Operating System

Bull DPS 7000

Getting Started With GCOS 7

Book 3 -
Operating your System

GCOS 7

Software

47 A2 22US Rev04

Preface

SCOPE AND OBJECTIVES

The three compact user's guides, *Getting Started With GCOS 7:*

- *Book 1, Setting Up Your System*

- *Book 2, Managing Your System*

- *Book 3, Operating Your System*

provide a simple approach to using GCOS 7 on the DPS 7000 computer. They explain the most important tasks for setting up and managing a standard GCOS 7 system. Background information on these tasks is provided only where it is necessary to clarify the purpose of a specific action.

Generally, you execute one or a series of commands. Often, these commands contain several parameters, with or without default values. These guides present command parameters for which there are no default values. They concentrate, therefore, on what you *must* do rather than on what you *can* do.

Whenever you need to depart from the standard approach to carrying out a specific task, you should refer to the relevant detailed documentation.

INTENDED READERS

These guides are for users of a DPS 7000 system running with the GCOS 7 operating system, release V5 or V6. You should have either some knowledge of microcomputers or of previous versions of GCOS 7. Many of the tasks described in these guides are normally performed by a System Administrator or main operator. You can gain a basic understanding of these tasks by working from the *Getting Started* guides.

HOW TO USE THESE GUIDES

Except for the first four chapters of Book 1, you don't have to read *Book 1, Book 2,* and *Book 3* in order. The division into three books provide a simple way of referring from one book to another. You will normally use all three books together.

Here are brief summaries of what is in each book.

Book 1 - Setting Up Your System explains how to use the System Console of the DPS 7000 and how to prepare GCOS 7 for everyday use. Since some knowledge of how to work under the Interactive Operator Facility (IOF) is necessary to perform these tasks, you should refer where necessary to *Book 3*. Your initial set-up also may include tasks discussed in *Book 2*.

The first four sections of Book 1 introduce the DPS 7000 hardware and software. You should read these sections first and carry out all the described procedures.

Book 2 - Managing Your System describes important tasks and ideas that relate to management of the system by the user acting as System Administrator. Note, however, that certain tasks described in *Book 2* may be part of your initial set-up.

Book 3 - Operating Your System describes the basic IOF activities that you do on a daily basis. You should refer to *Book 3* regularly while performing tasks described in *Books 1* and *2*.

STRUCTURE OF THE GUIDES

Book 1 - Setting Up Your System has eight sections.

Section 1	describes the physical components of your DPS 7000.
Section 2	tells you how to use the System Console and the window environment of the DPS 7000.
Section 3	tells you how to use GCOS 7 - the operating system of the DPS 7000.
Section 4	gives a detailed description of the hardware and software components of your DPS 7000.
Section 5	describes the main tasks of the System Administrator and tells you where each task is documented.
Section 6	explains how to set up the site catalog.
Section 7	explains how to build a library and create library members.
Section 8	explains how to write startup sequences.

Book 2 - Managing Your System has four sections and one appendix.

Section 1	provides an overview of the concept of jobs.
Section 2	tells you how to create UFAS indexed-sequential files, and how to manage files and volumes.
Section 3	explains how to manage catalogs and set access rights.
Section 4	describes how to create extra system files, update system domains, and use the back-up Production Set.
Appendix A	provides a list of all system files, with a description of each file and its volume location.

Book 3 - Operating Your System has three sections and one appendix.

Section 1   describes tasks related to starting and stopping the system every day.
Section 2   describes important concepts and conventions for working under IOF.
Section 3   describes some of the principal tasks of a main operator.
Appendix A   provides a listing of a sample IOF session.

NOTE: For ease of use, the glossary and index, which cover all three books, are repeated in each book.

ASSOCIATED DOCUMENTS

The DPS 7000 is delivered with the following documentation.

Four basic manuals

- Getting Started With GCOS 7:

- Book 1 - Setting Up Your System............................ 47 A2 20US
- Book 2 - Managing Your System............................ 47 A2 21US
- Book 3 - Operating Your System............................ 47 A2 22US

Information Directory (Rev3 - V5 or Rev4 -V6).........47 A2 11UG

Ten manuals required for further tasks

Simple Generation Guide Using FPG7.....................39 A2 46DM
Getting Started with Telecommunications 47 A2 70UC
System Installation Configuration and ....
Updating Guide (V6) 47 A2 17US
or
System Installation Configuration and ....
Updating Guide (V5) 47 A2 16US
GCOS 7 - V6 Software Release Bulletin 47 A2 15UG
or
GCOS 7 - V5 Software Release Bulletin 47 A2 09UG
GCOS 7 - V6 Evolution Guide 47 A2 16UG
or
GCOS 7 - V5 Evolution Guide 47 A2 12UG
System Administrator's Manual 47 A2 10US
System Overview... 47 A2 04UG
System Operator's Guide 47 A2 60UU
Console Messages Directory..................................... 47 A2 61UU
DPS 7000/An Operator's Guide 47 A1 71UU

Preface

Other manuals you may need

CONVENTIONS USED IN THESE MANUALS

Text Notation

Bold	is used for indicating numbered steps you carry out in procedures. It also indicates important words used for the first time. These words are usually referenced in the index.
<CAPITAL>	capital letters in angle brackets are used to indicate keys.
Text in a box	indicates a step in a procedure. The step is described in the pages that follow.
Bold text in a double box	indicates a step in a procedure. The step is described on the page opposite.

Syntax Notation

The following syntax conventions are observed in this manual:

Syntax	appears in the following font for commands: **DISPLAY_JOB;**
System Messages	appear in the following font: GCOS: End of Session
UPPER CASE	indicates a word or phrase to be entered exactly as shown. Commands, reserved words and keywords, and example filenames are in upper case.
Lower case	indicates a variable where you should substitute an appropriate value, such as the name of a file.
Square brackets [ ]	indicate optional parameters of keywords.
Curly braces { }	indicate a sequence of parameters or keywords, of which one **must** be chosen.
Vertical lines |	separate optional or mandatory choices.

Ellipses ...	indicate that the preceding item can be repeated. You may enter an arbitrary number of similar items.
<u>Default</u>	values are underlined.
<variable>	variables are sometimes enclosed by angle brackets.
<decx>	indicates the maximum number of digits that the value concerned can have. For example, <dec5> means that you can have values up to 99999.
Other punctuation	such as commas and parentheses should be entered as shown.

Table of Contents

Table of Contents

Appendix

Illustrations

Figures

Tables

Operating Your System

1. Starting and Terminating

This section describes the procedures and recommendations to follow for powering on and powering off the DPS 7000 and reloading GCOS 7.

