUPER option PSEUDO has been specified.

RESOURCES

```
    MT     DP     LP     CP     7T
B:  4      4      0      0      0
G:  4      4      0      0      0
O:  4      4      0      0      0
T:  16416  16416  16416  16416  0
```

These fields contain the set of allocatable system resources for this user.

The values in these fields are set and changed via suboptions specified after the SUPER option RESOURCES has been specified.

FACCOUNT

```
ACCOUNT  #GRANULES USED  ReadDelrecWnewUpdateScratchNolistFitmodExecCrea
ABC003     393 of  2000  Read=?, DEFAULT BACKUP, CGMEM=40, Nolist=Z?
```

(cont. next page)

These fields contain the user's file management account information. File management account information is assigned and maintained via the CP-6 PIG processor. PIG can be accessed through the SUPER option FACCOUNT so that matching logon account and file management account information can be processed as part of the same activity. However, the PIG processor must still be entered to make the appropriate entry in the Master Account Directory.

FE-MFPRG	FE-MAX-ACCT-MEM	FE-DBACCN
0	9999	

	FE-MINTS	FE-MAX-TIME	FE-MAX-MEM	FE-BILLING
U:	0	9999	128	1
C:	0	9999	128	1
H:	0	9999	128	1
G:	0	9999	128	1

FE-PRIVILEGES

	EXMM	EXPM	FMREAD	FMSEC	GPP	MAXM	MSYS	SPCLMM	SYSLOG	TND	INTCON
U:	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO
C:	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO
H:	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO
G:	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO

	CQ	SNAP	SCREECH
U:	NO	NO	NO
C:	NO	NO	NO
H:	NO	NO	NO
G:	NO	NO	NO

FE PROCESSOR PRIVILEGES

	CNTRLC	CNTRLD	EFT	EL	NETCON	LABEL	PADMIN	PIGC	PIGD	RATES	REPLAY
U:	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO
C:	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO
H:	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO
G:	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO

	SPIDERC	SPIDERD	SUPER	SUPERA	SUPERD	SUPERF	SUPERW	VOLINIT	SYSCON
U:	NO	NO	NO	NO	NO	NO	NO	NO	NO
C:	NO	NO	NO	NO	NO	NO	NO	NO	NO
H:	NO	NO	NO	NO	NO	NO	NO	NO	NO
G:	NO	NO	NO	NO	NO	NO	NO	NO	NO

	PIGETTE
U:	NO
C:	NO
H:	NO
G:	NO

These fields contain FEP resource, privilege and processor privilege information. Note that of the three settable FEP options — FEP resource, privilege and processor privilege options — only FEP resource and privilege options should be set. The third category, FEP processor privilege can be set in SUPER, but FEP processor privileges are not currently supported in the system. The FEP modes are: U(ser), C(omgroup), H(andler) and G(host). The values in these fields are set and changed via the appropriate SUPER FEP options and suboptions.

(cont. next page)

Default Record

The values in all fields can be assigned and changed via SUPER options (described below). In fact, the process of authorization is usually fairly simple because a default record is used to assign values for most of the fields in the user authorization record. When a user authorization record is created, a field not explicitly assigned a value via a SUPER option is implicitly assigned the value from the corresponding field in the default record.

If a project has been created, the project default record (DEFAULTP) will be used to make implicit value assignments to the fields in the user authorization record. The project default record is created by the project (or system) manager when the project is created (see Module 4–2). If a project hierarchy is not being used, the system default record will be used for such assignments. The system default record (named DEFAULT) is supplied with the system and can be modified by the system manager. (Module 4–1 contains a listing of the system default record.

Creating User Authorizations

Once SUPER has been invoked, creation of a user authorization is initiated by entering the following SUPER command:

```
CREATE account,name
```

This command:

- assigns the user logon id,
- initiates user authorization mode.

User Logon ID

The user logon ID identifies the user to the system. The logon ID is the key to gaining access to the CP–6 system. As such, the logon id is the system manager and project administrator's primary means of control over the users of the system. The facilities of a CP–6 system may not be used without a valid logon ID. As described in module 1–1, the user logon ID consists of:

```
account,name,password
```

The account defines the user's default file management account and determines a user's access to the files of other users. The account value consists of up to 8 alphanumeric characters. The name identifies the individual user. The name value consists of 1–12 alphanumeric characters and the symbols $ and :. These two fields are assigned via the SUPER command CREATE. The password is the user's tool for controlling access. An initial password can be assigned by using the SUPER option PASSWORD when defining the user.

The values assigned to these logon ID fields will result from a careful pre–installation planning of account groupings, of name conventions, and of policy regarding assignment of an initial password. Module 1–1 emphasizes the importance of accounts as the cornerstone of CP–6 security and file management procedures, and why grouping of users by account is important. Module 1–1 includes some examples of different grouping schemes.

Initiating User Authorization Mode

The SUPER command CREATE initiates user authorization mode. Before this command is issued, the project or system manager will have determined which fields in the user authorization record are to be explicitly assigned values and which fields are to default to the corresponding field values in the default record.

The online prompt to enter user authorization options is OPT*. Some options introduce a suboption mode. The online prompt to enter suboptions is SUB*. Note that in user authorization there are three levels of data entry and three prompts:

Data Entry Level	Prompt
The SUPER command level	CMD*
The option level	OPT*
The suboption level	SUB*

The following table lists the user authorization options and suboptions. Note that:

- most of the options parallel field names in the user authorization record.
- many options can be specified:

  field_name=YES
  or
  field_name=NO

  In these cases, specifying just

  field_name

  is the same as specifying field_name=YES.

- the BUDGET, FACCOUNT, FEPPRIVILEGE, FEPRIVILEGE, FEPSEUDO, FEPRESOURCES, PPRIVILEGE, PRIVILEGE PSEUDO, RESOURCES and SERVICES options introduce suboption mode.
- in general, keywords can be shortened by entering only the first few characters of the keyword. In this module, the minimum portion of a keyword appears outside square brackets, while the characters that are not required, that is, the extended portion of the keyword, are printed inside square brackets. The SUPER user must enter the minimum portion, and may choose to enter more. The reader should be aware that:

  - all of the minimum portion of a keyword must be entered and must be entered exactly as shown.
  - any number of characters in the extended portion may be added, but they must be entered in sequence.

  Looking at the option:

  FA[CCOUNT]

The SUPER user can enter:	The SUPER user CANNOT enter:
FA	F
FAC	FAT
FACC	FANT
FACCO	FCCOUNT
FACCOU	FACOUNT
FACCOUN	FACCXYZ
FACCOUNT	
	and so on.

Table 5. User Authorization Options and Suboptions

Options	Suboptions	Comment
AC[CESS]		
BANNERTEXT		
BA[TNUM]		The BATNUM option can be specified in SUPER but is currently not supported in the system.
BI[LLING]		
BU[DGET]	AC[HARGES] BK[ACCESS] BL[INDACCOUNTING] BU[DLIM] IC[HARGE] MC[HARGES] PC[HARGE]	
CP[ROC]		
EX[PIRE]		
FA[CCOUNT]	A[CUP] or NOA[CUP] B[ACKUP] or NOB[ACKUP] CG[MEM] C[HECKWRITE] or NOC[HECKWRITE] D[ATACHECKWRITE] or NOD[ATACHECKWRITE] D[EFAULT]B[ACKUP] G[RANLIM] M[ERGEACCESS] or NOM[ERGEACCESS] N[EWFDS] or NON[EWFDS]	

Table 5. User Authorization Options and Suboptions (cont.)

Options	Suboptions	Comment
	SH[ELFTIME] or NOSH[ELFTIME] O[WNER] PR[OTECTED] or NOT PR[OTECTED] PU[RGE] or NOPU[RGE] ST[OW] or NOST[OW] File Access Suboptions: D[ELR] DELF[ILE] E[XECUTE] F[ITMODE] N[OLIST] READ or RD REATTR SCRATCH U[PDATE] W[NEW] WR[ITE] Account Access Suboptions: C[REATE] NO[NE]	
FEBI[LLING]		
FEDB[ACCN]		
FEMA[CCTMEM]		
FEMF[PRG]		
FEMI[NTS]		
FEMME[MORY]		
FEMTI[ME]		
FEPP[RIVILEGE]	CNTRLC CNTRLD EF[T] EL LA[BEL] NETCON PADMIN PIGC PIGD PIGETTE RATES REPLAY SPIDERC SPIDERD SU[PER] SUPERA[UTH]	NOTE: The FEPPRIVILEGE suboptions can be set in SUPER, but currently are not supported in the system.

Table 5. User Authorization Options and Suboptions (cont.)

Options	Suboptions	Comment
	SUPERD	
	SUPERF[ORM]	
	SUPERW[SN]	
	SYSC[ON]	
	VOLINIT	
FEPR[IVILEGE]	CQ	
	EXM[M]	
	EXP[M]	
	FMR[EAD]	
	FMS[EC]	
	GP[P]	
	IN[TCON]	
	MA[XM]	
	MS[YS]	
	SCR[EECH]	
	SNAP	
	SP[CLMM]	
	SYSL[OG]	
	TN[D]	
FEPS[EUDO]	One or more FEP pseudo resources.	
FERE[SOURCES]	One or more FEP resources.	
HS[ET]		
KEY		
L[AST] CP[ROC]		
ME[MORY]		
NA[TIVEL]		
OU[TPUTPRIO]		
PA[SSWORD]		
PP[RIVILEGE]	CONTRLC	
	CNTRLD	
	EF[T]	
	EL	
	LA[BEL]	
	NETCON	
	PADMIN	
	PIGC	
	PIGD	
	PIGETTE	
	RATES	
	REPLAY	
	SPIDERC	
	SPIDERD	

Table 5. User Authorization Options and Suboptions (cont.)

Options	Suboptions	Comment
	SU[PER]	
	SUPERA[UTH]	
	SUPERD	
	SUPERF[ORM]	
	SUPERW[SN]	
	SYSC[ON]	
	VOLINIT	
PRIO[B]		
PRIV[ILEGE]	AS[AVE]	
	CF[EP]	
	DI[SPJOB]	
	EXM[M]	
	EXP[M]	
	FMD[IAG]	
	FMR[EAD]	
	FMS[EC]	
	GP[P]	
	IOQ	
	IOQW	
	JI[T]	
	MA[XM]	
	MF[EP]	
	MS[YS]	
	PM	
	SP[CLMM]	
	SYSC[ON]	
	SYSL[OG]	
	TN[D]	
PRO[FILE]		
PS[EUDO]	One or more pseudo resources.	
QU[AN]		
RE[SOURCES]	One or more resources.	
*S[_ACCOUNTING]		
SER[VICES]	DO	
	FP[OOLS]	
	LO	
	MA[XJOBPRIO]	
	P[DIS]	
	PO	
	TD[IS]	
SET[UP]		
ST[EPACCNT]		

Table 5. User Authorization Options and Suboptions (cont.)		
Options	Suboptions	Comment
TI[ME]		
W[SN]		

Generally, the system manager can assign any option or suboption, but a project administrator can assign options or suboptions only up to the level of authorization that the project administrator's own authorization record allows.

The following example illustrates the subset of SUPER options needed to set up a new user authorization record. In this example, all field values that can be taken from the default record are taken from the default record.

```
CMD*CREATE ABC001,001SMITH
OPT*HSET=USER                <— The home packset must always be specified
                                 (or it defaults to #SYS).
OPT*FACCOUNT                 <— The file management account may be created
SUB*READ=?,WRITE=ABC?            via SUPER or the PIG processor can be in-
SUB*GR=500                       voked separately.
SUB*
OPT*

More typically, a series of options like the following will be
used to create a user:

CMD*CREATE ABC002,002ROGERS
OPT*HSET=USER
OPT*PASSWORD=PHIL            <— Setting of initial user passwords is an
                                 installation policy.
OPT*PROFILE=VIP7801          <— Typically the account creator will need to
OPT*WSN=UPSTAIRS                 consider the choice of the default terminal
                                 and the default output destination for each
                                 user.
OPT*PRIVILEGE                <— Typically, the ASAVE privilege should be
SUB*DISPJOB 0,B,G,T              set to YES.  Note that suboption mode is
                                 entered and the modes that the privilege
                                 is to be applied to are specified.
SUB*
OPT*FACCOUNT
SUB*READ=?,WRITE=001?,CGMEM=40,NOLIST=XYZ?
SUB*
OPT*
```

A user can also be created from another user, as follows:

```
CMD*CREATE ABC003,003GOMEZ FROM ABC002,002ROGERS
OPT*BUDGET                       In this case, ABC003,003GOMEZ is created
SUB*MCHARGES=200.00              with all field values set the same as for
SUB*BKACCESS=NO                  ABC002,002ROGERS, except that the account
SUB*BUDLIM=YES                   is to have a maximum of accumulated charges
SUB*                             of $200.00, no bankruptcy access, budget
OPT*STEPACCNT                    limits are to be checked at each job step,
OPT*                             and job step accounting is to be performed.
```

The following example creates a different kind of user authorization record in which only batch jobs can run under the account. The account will be able to use two P6 pseudo resources and three disk packs. Its workstation of origin is named SHAMOKIN. The maximum time for any job in batch should be 6 hours, and the default time is 30 minutes.

```
CMD*C ABC004,004BATCH
OPT*ACCESS B, O=NO, G=NO, T=NO
OPT*PSEUDO
SUB*P6 B=2
SUB*
OPT*RESOURCES
SUB*DP B=3
SUB*
OPT*WSN=SHAMOKIN
OPT*TIME MAX B=360; TIME DEF B=30
OPT*
CMD*
```

Requesting Help

Online documentation on SUPER can be listed at the terminal as SUPER is being used. Online documentation can be requested both prior to entering a command and in response to an error diagnostic.

Requesting Online Documentation Before Entering a Command

Entering HELP in response to the CMD* prompt will result in display of the following message.

```
CP-6 communication management, project and user authorization, and
forms control functions are implemented through the SUPER processor.

To obtain more HELP information, see

HELP (SUPER) TOPICS                    Displays topics.
HELP (SUPER) COMMANDS                  Displays a list of SUPER
                                       commands.
HELP (SUPER) command PARAM             Displays parameters associated
                                       with a particular command.
HELP (SUPER) command DESCRIPTION       Displays description associated
                                       with a particular command.
?                                      Displays next level of HELP
                                       messages.
??                                     Displays all levels of HELP
                                       messages.
```

As this message suggests, SUPER HELP consists of different kinds of information about SUPER. The different topics of SUPER information for which online documentation exists can be listed by entering HELP TOPICS, which will result in the following display:

```
!HELP (SUPER) TOPICS
ALIGN_SUBOPTIONS       BAN_SUBOPTIONS       BUDGET_SUBOPTIONS
CHARSET_OPTIONS        COMMANDS   COMMUNICATIONS_COMMANDS
COMMUNICATION_PROFILE_OPTIONS    CONSOLE  CONSOLE_OPTIONS   CONTROLLER
CREATE     CREATE_CHARSET      CREATE_COMMANDS     CREATE_FORM
CREATE_FORM_GRAPHICS  CREATE_PROJECT       CREATE_PSEUDO
CURSOR_PROFILE_OPTIONS          DELETE_CHARSET      DELETE_COMMANDS
DELETE_FORMS          DELETE_USERS          DEVICE   DEVICE_ATTRIBUTE_OPTIONS
DEVICE_FOR_LINE       DEVICE_OPTIONS        ENTRY_OPTIONS     ENTRY_TEXT
FIELD_PROFILE_OPTIONS FORM_COMMANDS         FORM_OPTIONS
GRAPHICS_ALIGN_SUBOPTIONS     GRAPHICS_BAN_SUBOPTIONS
GRAPHICS_ENTRY_OPTIONS        GRAPHICS_FORM_OPTIONS GRAPHICS_PROFILE_OPTIONS
HELP       HELP_INDEX INVOKING_SUPER       LINE      LINK
LINK_PROFILE_OPTIONS  LIST    LIST_CHARSET          LIST_COMMANDS
LIST_FORM  LIST_PROJECT       LIST_TDEVICE          MAKEME
MISC_PROFILE_OPTIONS  MODIFY  MODIFY_COMMANDS       MODIFY_PROJECT
NOTATION   OPERATIONAL_PROFILE_OPTIONS  PACKSET_SUBOPTIONS    PRIVILEGES
PROFILE    PROFILE_OPTIONS    PROFILE_OPTION_SUMMARY          PROFILE_TYPES
PROJECT_COMMANDS      PROJECT_OPTIONS   RELATED_FILES         REMOVE
REMOVE_CHARSET        REMOVE_COMMANDS   REMOVE_FORM
REMOVE_PROJECT        STATION  TERMINAL TIE        TIE_PROJECT
TIMING_PROFILE_OPTIONS          UNTIE    UNTIE_PROJECT         USER_COMMANDS
USER_OPTIONS          VIRCIR_PROFILE_OPTIONS       VIRTUAL_CIRCUIT
WSN
```

The user can request:

- A list of SUPER commands
- Information about a specific SUPER command
- A list of SUPER options or suboptions
- Specific information about the options or suboptions in a category
- Other kinds of information.

The following is the display that results when HELP COMMANDS is entered in response to the CMD* prompt.

```
A list of SUPER commands and their definitions can be accessed by
typing

HELP (SUPER) COMMUNICATIONS_CMDS        for SUPER communications commands.
HELP (SUPER) PROJECT_CMDS               for SUPER project authorization
                                        commands.
HELP (SUPER) USER_CMDS                  for SUPER user authorization
                                        commands.
HELP (SUPER) FORM_CMDS                  for SUPER line-oriented and graphic
                                        form commands.
```

The following is the display that results when HELP USER_CMDS is entered in response to the CMD* prompt.

```
A list of user authorization commands and their definitions can be
accessed by typing a ? or a ?? after this message.  To obtain
definitions of individual user authorization commands, type

HELP (SUPER) USER_CMDS command

where command is any of the following: CREATE, DELETE_USERS, LIST, MODIFY and
REMOVE.
```

The following is the display that results when ?? is entered.

```
CREATE
        Invokes the user authorization mode of SUPER.
DELETE USERS
        Deletes all and rebuilds default user definitions.
LIST
        Displays user authorization information.
MODIFY
        Alters an existing authorization.
REMOVE
        Removes an existing authorization.
```

The following is the display that results when HELP CREATE is entered in response to the CMD* prompt.

```
Format:

C[REATE] [.]account1, name1 [FROM {[.]account2, name2}]
                               [     {DEFAULT          }]
```

Note that only the syntax is displayed. HELP information is organized hierarchically: the user can request each layer of information for a topic separately, or the user can request that all information for a topic be displayed. For example, the layers of information for a command include, in addition to syntax, parameter descriptions, a description of the command and, where appropriate, examples. Each of these layers can be displayed. For example.

```
CMD*?

Parameters:


account1, name1     is the logon ID.  This combination must be entered to gain access
to the CP-6 system.  The account is limited to 8 characters, the name to 12.  The
valid character set consists of alphanumeric characters plus the characters $ and :.

FROM account2, name2     indicates that each item not explicitly specified at the
option level is to be set to the value of the corresponding item of the
authorization record identified by this log-on ID.

FROM DEFAULT     indicates that each item not explicitly specified at the option
level is to be set to the value of the corresponding item of the default
authorization record.  (The default record may be initialized by substituting the
keyword DEFAULT for the account-name logon ID, i.e., CREATE DEFAULT.  Use of the
DEFAULT authorization permits the definition of many users with a minimum of
effort.)

If FROM is not specified, FROM DEFAULT is assumed.


CMD*?


Description:

This command invokes the user authorization mode of SUPER.  Options that may be
entered in this mode can be accessed by typing HELP (SUPER) USER_OPTS.


CMD*?


No message available.
```

Alternatively, ?? can be entered after the syntax is displayed to display all remaining information about the command. For example:

Format:

```
M[ODIFY] [.]account1, name1 [FROM {[.]account2, name2}]
                               [     {DEFAULT          }]
```

CMD*??

Parameters:

account1, name1 is the logon ID. This combination must be entered to gain access to the CP-6 system. The account is limited to 8 characters, the name to 12. The valid character set consists of alphanumeric characters plus the characters $ and :. Note that wildcarding is allowed for account1 and name1 (e.g., M[ODIFY] ?HOST, 102?).

account2, name2 indicates that each item not explicitly specified at the option level is to be set to the value of the corresponding item of the authorization record identified by this log-on ID.