You are shown how to:

- power on and log on;

- modify the seasonal time variance;

- reinitialize and reload GCOS 7;

- terminate the system and power off.

Commands and initiatives you will use:

DISPLAY/MODIFY_RESTART_OPTIONS

DISPLAY_JOB

RELEASE_JOB

DISPLAY/START/TERMINATE_TELECOM_SERVER

RELOAD

MODIFY_TIME

TERMINATE_SYSTEM

SEND

POWER ON

POWER OFF

Operating Your System

Most of the procedures that you will be using on a daily basis, such as signing on and off, are dealt with here. Many of the procedures in the 'hands on' sections in Book 1 reappear here, but they are treated more rapidly, and it is assumed that you have read the sections in Book 1.. If you require more information about some steps, refer to Book 1, Sections 2 and 3.

The procedures in this section do not flow sequentially from one to the other; each one demonstrates different parameter settings. Each procedure is complete in itself and explains how to carry out one series of operations.

1.1 STARTING UP AND LOGGING ON

Execute the POWER ON initiative

Log on with your name and password

Start the telecom server

Procedure 1-1. Everyday Start-up

Execute the POWER ON Initiative

Select the POWER ON initiative.

GCOS 7 should now reinitialize. This will take 15-20 minutes. If this does not happen, it is because the AUTO INIT parameter is set to 0. You can verify this with the DISPLAY INIT PARAMETERS initiative and alter it with the MODIFY INIT PARAMETERS initiative (see Book 1).

Log on with your name and password

1. Ensure that your console is in Caps Lock mode.

2. Enter your user name.

3. Enter your password.

When GCOS 7 is loaded, you will receive the GCOS READY message, which will be similar to:

```
GCOS 7 RELEASE V600 5002 READY OCT 11,1992
*****************************************
```

If your system start-up requests it, you may also receive further lines of text.

You are then prompted to enter your user name and password. Before doing this you should ensure that the light on the <CAPSLOCK> key is lit - indicating that you are in locked upper case mode. A successful sign-on is followed by the IOF banner:

```
>>> 10:13 IOF -
```

and any messages in your mailbox.

Execute the POWER ON initiative

Log on with your name and password

Start the telecom server

Procedure 1-1. Everyday Start-up

Start the telecom server

Type: STSVR TYPE=TNS;

Press <INS> to get the system prompt (S:) and then start the CNP7 telecom server with the START_TELECOM_SERVER (STSVR) command. If this command is not part of your system startup (see Book 1), then the user acting as main operator should execute this command in everyday startup.

1.2   MODIFYING THE RESTART

1.2.1   RESTART Options

The RESTART options determine the way in which the system is loaded during a power on, reinitialization, or reload.

Reinitialization means initializing the system again with a complete testing of the hardware.

Reload means to load the system again without running hardware tests. It is possible to do reloads only after the system has been initialized.

There are three kinds of system restart:

WARM   Brings back the system as near as possible to the state in which it was at the end of the previous session. This is the default.

COLD   Erases all information from the previous session, but preserves the initialization settings.

CLEAN   Reinitializes the entire system. Erases all information from the previous session and resets all system parameters to their default values. CLEAN is usually reserved for updates to GCOS 7 and for the initial load.

The restart always reverts to WARM. If you want to use COLD or CLEAN, you must specify it in the syntax of the command. Usually, COLD is used only if WARM fails. For detailed information on using COLD restart, refer to the *System Operator's Guide*.

There are other options that affect the way in which the system will perform a reload or reinitialization. You can modify some of these options either permanently or for the next RESTART only.

1.2.2   Modifying RESTART Options for the Next Reload

You modify the IL01 options REPEAT and AUTO for the next reload with the MODIFY_RESTART_OPTIONS (MDRSO) command. This command primarily relates to automatic versus manual loading. For example, at the end of the working day, you may wish to set the RESTART options to manual mode to prepare for modifications to GCOS 7 the next morning after power on.

The syntax of the command is:

```
MDRSO REPEAT={0|1} AUTO={0|1} SILENT={0|1}
```

REPEAT=1	means that after the next termination or crash, the system begins a reload automatically.
REPEAT=0	stops any automatic reload. You must type in a command to reload GCOS 7.
AUTO=1	means that the reload uses the current IL01 options and does not display the IL01 Options prompt.
AUTO=0	means that the reload is manual. The system displays the IL01 Options prompt and asks for replies from the operator.
SILENT=0	asks the system not to show the reload messages.

Use the DISPLAY_RESTART_OPTIONS (DRSO) command at the system prompt to display the current setting before using MDRSO:

```
DRSO;
```

The first line of the display will indicate the current settings, for this session and for the following session:

```
AUTO=1/0   SILENT=0/0   REPEAT=0
```

Examples:

```
MDRSO REPEAT=0 AUTO=1
```

The next system restart requires an operator to begin it, but the reload itself will proceed automatically up to GCOS READY, without displaying the IL01 options prompt.

```
MDRSO REPEAT=1 AUTO=1;
```

The next system reload is fully automatic.

```
MDRSO REPEAT=1 AUTO=0;
```

The next system reload is automatic, but it allows you to modify the IL01 OPTIONS.

```
MDRSO REPEAT=0 AUTO=0;
```

The next system reload is completely manual. This allows you to choose when to restart the system and what type of restart to use. You can also modify IL01 OPTIONS. This is the recommended setting for modifying RESTART options permanently (see below).

1.2.3 Modifying RESTART Options Permanently

Set manual mode

Terminate the system

Reload the system

Modify the IL01 Options

Procedure 1-2. Modifying IL01 Options

Set manual mode

1.   Press <INS> to get the S: prompt.

2.   Type: MDRSO REPEAT=0 AUTO=0;

You must set the system to manual mode in order to get the IL01 Options prompt.

Terminate the system

1.   Press <INS>.

2.   Type: TSYS GCOS;

3.   Wait until termination is completed.

Refer to the Section earlier in this book for details on the procedure for terminating GCOS 7.

Reload the system

1.   Select the SWITCH TO SERVICE initiative.

2.   Press the <INS> key.

3.   Type: MDOM MANU <ENTER>

4.   Press the <INS> key.

5.   Type: RL <ENTER>

The SWITCH TO SERVICE initiative moves you to the Service Window where you receive the SP: prompt instead of the S: prompt.   While at this prompt, you must not terminate commands with a semicolon. The MDOM command puts the service console into manual mode and the RL command starts the system reload.

Set manual mode

Terminate the system

Reload the system

Modify the IL01 Options

Procedure 1-2. Modifying IL01 Options

Modify the IL01 Options

Type: `INIT(AUTO NSILENT)`
            `REPEAT(30);<ENTER>`

Executing RL brings you to the System Window and the IL01 Options prompt:

```
IL01 OPTIONS?
I:
```

To select all the defaults for the IL01 options you should just press the <ENTER> key at the I: prompt.   For everyday operations, the important options are INIT and REPEAT. For a complete description of all the IL01 Options, refer to the *System Operator's Guide*.

INIT AUTO is the permanent version of MDRSO AUTO=1.   It specifies automatic reloading of GCOS 7.

INIT NSILENT displays all GCOS 7 messages on the terminal during initialization.   This is the recommended setting.   INIT SILENT gives a blank screen during the entire initialization process.

To set REPEAT, use the syntax:

```
REPEAT[(OFF|mm)]
```

Unlike REPEAT=(0|1) in the MDRSO command, the IL01 option REPEAT regulates how often to allow an automatic reload.

REPEAT(OFF) is the default. The system stops after a TSYS command or system crash. The operator must reload or reinitialize manually.

For REPEAT(mm), the parameter **mm** refers to the minimum number of minutes allowed between two system interruptions. The recommended value is between 30 and 60 minutes in order to allow time for a system dump when there is a crash.

For example, if you set:

```
IL01 OPTIONS?
I: REPEAT(30);
```

then there must be a lapse of at least 30 minutes between two consecutive system crashes or system terminations. If two consecutive terminations occur in under 30 minutes, then INIT AUTO is switched off. After the loading of GPOS and SPOS, the system displays the IL01 Options prompt again.

After you finish modifying the IL01 Options, press ENTER to load GCOS 7 and to display the GCOS READY message.

1.3 MODIFYING THE TIME VARIANCE

You may need to reset the seasonal time variance for the GCOS 7 clock twice a year to account for the changes in summer and winter time. You make this modification with the MODIFY_TIME (MDTIME) command. The syntax at the S: prompt is:

```
MDTIME +|-HH[mm]
```

You modify the time variance against Greenwich Mean Time (GMT). GMT is stored in the hardware clock.

+HH or -HH indicates the number of hours difference between GMT and your local time.

To set the time to Greenwich Mean Time (GMT) you type:

```
MDTIME 0;
```

mm is an optional parameter that pertains to time settings in certain parts of the world. In some localities the variance from GMT is in hours and minutes.

```
                    ┌─────────────────────────────────────────┐
                    │                                         │
                    │            ATTENTION                    │
                    │                                         │
                    │  Your system may include applications   │
                    │  that are set to run at certain times   │
                    │  of day. Be sure to take this into      │
                    │  account when you reset the seasonal    │
                    │  time variance.                         │
                    │                                         │
                    └─────────────────────────────────────────┘
```

1.4   REINITIALIZING THE SYSTEM

The procedure below describes a complete termination and reinitialization of GCOS 7. You may need to perform only some of these steps, depending upon your system status at the time.

Set INIT parameter and RESTART options

Terminate the system

Reinitialize the system

Procedure 1-3. Reinitializing the System

Set INIT parameter and RESTART options

1.   Verify that INIT parameter AUTO = 1.

2.   Type: MDRSO REPEAT=0;

From the Control screen, use the DISPLAY INIT PARAMETERS initiative to display the setting for AUTO. If necessary, use the MODIFY INIT PARAMETERS initiative to set AUTO=1 (see Book 1).

Then set REPEAT=0, otherwise the system will reload immediately when you close it down.

Terminate the system

Type: TSYS GCOS;

Refer to *Procedure 1-5* for a complete explanation of how to terminate GCOS 7.

Set INIT parameter and RESTART options

Terminate the system

Reinitialize the system

Procedure 1-3. Reinitializing the System

Reinitialize the system

1.   Select the SWITCH TO SERVICE initiative.

2.   Type: TS S0

3.   Type: IN

The SWITCH TO SERVICE initiative will move you into the Service Window.

Because REPEAT=0, system termination is followed by a reloading of GPOS and SPOS only. You can now perform reinitialization or reloading of GCOS 7.

The system performs a check of all the hardware and then loads GCOS 7.

If you replace the IN command by the INR command, GCOS 7 will reload much more quickly, as only a reduced set of hardware initialization checks are carried out.

1.5   RELOADING THE SYSTEM

The procedure below describes a complete termination and reload of GCOS 7. You may need to perform only some of these steps, depending upon your system status at the time.

Set INIT parameter and RESTART options

Terminate the system

Reload the system

Procedure 1-4. Reloading the System

Set INIT parameter and RESTART options

1.   Verify that INIT parameter AUTO = 1.

2.   Type: MDRSO REPEAT=0;

Use the DISPLAY INIT PARAMETERS initiative to display the setting for AUTO.   If necessary, use the MODIFY INIT PARAMETERS initiative to set AUTO=1.

Then set REPEAT=0, otherwise the system reloads immediately when you close it down.

Terminate the system

Type: TSYS GCOS;

Refer to *Procedure 1-5* for a complete explanation of how to terminate GCOS 7.

Set INIT parameter and RESTART options

Terminate the system

Reload the system

Procedure 1-4. Reloading the System

Reload the system

1.   **Select the SWITCH TO SERVICE initiative.**

2.   **Type: RL <Enter>**

Use the SWITCH TO SERVICE initiative to change to the Service Window and the SP: prompt.

The RL command reloads the system directly, without performing hardware checks.

1.6 TERMINATING AND SWITCHING OFF THE SYSTEM

Display jobs in execution

Send a message to users to log off

Terminate the telecom server

Check that REPEAT=0

Terminate the system and power off

Procedure 1-5. Terminating the System

Display jobs in execution

Type: DJ EX *;

You can use the DISPLAY_JOB (DJ) command to display all jobs that are currently in execution.

The following job types run constantly under GCOS 7. Except for TNS, the TSYS command terminates these job types:

- OMH
- READER
- JTRA
- WRITER (after current output, if any, is printed)
- IOF
- TNS

The TSYS command cannot terminate the **TNS** job. This job is the telecom server for the CNP 7. You must first terminate TNS with the TTSVR command (see below).

If there are any user jobs still running, you can interrupt them with the CJ (CANCEL_JOB) command after checking with the user.

Send a message to users to log off

Type: SEND 'TIME TO LOG OFF PLEASE' ROBERT;

The TSYS command does not abruptly terminate all jobs. However, it is good practice to remind all IOF users that system termination is about to take place. At the S: prompt, you send the message to each user with the SEND command in the form:

```
SEND '<your-message>' <user-name>;
```

Display jobs in execution

Send a message to users to log off

Terminate the telecom server

Check that REPEAT=0

Terminate the system and power off

Procedure 1-5. Terminating the System

Terminate the telecom server

1.   Type: TTSVR TNS;

2.   Type: DTSVR;

The TTSVR (TERMINATE_TELECOM_SERVER) command terminates the server after all users have logged off. If for some reason you need to terminate the server immediately with users still logged on, use the syntax:

```
TTSVR TNS OPTION=STRONG;
```

You use DTSVR (DISPLAY_TELECOM_SERVER) to check that the server has terminated.