DEFAULT indicates that each item not explicitly specified at the option level is to be set to the value of the corresponding item of the default authorization record.

Description:

This command alters an existing user authorization. The options which may be altered can be accessed by typing HELP (SUPER) USER_OPTS.

The following is an example of how to list options and then list specific information about an option.

```
CMD*HELP USER_OPTIONS
A list of user authorization options can be accessed by
typing a ? or a ?? after this message.  To obtain
definitions of individual options, type

HELP (SUPER) USER_OPTS option

where  option is any of the following:

ACCESS, BANNERTEXT, BILLING, BUDGET, CPROC, EXPIRE, FACCOUNT,
FEBILLING, FEDBACCN, FEMACCTMEM, FEMFPRG, FEMINTS,
FEMMEMORY, FEMTIME, FEPPRIVILEGE, FEPRIVILEGE, FEPSEUDO,
FERESOURCES, HSET, KEY, LAST_CPROC, MEMORY, NATIVEL, OUTPUTPRIO,
PASSWORD, PPRIVILEGE, PRIOB, PRIVILEGE, PROFILE, PSEUDO,
QUAN, RESOURCES, S_ACCOUNTING, SERVICES, SETUP, STEPACCNT,
TIME, WSN
CMD*HELP USER_OPTS BANNERTEXT
BANNERTEXTn={'[string]'[,user_alterable]|,user_alterable}
                Specifies the text of a user banner field and/or the alterable
                attribute of the text in the field, where:

                n    is a numeric value, 1 through 9, that identifies
                the user text field.

                'string'    is 0 to 80 ASCII printable characters that
                are the text for the field.  If string is null, the
                current contents of the text field will be deleted.

                ,user_alterable    is either:

                ,A[LTERABLE]    specifies that the user can modify the
                contents of the text field (via the IBEX command LET).
                ALTERABLE is the default.

                ,U[NALTERABLE]    specifies that the user cannot
                modify the contents of the text field.
```

Requesting Syntax Information After an Error Diagnostic

Once a line entry has been started in response to any SUPER prompt, the SUPER user can request help by entering a single or double question mark. SUPER will list the next expected entry on the line. For example:

```
CMD*CREATE ABC002,002ROGERS
OPT*HSET=?

     Syntax error
OPT*?
Acceptable input here is:
  an alphabetic name (plus '0123456789:$').
  #     DP
```

SUPER displays a list of the kinds of input it is expecting: either an alphabetic name (as qualified) or # or DP. (In fact, DP would be followed by # and then an alphabetic name, and # would be followed by an alphabetic name. Entering the expected data and then another question mark would reveal this.) The line can then be reentered with the correct value.

If a line is entered in response to any SUPER prompt and SUPER detects a syntax error, SUPER will diagnose the error. The SUPER user can request help by entering a single or double question mark. For example:

```
CMD*CREATE ABC002,002ROGERS
OPT*ABCDEF

     Syntax Error
OPT*?
Acceptable input here is:
  one or more blanks (and/or a "comment").
  the end of the command (possibly with a "comment").
  *S/_ACCOUNTING          ?           ??          AC/CESS     BANNERTEXT
  BA/TNUM     BI/LLING    BU/DGET     CP/ROC      E/ND        EX/PIRE
  FA/CCOUNT   FEBI/LLING  FECX/TMEM   FEDB/ACCN   FEMA/CCTMEM
  FEMF/PRG    FEMI/NTS    FEMME/MORY  FEMTI/ME    FEPP/RIVILEGE
  FEPR/IVILEGE            FEPS/EUDO   FERE/SOURCES             HEL/P     HS/ET
  KEY         L/AST       ME/MORY     NA/TIVEL    OU/TPUTPRIO
  PA/SSWORD   PP/RIVILEGE             PRIO/B      PR/IVILEGE  PRO/FILE
  PS/EUDO     QU/AN       Q/UIT       RE/SOURCES  SER/VICES   SET/UP
  ST/EPACCNT  TI/ME       WS/N
DPT*
```

SUPER displays a list of the keywords that can be entered. (In this case a list of the user authorization field names, plus the HELP single and double question marks and the keyword END. SUPER also explains that blanks can precede the next part of the entry, that a quoted comment can be entered, or that the response can be a null line ("the end of the command"). SUPER indicates the minimum portion of the keyword by terminating it with a slash.) The line can then be reentered with the correct value.

Listing User Authorization Records

The SUPER command LIST is used to list user authorization records. The user can list:

- an entire user authorization record.
- selected fields in a user authorization record.
- the default record (either the system or project default record) for a specified user.

The LIST command is entered, and then the option desired is specified. To request listing of an entire record, the following is specified:

```
CMD*LIST ABC001,001SMITH
OPT*ALL                    <— Requests a listing of the entire record.
OPT*
```

A user authorization record similar to the one pictured earlier in this module is listed.

The options that can be specified to request selected fields in a user authorization record are:

```
*S[_ACCOUNTING]        FEH[ANDLER]            OU[TPUTPRIO]
AC[CESS]               FEMA[CCTMEM]           PA[SSWORD]
ALT[ERABLE]            FEMI[NTS]              PPRIV[ILEGE]
BAN[NER]               FEMME[MORY]            PRIO[B]
BA[TCH]                FEMTI[ME]              PRIV[ILEGE]
BA[TNUM]               FEPP[RIV]              PRO[FILE]
BI[LLING]              FEPR[IV]               PS[EUDO]
BU[DGET]               FEPS[EUDO]             QU[AN]
CP[ROC]                FERE[SOURCE]           RE[SOURCES]
EX[PIRE]               FEU[SER]               SER[VICES]
FA[CCOUNT]             G[HOST]                SET[UP]
FE                     HS[ET]                 ST[EPACCNT]
FEBI[LLING]            KEY                    TI[ME]
FECG                   L[AST]                 TP
FECX[TMEM]             ME[MORY]               W[SN]
FEDB[ACCN]             N[ATIVEL]
FEG[HOST]              O[NLINE]
```

Most of the user authorization LIST options are the same as or are similar to user authorization record field names. However, note that when listing banner text the option specified is BANNER not BANNERTEXT. These options can be entered one per line or multiple options can be entered on one line, separated by semi-colons. For example:

```
CMD*LIST ABC001,001SMITH
OPT*ACCESS;PROFILE;WSN
OPT*
```

will cause a listing similar to the following:

```
ABC001,001SMITH

   PROFILE      WSN
    TTY          LOCAL

   ACCESS
 B: YES
 G: NO
 O: YES
 T: NO
 F: NO


 .. 1 users listed.
```

The user authorization LIST options FE, GHOST, ONLINE, and TP do not themselves request a listing. They are used as listing qualifiers to limit listing by mode. For example:

```
CMD*LIST ABC001,001SMITH
OPT*ONLINE
OPT*BILLING;MEM;TIME;QUAN;PRIOB
```

results in the following listing:

```
ABC001,001SMITH

   BILLING  MEM MAX  MEM DEF  TIME MAX  TIME DEF  QUAN      PRIOB
 O: 1        511      128      9999      9999      0         0


 .. 1 users listed.
```

The specification:

```
  CMD*LIST DEFAULT
```

will cause the system user default record (pictured in module 4–1) to be listed.

Administering User Authorizations

SUPER features a number of commands to enable the project administrator or system manager to maintain authorization records once they are created:

- the MODIFY command allows the authorized administrator to change the contents of the record.
- the REMOVE command allows the authorized administrator to delete an authorization record.
- the DELETE USERS command allows the project administrator to delete all user authorization records that that administrator is authorized to maintain. In the case of a project administrator, that is all records in the current project and all subordinate projects. (In the case of the system manager, this command must not be used. Its effect will be to develop all records except the system-built records (see Module 4-1).

The following is an example of using the MODIFY and REMOVE commands:

```
CMD*MODIFY ABC002,002ROGERS
OPT*PRIV
SUB*MAXM B,O,G        <--  Sets the MAXimum Memory allocation privilege for
                           batch, online, and ghost modes.
SUB*
OPT*PPRIV
SUB*EFT B,O,G         <--  Sets the EFT processor privilege for batch,
                           online, and ghost modes.
SUB*
OPT*SERVICE
SUB*MAXJOB B=9,O=9    <--  Sets the highest priority the user may assign to
                           a batch job at 9 in batch and online mode.
SUB*
OPT*
CMD*REMOVE ABC003,003GOMEZ(FACCOUNT)
                           Note that if FACCOUNT is not specified as
                           part of the REMOVE command.  The associated
                           file management account will not be removed.
                           The PIG processor will have to be invoked
                           to remove the file management account.
```

Module 5-1

Defining and Using a CP-6 Network

A CP-6 network consists of some combination of the following three types of nodes:

1. Host nodes. A host node is a CP-6 host system.

2. Local FEP nodes. A local FEP node is connected to a host via a coupler.

3. Remote FEP nodes. A remote FEP is connected to a local FEP through an HDLC line.

This module describes how to define and connect the nodes of a CP-6 network and how to manage the network once it has been created.

If the network is to consist only of host nodes and local FEPs, the network can be defined via the TIGR processor. If the network is to consist of host nodes, local FEPs and remote FEPs the process of defining and maintaining the network involves the use of several processors as follows:

Processor	Function	Command
NETCON	Defines the node number for remote FEPs and the node name for local and remote FEPs.	DEFINE
	Configures the HDLC channels for both local and remote FEPs and associates link and virtual circuit information with channel information.	CONFIG
	Defines the boot image.	SET
PIGETTE	Writes the boot diskette.	
SUPER	Defines physical link to network	DEF LINK
	Defines the link destination	DEF VIRTUAL CIRCUIT
TIGR	Defines the path connecting local FEPs to the system.	FEP

Creating Local FEPs on a Network

The TIGR processor is used to define the path connecting FEPs to the system. The FEP command:

- assigns a FEP a number
- defines the Input/Output Multiplexer (IOM) to which the FEP is connected. (IOM ports are defined via the TIGR processor IOM command.)
- specifies the channel over which the FEP is connected to the host.

- and can be used to specify other FEP attributes (i.e., the size of the input and output circular queues and the initialized partition status of the FEP.)

The TIGR processor MON command is used to establish a maximum number of nodes on the system.

There is only one default handler defined as part of the boot image for local FEPs: the ASYNC handler. The NETCON processor must be used to define handlers other than the ASYNC handler for a local FEP. The NETCON processor may also be used if the system manager wishes to assign a name (as well as a number) to the local FEPs in the system.

The following example illustrates use of the TIGR command to define local FEP 9 and use of the MON command to define a maximum of 50 nodes in a network:

```
FEP NAME=FEP 9,IOM#=0,CHAN=33,OUTQSZ=4;INQSZ=2
MON  ;                          |_____________|
 .                                      |
 .                                      |__Output and input queues
 .
 NODES=50,;
 .
 .
 .
```

These TIGR commands that help configure a network are included in a TIGR device configuration deck (see Module 3-1).

Adding Remote FEPs to a Network

Once the local FEPs are defined, the remote FEPs can be added to the system. Each remote FEP must be connected to a local FEP.

The physical connection between a remote FEP and a local FEP must be established via HDLCX25 linkage in both directions: that is, a remote FEP must be connected to a local FEP via an HDLCX25 physical link, and a local FEP must be connected to a remote FEP via an HDLCX25 physical link. The HDLCX25 linkage is defined via the SUPER CREATE VIRTUAL CIRCUIT command and the HDLCX25 handler must be explictly included in the boot image for both the local and remote FEPs.

The sequence of steps for attaching a remote FEP to a network is:

1. Use NETCON to define the remote nodes and node names.
2. Use SUPER to set up links and virtual circuits. That is, the SUPER processor is used to:
   - Create the link profiles required to connect a local FEP and a remote FEP.
   - Create virtual circuit profiles, one to be associated with each of the links that are to be creates.
   - Create the links between the local and remote FEPs.

- Create a virtual circuit for each link.

3. Use NETCON to:

   - associate FEPs, remote FEPs and remote FEP links with the SUPER- defined link names by using:

     - the CONFIG command to configure the channels.

     - the DEF LINK command to establish a communication channel for the remote FEP to use before communication with the host is establishd.

   - set the boot information for the remote FEP(s) and the local FEPs. being sure to include for both an HDLCX25 link.

4. Use PIGETTE to write the boot diskette and initialize a blank diskette for dumps.

5. Boot the system.

Defining the Remote Nodes and Node Names

Each node in the network has a unique number from 0–255 and a one to eight character name associated with it. Either the node number or the node name can be used in NETCON to identify the node. The node number for local FEPS is defined via the TIGR processor (see the section Defining Local FEPs, above). The node number for host nodes and for remote FEP nodes are defined via the NETCON processor DEFINE command. The names of all nodes are defined via the NETCON processor DEFINE command.

A remote FEP must be defined via the NETCON processor DEFINE NODE command before it can be booted.

In the following example, the NETCON processor is invoked to define a remote FEP, node 12 and assign it the node name L6XII.

```
!NETCON
NETCON C00 HERE
*DEFINE NODE=12,NODENAME=L6XII
```

The next section describes how the SUPER processor is then invoked to define the linkage for this node, which will be connected to local FEP 9.

Setting Up Links and Virtual Circuits

A link is a communication line that uses HDLC protocol and is used to connect the nodes in a network. A link constitutes the physical and the frame levels of an X.25 connection. A virtual circuit is a logical connection at a higher level than a link. Several virtual circuits can be multiplexed on to one link.

The links in the network must be defined via the SUPER processor CREATE LINK and CREATE VIRTUAL CIRCUIT commands before a remote FEP can be booted. For remote FEPs, one link must be defined for the local FEP and one link must be defined for the remote FEP. At least one virtual circuit must be defined on one of the end points of the linkage between two FEPs.

Once the SUPER processor has been used to create the HDLCX25 linkage, the NETCON processor is used to configure channels, local FEPs, remote FEPs and remote links via the link names assigned via the CREATE LINK and CREATE VIRTUAL CIRCUIT commands.

The SUPER processor CREATE LINK command creates link definitions. In the following example, links and virtual circuits are set up to link local FEP 9 to remote FEP 12. Note that links must be set up at both ends — at the local FEP and at the remote FEP ends.

```
!SUPER

*** CP-6 SUPER C00 ***

CMD*CREATE PROFILE LINK12PRO LINK  <— Creates a link profile for the
                                      remote FEP link (link 12).
                                      LINK profile options are specified
                                      in SUPER option mode.  The LINK
                                      profile options that can be
                                      specified are defined in the
                                      LINK Profile Option Table.
OPT*FRAME=512                      <— The maximum number of data bytes
                                      for the X.25 frame will be 512
                                      instead of the default of 128.
                                      Note that in this example the
                                      same frame size will be used for
                                      both end of the link and for the
                                      virtual circuits in the link.
                                      If different frame sizes are
                                      specified, the frame size used
                                      is determined by negotiating down.
OPT*CIRCUITS=10                    <— Specifies that 10 virtual circuits
                                      may be operational on the link at
                                      one time instead of the default
                                      of 1.
OPT*DEFAULT RESPONSE TIMER=0       <— Specifies that there is to be
                                      no wait time if an incoming
                                      call does not specify explicitly
                                      a response time for this circuit.
OPT*DEFAULT PACKET SIZE=512        <— Specifies that, if an incoming
                                      call does not explicitly specify
                                      a receive size, 512 bytes is to
                                      be used as the receive size
                                      instead of the default of 128.
OPT*MODE=DCE                       <— In this example, no public
                                      data network is used.  Therefore,
                                      one end in the connection must
                                      be declared to be DCE.
OPT*RETRAN=5                       <— Specifies that, before moving
                                      to alternative action, retransmission
                                      should be attempted five times
                                      instead of the default of 20
                                      times.
OPT*END
```

(cont. next page)

```
CMD*CREATE PROFILE LINK9PRO LINK    <— Creates another link profile for
OPT*FRAME=512                           the local FEP link (link 9).
OPT*CIRCUITS=10                         Note that all options specified
OPT*DEFAULT RESPONSE TIMER=0            for the local FEP link profile
OPT*MODE=DTE                            match the options specified for
OPT*RETRAN=5                            the remote FEP link profile
OPT*END                                 except that the mode for the
                                        local FEP link is DTE instead
                                        of DCE.

  The next series of SUPER commands and profile options create two
  virtual circuit profiles that are named so that they can be associated
  with the local and remote FEP links that are to be created.

CMD*CREATE PROFILE LINK12VCPRO VIRTUAL CIRCUIT
                                        Creates a profile for a virtual
                                        circuit for the remote FEP
                                        (required to enable HDLCX25
                                        communication links).
OPT*RECEIVE SIZE=512                <— Specifies that the maximum
                                        data size of packets from the
                                        call recipient is 512 bytes
                                        (the same as the frame size
                                        for the link).
OPT*SEND SIZE=512                   <— Specifies that the maximum data
                                        size of packets sent to the call
                                        recipient is also 512 bytes.
OPT*RESPONSE TIMER=0                <— Specifies that there is to be no
                                        wait after deciding to send an
                                        explicit flow control packet.
                                        The two kinds of flow control
                                        packets are positive acknow-
                                        ledgements (RRs) and negative
                                        acknowledgements (RNRs).
OPT*RECEIVE WINDOW=7                <— Specifies that the receive
                                        window size is 7 packets
                                        rather than the default of 2.
OPT*SEND WINDOW=7                   <— Specifies that the transmit
                                        window size to the call recipient
                                        is to be 7 rather than the default
                                        of 2.
OPT*RESPOND COMPLETE=NO             <— Specifies that any complete data
                                        packet sequence is not to be
                                        acknowleged immediately.  A
                                        complete data packet sequence
                                        is either a single packet that
                                        is not continued, or a sequence
                                        of continued packets that includes
                                        a final, non-continued packet.
OPT*RESPONSE DELAY=4                <— Specifies that 4 unacknowledged
                                        packets can be received before
                                        an explicit acknowledgement
                                        packet is generated.
OPT*TYPE=PRIMARY                    <— Specifies the virtual circuit
                                        usage type.
OPT*END
```

(cont. next page)

```
CMD*CR PRO LINK9VCPRO VIR CIR      <— Creates the profile for a virtual
OPT*RECEIVE SIZE=512                   circuit to be associated with the
OPT*SEND SIZE=512                      local FEP (LINK9).
OPT*RESPONSE TIMER=0
OPT*RECEIVE WINDOW=7
OPT*SEND WINDOW=7
OPT*RESPONSE COMPLETE=NO
OPT*TYPE=SECONDARY
OPT*END

CMD*CREATE LINK LINK12             <— Creates a link definition.  The
                                       LINK name can be 1-8 characters.
                                       Specifying the LINK command
                                       initiates SUPER option mode.
                                       Two options ADDRESS and PROFILE
                                       must always be specified.
OPT*PROFILE=LINK12PRO              <— Specifies the LINK profile for this
                                       link.  The profile specified must
                                       already have been created through a
                                       CREATE PROFILE...LINK command.
OPT*ADDRESS=90                     <— Specifies the address inserted in
                                       the calling address field of out-
                                       going call packets.  The value
                                       specified (90) is arbitrary.
                                       If a public data network is
                                       addressed, the PDN address would
                                       be specified.
OPT*END

CMD*CREATE VIRTUAL CIRCUIT 1 FOR LINK LINK12
                                       Creates a virtual circuit defini-
                                       tion for the remote FEP. Specifying
                                       the CREATE VIRTUAL CIRCUIT command
                                       initiates SUPER option mode.
OPT*PROFILE=LINK12VCPRO                For network nodes, the PROFILE and
OPT*ADDRESS=91                         ADDRESS options (as specified for
                                       the LINK example) are specified,
                                       and the DESTINATION option must be
                                       specified as well.  The address
                                       is the call packet called address.
OPT*DESTINATION=L6IX               <— Identifies the node name of the
                                       local FEP (9) (as defined via
                                       a NETCON DEFINE NODE command)
                                       as the circuit's destination.
                                       (In the case of a remote FEP,
                                       DESTINATION must point to a local
                                       FEP.)
OPT*END

CMD*CREATE LINK LINK9              <— Creates a link definition for
                                       the local to which the remote FEP
                                       is to be connected.

OPT*PROFILE=LINK9PRO               <— Specifies the LINK profile for this
                                       link.  The profile specified must
                                       already have been created through a
                                       CREATE PROFILE...LINK command.
OPT*ADDRESS=91                     <— Specifies the address inserted in
                                       the calling address field of out-
                                       going call packets.  Note that
                                       this address is the same as the
                                       link address for VIRTUAL CIRCUIT 1
```