If the server is still active, you receive the message:

```
TNS STATE:USED TYPE:TNS
```

When the server has terminated, the message reads:

```
TNS STATE:ENBL TYPE:TNS
```

Check that REPEAT=0

Type: DRSO;

You must have the setting REPEAT=0 for system termination, or GCOS 7 will reload.

Use DRSO to verify that REPEAT=0. If it is not, use MDRSO to change the setting.

Display jobs in execution

Send a message to users to log off

Terminate the telecom server

Check that REPEAT=0

Terminate the system and power off

Procedure 1-5. Terminating the System

Terminate the system and power off

Type: TSYS GCOS PWROFF;

This command executes the system request to shut down. No further jobs can start. All jobs that are currently running will continue until they reach either the end of their current step or a checkpoint. When the termination process finishes, the system is automatically powered off.

Note that the TSYS command is denied if:

- the TNS server is active (see the previous step);

- one or more jobs are suspended.

If one or more jobs are suspended when you execute TSYS, you receive the message:

```
JOBS SUSPENDED, END NOT ALLOWED
```

If this happens, reactivate the job(s) with the RELEASE_JOB command. Type:

```
S: RELEASE_JOB STRONG=1;
```

Then re-enter the TSYS command.

When termination is complete, the DPS 7000 powers off automatically. (The console, on the other hand, remains on, as its power supply is separate).

```
AV02 GCOS 7: END OF SESSION
```

Note that if a WRITER job is printing output when the TSYS command is executed, the system will wait for the job to finish printing before executing TSYS. During this delay the system does not display any message.

Recommendations for Power Off

At the end of your working day, use the following procedure to terminate GCOS 7. After successfully completing this procedure, you can, if necessary, switch off the DPS 7000. Here are a few recommendations regarding the powering off of the DPS 7000:

- It is preferable to leave the A/C power supply ON 24 hours a day, unless you are planning to shut down the machine and leave it unattended for a long period (one week or more).

- *Never cut off power to the DPS 7000 from a switch other than the mains switch of the machine itself.* If you do, the DPS 7000 thinks that a power failure has occurred, and it activates the back-up power supply. This drains the battery (if present), and the system continues operating.

- Never switch the mains switch OFF without carrying out all the steps in the termination process. If for example you cut off the power before all jobs have terminated, much work could be irretrievably lost.

2. Using the Interactive Operation Facility (IOF)

This section introduces the GCOS 7 Interactive Operation Facility (IOF). It is designed to familiarize you with important aspects of using IOF.

This section tells you how to:

- use Menu mode and Line mode;

- use Help texts;

- read and send messages;

- use IOF command syntax;

- manipulate Source Language libraries;

- use the Text Editor;

- handle outputs.

Commands you will use:

MODIFY_PROFILE   DPRINT
MAIL   SEND
EDIT   SCANNER
MAINTAIN_LIBRARY commands

NOTE: To use Menu mode, you must have a screen-mode terminal. The System Console of the DPS 7000 does not provide screen-mode capabilities.

2.1   USING COMMANDS

2.1.1   Menu Mode and Line Mode

Menu mode, or Screen mode, is the menu-driven interface for IOF commands.   Each level in IOF presents the available commands on a series of **menu screens**.

The commands on each page are numbered from 1 upwards. You activate a command by choosing its number from the list. This displays one or more prompt screens for entering the necessary parameter values for that command.

Line mode is the standard interface for GCOS 7. In Line mode, you enter commands at the prompt according to the syntax required by the command. The different GCOS 7 prompts (S:, C:, R: and so on) indicate that you are in a different processor. The examples in these compact guides for using IOF commands are given in Line mode.

All commands operate the same way in Menu mode as in Line mode.

2.1.2   Mixed Mode and Prompt Mode

Mixed mode combines the two ways of working under IOF. When you are using Mixed mode, GCOS 7 presents the standard Line mode interface, but it enables you to call up menu screens and command prompt screens on request by typing the question mark character (?).

To call for menu screens at the current level, simply type **?** at the prompt.

For example:

```
S: ?
```

displays a menu screen for system level commands.

To call for a command prompt screen, type in the name of the command, or its abbreviation, followed by the question mark. ***For example:***

```
DPRINT?
```

displays the prompt screen for the command DPRINT.   More details about using DPRINT are given later in this Section.

Prompt mode is a way of describing the use of the **?** character in Line mode.

When you type in **?** at the current prompt in Line mode, the system generates a list of the available commands at that level, but it does not display a menu screen for making a numbered selection. This is still a helpful way of reminding yourself of the commands you can use at any given level.

When you type in a command name followed by **?**, the system prompts you one by one for values for each parameter of the command. As you type in each value and press ENTER, the next value is displayed. For example:

```
DPRINT?
SUBFILES: SAMPLE
LIB: MYPROJ.REPORTS
CLASS:
```

If a parameter is **mandatory**, that is, you must supply a value for the parameter in order to execute the command, then the parameter is preceded by a (**+**).

If a parameter contains a default value or a previously assigned value, the value is included in parentheses:

```
SUBFILES (SAMPLE) :
```

To retain a previous value, simply press ENTER without typing in anything.

2.1.3   Setting the Mode

You set your working mode with the parameters NOVICE (0|1) and MENU (0|1) of the user profile. To display the values for all parameters of the user profile, use the DISPLAY_PROFILE (DP) command:

```
DP;
```

The working modes and their corresponding values for NOVICE and MENU are:

Menu mode: NOVICE=1; MENU=0 or 1

Mixed mode: NOVICE=0; MENU=1

Line mode: NOVICE=0; MENU=0

To change either of these values, use the MODIFY_PROFILE (MDP) command. For example, to change the working mode from Line to Menu, type after the S: prompt:

```
MDP NOVICE=1;
```

Note that when NOVICE=1, the value of MENU is not important and can be either 0 or 1.

2.1.4   **Using a Menu Screen**

Figure 2-1 provides an example of an IOF Menu screen. The top center of the screen displays the title. The top left corner displays the page number of the screen in the form **n/p**, where **n** is the current page number and **p** is the total number of pages available. In the example, this is page 9 of 12 pages.

You move around the pages of menu screens with the help of **move requests**. You type in the desired request in the **action field** and press ENTER. The move requests are:

> or Blank	Moves forward one screen.
>N	Moves forward **N** screens.
<	Moves back one screen.
<N	Moves back **N** screens.
=N	Moves to the **N**th screen
/	Exits Menu mode and returns to Line mode.

IOF treats any character string other than a move request, a Help request (?), or a menu selection as a request for the next screen.

The main body of the screen contains four columns. From left to right these are:

• the number of the command;

• the command name;

• the command abbreviation;

• a brief description of the command function.

To execute a command, type its number in the action field and press ENTER.

In the example, you choose the DPRINT command by typing the number 6 in the action field.

Prompt lines

The prompt lines at the bottom of the screen allow you to type in commands using Line mode syntax. This enables you to enter commands for which you already know the full syntax. In addition, you can type in any command available at the current level. The use of prompt lines does not restrict you to the commands displayed on the current screen page.
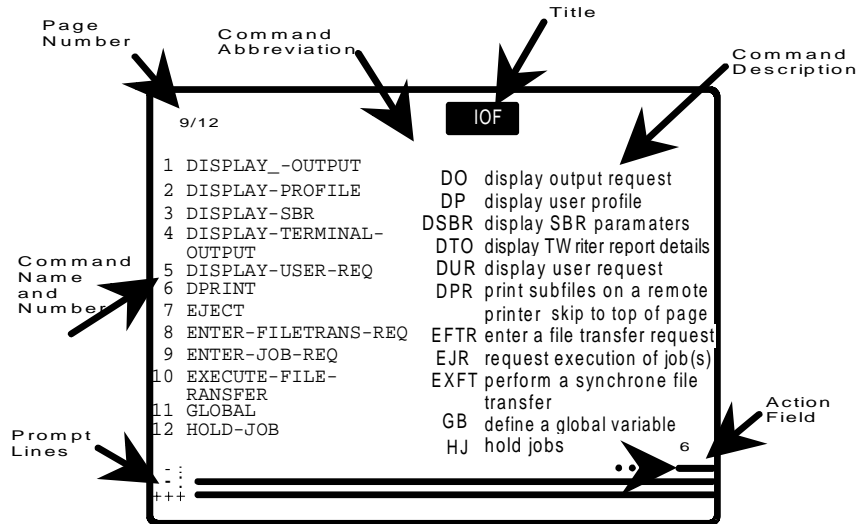

Figure 2-1. The IOF Menu Screen

2.1.5   Using a Prompt Screen

Figure 2-2 below provides an example of the prompt screen for the DPRINT command. For most commands, the action field on a prompt screen is for entering one of two requests:

- Help request (**?**) to call for the command's Help text.

- Exit request (**/**) to return to the menu screen.

In the few cases where a command consists of more than one page of prompt screens, you can also enter any of the available move requests.

The layout of all prompt screens is the same. The command title and command description are at the top center, the action field top left, and the parameters, parameter descriptions, and value fields in the main body of the screen.

The length of each field is equal to the maximum length allowable for a parameter value.

If a parameter contains a default value, this value is displayed on the screen.

Figure 2-2. Prompt Screen for the DPRINT Command

Executing a command

You move the cursor down the screen from field to field with the <TAB> key. You move back up the screen with <SHIFT> <TAB>.

To type in values for parameters, move the cursor to the desired field, type in the value(s), then move to the next field. It is not necessary to fill in fields in any given order.

After you have typed in all necessary parameter values, press <ENTER> to execute the command.

After executing a command, the system re-displays the current menu screen. You are then free to choose another command.

2.1.6 Help Texts

There are Help texts for every command and every command parameter under IOF. Each text contains one or more pages of information on how to use the command, together with appropriate Line mode examples.

Help texts give you quick access to information. You can read the Help text that relates to the particular command or parameter you are working with and then return immediately to your current screen, just where you left off. The use of Help texts never interferes with your work.

You call for a Help text with the **?** request.

In Menu mode, you type ? in the command's action field or parameter field and press <ENTER>.

In Line mode, you type in the command preceded by the ?:

```
?DPRINT
```

Or in prompt mode, you type ? in the parameter field:

```
DPRINT?
SUBFILES: ?
```

Using the Help

After you type in the request and press <ENTER>, your active screen disappears and the first page of the Help text appears. Figure 2-3 below illustrates the first page of the Help text for the DPRINT command.

The top left corner displays the page number and the number of pages of text. The top right corner displays the name of the command or parameter that the text refers to.

You can move around the pages of Help text with the same move requests that you use for menu screens.

The standard arrangement of the content of a Help text is:

- command or parameter definition

- parameter description (for commands)

- special recommendations or cautionary remarks

- line mode examples

When you reach the last screen and press <ENTER>, you return to your current screen. Anything you had typed on the screen prior to calling the Help text remains intact. You can also return to the active screen at any time with the Exit request (/).
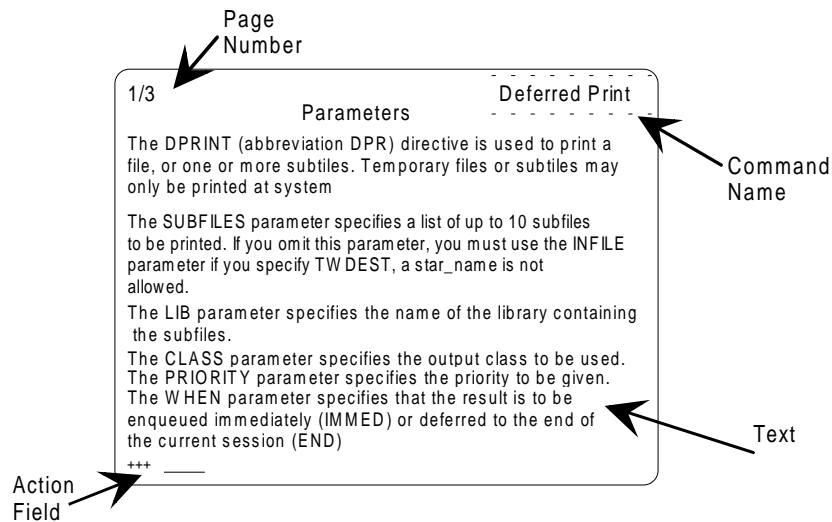
Figure 2-3. HELP Text for the DPRINT Command

2.2 MESSAGES

2.2.1 Using the Mailbox

Every user of IOF is provided with a **mailbox**. A mailbox is a container that holds your messages until you read them. Your mailbox can receive messages even when you are not logged on to IOF. When you log on, the system usually displays the contents of your mailbox, that is, all messages that have been sent to you since you last logged off.

You regulate the saving and displaying of the messages you receive with the MAIL command. You send messages to other users with the SEND command.