(cont. next page)

```
                                        for LINK 12.  When LINK 9 receives
                                        a call packet from LINK 12, the
                                        addresses will match.
OPT*END
CMD*CREATE VIRTUAL CIRCUIT 1 FOR LINK LINK9
                                        Creates a virtual circuit defini-
                                        tion for the local FEP.

OPT*PROFILE=LINK9VCPRO                  For network nodes, the PROFILE and
OPT*ADDRESS=90                          ADDRESS options (as specified for
                                        the LINK example) are specified,
                                        and the DESTINATION option must be
                                        specified as well.
OPT*DESTINATION=L6XII                <- Identifies the virtual circuit's
                                        destination.
OPT*END
CMD*END
```

The SUPER processor MODIFY, REMOVE and LIST LINK commands are used to modify, remove and list link definitions. The SUPER: Communications Management section of the System Support Reference manual contains descriptions of the these commands. Presented below are some examples of the types of LINK definition and LINK profile displays that can be listed through the SUPER processor LIST LINK and LIST PROFILE commands.

```
!SUPER

*** CP-6 SUPER C00 ***

CMD*L LINK LINK9                     <- Lists the LINK definition for
                                        local FEP 9.
LINK LINK9  PRO=LINK9PRO  ADDRESS=91000000000000

  CIRCUIT  PROFILE     CUG  DEST    ADDRESS
   1       LINK9VCPRO  00   L6XII    90000000000000

CMD*L LINK LINK12                 <- Lists the LINK definition for
                                     remote FEP 12.
LINK LINK12  PRO=LINK12PRO  ADDRESS=90000000000000

  CIRCUIT  PROFILE     CUG  DEST    ADDRESS
   1       LINK12VCPRO 00   L6IX     91000000000000

CMD*L PRO LINK9PRO                <- Lists the LINK profile for
                                     local FEP 9.
OPT*ALL                           <- Values for all LINK Profile
                                     options (see the LINK Profile
                                     Options table) are requested.  A
                                     display of specific options can be
                                     requested instead.
LINK9PRO          Type = LINK

   CALLS            CIRCUITS         DFLT PACKET SET  DFLT_RESPOND
    ALL              10               512              0
```

(cont. next page)

```
   DFLT WINDOW       FRAME             LCGN              MAX WINDOW
    2                 512               0                 2

   MODE              RESPONSE DELAY    RETRANSMISSIONS   REVERSE
    DTE               1                 5                 Yes

   TIMEOUT           WINDOW
    3                 7


CMD*L PRO LINK12PRO              <-- Lists the LINK profile for
                                     remote FEP 12.
OPT*ALL                          <-- Requests a display of values for
                                     all LINK profile options for the
                                     remote FEP.
LINK12PRO          Type = LINK

   CALLS             CIRCUITS          DFLT PACKET SET   DFLT_RESPOND
    ALL               10                512               0

   DFLT WINDOW       FRAME             LCGN              MAX WINDOW
    2                 512               0                 2

   MODE              RESPONSE DELAY    RETRANSMISSIONS   REVERSE
    DCE               1                 5                 Yes

   TIMEOUT           WINDOW
    3                 7


CMD*L PRO LINK12VCPRO            <-- Lists a VIRTUAL CIRCUIT profile
                                     for a virtual circuit defined for
                                     the remote FEP.
OPT*ALL                          <-- Requests a display of values for
                                     all profile options for the
                                     remote FEP virtual circuit (see
                                     the Virtual Circuit Profile Options
                                     table).  A display of specific
                                     options can be requested also.
LINK12VCPRO         Type = VIRCIR

   BACKLOG           COST              DELAYS            RECEIVE_SIZE
     5                100               60                512

   RECEIVE_THR       RECEIVE_WND       RESP_TO_CMP       RESPONSE DELAY
    NONE              7                 No                4

   RESP_TIMER        RETRYS            REVERSE           SEND_SIZE
    0                 10                No                512

   SEND_THR          SEND_WINDOW       TIMEOUT           TYPE
    NONE              7                 200               PRIMARY
```

Table 6. LINK Profile Options

Option	Description
CA[LLS]={NONE|X25|X29|ALL}	Specifies the usage of a link for incoming calls. NONE indicates that no incoming calls are allowed. X25 indicates that all incoming calls except X29 calls are allowed. X29 indicates that only X29 calls are allowed. ALL is a combination of X25 and X29. The default is ALL.
CIR[CUITS]=value	Specifies a value (1–255) that is the maximum number of virtual circuits that may be operational at one time on a link. The default is 255.
DEF[AULT] PACK[ET SIZE]={128|256|512|1024}	Specifies the packet size to use if an incoming call does not explicitly specify a receive size. The default is 128.
DEF[AULT] RES[PONSE TIMER]=value	Specifies the number of second (0–255) to wait before sending a packet-level acknowledgement for circuits defined on the link. This option specifies the wait time to be used if an incoming call does not explicitly specify a response time for this circuit.
DEF[AULT] WIN[DOW]=value	Specifies the packet window size (2–7) to use if an incoming call does not explicitly specify a Receive Window. The default is 2.
FR[AME]={128|256|512|1024}	Specifies the maximum number of data bytes that an X.25 information frame can contain. The default is 128.
LCGN=value	Specifies the logical channel group number (0–15) for outgoing calls. The default is 0.
MAX[IMUM] WIN[DOW]=value	Specifies the maximum send window (2–7) on incoming calls for circuits on this link. When making or receiving calls, if the send window size parameter exceeds this value, this value is used. The default is 7.

Table 6. LINK Profile Options (cont.)

Option	Description
RESP[ONSE] DEL[AY]=value	Specifies the number (0–7) of unacknowledged information frames that should be received before an explicit acknowledgement frame is generated. A value of 0 indicates that explicit acknowledgement frames should only be generated in response to received commands with the poll bit set. The default is 1.
RET[RANSMISSION]=value	Specifies the number of retransmissions (0–255) that should be done before moving to an alternative action. This is CCITT parameter N2. A value of 0 indicates an infinite number of retransmissions should be done. The default is 20.
REV[ERSE][={Y[ES]|N[O]}]	Specifies whether or not to accept incoming calls that specify reverse charges. The default is YES.
TIME[OUT]=value	Specifies the time interval in seconds (1–255) that should expire after transmission of a frame before corrective action should be taken because a response was not received. This is CCITT parameter T1. The default is 3.
WI[NDOW]=value	Specifies the frame transmit window size (1–7). This is CCITT parameter K. The default is 7.

Table 7. VIRTUAL CIRCUIT Profile Options

Option	Description
DEL[AYS]=value	Specifies the amount of time in seconds (0–255) before an attempt is made to reconnect a VC that has been cleared for reasons other than a higher level initiated clear. A value of 0 indicates that no delay should be used. The default is 60.
MAXV[IRCIR]=value	Specifies the maximum number (0–255) of virtual circuits to be used in a connection between a pair of nodes. If MAXVIRCIR is less than MINVIRCIR, MINVIRCIR is used. The default is 255. If only one virtual circuit is to be defined, efficiency will be gained if MAXVIRCIR is set to 1.
MINV[IRCIR]=value	Specifies the minimum number (0–255) of virtual circuits to be used in a connection between two nodes. The default for MINVIRCIR is 1. MINVIRCIR and MAXVIRCIR values are specified or defaulted for each virtual circuit between two nodes. If multiple virtual circuits are defined between a pair of nodes, the values for all MINVIRCIR/MAXVIRCIR for all the virtual circuits defined between the two nodes must be the same. Otherwise, the MIN/MAX range selected will be somewhat unpredictable.
MODE={DTE|DCE}	Determines the command and response addresses for frames, and affects logical channel number assignment and the outcome of a call collision. If a connection involves a PDN, DTE should be specified. If no PDN is involved, one end must specify DTE, and the other end must specify DCE. The default is DTE.
REC[EIVE] SIZ[E]={128|256|512|1024}	Specifies the maximum data size of packets from the call recipient. If this value is larger than the frame size of the LINK profile, the frame size value is used. The default is 128.
REC[EIVE] THR[OUGHPUT]={NONE|75|150|300|600|1200|2400|4800|9600|19200|48000}	Specifies the throughput class of the VC from the call recipient. This option is meaningful only if the connection is made through a PDN. The default is NONE.

Table 7. VIRTUAL CIRCUIT Profile Options (cont.)

Option	Description
REC[EIVE] WIN[DOW]=value	Specifies the transmit window size (2–7) for the call recipient. The default is 2.
RES[POND TO] COM[PLETE][={YES|NO}]	Specifies whether (YES) or not (NO) any complete data packet sequence should be acknowledged immediately, regardless of the response delay value. The default is YES.
RESP[ONSE] DEL[AY]=value	Specifies the number of unacknowledged packets (1–7) that should be received before an explicit acknowledgement packet is generated. If this value is larger than the receive window, the receive window value is used. The default is 1.
RES[PONSE] TI[MER]=value	The number of seconds (0–255) to wait after deciding to send an explicit flow control message (based on response delay and response to complete). If an implied flow control message is sent before the time expires, no flow control message needs to be sent. The default is 3.
RETR[YS]=value	Specifies the number of consecutive unsuccessful call packets (0–10) that will be transmitted before the VC is declared lost. A value of 0 indicates no maximum number of retries. The default is 10.
RE[VERSE][={Y[ES]|N[O]}]	Specifies whether or not reverse charging should be specified in the call packet. The default is NO.
SEN[D] SIZ[E]={128|256|512|1024}	Specifies the maximum data size of packets sent to the call recipient. The default is 128.
SEN[D] THR[OUGHPUT]={NONE|75|150|300|600|1200|2400|4800|9600|19200|48000}	Specifies the throughput class of the VC to the call recipient. This option is only meaningful if the connection is made through a PDN. The default is NONE.

Table 7. VIRTUAL CIRCUIT Profile Options (cont.)

Option	Description
SEN[D] WIN[DOW]=value	Specifies the transmit window size (2–7) to the call recipient. The default is 2.
TIME[OUT]=value	The timeout value (3–255) used for timing out responses to call reset and clear packets. The default is 200.
TY[PE]={P[RIMARY]|S[ECONDARY]|B[ACKUP]}	Specifies usage type of the VC. PRIMARY specifies that the VC should always be used. SECONDARY is intended to meet peek loads. BACKUP is intended as a temporary replacement for primary and secondary VCs. This option applies only to VCs associated with a multilink network connection. The default is PRIMARY.

Configuring the Network

NETCON determines the channels that can be supported and thereby determines the kinds of handlers that can be downline loaded.

A channel table is created in each FEP. The channel table contains one entry for each channel connected to the FEP. Each entry contains flags and parameters that specify how a given channel is to be used. These values for the channel table are defaulted. (The DEFAULT command is used to set and change the default line configuration criteria for a handler in the FEP.) The system manager can change the defaults. The channel table entry also contains the current channel status (i.e., availability) data. These values can also be changed by any of several NETCON commands. Refer to the section Maintaining a Network, below, for more information on changing channel table defaults and changing channel status.

After the NETCON processor has been used to define remote node numbers and all node names and the SUPER processor has been used to define the links and virtual circuits, the NETCON processor is again invoked to configure the system. In the following example, the remote FEP defined as node 12 with the node name L6XII is configured.

```
!NETCON
NETCON C00 HERE
*SEL N=12                        <— Specifies which node is to be configured.
*DEFINE LINK .5600               <— Specifies that channel 5600 is to be
                                    used to communicate with the network
                                    before establishing communication
                                    with the host.
*CONFIG .5600 LOGON='LINK12',ENABLE=YES,REENABLE=YES
                                    Specifies the configuration for channel
                                    5600 (the channel on FEP 12 that is one
                                    of the end points of the 9<->12 link).
*SEL N=9                         <— Specifies node 9, the local FEP, is to be
                                    configured.  NO DEFINE LINK command is
                                    required on the local side.
*CONFIG .5400 LOGON='LINK9',ENABLE=YES,REENABLE=YES
                                    Specifies the configuration for channel
                                    .5400 (the channel on FEP 9 that is one
                                    of the end points of the 9<->12 link).
```

Note that although there is only one link being configured, the two ends of the link (that is the local end and the remote end) must be configured.

The following figure indicates how the FEP linkage is configured.

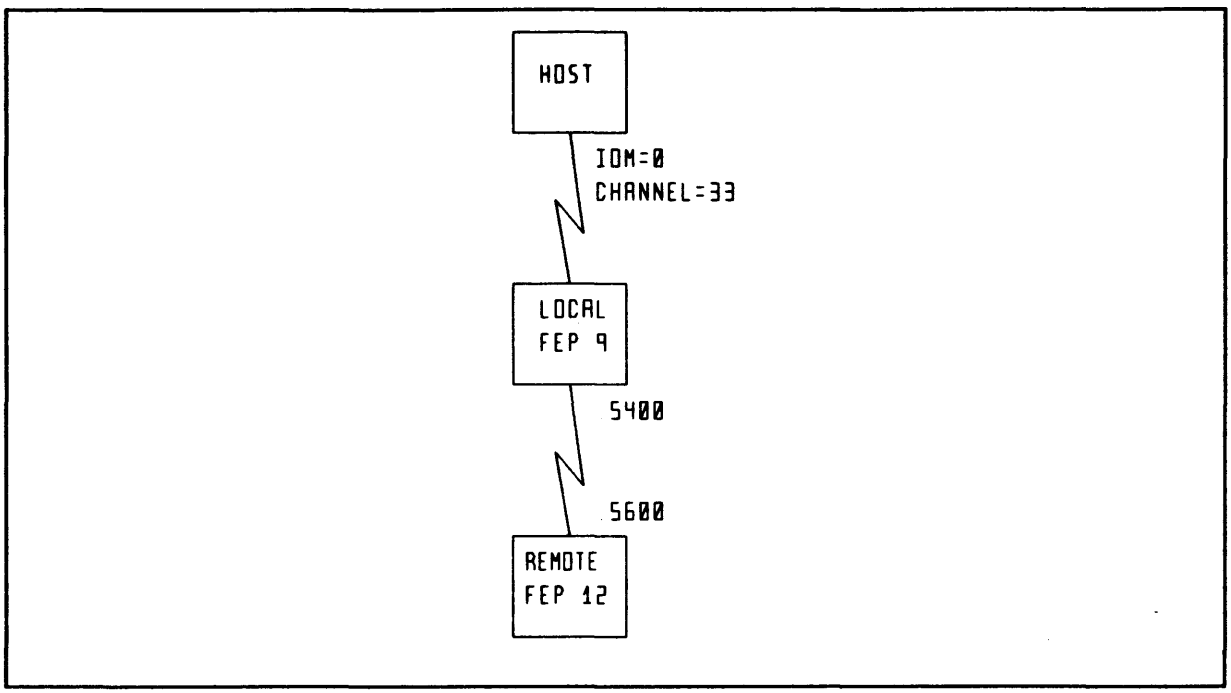


Figure 5. Sample Network Linkage

Setting the Boot Information

NETCON maintains the boot information about each FEP and each FEP's handlers. Through NETCON, control parameters can be set and changed that will influence performance characteristics for both the FEP and the handlers in the FEP.

For a local FEP, the handler information need be set only if the handler is a special handler. For a remote FEP, all handlers must be explicitly defined before the remote FEP can be booted. Handler information is configured via the NETCON processor SET BOOTINFO command.

In the following example, the boot image is created for the remote FEP and for the local FEP.

```
*SEL N=12                      <— Selects the remote FEP for further
                                  processing.  Note that FEP 12 is going
                                  to have a boot diskette.  It will not
                                  be booted over a coupler from a host
                                  boot image.
*SET BOOTINFO                  <— The boot image information is defined.
Monitor fid (M:FEP.:SYS)       <— Identifies the LCP6 monitor run unit.
Number of handlers 4           <— Specifies the number of handlers to boot.
Handler name NODEADMN          <— Specifies the name of the first handler.
Handler fid NODEADMN.:SYS      <— Specifies the fid of the first handler.
Handler name HDLCX25           <— Specifies the name of the second handler.
Handler fid HDLCX25.:SYS       <— Specifies the fid of the second handler.
Handler name ASYNC             <— Specifies the name of the third handler.
Handler fid ASYNC.:SYS         <— Specifies the fid of the third handler.
Handler name UNITREC           <— Specifies the name of the fourth handler.
Handler fid UNITREC.:SYS       <— Specifies the fid of the fourth handler.
Library account :SYS           <— Specifies the Library Account.

*SEL N=9                       <— Selects the local FEP for further
                                  processing.
*SET BOOTINFO                  <— The boot image information is defined.
Monitor fid (M:FEP.:SYS)       <— Identifies the LCP6 monitor run unit.
Number of handlers 5           <— Specifies the number of handlers to boot.
Handler name NODEADMN          <— Specifies the name of the first handler.
Handler fid NODEADMN.:SYS      <— Specifies the fid of the first handler.
Handler name COUPLER           <— Specifies the name of the second handler.
Handler fid COUPLER.:SYS       <— Specifies the fid of the second handler.
Handler name HDLCX25           <— Specifies the name of the third handler.
Handler fid HDLCX25.:SYS       <— Specifies the fid of the third handler.
Handler name ASYNC             <— Specifies the name of the fourth handler.
Handler fid ASYNC.:SYS         <— Specifies the fid of the fourth handler.
Handler name BISYNC            <— Specifies the name of the fifth handler.
Handler fid BISYNC.:SYS        <— Specifies the fid of the fifth handler.
Library account :SYS           <— Specifies the Library Account.
```

Note that in interactive mode the system manager is prompted to enter the monitor fid, the number of handlers, and, for each handler, the handler name and the handler fid.

Writing the Boot Diskette

Before a remote FEP can be booted, the PIGETTE processor is used to create a bootstrap diskette for the remote FEP connected to a local FEP. The boot diskette contains the LCP-6 monitor, and temporary diagnostic dump areas, as well as the boot image for the remote FEP. The PIGETTE processor uses records in the :NETCON file to describe the boot image. The diskette is a double-sided, double-density, 5-1/4 inch diskette.

This diskette must be loaded at the remote FEP. This diskette will also be read when a software crash is detected by a remote FEP to recover the remote FEP's monitor.

When a remote FEP detects a software crash, the remote FEP automatically dumps information to its diskette. This information will be sent to the FROG processor.

The PIGETTE processor BUILD command is used to create a boot image for a remote FEP. To create the boot image:

- The NETCON processor DEFINE command must be used first to define the remote FEP.
- The PIGETTE processor is then invoked. When it is, the VOLINIT command must be used first to initialize the boot diskette.
- Then, the PIGETTE processor BUILD command is used to build the boot image on the diskette.

For example, to build the boot diskette for FEP 12 (the remote FEP), the system manager does the following.

```
!PIGETTE
C00 PIGETTE here at 15:57:10.28 on 08/24/84

Oink: VOLINIT FEP 1 DRIVE 1 (TYPE=RFEP)
 ..VOLINITing
VOLINIT complete.
Oink: BUILD 12 OVER FEP 1 DRIVE 1
  ..COPYing
COPY complete..
Oink: L FEP 1
DIskette on FEP:  1 Drive: 1
Created on 15:58 08/24/84
Built for use on FEP 12
Oink: END
PIGETTE exiting.
```

These commands load and initialize a diskette in drive 1 of FEP 1, and create a boot diskette that will eventually be used in FEP 12. Note the use of the PIGETTE processor LIST command to display information about the diskette.

The PIGETTE processor MOVE command is used to move data between an actual diskette in a FEP drive and a virtual diskette stored in a CP-6 keyed file. This use of PIGETTE is described in the Section PIGETTE: Diskette FEP Initialization in the System Support Reference Manual.

Displaying NETCON Information

The NETCON processor DISPLAY command can be used to display different kinds of information about the network. Some of the more common forms of the NETCON processor DISPLAY command, and the resulting displays are listed below. Refer to the Section "NETCON: Network Configuration" in the System Support Reference Manual for a more complete treatment of the DISPLAY command syntax and for additional samples of DISPLAY output.

The DISPLAY NETWORK command displays a listing of the network nodes.

```
*DI NETWORK
Node#    Name     Type Control
   0   L66A       Host
   1   L6I        Fep L66B
   2   L6II       Fep L66B
   3   L6III      Fep L66B
   4   L6IV       Fep L66A
   5   L6V        Fep L66B
   6   L6VI       Fep L66C
   8   L6VIII     Fep L66A
   9   L6IX       Fep L66B
  10   L6X        Fep L66B
  11   L6XI       Fep L66B
  12   L6XII      Fep L66B
  13   L6XIII     Fep L66B
  15   L6XV       Fep L66B
  20   L66B       Me
  21   L66D       Host
  22   L66C       Host
  30   DVFEP      Fep L66A
  32   DVFEP      Fep L66A
```

The DISPLAY LINKS command displays link information (including the number of links for a node, the number of each link and the link line specification).

```
*SEL N=12
*DI LINKS

Link information for node L6XII
      Number of links = 1
         Link# 0 = .5600
```

The DISPLAY CONFIG command displays configuration information for the specified line on the selected node.

```
*DI CONFIG .5600
Line Configuration for Node# 12(L6XII    ) CHANNEL# .5600 Terminal id
      LINE SPEED:
           Auto Speed....D/C              Speed....0
      LOGON:
           Logon String....LINK12
           Echo Logon......D/C
           Profile.........None

      Flow control..........D/C
      Input.................D/C           Output.................D/C
      Salutation............D/C           Break Required.........D/C
      Remote................Term          Buffer Size............0
      Enable................Yes           Reenable...............Yes
      Read time out.........0
      Logon Time out.........0            Trans Proc Timeout.....0
      Hardwire..............No            Clocking...............D/C
      Drop DTR..............D/C           Kill on Host Down......No
      Disable on Host Down...No
      Resource code.........None         Attribute..............0
      Block.................0            Unblock................0
      Remote Debug..........D/C          Debug..................D/C

*DI CONFIG .5400
Line Configuration for Node# 9(L6IX     ) CHANNEL# .5400 Terminal id
      LINE SPEED:
           Auto Speed....D/C              Speed....0
      LOGON:
           Logon String....LINK9
           Echo Logon......D/C
           Profile.........None

      Flow control..........D/C
      Input.................D/C           Output.................D/C
      Salutation............No            Break Required.........D/C
      Remote................Term          Buffer Size............0
      Enable................Yes           Reenable...............Yes
      Read time out.........0
      Logon Time out.........0            Trans Proc Timeout.....0
      Hardwire..............No            Clocking...............D/C
      Drop DTR..............D/C           Kill on Host Down......No
      Disable on Host Down...No
      Resource code.........None         Attribute..............0
      Block.................0            Unblock................0
      Remote Debug..........D/C          Debug..................D/C
```

The DISPLAY BOOTINFO command displays FEP boot handler and library data for the specified node. For example:

```
*SEL N=9
*DI BOOTINFO
Boot Information for Node L6IX

Monitor Fid = M:FEP.:SYS
Number of Handlers = 5
Handler #1 Name = NODEADMN
Handler #1 Fid = NODEADMN.:SYS
Handler #2 Name = COUPLER
Handler #2 Fid = COUPLER.:SYS
Handler #3 Name = HDLCX25
Handler #3 Fid = HDLCX25.:SYS
Handler #4 Name = ASYNC
Handler #4 Fid = ASYNC.:SYS
Handler #5 Name = BISYNC
Handler #5 Fid = BISYNC.:SYS
Library account = :SYS
```

Booting the System

The boot process differs for local FEPs and remote FEPs. Further, the booting differs for a tape boot/hardware recovery and for a software recovery. Regardless of these differences (described in section 4 of the System Support Reference Manual), once contact is established via the link channel to a local or remote FEP, connection is established throughout the CP-6 network: all nodes are informed of the connect point.

For initial booting, an operator must always hardware boot a remote FEP. The remote FEP boot diskette is always placed in channel 400 at the remote FEP to perform the hardware boot.

Maintenance Through NETCON

Through NETCON, an online user can enable and disable lines and otherwise affect a line's availability, and can change FEP control parameters that will influence performance characteristics.

The NETCON commands used to modify the network and their functions are listed below:

Command	Change Function
CONFIG	Change line configuration parameters (i.e., the default flags and parameters in a FEP Channel Table that specify how the channel is to be used.
DELETE	Delete a node or pseudo resource.

DISABLE DISCONNECT ENABLE KILL	Change channel line availability.
DISPLAY	Display the current values of boot information, the network nodes, pseudo-resources, or NETCON controlled parameters.
PARTITION RETURN	Remove channels from system use and make them available for test and diagnostics, and return them to system use.
SELECT	Select a network node or handler for display/ change operations.
SET	Change NETCON controlled parameters on a per host, per FEP or per handler basis.

Note that the CONFIG, DISABLE, DISCONNECT, ENABLE, KILL, PARTITION, and RETURN commands can all identify terminals, controllers, and subdevices via:

- a hexadecimal channel number (which must be preceded by a period) that is the address of the physical channel corresponding to the terminal.
- the terminal name as defined via SUPER.

Changing Line Configuration Parameters

Line configuration options define the class of device, set line values such as buffer sizes, character transmission blocks, line status, input/ output characteristics, CP-6 environment characteristics (the logon, the profile) for the line and many others. The NETCON Line Configuration Option Table in the System Support Reference Manual details the options and the parameters that they control.

Changing Boot and Handler Parameters

The procedure to change a node definition is to:

1. Use the NETCON processor SELECT command to select the node.
2. Use the NETCON processor SET BOOTINFO command to change the handler or other boot information.

In online mode, the user is prompted to enter the information. As each item is prompted for, its current value is listed.

The NETCON processor SET command can also be used to change a number of boot-time control parameters, general handler parameters, ASYNC-specific handler parameters, FEI handler parameters, and node administrator parameters. These parameters are all described in the SET Command Parameters table in the NETCON section of the System Support Reference Manual.

Module 6-1

Introduction to Response and Throughput Tuning Tools

This module introduces the reader to various tools contained in the CP-6 system that are used to obtain the maximum performance from a CP-6 system. System tuning techniques differ depending on how the CP-6 installation has been configured and the nature of the inefficiencies the tuning is meant to correct. But, the basic approach for use of the tools introduced in this section is always the same: the recommended CP-6 system tuning approach involves establishing a baseline of performance for a particular CP-6 installation, and then recognizing deviations from this baseline.

Once the baseline is established, the deviations need to be examined more closely: first, determining the type of problem which has presented itself; then, focusing even more closely in order to determine the cause, and thus the remedy, for the deviation. Occasionally the detecting of problems is not quite so analytical, and requires vigilance and intuition to ferret out the cause.

This module:

- categorizes the kinds of system performance problems that can arise and
- identifies in a general way procedures and tools available for detecting these problems.

Module 6-2 is an example of how system statistics used in analyzing system performance problems can be gathered. Module 6-3 is a more detailed examination of how CP-6 processors can be used to tune a system.

There are several aspects to performance of a CP-6 system. Each aspect may exhibit itself as a different type of problem, even though the problems may superficially appear to be the same. It is very important to identify the particular type of problem, in order to administer the proper corrective action. The following aspects of performance will be examined in this Introduction:

- Responsiveness
- Throughput
- Memory utilization
- Input/output throughput
- FEP throughput

As a general technique it is recommended that most CP-6 installations constantly run STATS as a batch or ghost job to keep a log of system performance activity. Module 6-2 contains an example of how to run STATS as a ghost job. If a STATS log of system performance activity is maintained, STATS can be used to analyze the normal operating profile of the installation. This technique can also be used to assist in predicting the need for additional equipment.

Given that STATS is being used to analyze the normal operating profile of an installation, the next step is to identify the existence of problems. Basically, there are two ways to do this. The first and recommended approach is to regularly watch the statistics being gathered to detect anomalies. The second, and generally less satisfactory, is to wait for complaints from the user of the system.

Once a problem has been identified, the next step is to classify the problem into one of the five areas mentioned above. For each type of problem, the next step is to use the tools to focus in on the particular problem, solve the most severe problem, and then re-examine the situation to see if a serious problem still exists.

Ways to recognize each of the types of problems are discussed below. The discussions which follow all assume that a system is adequately configured and has experienced adequate performance. The techniques given are intended to identify changes in performance characteristics in a system and to assist in obtaining the maximum from a CP-6 system.

Responsiveness

CP-6 responsiveness performance can be divided into three areas: time-sharing responsiveness, batch responsiveness, and transaction processing responsiveness.

Time-sharing Responsiveness

Time-sharing responsiveness ("Gee, the system seems to be responding slowly today") can be divided into three categories:

- Host response
- FEP saturation
- Input/output bottleneck

This section examines Host response problems. If a time-sharing responsiveness problem is not related to host response, then it is related to either FEP saturation or an input/output bottleneck. These aspects of performance are discussed as separate topics, below.

A host response problem manifests itself in one of two ways: slow response to trivial tasks, and slow response to more substantial compute-bound tasks. The response time to trivial tasks is the response time printed by the STATS processor, and problems of this type can be diagnosed directly from STATS. If this value is not in a desirable range, it can be affected by adjusting the various QUAN, QMIN, and PRIO values using the CONTROL processor (see Module 6-3).

The response time to more substantial tasks is reflected in the ETMF (Elapsed Time Multiplication Factor) figure as reported by STATS. Abnormally high ETMF values signify throughput problems which are discussed below.

TP Responsiveness

This class of problems can be due to Host transaction bottleneck, FEP saturation, or an input/output bottleneck on the files or database in question.

If a number of transactions are not queuing up in the host for processing, then the problem must be an FEP saturation problem. The FPL programs should be examined for inefficient code.

If the problem is not in the FEP, the problem may be caused by an input/output bottleneck contention for TPAPs or a throughput problem. Input/output bottleneck and throughput problems are discussed in a separate topic, below. If the problem is due to contention for TPAPs, the transaction load should be analyzed carefully, and the most heavily used TPAPs should be considered for PERM status. (Refer to the publication CP-6 TP Administrator Guide (CE50) for more information on TPAPs and PERM status.)

Batch Responsiveness

Batch responsiveness is basically batch turnaround. The CONTROL processor can be used for batch turnaround problems to examine the particular definitions and the number of jobs running in each partition. Consideration should be given to establishing express partitions for fast turnaround of jobs with limited resource requirements.

Throughput

This section is concerned with CPU utilization problems and CPU users that create such problems. Generally, a CPU throughput problem is signalled by anomalously high ETMF. The high ETMF is usually due to one or more heavily used programs that are somewhat inefficient. To proceed further in this kind of problem analysis, the heaviest user of CPU time must be identified. The program ST.X(H) is the most useful tool for this purpose. ST.X(H) will take a 30-second snapshot of the system, and report the top six users of CPU time, memory, and input/output.

Having identified the suspects, the next step is to determine whether the problem is an execution time or service time problem. The easiest way to make this determination is with ST.X (sysid), which will report the usage of a particular user of execution and service time.

If it is determined that the throughput problem is a service time problem with a particular user, the best way to proceed is to use the MOUSE feature of STATS, reporting on the user in question. MOUSE provides a report of all of the monitor services used, along with various statistics about the monitor services. (Refer to the CP-6 System Support Reference (CE41) for more information on the MOUSE feature of the STATS processor.) The program can then be analyzed to determine if it is doing unnecessary or inefficient operations.

If it is determined that the throughput problem is an execution time problem, the best way to proceed is to use PMON.X to determine the execution time profile for the program.

Usually, these steps will be enough to locate the problem so that the inefficient code can be eliminated without further difficulty. After having eliminated the principal offenders, the program should be re-examined for further problems.

Frequently, a program will have a problem with both service and execution time, in which case both of the just described techniques should be used.

Memory Utilization

If the problem appears to be that more memory is being used to support a system load than should be necessary, the following approaches should be used.

First, an overall global picture of memory utilization should be obtained. The STATS processor RESOURCE display provides this information. As a result of this information, some adjustments to TIGR and CONTROL parameters may be desirable to reduce certain memory usages. (See Module 6-3 for an example of a STATS processor RESOURCE display and for more discussion of tuning a system using these parameters.)

Then, the following steps should be performed:

- The memory used by users of the system should be examined carefully. It may be necessary to constrain the memory available to online users, thus forcing jobs which use more memory into batch.
- ST.X(H) should be used to examine the users using the most memory and ST.X(M) should be used to examine the profile of all user's memory.

- If there are applications using large amounts of memory, they should be examined in more detail, especially if they are typically in use by multiple users. Remember that procedure is usually shared, but that data is not.

- The next step is to examine the LINK map of the user in question to identify the offending modules within the run unit.

- Then the data map of the offending modules can be examined for extraneous data usage.

Input/Output Bottlenecks

The STATS input/output display can be used to examine the load on various devices to determine if a system is experiencing an input/output bottleneck problem. If such a problem is detected, the next step is to determine if the problem is user-associated or configuration-associated.

ST.X(H) can be used to list the top six input/output users. If there are some standouts, there may well be inefficiency in the programs. The MOUSE feature of the STATS processor can be used to further identify the source of excessive input/output.

If there are no clear excessive input/output users, the bottleneck may be due to an imbalance among packs. Consideration should be given to moving heavily-used accounts to lightly-used packsets, or to splitting heavily-used packsets among several devices.

FEP Bottleneck

If an FEP bottleneck is suspected, the short form of the STATS processor FEP display can be used to determine the load on all FEPs. (Refer to Module 6-3 for an example of a FEP display.) Any FEP for which the utilization approaches 100% will be a probable cause of a bottleneck. Such an FEP should be examined more closely using the long form of the STATS processor FEP display to get a detailed breakdown of usage to determine the cause of the overload. Then, consideration should be given to moving lines in order to balance loads among FEPs.

Module 6-2

Collecting CP-6 Statistics

To determine if a CP-6 system is meeting its goals and objectives, the system manager must measure system performance. CP-6 system performance is measured using the STATS processor (described fully in the CP-6 System Support Reference Manual (CE41)) to collect statistics on how the system is operating.

The statistics that are collected by STATS are used for several purposes. The statistics are used:

- To determine if the CP-6 system is meeting its goals and objectives defined in terms of STATS items.
- When modifying the operation of the CP-6 system.
- For long-term system planning. As illustrated in the example in this module, long term system planning can be automated if a synopsis of daily STATS data reduction is kept on a data file.

All of these uses of statistics require that STATS data files be built.

The STATS processor can be run in online, batch, or ghost mode. When collecting statistics for long-term system tuning and planning, STATS is usually run continuously in the ghost mode because:

- STATS requires fewer resources in the ghost mode: A terminal or a FEP port is not required and a batch partition is not required.
- The system manager can ensure that STATS is started whenever the system is booted, by putting commands in the GOOSE_EGG file (described below).

Since the collection of system statistics requires some system resources, use of STATS commands needs to be planned to minimize the resources required, and to ensure collection of sufficient data so that the statistics are meaningful.

The CPU resource requirement to collect the system statistics is negligible. Host and FEP statistics can be collected for an entire 24-hour period and use less than 10 minutes of CPU time.

The memory resource requirement for the STATS processor to log the host and FEP statistics is less than 128KW. The amount of memory actually required by STATS is dependent upon the configuration of the system and the statistics that are logged.

The disk space resource requirement for collection of the statistics is dependent upon the system configuration, the statistics that are logged, and the frequency with which the statistics are logged.

The interval that is used to log the statistics to the file is chosen to keep the CPU resource requirement very low, to keep the disk space requirement reasonable, and to collect sufficient data to make the statistics meaningful. The interval is generally chosen to be 15 to 30 minute during the periods of high usage. A larger interval can be used during periods of light usage (e.g., a 120 minute during third shift or during a weekend).

To ease the management of the amount of disk space used in the collection of statistics, a new file can be created every day. Daily file creation allows:

- Batch jobs to be scheduled to do the STATS data reductions.
- Files to be moved from the current disk to another disk or tape for long-term storage in a timely manner.

The statistics files can be moved to long-term storage on either a daily, weekly, or monthly basis. Once data reductions are performed and the files are moved to long-term storage, the files can be deleted from the current disk. The frequency with which files are moved to long-term storage regulates the total amount of disk space used for the collection of system statistics.

This module describes how to:

- Create a ghost STATS user.
- Create an XEQ file to gather statistics and perform data reduction on the statistics.
- Ensure statistics gathering by using the GOOSE processor.

Creating a Ghost STATS User

The process of running STATS in the ghost mode to collect statistics is started by choosing a user id for the ghost user. This user id can be an existing user id or a special user id that is created for this purpose. In either case, the user id must be authorized with the appropriate resources, access modes, and permissions to run STATS. In particular, the user id must be authorized for access in the ghost mode, must have sufficient file space to save the STATS files, must have sufficient memory to run STATS, and must have the PM PRIV to run STATS. The following figure is an example of how a special user id would be created to run STATS in the ghost mode.

```
!SUPER

*** CP-6 SUPER C00 ***

CMD*CREATE :STATS,STATISTICS FROM DEFAULT
OPT*ACCESS B=YES, O=YES, G=YES, TP=NO
OPT*HSET = DP#SYS
OPT*FACCOUNT
SUB*GR=15000, NOLIST=?, DEFAULT BACKUP, NO ACUP
SUB*
OPT*MEMORY MAX B=256, O=256, G=256
OPT*MEMORY DEF B=128, O=128, G=128
OPT*PASSWORD = PASSWORD
OPT*PRIV
SUB*PM B=YES, O=YES, G=YES, TP=NO
SUB*END
OPT*SETUP G = !XEQ STATS_XEQ
OPT*END
CMD*END
```

Figure 6. Sample STATS User ID Creation

In the example:

- The user id is authorized to run in batch, online, or ghost modes. The ghost mode can be used to collect the statistics; the batch and online modes can be used to do data reductions on the statistics.
- A file management account is created on DP#SYS to hold the STATS files. DP#SYS is chosen as the home packset because that packset is always mounted when the system is booted. The STATS ghost requires that the home packset be mounted when initiated because an XEQ file is used and because STATS will create statistics files.
- The user id is given enough default memory to run the STATS processor.
- The user id is given the PM privilege so that the STATS processor can be run.
- The user id is given a setup command for the ghost mode. This setup command executes the STATS_XEQ.:STATS file when the ghost is started.

Creating an XEQ Command File

The ghost STATS user is usually set up to execute an XEQ file. The XEQ file controls how the ghost STATS user collects the system statistics. In the previous user authorization, the XEQ file was specified in the SETUP option. The XEQ file name was STATS_XEQ.:STATS. If the SETUP option is used as shown, the STATS_XEQ.:STATS must be created. The following figure is an example of what that XEQ file could be like. The comments in the XEQ file describe how it functions.