If for some reason you do not wish to be disturbed by incoming mail during your IOF session, type:

```
MAIL OFF;
```

This stores mail in your mailbox, instead of displaying it on the terminal when it arrives. To switch back to normal operation, type in:

```
MAIL ON;
```

To print the accumulated contents of the mailbox at once, type:

```
MAIL;
```

To send a message to another user, use the SEND command in the form:

```
SEND '<text of message>' <user-name>;
```

If you specify MAIN, the message is sent to the main operator only.

Examples:

```
SEND 'Ready to start YOURPROG' SMITH;

SEND 'Mount special paper and reply READY' MAIN;
```

If you are working at R: level (for example in Text Editor) or at any level where SEND is not a standard command, you must prefix SEND with the characters **$$**, or whichever characters are used as the **directive identifier (DI)** in the user profile. For example, at the R: prompt:

```
$$SEND 'Group meeting today at 10:00' JONES;
$$SEND 'Group meeting today at 10:00' SMITH;
$$SEND 'Group meeting today at 10:00' BROWN;
```

2.2.2   Console Messages

In general, console messages are divided into two parts: a message label and the text of the message itself.

The message label is either a **system identifier** or a **time stamp**.

A system identifier is a four-digit string composed of two alphabetic characters and two numbers. For example:

```
LD10   exception message IN bhf1-1.bhf1-2 AT ADDRESS
addr
```

```
DF01   ron FILE efn volnm USED BY: ronlist
```

LD10 is a system identifier issued by Load Module Management. **DF01** is a system identifier issued by Data File.

The time stamp takes the form **hh.mm**, where hh is the hour and mm the minute:

```
12.30 cssnm OUT (BY SYSTEM)

09.45 dvnm ALLOCATED TO ronlist

14.15 fon TRANSFER STARTED
```

If a message requires immediate operator action, the system displays an asterisk (*) directly before the message. The action required is usually either a console response or an action at the installation, such as the mounting of a device.

When a console response is required, the system displays an I: prompt below the message for you to type in your reply:

```
*IL44 DAY/TIME D.M.Y/H.M.<.S>?
I:
```

When an action is requested that does not involve a console response, the prompt is not displayed:

```
*hh.mm dvnm I/O ERROR ON volnm FOR ron

*AV53 dvnm NOT OPERATIONAL
```

You will find a complete list of system identifiers and messages in the *Console Messages Directory*.

2.3   IOF RULES AND CONVENTIONS

2.3.1   Naming Conventions

The maximum length of the name of an object under IOF is 44 characters. However many object names have a shorter maximum length. Some general categories of object type, with their maximum name length, are:

- names of catalogs, directories, master directories, and files - 44 characters;

- program names, command names, library member names, and global variables - 31 characters;

- output names - 14 characters;

- job names - 8 characters.

Names can contain only the following characters:

- uppercase letters A through Z;

- lowercase letters a through z (which are then converted to upper case by the system);

- digits 0 through 9;

- the underscore (_);

- the minus sign (-).

Names must begin with a letter or a digit. Names of certain objects, however, such as commands and keywords, must begin with a letter.

Program names can begin with a digit. All of the following are valid names:

MYPROGRAM

PG-1234

ORDER-6

PAY_ORDER

0126 and 02PG are not valid as command or keyword names, but are valid as program names.

NOTE: The library names used in this section are all example names.

2.3.2   Star-names

You can use star-names with certain parameters to denote a group of names that meet certain criteria. A **star-name** acts as a pattern where the asterisk (*) character in the name is used to match any occurrence, including none, of characters within a name. The following examples illustrate how star-names work.

ABC*	matches all names that begin with ABC.
*XYZ	matches all names that end with XYZ.
A*Z	matches all names that begin with A and end with Z.
*	matches all names.
A*I*Z	matches all names that begin with A, end with Z, and contain an I somewhere in between.
ABC	matches only name ABC.

You can prefix a star-name with a not sign (^) to mean "all names **except**...".

Examples:

^*S	match all names except those that end with S.
^X	match all names except X.
^XY*	match all names except those that begin with XY.

A star-name, including qualifiers and the not (^) prefix, cannot exceed 88 characters in length.

2.3.3   Conventions for Using Commands

GCL Commands

There are GCL (GCOS Command Language) commands at system level (S:) and at every processor level. Examples of system-level GCL commands are EDIT (ED), BUILD_FILE (BF), LIST_FILE_SPACE (LSFSP), and CREATE_CATALOG (CRCAT). A **processor** is a group of commands that perform a specific set of functions, such as the Full Screen Editor (FSE) and MAINTAIN_CATALOG. While inside a processor, you can use only those commands which belong to it.

Directives

Directives are special GCL commands that are available anywhere in the system. They are not related to any specific processor. Examples of directives are DISPLAY_JOB (DJ), MODIFY_PROFILE (MDP), and SEND.

When you use a directive within a processor that does not contain directives by default, then the syntax of the command must be preceded by the **directive identifier** (**DI**).

The default directive identifier is **$$**. For example, to display the user profile while inside the Text Editor, at the R: prompt type:

```
$$DP;
```

To change the directive identifier, use MODIFY_PROFILE (MDP) at the S: prompt. The DI can be up to 3 characters long:

```
MDP DI=*$*;
```

Syntax of Directives and System-level GCL Commands

The syntax of directives and system-level commands is the same. A GCL command consists of a command name (or its abbreviation) followed by one or more parameters. Some commands have no parameters, but this is the exception rather than the rule. A parameter is either mandatory (you must specify a value for it) or optional.

Most commands have an abbreviated form. For example, the short form of the LIST_FILE_SPACE command is LSFSP. It is good practice to become acquainted with and to use abbreviated forms.

The conventions used to indicate command syntax in documentation are illustrated here using the LIST_FILE_SPACE command as an example:

```
LSFSP FILE=file-desc [SILENT=0|1]
      [PRTFILE=file-desc];
```

This command contains one mandatory parameter named FILE, and two optional parameters named SILENT and PRTFILE. The bar, |, between 0 and 1 in SILENT indicates that you can specify 0 or 1. An underlined value indicates a default value. The square brackets, [ ], indicate that the parameter is optional.

Command Execution

To execute a command, you enter its name (or its abbreviation) followed by its parameter values (if any). You can specify the parameter values in keyword form, positional form, or mixed form.

Keyword form - In keyword form, each parameter value is preceded by <keyword>=, where keyword is the name of the parameter. The parameter values can be specified in any order. For example:

```
LSFSP FILE=P1.F1 SILENT=0 PRTFILE=P1.F2;
```

Positional form - In positional form, the keywords are omitted and the values are given in the order required by the command syntax. For example:

```
LSFSP P1.F1 0 P1.F2;
```

Mixed form - You can mix positional and keyword forms in the same command. In this case, the positional parameters are positioned by the last keyword parameter that precedes them. For example:

```
LSFSP SILENT=0 P1.F2 FILE=P1.F1;
```

P1.F2 refers to PRTFILE, as this is the first parameter after SILENT in the command syntax.