In this example:

- The STATS commands to collect the system and FEP statistics are embedded in the STATS_XEQ.:STATS XEQ file. These STATS commands are put in a temporary star file (*STATS_COMMANDS). They are later used via an XEQ statement with the appropriate substitutions.
- The XEQ file submits one or two batch jobs to perform data reduction after collecting the statistics: data reduction of the statistics can be an automated part of statistics collection.

```
!DEFAULT PERIOD1_STOP$=0800, PERIOD1_INT$=60
!DEFAULT PERIOD2_STOP$=1800, PERIOD2_INT$=30
!DEFAULT PERIOD3_STOP$=2359, PERIOD3_INT$=60
!DEFAULT WEEKEND_STOP$=2359, WEEKEND_INT$=120
!"
!"    This is an XEQ file that controls the execution of the
!"    STATS ghost that collects system statistics.  The STATS
!"    data is logged into a file whose name is of the form
!"    STATSDATA_yymmdd where yymmdd is the date as given by
!"    the IBEX $DATE function.  This file is created in the
!"    current file directory (file management account).
!"
!"    The DEFAULT substitution variables PERIODn_STOP$ and
!"    PERIODn_INT$ specify the stop time and interval size
!"    for a period of every weekday.  The start time for
!"    PERIOD1 is 0000.  The start time for all other periods
!"    is the stop time of the previous period.  The stop
!"    time, specified using a 24-hour clock, must be specified
!"    as four decimal digits (with leading zeroes as necessary).
!"
```

Figure 7. STATS_XEQ: Sample STATS XEQ File (cont. next page)

```
!"    The DEFAULT substitution variables WEEKEND_STOP$ and
!"    WEEKEND_INT$ specify the stop time and interval size
!"    for the weekend.  The start time for the interval is
!"    the STOP time of the last period of the last weekday
!"    (i.e., 2359).  The stop time for the interval must be 2359.
!"
!"    Additional intervals can be added to the weekday and/or
!"    weekend by adding the appropriate DEFAULT substitution
!"    variables, command variables, and the appropriate IBEX
!"    commands to this XEQ file.
!"
!"    This command file will not function correctly for
!"    collecting STATS data on a noncontinuous basis.
!"    That is, the first period of the weekday/weekend must
!"    start at 0000 and the last period of the weekday/weekend
!"    must end at 2359.
!"
!"    WARNING:  If the length of a period is not an even
!"    multiple of the interval, the switching of interval
!"    size will not occur at the specified start/stop times.
!"
!"
!"
!"    Move all DEFAULT and SUBSTITUTION parameters into command
!"    variables.  The delete all DEFAULT parameters.
!LET     PERIOD1_STOP = PERIOD1_STOP$, PERIOD1_INT = PERIOD1_INT$
!LET     PERIOD2_STOP = PERIOD2_STOP$, PERIOD2_INT = PERIOD2_INT$
!LET     PERIOD3_STOP = PERIOD3_STOP$, PERIOD3_INT = PERIOD3_INT$
!LET     WEEKEND_STOP = WEEKEND_STOP$, WEEKEND_INT = WEEKEND_INT$
!DEFAULT DELETE
!"
!BUILD:                 "       Build command file.
!"
!"    Build the STATS command file.  XEQ substitutions will
!"    supply the appropriate values.
!IF      $FID_EXIST ( '*STATS_COMMANDS' ) THEN DELETE *STATS_COMMANDS
!EDIT
 BUILD   *STATS_COMMANDS
$STATS
 MESSAGE STATS collecting statistics using INT$ minute intervals
 FILE    FILE$
 INT     INT$
 DI      NONE
 LOG     PMDAT, FEP
 GO      N TIMES
 END
!EOD                    "       Terminate EDIT file data.
SE; CL1,1; /$/S/!/      "       Substitute ! for $ in STATS
                        "       command, so it will execute.
END                     "       Terminate EDIT processor.
!"
!BEGIN:                 "       Beginning of iterative loop.
!"
!"    Get current date, time, and day of the week.  Also calculate
!"    the current time as minutes since the beginning of the day.
!LET     DATE = $DATE
!LET     TIME = $TIME
!LET     DAY  = $DAY
!LET     NOW  = TIME / 100 * 60 + $SUBSTR  ( TIME, 2, 2 )
!"
!WEEKEND:               "       Weekend decision.
!"
```

Figure 7. STATS_XEQ: Sample STATS XEQ File (cont. next page)

```
!"    If this is not Saturday or Sunday, go to the weekday decision.
!"    Otherwise, set the interval size and stop time based upon
!"    the weekend parameters.
!IF       DAY ~= 'SAT'  ↑  DAY ~= 'SUN'  THEN GOTO WEEKDAY
!LET      INT  = WEEKEND_INT
!LET      START = '0000'
!LET      STOP = WEEKEND_STOP
!GOTO     CALCULATE
!"
!WEEKDAY:                 "       Weekday decision.
!"
!"    Determine which period of the day this is.  Then set
!"    the interval size and stop time based upon period of
!"    the day.
!PERIOD1:
!IF       TIME >= PERIOD1_STOP           THEN GOTO PERIOD2
!LET      INT = PERIOD1_INT
!LET      START = '0000'
!LET      STOP = PERIOD1_STOP
!GOTO     CALCULATE
!PERIOD2:
!IF       TIME >= PERIOD2_STOP           THEN GOTO PERIOD3
!LET      INT = PERIOD2_INT
!LET      START = PERIOD1_STOP
!LET      STOP = PERIOD2_STOP
!GOTO     CALCULATE
!PERIOD3:
!LET      INT = PERIOD3_INT
!LET      START = PERIOD2_STOP
!LET      STOP = PERIOD3_STOP
!"
!CALCULATE:               "       Calculations.
!"
!"    Calculate the number of intervals to be done.  Also
!"    generate the name of the file that is to be used.
!"
!"    WARNING:  If the length of a period is not an even
!"    multiple of the interval, the switching of interval
!"    size will not occur at the specified start/stop times.
!LET      THEN = STOP / 100 * 60 + $SUBSTR ( STOP, 2, 2 )
!LET      N = ( THEN - NOW + INT - 1 ) / INT
!LET      FILE = 'STATSDATA_' || DATE
!"                                 Concatenate today's date.
!EXECUTE:                 "       Execute command file.
!"
!"    Execute the STATS command file with the appropriate
!"    substitutions.
!XEQ      *STATS_COMMANDS          FILE$ = '%FILE', ;
!                                  INT$ = %INT, ;
!                                  N = %N
!"
!REDUCTION:               "       Data reduction decision.
!"
!"    Submit the batch data reduction job for this period.
!"    Transfer variables from this XEQ file to the batch job.
!LET      NAME = 'STATS_' || DATE || '_' || START || '_' || STOP
!LET      YY = $SUBSTR ( DATE, 0, 2 )
!LET      MM = $SUBSTR ( DATE, 2, 2 )
!LET      DD = $SUBSTR ( DATE, 4, 2 )
!LET      MMDDYY = MM || '/' || DD || '/' || YY
!LET      FROM = $SUBSTR ( START, 0, 2 ) || ':' || ;
!               $SUBSTR ( START, 2, 2 )
```

Figure 7. STATS_XEQ: Sample STATS XEQ File (cont. next page)

```
!LET      TO    = $SUBSTR ( STOP,  0, 2 ) || ':' || ;
!                 $SUBSTR ( STOP,  2, 2 )
!BATCH    STATS_REDUCTION          NAME$ = '%NAME', ;
!                                  FILE$ = '%FILE', ;
!                                  MMDDYY$ = '%MMDDYY', ;
!                                  FROM$ = '%FROM', ;
!                                  TO$ = '%TO'
!"
!"     Determine if the stop time of this interval is at the end
!"     of the day.  If not, go to begin the next interval.
!"     Otherwise, submit another batch data reduction job to
!"     do the data reduction for the entire day.  Then go to
!"     begin the next interval.
!IF       STOP ~= 2359             THEN GOTO BEGIN
!LET      NAME = 'STATS_' || DATE || '_0000_2359'
!BATCH    STATS_REDUCTION          NAME$ = '%NAME', ;
!                                  FILE$ = '%FILE', ;
!                                  MMDDYY$ = '%MMDDYY', ;
!                                  FROM$ = '00:00', ;
!                                  TO$ = '23:59', ;
!                                  SCHED$ = 'RERUN, ORDER'
!GOTO     BEGIN
```

Figure 7. STATS_XEQ: Sample STATS XEQ File

Gathering Statistics

STATS commands are used to gather statistics. In the example, STATS commands in the XEQ star file are used to log all system and FEP statistics. In the collection of statistics for system tuning and planning, all statistics are usually collected. Since all statistics are collected, the system manager has the ability to generate any or all statistics displays to meet any unexpected requirements that might arise.

The collected statistics are either displayed as is, or a data reduction is performed. A display of the collected statistics (i.e., a REPLAY) will print the statistics as they would have been printed at the time they were being collected. Generally, statistics are replayed only if the system manager wishes to see how the system was operating during a period of particular interest. In this circumstance, a subset of the statistics for the time span in question can be replayed on a terminal.

STATS Data Reduction

For system tuning and system planning, data reduction is generally performed on the collected statistics. System tuning and planning decisions are usually based on the results of data reductions performed in either online or batch mode. In online mode, the system manager must use a terminal to perform data reductions. In batch mode, the data reductions can be submitted automatically by the XEQ file that controls the collection of the statistics.

Data reduction results need to be examined as they are produced. From these data reductions, the system manager can determine what is normal for the system. These data reductions will show the system manager if the CP-6 system is meeting its goals and objectives that are defined in terms of STATS items. As the workload and system configuration change, the system manager will know from the data reductions when changes will have to be made in the system tuning parameters. In addition, the examination of these data reductions may also reveal when the system configuration must be enlarged or upgraded to meet performance requirements.

An example of a batch data reduction job is shown in the following figure. This is the batch job that would be submitted by the STATS_XEQ.:STATS XEQ file.

```
!DEFAULT NAME$=STATS_REDUCTION
!DEFAULT WSN$=LOCAL, DEFER$=0:00, SCHED$=RERUN
!DEFAULT TIME$=30:00, MEM$=128
!DEFAULT FPOOLS$=31
!JOB     NAME=NAME$, WSN=WSN$, DEFER=DEFER$, SCHED$
!RES     TIME=TIME$, MEM=MEM$
!LIMIT   FPOOLS=FPOOLS$
!STATS
 FILE    FILE$
 SPAN    FROM$, MMDDYY$ - TO$, MMDDYY$
 HISTOGRAM RESPONSE(SNAP), USER SIZE(SNAP), INTERACTION(SNAP)
 ALSO DI CPU, RESOURCES, DEVICES, CHANNELS, PROCESSOR, FEP SUMMARY
 GLOM
 STATISTICS ALL
 END
```

Figure 8. STATS_REDUCTION: Sample STATS Data Reduction Job

In this example, selected STATS data reductions are performed. These data reductions provide a starting point for system tuning and planning in less than 20 pages of output. The small amount of output can be quickly read by the system manager. If some of these data reductions are not useful, they can be removed from the data reduction commands. If other data reductions are required, they can be added to the data reduction commands.

The reduction job example will function correctly under normal circumstances. If the system has an interruption (i.e., ZAP!, DIE!, or SCREECH), the data reduction job will not function correctly because the STATS processor GLOM command cannot perform calculations for an interval during which a system interruption occurred. In this case, the system manager will have to manually perform the data reductions across the partial intervals. This procedure can be performed either at an online terminal or by submitting a batch job (i.e., the STATS_REDUCTION job) with modified substitution parameters.

GOOSE commands

After the user id has been created and the appropriate files have been built, the system manager can start the STATS collection ghost by using the GOOSE processor (described fully in the CP-6 System Support Reference Manual (CE41)). The following figure is an example of starting the STATS ghost.

```
!GOOSE
Goose here
:START :STATS,STATISTICS,PASSWORD
:END
```

Figure 9. Starting STATS Ghost Immediately

To ensure that the STATS ghost is started whenever the system is booted, the system manager can schedule the starting of the ghost in the GOOSE_EGG file, as shown in the following figure.

```
!GOOSE
GOOSE HERE
:UPDATE
EDIT C00 here
*AP
    1.000 START :STATS,STATISTICS,PASSWORD AT STARTUP
    2.000
*END
   GOOSE_EGG updated
   Automatic scheduling updated
:END
```

Figure 10. Scheduling Start of STATS Ghost

Using the specified command in the GOOSE_EGG file, the GOOSE processor will start the STATS ghost following every system boot or recovery.

Module 6-3
Using CP-6 Statistics

The process of modifying the operation of a CP-6 system to meet specific goals and objectives is referred to as tuning the system. After CP-6 statistics have been collected and data reductions have been performed, the system manager can then use the data reductions to do system tuning. The data reductions are used either individually or in combination with others to provide the data to make tuning decisions.

This module discusses how to use the various STATS displays for system tuning, and how to use TIGR, CONTROL, NETCON and SUPER processor parameters to modify the performance of a CP-6 system. The values that the system manager uses for these tuning parameters are based upon the current parameter values and upon the statistics gathered from the running CP-6 system.

Note that to perform system tuning:

- Several data reductions should be available. Usually, system tuning is aimed at providing the best CP-6 system performance for a normal workload. The normal workload is unique for each CP-6 system. The system manager determines the normal workload by examining the statistics continuously on a long-term basis. The system manager determines the normal workload for the CP-6 system and sets the tuning parameters accordingly.

- System tuning parameters are set in TIGR, CONTROL, NETCON, and SUPER. Changes in TIGR parameters are only effective after a reconfiguration boot has been performed. Changes in CONTROL and NETCON parameters are stored in the host and FEP monitor tables. Some of these changes become effective immediately. Other changes are effective only for new users as they log on. Changes in SUPER parameters are always effective the next time a user logs on. The description of the individual processor commands and, in some cases, the tuning parameters are documented in the CP-6 System Support Reference Manual (CE41). That manual must be examined to determine when tuning parameter changes become effective.

Normally, system tuning parameters are not changed for transient workload fluctuations. However, the system manager may find that the workload changes according to the time of day or day of the week. In this case, the system manager can establish separate tuning parameters for CONTROL (and, if appropriate, NETCON) for each of these periods. The system manager can then put commands in the GOOSE_EGG file that will cause the GOOSE processor to start a ghost user at selected times on selected days. (Refer to Module 6-2.) This ghost user will execute XEQ files that will set the system tuning parameters in CONTROL and/or NETCON.

If the tuning parameters are changed at selected times, the STATS collection XEQ file (e.g., STATS_XEQ in Module 6-2) should be changed so that the periods coincide with the selected times. This allows STATS data reductions from each period to be used to adjust the tuning parameters for that period. In this situation, the system manager is actually tuning the CP-6 system for optimum performance during different workload conditions.

Note too that the WASP tool in the X account can be used to perform online monitoring of many of the display items reported via the STATS processor.

STATS CPU display and CPU Tuning

The STATS processor CPU display provides an overall snapshot of what is happening in the system. The STATS RESPONSE histogram is used as detailed information to help in setting the CPU tuning parameters.

STATS CPU Display

An example of a STATS CPU display is shown in the following figure.

STATS interval from 08:08:36.94 to 15:39:12.98

	{all}	{snap}		{all}	{snap}
% batch execution	10.3	31.7	ETMF	1	1
% batch service	17.5	30.2	90% response time	50	50
% online execution	3.8	37.0	I/O load factor	10	8
% online service	8.7	90.3	# of batch users	0	4
% ghost execution	6.9	26.0	# of online users	7	55
% ghost service	9.9	30.4	# of ghost users	18	19
% TP execution	0.0	0.0	# of TP users	0	2
% TP service	0.0	0.0	I/Os per minute	682	2876
% monitor execution	4.4	17.3	Schedules per minute	708	3244
% I/O wait	23.4	20.1	Interactions per min	12	105
% resource wait	0.0	0.1	Events per minute	1371	5814
% I/O+resource wait	0.0	0.1	PMMEs per minute	8002	36902
% true idle	214.0	9.6	Avg. usec per PMME	2813	2561
Total	300.9	300.3	Minutes in interval	6517	451

Figure 11. STATS CPU Display

When tuning a system, the system manger should use the STATS processor GLOM command to perform data reduction and examine the statistics in the snap column of the data reduction. These statistics are the averages of the workload for the selected period. The statistics in the all column are the averages of the workload since the system was last booted. Since the all column may contain the statistics from several distinct periods, in most cases the all statistics are not used to make system tuning decisions.

The system manager should examine the percentages in the first eight lines of the left side of the display and the number of users in each mode to determine whether each CP-6 access mode is getting the correct total percentage of the CPU for the period spanned by the statistics.

The following table defines the items in a STATS CPU display.

Table 8. STATS CPU Display Definitions

Display Entry	Definition
% mode execution	The percentage of CPU time spent executing user code in each mode. The mode is specified as either batch, online, ghost, or TP. This percentage is directly chargeable to users in the form of CPU execution time.
% mode service	The percentage of CPU time spent executing monitor code to satisfy user monitor service requests. The mode is specified as batch, online, ghost, or TP. This percentage is directly chargeable to users in the form of CPU service time.
% monitor execution	The percentage of CPU time spent executing monitor code for internal monitor functions (e.g., scheduling, interrupt handling, etc.). This percentage of the CPU time is the monitor overhead.
% I/O wait	The percentage of CPU time spent idle waiting for I/O operations to complete. If the system did not have to go into an idle state waiting for I/O to complete, this percentage will be zero.
% resource wait	The percentage of CPU time spent idle waiting for an internal monitor resource to become available. If the system did not have to go into an idle state waiting for an internal monitor resource to become available, this percentage will be zero.
% I/O resource wait	The percentage of CPU time spent idle waiting for I/O to complete or for an internal monitor resource to become available. If the system did not have to go into an idle state waiting for an I/O to complete or an internal monitor resource to become available, this percentage will be zero.

Table 8. STATS CPU Display Definitions (cont.)

Display Entry	Definition
% true idle	The percentage of CPU time that was spent idle with nothing to do.
ETMF	Execution time multiplication factor. This factor is multiplied times the required CPU time to give an estimate of how much elapsed time will be required to execute a task. For example, if a task requires 2 CPU minutes and the ETMF is 3, then the task will require approximately 6 minutes of elapsed time to complete.
90% response time	The time in milliseconds between the receipt of an activation character (normally a carriage return) by the host and the start of processing for 90% of the activation characters received. The 90% response time is calculated only for online users. Note that this is not the response time from last character entered to first character received, and does not include delays in the FEP.
I/O load factor	This factor is the measure of the I/O load on the system. This factor is the probability that an I/O request will be queued rather than executed immediately. This number is the average of the I/O load factors of all devices active during the interval.
# of mode users	The number of users on the system for each mode. The mode is specified as batch, online, ghost or TP. For a GO or REPLAY command, the number in the snap column is the number of users on the system in the respective mode at the end of the interval. The number in the all column is the average number of users on the system in the respective mode since the last system boot or recovery. For a GLOM command, each column is the average number of users on the system in the respective mode during the interval specified in each column.

Table 8. STATS CPU Display Definitions (cont.)

Display Entry	Definition
I/Os per minute	The number of I/O connects per minute to all IOM-connected controllers and system consoles. Each connect may contain several I/O commands for the controller.
Schedules per minute	The number of passes per minute through the system scheduler. The number of schedules per minute is influenced by the number of I/Os per minute, the number of interactions per minute, and tuning parameters.
Interactions per min	The number of activation characters (e.g., carriage return, line feed, etc.) received by the host per minute.
Events per minute	The number of scheduler events that occurred per minute.
PMMEs per minute	The number of monitor service requests per minute. Every monitor service request is a Privileged Master Mode Entry (PMME) CLIMB instruction.
Avg. usec per PMME	The average number of microseconds required to complete the monitor service request.
Minutes in interval	The number of minutes in the interval. The all column contains the number of minutes since the last system boot or recovery. The snap columns contains the number of minutes in the interval of the INT or GLOM command.

STATS RESPONSE Histogram

The STATS RESPONSE histogram provides detailed information about the interactive response time for online users. The following figure is an example of the STATS RESPONSE histogram.

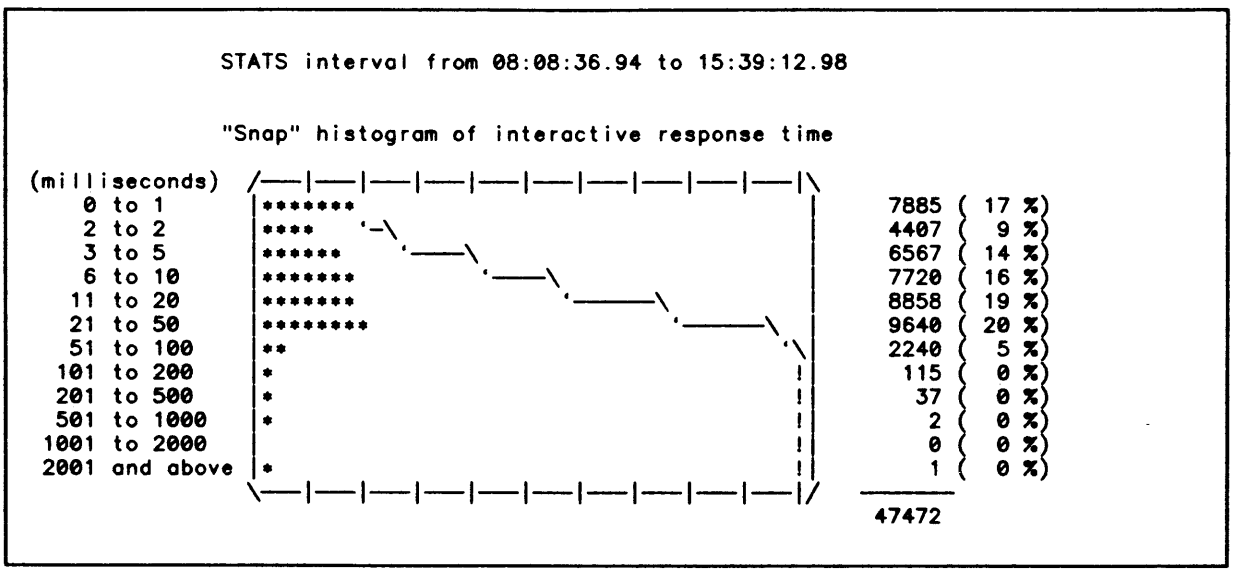


Figure 12. STATS RESPONSE Histogram

The STATS RESPONSE histogram creates a histogram plot from the response time to every interaction that occurred during the requested interval.

This histogram provides:

- a detailed report of the 90% response time reported in the STATS CPU display.
- a count and a percentage of the interactive response time for various time intervals.

CPU Tuning

If analysis of the STATS CPU display and the STATS RESPONSE histogram indicates that one or more modes is receiving too much or too little of the CPU for the number of users in that mode, the system manager must change some of the TIGR, CONTROL, NETCON, and SUPER tuning parameters.

USERS TIGR Parameter

One of the first tuning parameters to be established by the system manager is the maximum number of concurrent users ever to be allowed on the system. The maximum number of concurrent users allowed on the system is specified by the TIGR processor USERS parameter. The USERS parameter is an option on the TIGR processor MON command. The total number of users concurrently on the system in all of the access modes cannot exceed the number specified by the USERS parameter. Therefore, the system manager usually sets this parameter slightly greater than the total number of users that are expected on the system at any one time.

LIMITU CONTROL Parameter

After the TIGR processor USERS parameter has been set, the system manager still has the ability to reduce the total number of concurrent users on the system. The system manager can limit the total number of concurrent users on the system by using the CONTROL processor LIMITU parameter or the operator ON keyin. The total number of concurrent users specified by either of these methods must be less than or equal to the maximum number of concurrent users specified by the TIGR processor USERS parameter.

UM CONTROL Parameter

After the total number of concurrent users has been specified, the system manager must determine the maximum allowable number of users in each of the CP-6 access modes. The maximum number of users in each mode is specified using the CONTROL processor UM parameter or various forms of the operator ON keyin. The sum of the maximum number of users in each mode (UM) can exceed the total number of concurrent users on the system (LIMITU). However, the system will not allow any user in any mode to log onto the system once the total number of concurrent users on the system has been reached. This can produce undesirable results for ghost and TP users. Therefore, the system manager must be very cautious when such a situation arises. Observing the following three guidelines will help avoid such undesirable results.

1. The maximum number of ghost users should be larger than the maximum number of concurrent system- and installation-supplied ghosts. This ensures that ghost users are allowed to logon when they are initiated. Ghost users are usually initiated by the system, by GOOSE commands in the GOOSE_EGG file, or by the START TP keyin. These types of activities generally should not fail because the maximum number of ghost users has been exceeded.

2. The maximum number of online and TP users should be larger than the maximum number of concurrent online and TP users expected on the system. If the current number of online and/or TP users reach the maximum number for that mode, the additional users in that mode that attempt to logon will receive a message that no more users in that mode are being accepted. This can be very frustrating to a user sitting at a terminal. Only a lack of system resources (e.g., memory) or specific system goals and objectives should cause this guideline to be ignored.

3. The maximum number of batch users is chosen based upon the system goals and objectives. If the system goals and objectives specify that online and/or TP users are to receive good response time, the number of batch jobs must be limited to approximately two batch jobs per CPU. If the system goals and objectives specify that batch jobs are of primary importance, a larger number of batch jobs can be run. However, dramatically increasing the number of batch jobs that are run concurrently actually increases the elapsed times of individual jobs and reduces total batch throughput. In other words, running three batch jobs with an ETMF of 1 will produce shorter elapsed times and more throughput than running six batch jobs with an ETMF of 3 or 4.

MAXACCT CONTROL Parameter

The system manager can use the MAXACCT CONTROL parameter to prevent a single user from monopolizing the batch user slots. This parameter specifies the number of jobs that will be run concurrently from the same account. For example, if the maximum number of batch users is 2, MAXACCT can be set to 1 to prevent a single user from running two batch jobs at the same time. Any user may submit several jobs, but in this case, only one job from any one account will be run at one time.

NPART CONTROL Parameter

In the CP-6 system, the batch jobs to be run are selected through the use of batch partitions. These batch partitions do not represent any real, physical resource. They are used as a selection mechanism to choose the next batch job to run. When a batch job is actually in execution, it does not really run in a partition. Rather, a batch job is associated with a batch partition during execution to control only the selection of other batch jobs for execution. Therefore, each of these batch partitions may have more than one executing batch job associated with it.

Each batch partition has several selection limits. If a batch job falls within all of the selection limits of a partition, the job is eligible to be selected from that partition. A batch job may be eligible to be selected from more than one partition. However, when placed in execution, it will be associated only with the one partition that it was selected from.

The system manager selects the number of batch partitions that will be used to select batch jobs by specifying the CONTROL processor NPART parameter. The number of batch partitions (NPART) may exceed the number of concurrent batch jobs allowed on the system (UM(B)). Up to 16 batch partitions may be used to select batch jobs. The number of partitions used is dependent upon the selection criteria used to select batch jobs. The partitions that are selected are numbered from 1 through NPART.

PLOCK CONTROL Parameter

Even after NPART partitions are selected, the system manager can prevent the selection of batch jobs from one or more partitions by locking the partition. If a partition is locked, no additional jobs can be selected for execution from that partition. If a partition is unlocked, jobs can be selected for execution from that partition.

The system manager locks and unlocks partitions by specifying a value for the PLOCK CONTROL parameter. There is a PLOCK CONTROL parameter associated with each partition. Therefore, each partition can be locked or unlocked on an individual basis.

Partition Criteria CONTROL Parameters

The selection criteria for each partition are things such as CPU time, memory, real resources, pseudo resources, maximum number of jobs allowed to run in a partition, and maximum number of jobs from a single account allowed to run in a partition. The time, memory, and resources criteria have minimum and maximum values for each partition. The maximum number of jobs and the maximum number of jobs for a single account only have a maximum value for each partition.

When selecting a job for execution, the available partitions are scanned from partition 1 through partition NPART. If the first available partition contains a job that can be run, that job is placed in execution. If the first available partition does not contain a job that can be run, the next partition is examined. If none of the partitions contain a job that can be run, no job is placed in execution.

The selection criteria for each partition must be established by the system manager. These selection criteria should be established to meet the system goals and objectives for batch jobs. For example, if an objective is to provide fast turn-around for small, short batch jobs, partition 1 can be set up to run only jobs that ask for 32KW of memory and less than 1 minute of CPU time. Partition 1 is used so that these small, short jobs will be considered before any other jobs. In addition, this partition can be set up to run as many as 511 of these small jobs at once. The actual number of jobs run will then be controlled by the maximum number of batch jobs on the system (UM(B)) and the number of batch jobs already running in other partitions.

Other sets of criteria can be established for other partitions to accommodate other batch jobs. These criteria can include time, memory, real resources, pseudo resources, and maximum number of jobs. The number of sets of criteria helps to select the number of partitions (NPART).

Various CONTROL parameters are used to specify the partition criteria parameters. PMINTI and PMAXTI are used to specify the minimum and maximum time for each partition. PMINMM and PMAXMM are used to specify the minimum and maximum memory for each partition. PMINres and PMAXres are used to specify the minimum and maximum resources (both real and pseudo) for each partition. PJMAX is used to specify the total number of jobs that can be in execution for each partition. PMAXACCT is used to specify the maximum number of jobs from a single account that can be executed for each partition.

QMIN CONTROL Parameter

After setting the limits on the number of users, the next CONTROL tuning parameter that should be established is QMIN. This parameter specifies the minimum time-slices that are given to users.

The same or a different QMIN value can be established for each mode. If the percentage of CPU usage by each mode is acceptable, the same QMIN value can be used for each mode. Different QMIN values can be used to help change the percentage of CPU usage by each mode. In this case, the different QMIN values are based upon the online QMIN value. The value for QMIN for the online mode is chosen based upon the STATS histogram of the compute time between interactions.

STATS INTERACTION Histogram

The STATS INTERACTION histogram provides detailed information about the amount of compute time used between interactions. The following figure is an example of the STATS INTERACTION histogram.

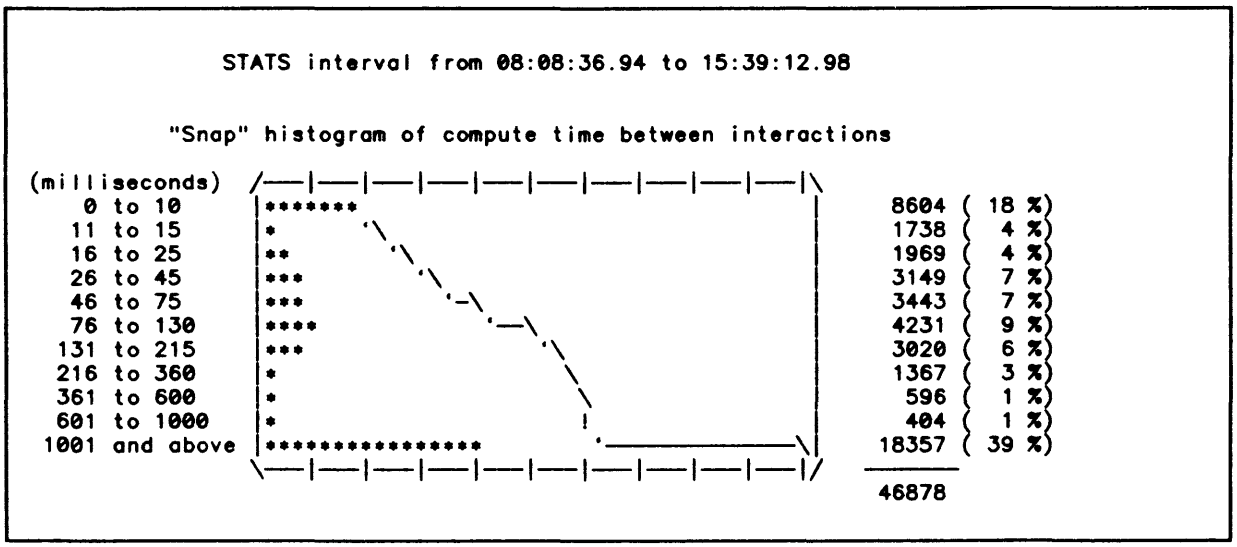


Figure 13. STATS INTERACTION Histogram

The STATS INTERACTION histogram creates a histogram plot from the amount of CPU time each online user uses between each interaction. This histogram provides a count and a percentage of the compute time between interactions for various compute time intervals.

Establishing the Online QMIN Value

The online QMIN value should be set large enough so that trivial interactions are completed within one QMIN. This ensures fast response to all trivial interactions. The STATS INTERACTION histogram is used to determine what the trivial interactions are for each particular system. The online QMIN value is then chosen after the trivial interaction compute time is determined from this histogram.

For the example histogram, the QMIN value could be chosen anywhere from 20 (the minimum) to 130. The smaller value of QMIN will ensure fast response time to slightly less than 26% (i.e., 18%+4%+4%) of the interactions. The longer compute time interactions will receive slower response time. The smaller QMIN value will also cause more trips through the system scheduler, and therefore, more system overhead in the form of monitor execution.

The larger value of QMIN will ensure fast response time to approximately 49% (i.e., 18%+4%+4%+7%+7%+9%) of the interactions. All interactions with compute time up to the QMIN value will receive fast response time. The larger QMIN value will cause fewer trips through the system scheduler, and therefore, less system overhead in the form of monitor execution. However, the larger value of QMIN could also cause the 90% response time to increase.

Values of QMIN larger than 130 would not bring substantial throughput or response time improvements for the system in the sample histogram. The percentages of interactions for the various compute times are decreasing above 130 milliseconds.

The large number of interactions with greater than one second of compute time represent the compute-bound online users. Increasing QMIN to such large values will not actually help these compute-bound users. Such large QMIN values allow compute-bound users to monopolize the system. Such large values of QMIN will also dramatically increase the 90% response time.

The system manager should choose the online QMIN value from the range of values presented in this histogram. The QMIN value must be chosen to balance response time and throughput with the system goals and objectives. Usually, the QMIN value that provides the best response time will not provide the best throughput and vice versa.

Establishing QMIN Values for Other Modes

After the online QMIN value is established, the QMIN values for the other modes can be established. If the percentage of CPU usage by each mode is acceptable, the same QMIN value can be used for each mode. If the percentage of CPU usage by a mode is too large, a smaller QMIN value can be used for that mode. If the percentage of CPU usage by a mode is too small, a larger QMIN value can be used for that mode. However, the larger QMIN value can also have the effect of increasing the 90% response time for the online users. Therefore, larger QMIN values must be used with caution.

Normally, the batch QMIN value is set the same or smaller than the online QMIN value. This setting helps to prevent batch users from taking the CPU away from online users. If the batch QMIN value is larger than the online value, online users may have difficulties completing their tasks. For the ghost and TP modes, the QMIN value is usually set the same or larger than the online value. For ghost users, this setting helps to ensure that the system ghosts can accomplish their functions quickly and efficiently. For TP users, this setting helps to ensure that transactions are processed quickly.

QUAN and PQUAN CONTROL and SUPER Parameters

After setting the CONTROL processor QMIN parameter, the system manager must set the QUAN and PQUAN tuning parameters. These parameters specify the maximum time-slices that are given to users. The CONTROL processor QUAN parameter establishes the default tuning parameter for online, ghost, and TP users. The CONTROL processor PQUAN parameter establishes the default tuning parameter for batch users. The SUPER processor QUAN parameter establishes a specific tuning parameter for each mode for a user id. If a specific QUAN value has not been established for a user in a particular mode, the appropriate default CONTROL tuning parameter is used.

The value chosen for the CONTROL processor QUAN parameter is based upon the QMIN values for online, ghost, and TP modes. The value for QUAN is usually a factor of 3 to 10 (or more) times the largest of these three QMIN values. The larger QUAN values tend to allow compute-bound users to dominate the system by locking out other compute-bound users. The smaller QUAN values tend to spread the CPU resource among the compute-bound users by causing schedules to occur more often.

The value chosen for PQUAN is based upon the QMIN value for batch and the CONTROL processor QUAN parameter. The PQUAN value is used in conjunction with the QMIN parameter to regulate the percentage of CPU used by batch users. If PQUAN is smaller than QUAN, the batch CPU percentage will be reduced. If PQUAN is equal to QUAN, the batch users will be treated the same as online, ghost, and TP users. If PQUAN is greater than QUAN, the batch CPU percentage will be increased.

A different PQUAN value can be specified for each batch partition. If a partition is running very short batch jobs, a PQUAN equal to or slightly greater than QUAN can be used to help process these jobs quickly. If a partition is running large, long batch jobs, a PQUAN value smaller than QUAN will help to ensure that the batch job does not lock out other (i.e., online, ghost, or TP) compute-bound users.

The SUPER processor QUAN parameter can be used to give a user or a group of users a special QUAN value for one or more access modes. These special QUAN values can be used to give a user or a group of users more or less of the CPU resource. If the special QUAN value is smaller than the default QUAN (or PQUAN) value, the user or group of users will receive less of the CPU compared to other users in that mode. If the special QUAN value is larger than the default QUAN (or PQUAN) value, the user or group of users will receive more of the CPU.

IOTA CONTROL Parameter

While the QMIN and QUAN parameters are used to control the relative total percentage of CPU utilization, the CONTROL processor IOTA parameter is used to control the relative I/O rates of each of the modes. The value of IOTA is used to reduce the current effective QMIN value for a user for each physical I/O. The larger the IOTA value, the slower the effective I/O rate for that mode. The smaller the IOTA value, the higher the effective I/O rate for that mode.

The value for IOTA is chosen for each mode based upon the QMIN value and the desired effective I/O rate for that mode. A larger IOTA value can be used for the batch mode to prevent batch users from monopolizing system I/O. A smaller IOTA value can be used for TP users to allow the processing of transactions more rapidly despite database disk accesses. For example, with a QMIN value of 60 and an IOTA value of 10, a maximum of 6 physical I/Os can be performed before the effective QMIN is reduced to zero and the user will be rescheduled. On the other hand, if the IOTA value is 4, a maximum of 15 physical I/Os can be performed before the rescheduling will occur.

PRIOB and PPRIO CONTROL Parameters and SUPER Parameters

The QMIN, QUAN, PQUAN, and IOTA tuning parameters provide a very delicate tuning ability. The PRIOB and PPRIO tuning parameters provide a much coarser tuning ability. The PRIOB and PPRIO tuning parameters are used to specify the base execution priority. The base execution priority can shift dramatically the CPU and I/O resources between access modes and/or users. A higher base execution priority is given to a mode, user, or group of users only if their tasks are to be performed before anything else on the system. A lower base execution priority is given if the tasks are to be performed only after everything at a higher base execution priority has been given CPU resources.

Since the base execution priorities have such a dramatic effect, they must be changed with caution. In particular, no mode, user, or group of users should be given a base execution priority such that they are able to reach an execution priority above the system ghost users. If this situation occurs, the system may hang.

The system manager can give all access modes a default base execution priority of 2. This will allow the system manager to set the actual base priority of an access mode, user, or group of users below the system default base execution priority. Since the system default priority does not have to be changed to do this, all users do not have to be removed from the system. The new lower base execution priority will take effect as the users log onto the specified mode.

The default base execution priority for online, ghost, and TP users is set using the CONTROL processor PRIOB parameter. The online and ghost default base execution priorities are usually set to the same value. The TP default base execution priority is either set the same as or higher than the online value. If the response time for TP users is a major system goal or objective, the TP default base execution priority can be set higher than the online value. For example, if the online value is 2, the TP value can be set to 4.

The default base execution priority for batch users is set using the CONTROL processor PPRIO parameter. A default base execution priority can be established for each batch partition. A batch partition running short, small batch jobs can be given a default base execution priority equal to the online value. A batch partition running large, long batch jobs can be given a default base execution priority less than the online value if this does not violate the system goals and objectives.

In general, no batch default base execution priority should be greater than the online value. If the batch value is greater, online users will encounter great difficulties in completing their online tasks.

The SUPER processor PRIOB parameter can be used to give a user or a group of users a special base execution priority for one or more access modes. These special PRIOB values can be used to give a user or group of users more or less of the CPU resource. If the special PRIOB value is smaller than the default PRIOB (or PPRIO) value, the user or group of users will receive less of the CPU than users with a greater base execution priority. If the special PRIOB value is larger than the default PRIOB (or PPRIO) value, the user or group of users will receive more of the CPU than users with a lower base execution priority.

STATS RESOURCE Display for Resources and Resource Tuning

The STATS RESOURCE display shows where and how much of various system resources are used. For this discussion, the STATS resource display is divided into two parts: monitor resources and memory utilization.

STATS RESOURCE Display for Monitor Resources

The following is an example of a STATS RESOURCE display of monitor resources.

```
            STATS interval from 08:08:36.94 to 15:39:12.98


                 CP-6 monitor resource utilization

{Resource name}          { # in }  {---since system boot---}  { Total   }
                         {use now}  { (max) (min) (average) }  {available}

IOQ packets                 42        110   0        38          110
IOS packets                 102       107   58       96          397
I/O cache entries           3676      4189  3        3337        4096
Enqueue/Dequeue data blocks 622       1791  11       476         2560
Scheduler Do-list entries   0         19    0        0           50
```

Figure 14. STATS RESOURCE Display of Monitor Resources

The first lines in this display show the current usage, minimum, average, and maximum usage since the last system boot, and the total number of various internal monitor resources. With the exception of the I/O cache entries, all resource usage is reported on a single line. The I/O cache entries item is expanded into a table that follows the single line resource usage reports. The items in the I/O cache part of the table are defined below in the section "I/O Cache Tuning".