Although all three forms are valid, for the sake of clarity the keyword form is recommended. All examples of command usage in these compact guides are given in keyword form.

Separators

You can use commas instead of (and as well as) spaces as separators between parameters. Two consecutive commas indicate that a value is omitted for that parameter. The parameter concerned takes its default value, if any. For example:

```
LSFSP FILE=P1.F1,,PRTFILE=P1.F2;
```

Here the consecutive commas indicate an omitted positional parameter value, SILENT, which takes the default 0.

Boolean Parameters

SILENT is an example of a **boolean parameter**. A boolean parameter takes only the value 0 or 1. 0 means NO (do not take this option) and 1 means YES (take this option).

Self Identifying Value (SIV) - You can specify a Boolean parameter in keyword or positional form, or as a Self Identifying Value (SIV). SIV means that the boolean parameter name itself represents the value 1. If the parameter name is preceded by **N**, this represents the value 0.

For example, all of the following positive responses are synonymous and valid for the boolean parameter SILENT:

SILENT=1   (keyword value)
1   (positional value)
SILENT   (SIV)

The corresponding negative responses are:

SILENT=0   (keyword value)
0   (positional value)
NSILENT   (SIV)

Line format and free format

You can enter commands in either line format or free format.

In **line format**, the end of the command line acts as terminator. You are in line format if GCLFORM=LINE is in your user profile.

In **free format**, the end of a line does not act as a terminator for a command. You must explicitly terminate a command with a semi-colon (**;**). You are in free format if GCLFORM=FREE is in your user profile. All examples in these compact manuals use free format.

Continuation prompt

Sometimes the length of your command syntax exceeds the width of the terminal screen. GCOS 7 provides a **continuation prompt** (**-:**) to allow you to continue typing in a single command over more than one screen width.

To call a continuation prompt, insert a space and a hyphen (-) at the end of the first line of text and press <ENTER>. The cursor reappears on the second line after the continuation prompt. You can now continue typing in the rest of the syntax for the command. You can call for another prompt if necessary.

For example, to type in the following at the S: prompt:

```
EJ PROC=CBL VALUES=(MYPG INLIB=.MINE.SL
CULIB=.MINE.CU MAP PRTLIB=.MINE.LIS DEBUG
XREF);
```

use two continuation prompts:

```
EJ PROC=CBL -
-: VALUES=(MYPG INLIB=.MINE.SL CULIB=.MINE.CU -
-: MAP PRTLIB=.MINE.LIS DEBUG XREF);
```

2.4   SOURCE LANGUAGE LIBRARIES

2.4.1   Commands for Managing SL Libraries

The Library Maintenance utility is a processor that provides a number of commands for the efficient handling of source language (SL) libraries and their members. It is important to note that the commands work with already-existing libraries; they do not create them. To create a library, use the GCL command BUILD_LIBRARY (BLIB). The library-maintenance commands cover:

- assigning/deassigning libraries or files;

- listing library contents;

- displaying library directory information;

- creating, renaming, modifying, deleting members;

- printing members;

- moving members from one library to another.

To access the Library Maintenance utility, use the command MAINTAIN_LIBRARY (MNLIB) at the S: prompt:

```
MNLIB;           Library Maintenance displays
its prompt:
C:
```

The next few pages describe the most important commands of the Library Maintenance utility.

Assigning Libraries

To assign output and input libraries and files, use the LIB, INLIB, INFILE and OUTFILE commands, at the C: prompt:

```
LIB MYPROJ.HELPC;

INLIB MYPROJ.SL-LL;

INFILE MYPROJ.SEQ1;
```

The output and input libraries must be created with the BLIB command before you assign them.

Deassigning Libraries and Files

To deassign a library or file, simply type in the appropriate command: LIB, INLIBi, INFILE, or OUTFILE without any file or library name:

```
LIB;
INLIB2;
INFILE;
OUTFILE;
```


Displaying Libraries and Files

To display the name and location of your libraries and files, use the D LIBS and D FILES commands at the C: prompt:

```
D LIBS;
D FILES;
```

Listing Library and File Information

To display information about your libraries and files, use the LIST command. This command offers several options for displaying information. The parameters are:

NAMES   member names only (when used alone)

CREATE   creation date and time

MODIF   last modification date and time

LINES   number of lines in each member

LANG   language type

DIRINF   file, media, and library occupancy information

SIZE   space allocated (in blocks) to each member (ignored with INFILE)

If you use the command without specifying any of the above parameters, all of the above information is listed, except library occupancy information.   You must specify the DIRINF parameter to display library occupancy information.

To find out how full a library is, use either the DIRINF parameter or the following syntax to omit file and media information:

```
C: LIST ^* DIRINF;
```

Use the NAMES parameter to verify whether or not a certain member is in your library or to remind yourself of the name of a member. In the latter case, you can use a star-name to limit the display to a range of possible names. For example:

```
C: LIST LL_SIM* NAMES;
```

Printing a Member On Your Terminal

To print a member on the terminal, use the PRINT command:

```
C: PRINT LL_LISTALL;
```

Using Dprint to Print a Member On the System Printer

Whenever you want to print members on your system printer, you will find it convenient to use the DPRINT directive at either Library Maintenance (C:) or system (S:) level. In most instances you will only need to use the parameters SUBFILES, LIB, COPIES, and possibly BANINF (banner information). The banner information appears on the first and last pages of your listing and is used to identify the owner of a listing.

DPRINT is discussed in more detail below.

Moving Members from One Library to Another

To move members from one library to another, use the MOVE command. Assign your input and output libraries first with the LIB and INLIB commands. You move members **from** an input library **to** an output library. If you want to move other members, you must reassign the input and output libraries.

If you are moving a new member name, at the C: prompt use the syntax:

```
MOVE INLIB:member-name;
```

If the member name already exists, add the REPLACE parameter

```
MOVE INLIB:member-name REPLACE;
```

Moving Members Between a Library and a Sequential File

Use the SAVE and RESTORE commands to move members to and from a sequential file. To save a member, you must first assign the sequential file with OUTFILE and the input library with INLIB. At the C: prompt:

```
INLIB1 SD3.SL-LL;
OUTFILE SD3.SEQ1;
SAVE INLIB1:*;
```

When members are to be restored from a sequential file to a library, assign the sequential file with INFILE and the output library with LIB:

```
INFILE SD3.SEQ1;
LIB SD3.HELPC;
RESTORE INFILE: * REPLACE;
```

Rename, Renumber, and Delete

Use the RENAME, RENUMBER, and DELETE commands to rename, renumber, and delete members. Here is a brief look at these three commands, entered at the C: prompt:

```
RENAME LL_SIMPLE LL_EASY;
```

Rename member LL_SIMPLE as LL_EASY

```
RENUMBER LL_SIMPLE;
```

Renumber LL_SIMPLE as 10, 20, 30, 40...

```
RENUMBER LL_SIMPLE NUMBER=(1,1);
```

Renumber LL_SIMPLE as 1, 2, 3, 4...

```
DELETE LL_SIMPLE;
```

Delete LL_SIMPLE.