Resource Tuning

The following system tuning actions should be taken for all resources displayed except I/O cache entries. If the RESOURCE display shows that the maximum usage of a resource is the same as the total number of the resource in the system, the system manager must closely monitor the resource. If the RESOURCE display shows that the current and/or average usage of a resource is at or near the total number of the resource in the system, the system manager must increase the total number of that resource in the system. If the RESOURCE display shows that the maximum usage of a resource never reaches the total number of the resource in the system, the system manager may cautiously decrease the total number of that resource in the system.

DOLIST, ENQ, and QUEUE TIGR Parameters

The total number of the resources displayed in the RESOURCE display (except the I/O cache entries) is controlled by entries on the MON card in the TIGR deck. The total number of a resource is increased by increasing the corresponding parameter via the MON command. The total number of IOQ and IOS resources are controlled by the parameters of the QUEUE option on the MON command. The total number of the Enqueue/Dequeue data blocks is controlled by the parameters of the ENQ option on the MON command. The total number of Scheduler Do-list entries is controlled by the parameter of the DOLIST option on the MON command.

Since these resources are controlled by TIGR parameters and changes to TIGR parameters are not effective until after a reconfiguration boot is performed, these TIGR parameters are usually made slightly larger than required by the system to allow some room for growth, which will eliminate having to make TIGR changes and perform reconfiguration boots frequently.

I/O CACHE Tuning

The I/O cache provides a way to reduce the number of I/Os in the CP-6 system. The system manager can tune the caching system to the particular CP-6 environment. The system manager can control the amount of caching done, as well as the type of caching. The system manager can control:

- the size of the I/O cache table (through the TIGR deck).
- expire times, which determine which type of granules are retained the longest. The expire times may be adjusted individually for each granule type as the system is running (through the CONTROL processor EXPTIME system parameter).
- the update limit, which eliminates many of the writes to disk (through the CONTROL processor UPDLIMIT system parameter).

As the caching system runs, it gathers statistics that indicate how well the system is running, and provides information to help the system manager tune the system to the specific environment. These statistics are made available through the STATS processor.

When a CP-6 system is booted a set of default values for the I/O cache expire times, and cache table size is used. STATS can be used to tune these values, so the I/O cache is used more efficiently.

The I/O cache uses main memory to cache disk granules. If CP-6 memory management needs memory, it may make a request to the I/O cache system for memory. The I/O cache system decides which memory to give to memory management based on the values for expire time. If the values for expire time are set too high, then memory management will have to make hundreds of calls to the I/O cache system to get the memory it needs. If the values for expire time are too low, then the I/O cache will give memory management more memory than it actually needs, and the extra memory will be left unused, when it could be used for caching granules.

STATS I/O CACHE Displays

The STATS processor DISPLAY RESOURCE and SUMMARIZE CACHE commands are used to produce displays that can be used for tuning the cache.

The SUMMARIZE CACHE displays looks as follows:

```
    Interval end  IOC Trnc
    11:43:18.56         0
```

The display indicates the number of truncs per minute. A trunc occurs each time the memory management system requests memory from the I/O cache system. A value of 0 indicates that the I/O cache system is called less than once per minute for memory. High trunc values indicate that the I/O cache is not being used efficiently.

The I/O cache activity table in the STATS RESOURCE display is pictured below.

STATS interval from 08:08:36.94 to 15:39:12.98

I/O cache activity (actions per minute)

	Attempted Gets	Hits UC=0	Hits UC>0	Percent Hits	Attempted Puts	Failed Puts	Unused Pages	
MAD	10	10	0	99	0	0	12	{all}
	111	111	0	99	0	0	11	{snap}
PAD	2	2	0	95	0	0	12	{all}
	19	18	0	95	1	0	15	{snap}
GP	33	29	4	99	0	0	15	{all}
	134	119	15	99	0	0	21	{snap}
FD	174	170	1	98	3	0	244	{all}
	460	433	6	95	22	0	169	{snap}
FIT	207	130	3	64	74	0	164	{all}
	588	453	15	79	121	0	280	{snap}
UL	17	17	1	98	1	0	18	{all}
	75	66	7	96	6	0	36	{snap}
INDEX	74	58	8	88	13	0	83	{all}
	468	346	42	82	107	0	484	{snap}
DATA	331	266	19	86	71	0	697	{all}
	2041	1432	72	73	706	0	2179	{snap}
REL	0	0	0	65	0	0	0	{all}
	0	0	0	0	0	0	0	{snap}
CONSEC	40	28	0	70	36	0	2512	{all}

Figure 15. STATS RESOURCE Display of I/O Cache Activity Table (cont. next page)

	161	90	6	59	175	0	208	{snap}
ELSE	13	0	0	0	14	0	0	{all}
	33	0	0	0	36	0	0	{snap}
Total	902	711	35	82	210	0	3757	{all}
	4094	3071	162	78	1173	0	3403	{snap}

Figure 15. STATS RESOURCE Display of I/O Cache Activity Table

The left column indicates the types of granules that are being cached. (See the table on I/O cache granule types.) The Attempted Gets column contains the number of times an attempt was made to get an item from the cache. A Hit occurs when the attempt was successful. The Attempted Puts column contains the number of times an attempt was made to put an item into the cache; Failed Put is the number of times the attempt failed. The Unused Pages column contains the number of pages in the I/O cache that are currently not in use.

The Failed Puts column is useful for determining if the size of the I/O cache table, as set by the TIGR command, was large enough. If any of the rows indicate non-zero values regularly, then the size of the cache should probably be doubled.

The Unused Pages column, and the truncs information from the SUMMARIZE CACHE display are used for tuning the CONTROL processor EXPTIME parameter. Items that have a low number of unused pages should probably remain in the cache longer, and therefore have higher values for expire times. Items with a higher number of unused pages should probably have lower expire times. Evaluation starts with the item that has the highest number of unused pages. Typically, the DATA row will have over half of the unused pages in the cache. If the cache is tuned properly, the expire time for DATA items will be about half of the time between cache truncs. In this case, whenever the memory management requests memory from the I/O cache, about half of the I/O cache will be freed for use by memory management.

Table 9. I/O Cache Granule Types

Granule Types	Description
CONSEC	Consecutive files, which contain data records(only).
DATA	Data granules, which contain keyed file data.
ELSE	All other granules (including symbiont files and unit record files).
FD	File Directories, each of which contains a list of all files in an account.
FIT	File Information Table, containing access controls and other file specific information for all files.
GP	Granule Pool, a list of free granules on the packset.
INDEX	Index granules, which contain the record keys for keyed files.
MAD	Master Account Directory, a global list of all accounts on the system.

Table 9. I/O Cache Granule Types (cont.)

Granule Types	Description
PAD	Packset Account Directories, each of which contains a list of all accounts in the packset.
REL	Relative files, which contain fixed length data records (only).
UL	Upper level index, which contains pointers to index granules.

For example, suppose during the system's peak load, the number of truncs per minute is about 50. This means that the I/O cache is truncated about once every 1.2 seconds. The system manager can set the value for the DATA expire time to about 60 (the value is in hundredths of a second) and use the Unused Memory column to set the remaining values. If the MAD row has about 1/100 the unused pages of the DATA row, the system manager can set the expire time for the MAD to about 6000.

After the new values are set, the cache should be monitored again. The new expire times will affect how often the cache will be truncated. This first guess may be too low, or too high. Eventually, a balance will be reached. Exact values for expire times will not be possible, since the I/O load on a system often varies depending on the time of day, or day of the week. Two or three iterations of the above procedure will probably be sufficient.

STATS RESOURCE Display for Memory and Memory Tuning

The second part of the STATS RESOURCE display shows how system memory is used. The STATS RESOURCE display and the STATS USER SIZE histogram are used to set the resource memory utilization tuning parameters.

STATS RESOURCE Display for Memory Utilization

In the following example of a STATS RESOURCE display of memory utilization, note the zero-valued items. Zero-value items are normally suppressed. They are included here to acquaint the reader with these possible display items.

```
STATS interval from 08:08:36.94 to 15:39:12.98

CP-6 memory utilization

AARDVARK and RECOVERY                                45
XDELTA and monitor debug schema                     112
Monitor procedure and static data                   259
Monitor context (JITs, HJITs, PPUT, page tables)     47
Monitor dynamic data segments                        27
TIGR-built tables                                    50
Communications WSQs                                  30
Comgroup queue                                       51
Total pages held back for monitor use                15
Resident system ghosts                              384
Required processors (IBEX, DELTA, LOGON)            228
All other special shared (resident) processors      599
                                                    ----
Total dedicated memory                             1847

Available to users                                 6345
Currently allocated to users                       1620
Automatically shared run units in use               394
Shared data segments in use                          10
Free pages                                           99
Automatically shared run units not in use           685
I/O cache pages (Use Count = 0)                    3402
Total pages currently available                    4201
Suspected bad physical pages                          0
Physical pages being tested                           0
Confirmed bad physical pages                          0
I/O cache pages                                    3529
Number of pages not accounted for                     8
Total physical pages in system                     8192
```

Figure 16. STATS RESOURCE Display of Memory Utilization

The items in this part of the STATS RESOURCE display are described in the following table. For a GLOM data reduction, this part of the RESOURCE display will show the average memory usage for various system functions. During operation, the total dedicated memory remains constant. The remaining memory categories are changing very rapidly.

Table 10. STATS RESOURCE Memory Display Definitions

Item	Definition
AARDVARK and RECOVERY	Pages required for the boot processor (AARDVARK) and the automatic recovery processor (RECOVERY).
XDELTA and monitor debug schema	Pages required for the monitor debugger (XDELTA) and the debug schema used to debug the monitor. The size of the debug schema area may change by specifying different values to the FUNCTIONAL CODE GROUPS question at boot time.
Monitor procedure and static data	Pages required for the monitor executable procedure (i.e., PROC:%...) and static data (i.e., DCL%...%STATIC).
Monitor context (JITs, HJITs, PPUT, page tables)	Pages required for the monitor context area, which includes the monitor Job Information Table (JIT), House-keeping JIT (HJIT), physical page use table (PPUT), and page tables (PT).
Monitor dynamic data segments	Pages required for monitor dynamic segments, which includes CFUs, LDCTs, ASAVE and ENQ.
TIGR-built tables	Pages required for tables built by TIGR at boot time. This includes the user table, I/O cache, device tables and autoshare tables.
Communications WSQs	Pages used in communication with FEPs. These pages are controlled by the INQSZ and OUTQSZ parameters on the TIGR processor FEP command.
Comgroup queue	An area for comgroup context.

Table 10. STATS RESOURCE Memory Display Definitions (cont.)	
Item	Definition
Total pages held back for monitor use	Pages reserved for CFUs, LDCTs, ASAVE, ENQ and stealable pages. These pages are affected by the CFU, DEVMAX, ENQ, and STEALPGS options on the TIGR processor MON command.
Resident system ghosts	Pages required for resident system ghosts. These are ghost users that perform part of the operating system's functions such as ELF, PIG, SLUG, INSYM, and CUTSYM (and are referred to as the MING ghosts).
Required processors (IBEX, DELTA, LOGON)	Pages required for the IBEX, DELTA, and LOGON processors. These processors must be present for the system to function.
All other special shared (resident) processors	Pages required for all processors, shared libraries, alternate shared libraries, and debuggers. These items are specified on the SPROC options on the TIGR processor MON command.
Total dedicated memory	Total pages required by the monitor, its tables, ghosts, and processors.
Available to users	Pages available to users. This number is the difference between the total pages in the system and the total dedicated memory.
Currently allocated to users	Pages that are currently actually allocated to users. These pages cannot be shared. These pages plus the automatically shared run unit pages in use, plus the shared data segments in use constitute all user memory.

Table 10. STATS RESOURCE Memory Display Definitions (cont.)

Item	Definition
Automatically shared run units in use	Pages of procedure of automatically shared run units that are currently being used by one or more users.
Shared data segments in use	Pages of shared data segments currently being used by one or more users.
Free pages	Pages in the system not currently used for any purpose.
Automatically shared run units not in use	Pages of the procedure of automatically shared run units that are not being used by any user. The pages are candidates to be used for other purposes if the free pages are exhausted.
I/O cache pages (Use Count=0)	I/O cache pages not currently being used.
Total pages currently available	Pages on the system that are currently available for use. This number is the sum of the free pages and pages for the automatically shared run units not in use.
Suspected bad physical pages	Pages that have been marked as suspect by TOLTS.
Physical pages being tested	Pages that are currently being tested by TOLTS.
Confirmed bad physical pages	Pages that have been partitioned by SYSCON.

Table 10. STATS RESOURCE Memory Display Definitions (cont.)	
Item	Definition
I/O cache pages	Total I/O cache pages, both used and unused.
Number of pages not accounted for	Pages that cannot be currently accounted for. Because of the dynamic page usage of CP-6, pages are constantly being assigned to new functions or usages. This number reflects the number of pages that are currently in a status that STATS is unaware of.
Total physical pages in system	Total pages in the system. This is the system memory size as found by AARDVARK at the last system boot or recovery.

STATS USER SIZE Histogram

The STATS USER SIZE HISTOGRAM provides detailed information about the memory sizes of all users. It creates a histogram plot from the memory sizes of all users in the system. This histogram provides:

- a detailed report of the memory currently assigned to users in the STATS RESOURCE display for memory.
- a count and a percentage of the memory sizes for various memory size ranges.

The following figure is an example of a STATS USER SIZE histogram.

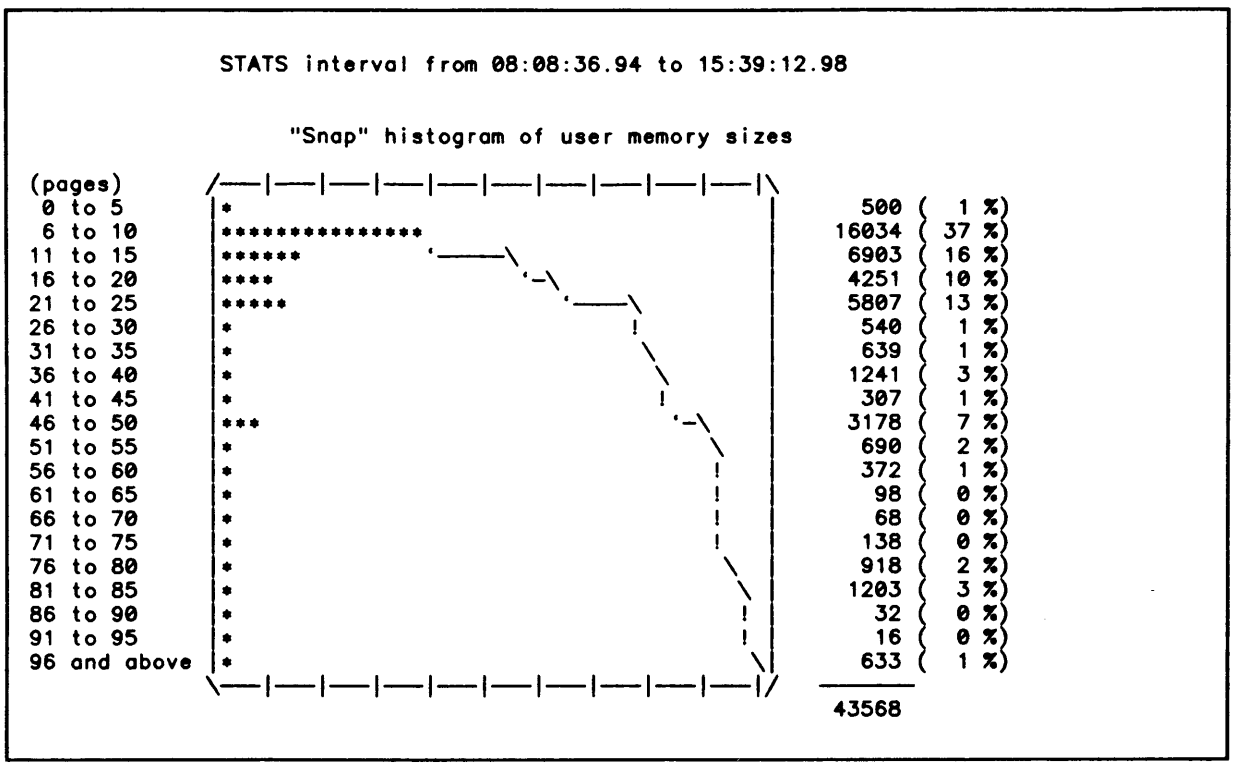


Figure 17. STATS USER SIZE Histogram

Memory Tuning

The CONTROL processor AUTOSHARE and MAXMM parameters are used in memory tuning.

AUTOSHARE CONTROL Parameter

The CP-6 system was designed and developed to share a single copy of the procedure area among all users of a program. However, the system manager can control whether or not the procedure area for run units is shared. The sharing of the procedure area in run units is controlled by the CONTROL parameter AUTOSHARE. Either none, some, or all procedure areas can be shared. The default is that some procedure areas are shared based upon the LINK options in the run unit.

Use of the NONE option is not advised unless absolutely required by system goals and objectives. Use of this option will dramatically increase memory requirements, and will also cause a degradation in system performance.

MAXMM CONTROL Parameter

Because of the extensive use of memory sharing, the default maximum memory for each mode is greater than the total physical memory on the system. However, the system manager can specify the total amount of memory available to each mode. The total amount of memory for each mode is specified using the CONTROL parameter MAXMM. In most normal system operations, the system manager will not need to change the default setting of these parameters.

Limiting the total amount of memory for online, ghost, and TP users can have undesirable results. Ghost and TP users may not be able to recover from a memory limit exceeded error. Online users may acquire memory to do their tasks and then not release the extra memory when no longer required, which can lead to reduced memory utilization on the system.

The total amount of memory for batch users can be limited. Since batch jobs must specify the maximum amount of memory they will use on their RESOURCE command, batch jobs will not be aborted because a total memory limit is exceeded. However, batch jobs will still be aborted if their individual memory limit is exceeded.

The total amount of memory for batch users can be limited to control the batch memory usage regardless of hom many batch jobs are executing. For example, total batch memory could be restricted to 1024 pages (i.e., 1024KW). This means that if 768KW of batch memory is already in use, a job requiring more than 256KW of memory will not be eligible to begin execution. However, a job requiring 256KW or less memory will be eligible to begin execution.

STATS DEVICE Display and Overload Problems

The STATS DEVICE display provides detailed information about input/output activity on IOM-connected consoles, controllers, and devices connected to IOM controllers.

STATS DEVICE Display

The following is an example of part of a STATS DEVICE display.

```
                STATS interval from 08:08:36.94 to 15:39:12.98

             # of     connects      %       %       %       %       load
 name      connects  per min.     idle    wait    busy    backlog  factor

SC010000     4387         9      100.0     0.0     0.0     0.0      0.0   {snap}
             16159        2      100.0     0.0     0.0     0.0      0.0   {all}

DC010000     168217     373      100.0     0.0     0.0     0.0      0.0   {snap}
             450138      69      100.0     0.0     0.0     0.0      0.0   {all}

DC020000     341922     758      100.0     0.0     0.0     0.0      0.0   {snap}
             903882     138      100.0     0.0     0.0     0.0      0.0   {all}

DP010000     151114     335       83.9     0.1    12.8     3.3     20.9   {snap}
             426567      65       97.1     0.0     2.4     0.5     18.3   {all}
```

Figure 18. STATS DEVICE Display (cont. next page)

DP020000	143869	319	80.4	0.1	15.9	3.6	18.9	{snap}
	293425	45	97.6	0.0	2.0	0.4	16.8	{all}
DP030000	167480	371	81.0	0.1	15.8	3.1	16.8	{snap}
	564699	86	96.5	0.0	3.1	0.4	12.4	{all}
DP040000	6316	14	99.5	0.0	0.5	0.0	0.6	{snap}
	9030	1	100.0	0.0	0.0	0.0	0.6	{all}
DP050000	4709	10	99.5	0.0	0.4	0.1	14.1	{snap}
	7406	1	99.9	0.0	0.0	0.0	13.9	{all}
DP060000	24	0	100.0	0.0	0.0	0.0	1.3	{snap}
	6838	1	99.9	0.0	0.1	0.0	26.5	{all}
DP070000	140	0	100.0	0.0	0.0	0.0	1.1	{snap}
	6789	1	100.0	0.0	0.0	0.0	6.8	{all}
DP080000	48	0	99.8	0.2	0.0	0.0	98.3	{snap}
	70	0	100.0	0.0	0.0	0.0	98.7	{all}
DP090000	3174	7	99.5	0.0	0.5	0.0	1.0	{snap}
	5929	0	99.9	0.0	0.1	0.0	5.4	{all}
DP100000	12602	27	98.8	0.0	1.0	0.1	10.0	{snap}
	12603	1	99.9	0.0	0.1	0.0	10.0	{all}
DP110000	20663	45	98.3	0.0	1.6	0.1	7.5	{snap}
	20664	3	99.9	0.0	0.1	0.0	7.5	{all}
TC010000	82331	182	100.0	0.0	0.0	0.0	0.0	{snap}
	264303	40	100.0	0.0	0.0	0.0	0.0	{all}
MT020000	575	1	99.8	0.0	0.2	0.0	1.2	{snap}
	575	0	100.0	0.0	0.0	0.0	1.2	{all}
MT030000	14745	32	98.3	0.1	1.6	0.0	4.3	{snap}
	79599	12	98.9	0.0	1.0	0.0	5.1	{all}
MT040000	51585	114	90.0	0.2	9.8	0.0	1.7	{snap}
	121087	18	98.3	0.0	1.7	0.0	2.6	{all}
MT050000	15426	34	98.7	0.1	1.2	0.0	4.7	{snap}
	63042	9	99.3	0.0	0.7	0.0	7.4	{all}
UC010000	3743	8	100.0	0.0	0.0	0.0	0.0	{snap}
	9508	1	100.0	0.0	0.0	0.0	0.0	{all}
LP010000	2386	5	76.1	0.0	23.9	0.0	0.0	{snap}
	7253	1	95.6	0.1	4.3	0.0	1.8	{all}
LP020000	1357	3	88.2	0.0	11.8	0.0	0.0	{snap}
	2255	0	98.6	0.0	1.4	0.0	1.4	{all}
UC020000	24	0	100.0	0.0	0.0	0.0	0.0	{snap}
	24	0	100.0	0.0	0.0	0.0	0.0	{all}
CR010000	24	0	100.0	0.0	0.0	0.0	95.6	{snap}
	24	0	100.0	0.0	0.0	0.0	95.6	{all}

Figure 18. STATS DEVICE Display

The following table describes the headings in this STATS DEVICE display. For a GLOM data reduction, there will be an all and a snap line for each device. The all line shows the average device activity since the last system boot or recovery. The snap line shows the average device activity for the requested interval.

Table 11. STATS DEVICE and CHANNEL Display Definitions

Heading	Definition
# of connects	The number of connects to the controller, device, or channel. A connect is performed at the beginning of an input/output request. A connect may contain more than one input/output operation.
connects per min.	The number of connects per minute to the controller, device, or channel; that is the number of connects divided by the number of minutes in the interval.
% idle	The percentage of time the controller, device, or channel is idle. Idle means no input/output is in progress and no input/output is waiting. For controllers this value is always 100%.
% wait	The percentage of time the controller, device, or channel is waiting. Waiting means no input/output is in progress, and one or more input/output requests are waiting but cannot be started because no IOM channel or controller slot is available. For controllers and channels, this value is always 0%.
% busy	The percentage of time the controller, device, or channel is busy. Busy means input/output is in progress and no other input/output request is waiting. For controllers, this value is always 0%.

Table 11. STATS DEVICE and CHANNEL Display Definitions (cont.)	
Heading	Definition
% backlog	The percentage of time the controller, device, or channel input/output is backlogged. Backlog means one input/output is in progress and one or more input/output requests are waiting. For controllers and channels, this value is always 0%.
load factor	The percentage of time an input/output request for a controller, device, or channel will be queued instead of initiated immediately. This percentage is calculated as LOAD FACTOR = 100% * (wait + backlog) / (100 - idle). For controllers and channels, this value is always 0%.

Handling Overload Problems

The sum of the idle, wait, busy, and backlog percentages should be 100%. If the sum is not exactly 100%, the problem is usually caused by rounding errors.

The load factor can be converted from a percentage to a probability by dividing by 100. Then the load factor can be viewed as the probability that an input/output request will be queued instead of initiated immediately.

The load factor can be used to determine if a disk spindle is being overloaded. If this figure is greater than 50% for any individual disk spindle, there is a substantial amount of contention for access to that spindle. Users attempting to access that spindle will suffer limited input/output throughput. If this figure is greater than 75% for any individual disk spindle, the throughput limitations will be very severe.

There are no system tuning parameters than can correct a disk spindle overloading problem. Rather, the system manager may move some heavily used accounts to another packset. The system manager determines the files and accounts to be moved by using the ANLZ processor CFU command when the overloaded situation is occurring on the disk spindle. The system manager then uses PIG and/or EFT to move the heavily used account to another packset on another disk spindle.

If the account cannot be moved to another packset (e.g., the accounts must reside on DP#SYS), the system manager may change the physical organization of the packset. This is done by using PIG either to EXTEND the existing packset or to SCRATCH and then BUILD a new packset spread across additional spindles. By spreading a packset across additional spindles, greater input/output throughput for the packset can be realized through the additional disk access mechanisms.

While extending or building of a packset across multiple spindles provides additional input/output throughput for the packset, it also causes a reduced reliability for the packset. If one volume of a multivolume packset cannot be mounted, the entire packset is unavailable for use. Therefore, the additional input/output throughput is gained for a packset at the cost of reduced reliability for the packset. The system manager must use the system goals and objectives to determine whether the disk overloading problem should be solved by building a multivolume packset on multiple spindles.

STATS CHANNEL Display and Channel Loading Problems

The STATS CHANNEL display provides detailed information about input/output activity on all IOM channels connected to consoles and controllers.

STATS Channel Display

The following is an example of a STATS CHANNEL display.

STATS interval from 08:08:36.94 to 15:39:12.98

IOM-chan number	# of connects	connects per min.	% idle	% wait	% busy	% backlog	load factor	
0-08	41798	92	95.2	0.0	4.8	0.0	0.0	{snap}
	112750	17	99.3	0.0	0.7	0.0	0.0	{all}
0-09	42275	93	95.3	0.0	4.7	0.0	0.0	{snap}
	112272	17	99.3	0.0	0.7	0.0	0.0	{all}
0-10	42213	93	95.3	0.0	4.7	0.0	0.0	{snap}
	111541	17	99.3	0.0	0.7	0.0	0.0	{all}
0-11	41931	93	95.3	0.0	4.7	0.0	0.0	{snap}
	113571	17	99.3	0.0	0.7	0.0	0.0	{all}
0-12	43314	96	95.0	0.0	5.0	0.0	0.0	{snap}
	113517	17	99.2	0.0	0.8	0.0	0.0	{all}
0-13	43308	96	95.1	0.0	4.9	0.0	0.0	{snap}
	113344	17	99.2	0.0	0.8	0.0	0.0	{all}
0-14	42443	94	95.2	0.0	4.8	0.0	0.0	{snap}
	112298	17	99.2	0.0	0.8	0.0	0.0	{all}
0-15	43217	95	95.0	0.0	5.0	0.0	0.0	{snap}
	112723	17	99.2	0.0	0.8	0.0	0.0	{all}
0-16	41167	91	93.3	0.0	6.7	0.0	0.0	{snap}
	133127	20	98.3	0.0	1.7	0.0	0.0	{all}
0-17	41164	91	93.9	0.0	6.1	0.0	0.0	{snap}
	131176	20	98.3	0.0	1.7	0.0	0.0	{all}
0-20	86427	191	92.4	0.0	7.6	0.0	0.0	{snap}
	349929	53	97.8	0.0	2.2	0.0	0.0	{all}
0-21	86542	192	92.4	0.0	7.6	0.0	0.0	{snap}
	350190	53	97.8	0.0	2.2	0.0	0.0	{all}

Figure 19. STATS CHANNEL Display (cont. next page)

0-22	86598	192	92.4	0.0	7.6	0.0	0.0	{snap}
	349824	53	97.8	0.0	2.2	0.0	0.0	{all}
0-23	86136	191	91.8	0.0	8.2	0.0	0.0	{snap}
	349372	53	97.8	0.0	2.2	0.0	0.0	{all}
0-24	2386	5	71.1	0.0	28.9	0.0	0.0	{snap}
	7253	1	94.8	0.0	5.2	0.0	0.0	{all}
0-25	1357	3	87.6	0.0	12.4	0.0	0.0	{snap}
	2255	0	98.5	0.0	1.5	0.0	0.0	{all}
0-26	24	0	99.7	0.0	0.3	0.0	0.0	{snap}
	24	0	100.0	0.0	0.0	0.0	0.0	{all}
0-28	1845	4	85.3	0.0	14.7	0.0	0.0	{snap}
	2420	0	98.6	0.0	1.4	0.0	0.0	{all}
0-30	4387	9	84.5	0.0	15.5	0.0	0.0	{snap}
	16159	2	96.5	0.0	3.5	0.0	0.0	{all}

Figure 19. STATS CHANNEL Display

The previous table describes the headings in this display. For a GLOM data reduction, there will be an all and a snap line for each channel. The all line shows the average channel activity since the last system boot or recovery. The snap line shows the average channel activity for the requested interval.

Handling Channel Loading Problems

There are no system tuning parameters that can dramatically change the channel loading. Channel loading can be effectively changed by moving disk or tape devices to another subsystem. (Each subsystem is specified by a DISK or TAPE command in TIGR.) Channel loading can also be changed by adding additional physical and/or logical channels to controllers.

STATS PROCESSOR Display and Processor Tuning

The STATS PROCESSOR display provides detailed information about the CPU usage by shared processors (i.e., run units).

STATS Processor Display

The following is an example of a STATS PROCESSOR display.

STATS interval from 08:08:36.94 to 15:39:12.98

Processor name	Type	Users	snap % exec	snap % serv	all % exec	all % serv
LOGON	icp	6	0.0	0.2	0.0	0.0
IBEX	icp	28	3.2	12.2	0.9	3.2
TPCP	icp	2	0.0	0.0	0.0	0.0
DELTA	idb	2	2.0	1.6	0.1	0.1
PIG	std	1	0.0	0.2	0.0	0.0
PCL	std	0	3.8	24.1	1.2	9.7
DOG	std	1	0.0	0.1	0.0	0.0
GOOSE	std	1	0.0	0.0	0.0	0.0
JAYS	std	1	0.0	0.0	0.0	0.0
THING	std	1	0.3	3.1	0.0	0.2
EDIT	std	12	5.1	19.1	0.7	2.3
STARGHDSC	std	1	7.3	4.6	0.9	0.5
CBAU	std	1	0.0	0.0	0.0	0.0
STARGHST	std	1	15.7	11.8	4.9	3.2
STATS	std	4	0.0	0.2	0.0	0.1
STARLOG	std	1	1.3	2.0	0.3	0.5
MAIL	std	7	1.4	9.6	0.2	0.9
SEND	std	1	0.1	0.9	0.0	0.1
TEXT	std	0	1.4	0.4	4.1	0.2
SOLAR	std	2	0.2	0.5	0.0	0.1
IMP	std	0	0.2	0.3	0.0	0.0
EDGEMARK	std	1	0.0	0.2	0.0	0.0
ARES	std	0	2.8	4.7	0.5	0.7
6EDIT	std	0	1.1	0.5	0.1	0.1
GOPHER	std	0	1.0	2.4	0.1	0.2
MODMOVE	std	0	0.0	0.1	0.0	0.0
ANLZ	std	0	0.1	0.3	0.0	0.0
APL	std	0	3.2	0.3	0.2	0.0

Figure 20. STATS PROCESSOR Display

The following table describes the headings in the STATS PROCESSOR display. For a GLOM data reduction, the all column shows the average CPU usage since the last system boot or recovery. The snap column shows the average CPU usage for each processor during the requested interval.

Table 12. STATS PROCESSOR Display Definitions

Heading	Definition
Processor name	The processor name is taken from the name of the run unit. The account is not reported as part of the processor name. If run units with the same name are run from separate accounts, there will be multiple entries with the same processor name.
Type	Specifies the type of the processor. The type is one of the following: icp Interactive command program. This program executes in the command processor domain. idb Interactive debugger. This processor executes in the debugger domain. std standard run unit. This processor executes in the user domain.
Users	Specifies the number of users of the processor. For the GO or REPLAY commands, this is the number of users of the processor during the interval. For the GLOM command, this is the average number of users of the processor during the interval.
% exec	Specifies the percentage of CPU time spent in execution in the processor for all users of the processor. For the snap column, this is the percentage of CPU execution time used by this processor during the interval. For the all column, this is the average percentage of CPU execution time used by this processor since the last system boot or recovery.
% serv	Specifies the percentage of CPU time spent in processing monitor service requests for this processor for all users of the processor. For the snap column, this is the percentage of CPU service time used by this processor during the interval. For the all column, this is the average percentage of CPU service time used by this processor since the last system boot or recovery.

Processor Tuning

The STATS PROCESSOR display shows the percentage of CPU time that is being used by various shared run units for all users of the run units. The shared run units that are reported upon are the most used shared run units. The system manager can use this display in several ways.

This display can be used to ensure that multiple copies of the same program are not used from separate accounts. The system manager should attempt to collect all commonly used programs and place them in one or more library accounts. If commonly used programs are used from common accounts, memory requirements will be decreased and system performance and throughput will be increased.

This display can also be used to examine those installation-supplied run units that are consuming large amounts of CPU time. In this case, the system manager should examine the efficiency of the installation-supplied run units. The system manager can do this by using the PMON and PMDISP tools in the X account. If heavily used installation-supplied run units have sections where efficiency can be improved, system performance and throughput can be increased by improving these programs.

This display can also be used to control the maximum number of concurrent users of a processor. If the amount of CPU usage of a processor violates the system goals and objectives, the system manager can control the maximum number of concurrent users of a processor. This control is started by creating a pseudo resource via the TIGR processor MON command. Then, the run unit is modified by CONTROL or by relinking with a LINK option to require the pseudo resource. The system manager can then use CONTROL to control the maximum number of the pseudo resource in each of the access modes. The system manager can use SUPER also to control which users are able to acquire the pseudo resource.

This process means that the user must acquire a pseudo resource before running the program. In batch mode, the pseudo resource must be requested via the IBEX processor RESOURCE command. In the online mode, the user must acquire the resource by using the IBEX processor ACQUIRE or ORES command.

STATS FEP SUMMARY Display and FEP Tuning

The STATS FEP SUMMARY display summarizes the performance of the FEP(s) on the system.

STATS FEP Summary Display

The following is an example of the STATS FEP SUMMARY display.

FRI. NOV 09 '84 at 08:57	FEP 04	FEP 08	FEP 32	FEP 33
Async terminals	13	14	11	13
Async bytes output/min	4278	1584	524	3560
Async bytes input/min	148	42	19	125
RBT terminals	10	8	0	0
RBT bytes output/min	1240	1623	0	0
RBT bytes input/min	479	75	0	0
urp devices	0	0	3	1
urp bytes output/min	0	0	1500	0
% FEP busy	32	12	6	23

Figure 21. STATS FEP SUMMARY Display

For the GLOM data reduction, these numbers are the averages for the specified period. The system manager uses these number to set various FEP tuning parameters. The STATS processor FEP DATA or FEP EXTENDED commands can be used to create more detailed displays on FEP performance.

Tuning FEPs

Both NETCON and TIGR processor parameters can be used to support FEP tuning.

BLOCK and UNBLOCK NETCON Parameters

If the STATS FEP SUMMARY display shows that one type of terminal or device is doing too much output, the system manager can throttle the line(s) and/or terminal or device type. The system manager can also increase the throughput for certain line(s) and/or terminal or device type(s). This throttling or unthrottling is done by using the NETCON processor BLOCK and UNBLOCK parameters.

The system manager can use the BLOCK and UNBLOCK options on the NETCON processor CONFIG command to throttle a particular line. The system manager can use the BLOCK and UNBLOCK options on the NETCON processor DEFAULT command to throttle a particular type of terminal or device. The system manager can use the BLOCK and UNBLOCK options on the NETCON processor SET command to throttle asynchronous terminals based upon their line speed.

The system manager uses these BLOCK and UNBLOCK parameters to speed up or slow down the effective line speed of a terminal or device. The effective line speed needs to be changed only if necessary to meet the system goals and objectives.

BUFSIZE NETCON Parameter

The system manager can use the BUFSIZE option on the NETCON processor CONFIG and DEFAULT commands to change the default buffer size of an asynchronous line. This parameter needs to be specified for those lines on which an asynchronous device (e.g., a PC) is transmitting to the FEP at line speed. When a device transmits to the FEP at line speed, the default buffer size needs to be changed to equal the largest record sent from the device to the FEP. This prevents the FEP from losing input characters while trying to switch to a larger input buffer.

INQSZ TIGR Parameter

The system manager can control the number of pages of host memory that is to be used for the input circular queue for each local FEP. A minimum of two pages should be specified. If the local FEP has one or more remote FEPs connected to it or if the local FEP has several high-speed input devices in operation concurrently (e.g., HASP link, PC file transfer, etc.), more than two pages should be specified.

The number of pages in the input circular queue is specified by the INQSZ option on each TIGR processor FEP command. If INQSZ is not specified, a value of one is used. The maximum value that may be specified is four.

OUTZSZ TIGR Parameter

The system manager can control the number of pages of host memory that is to be used for the output circular queue for each local FEP. A minimum of two pages should be specified. If the local FEP has one or more remote FEPs connected to it or if any FEP programs are used on the FEP or its RFEP(s), more than two pages should be specified.

The number of pages in the output circular queue is specified by the OUTQSZ option on each TIGR processor FEP command. If OUTQSZ is not specified, a value of one is used. The maximum value that may be specified is four.

STATS STATISTICS Display

The STATS processor STATISTICS command can be used to display the minimum, maximum, average, and standard deviation of all STATS items. The following is an example of the STATS STATISTICS display.

STATS interval from 08:08:36.94 to 15:39:12.98

Mean	Std dev.	Minimum	Maximum	Item or expression
1.3	0.4	1	2	ETMF
51.6	14.3	25	100	90% resp
11.2	6.4	4	23	I/O load
95.9	14.8	67	114	Users
3.1	1.0	1	4	Batch
74.2	14.0	47	92	Online
18.3	0.6	18	20	Ghost
0.4	0.8	0	2	TP
89.9	27.5	1	114	% exec
147.6	50.8	3	202	% serv
16.1	4.7	0	19	% mon
9.1	20.9	0	85	% idle
18.9	20.4	0	65	% I/O
0.1	0.2	0	1	% Res
0.0	0.0	0	0	% IOtres
3040.7	923.7	53	3638	Scheds
98.9	37.4	1	149	Ints
5449.1	1587.1	98	6580	Events
34803.1	3567.6	807	52303	PMMEs
3.0	0.0	3	3	CPUs
2695.6	780.1	51	3332	I/Os
2503.3	741.7	51	3134	Disk IOs
171.6	148.3	0	523	Tape IOs
20.8	14.9	0	48	Misc IOs
3593.8	531.7	2763	4813	Free Pgs

Figure 22. STATS STATISTICS Display

The values shown in the STATISTICS command display are for the selected interval. While the GLOM command displays do their calculation using only the first and last STATS log record within the interval, the STATISTICS display is based upon calculations using every record within the interval. Therefore, the STATISTICS values are generally more accurate than the GLOM values. The STATISTICS values are used as more detailed information for the various GLOM displays.

Index

HONEYWELL INFORMATION SYSTEMS
Technical Publications Remarks Form

TITLE: CP-6
SYSTEM MANAGER HANDBOOK

ORDER NO. CE60-00

DATED MARCH 1985

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 _____

_____

LEASE FOLD AND TAPE–
OTE: 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

Together, we can find the answers.

Honeywell

Honeywell Information Systems
U.S.A.: 200 Smith St., MS 486, Waltham, MA 02154
Canada: 155 Gordon Baker Rd., Willowdale, ON M2H 3N7
U.K.: Great West Rd., Brentford, Middlesex TW8 9DH **Italy**: 32 Via Pirelli, 20124 Milano
Mexico: Avenida Nuevo Leon 250, Mexico 11, D.F. **Japan**: 2-2 Kanda Jimbo-cho, Chiyoda-ku, Tokyo
Australia: 124 Walker St., North Sydney, N.S.W. 2060 **S.E. Asia**: Mandarin Plaza, Tsimshatsui East, H.K.

42646, 5C485, Printed in U.S.A.

CE60-00