2.4.2 Introduction to Text Editor

The Text Editor processor enables you to create and to modify members of Source Language (SL) libraries. You create and modify images of members in temporary storage areas called **workspaces**. When you are finished, you write the contents to a library member in the SL output library.

You call the Text Editor with the EDIT command. Once you are within the editor (R: level), you have many requests available for editing text. The most commonly-used requests are:

R   Read
L   List
P   Print
A   Append
I   Insert
C   Change
S   Substitute
D   Delete
J   Renumber
X   Workspace Status
T   Top of Page
W   Write
Z   Overwrite
B   Change Workspace
Y   Alter other commands
Q   Quit

You can type in requests in either uppercase (A,R,S,I...) or lowercase (a,r,s,i...).

Addressing Lines in a Workspace

The Text Editor is a line-oriented editor. Requests normally operate on one or more lines. If a request operates on a single line, the format is:

<line#> <request>   100 L

If a request operates on a range of lines, the format is:

<line#>,<line#>...<request>   100,200 L

You can also address a line by its position relative to the current, first, or last line. This is called **relative addressing**. The current, first, and last lines of a workspace are represented by the following characters:

.	(dot)	current line
^	(not)	first line
$	(dollar)	last line

If you omit these characters, the current line is assumed.

Here are some examples of relative addressing using these characters:

.-2	2 lines before the current line
.+1	I line after the current line
.3	3 lines after the current line
^+1	the second line
$	the last line
-5	5 lines before the current line
+4	4 lines after the current line

You can also address a line by searching for a string of characters on that line. Your search can include star-names.

Here are some examples of addressing by context.

/A /	addresses a line containing A, followed by a space.
/AB*C/	addresses a line containing AB, followed by any number of characters, and then C.

You can add the Global (G) request to your character string search. The Global requests include: Global List (GL), Global Delete (GD), and Global Substitution (GS).

GL/ABC/	lists all lines in the workspace that contain ABC.
GS/ABC/DEF/	replaces ABC with DEF in all lines of the workspace.
GD/AB*DEF/	deletes all lines that have AB, followed by any characters, and then DEF.

2.4.3   A Sample Text Editor Session

In this sample session, you will see the following operations: create a member, write it to an output library, append and insert lines, use the substitute request, and modify and re-save the member.

CONSOLE DISPLAY

```
S: EDIT
R: A
I: This is a sample Text Editor member.
I: This is the second line of the member.
I: This is the third line of the member.
I: This is the fourth line of the member.
I: This is the last line of the member.
I: /
R: ^,$J
R: ^,$L
10 This is a sample Text Editor member.
20 This is the second line of the member.
30 This is the third line of the member.
40 This is the fourth line of the member.
50 This is the last line of the member.
R: W SAMPLE
* NO LANGUAGE TYPE, DAT ASSUMED
R: 30A
I:This line has been inserted after line 30.
I: /
R: 20I
I: This line has been inserted before line 20.
I: /
R: X
*WORKSPACE(0) -->   7
IMPLICIT NAME : SAMPLE
```

COMMENTARY

You call the Text Editor by typing EDIT. You call for an I: prompt with the **A** (=Append) request. When you receive the I: prompt, you are in text entry mode.

After you enter a line of text, press <ENTER> to receive another I: prompt. Each line of text cannot exceed the width of the screen. You cannot use a continuation prompt for extending text entry.

You leave text entry mode by entering a slash (/) character.

To number the lines of the text, use the **J** request. By default, J numbers lines in increments of ten. **^,$** indicates that you want to renumber all lines, that is, from the first (^) to the last ($).

You list the numbered lines of text with the **L** request.

You write the current text to the member SAMPLE with the **W** request. Note that the text in the workspace also receives the name SAMPLE. This is called the **implicit name**.

You use **A** (append) to add another line *after* the specified line. You use **I** (insert) to add a line *before* the specified line.

You list information about your workspaces with the **X** request. You can open up to 10 workspaces in any one session. They are named **B0** to **B9**. These are the **explicit names** of the workspaces.

By default, the first workspace is named **B0**.

```
R: T
R: ^,$L
 10 This is a sample Text Editor member.
  0 This line has been inserted before line 20.
 20 This is the second line of the member.
 30 This is the third line of the member.
  0 This line has been appended to line 30.
 40 This is the fourth line of the member.
 50 This is the last line of the member.
R: YJ/1/1/
R: ^,$J
R: ^,$L
  1 This is a sample Text Editor member.
  2 This line has been inserted before line 20.
  3 This is the second line of the member.
  4 This is the third line of the member.
  5 This line has been appended to line 30.
  6 This is the fourth line of the member.
  7 This is the last line of the member.
R: B1
R: A
I: This is a sample member in workspace B1.
I: /
R: X
*WORKSPACE(0)          7
*WORKSPACE(1) -->      1
NO IMPLICIT NAME
R: B0
R: X
*WORKSPACE(0) -->      7
*WORKSPACE(1)          1
NO IMPLICIT NAME
```

COMMENTARY

When you execute **X**, each workspace is listed, together with the number of lines in that workspace and, if you have entered and saved text, the implicit name.

Move to the top of the screen with the **T** request and list the entire contents of the workspace.

The **Y** request alters the actions of another request. Here you use **YJ** to respecify the line numbering. The new line numbering will begin with number 1 and increase in increments of 1 (1,2,3...). This is followed by a relisting.

Now open the second workspace **B1** and add a line of text.

The **X** command now lists both workspaces. The arrow (**-->**) indicates the current workspace. When you return to the original workspace and list the workspace information, the arrow now points to **B0** as the current workspace.

The previous implicit name is removed by opening a new workspace.

```
R: 7L
7 This is the last line of the member.
R: 7S/the last/the 7th and last/L
7 This is the 7th and last line of the member.
R: 6P
The fourth line of the member.
R: 6L
6 The fourth line of the member.
R: 1,6D
R: ^,$L
7 This is the 7th and last line of the member.
R: 7C
I: The only remaining line of the member.
I: /
R: ^,$L
0 The only remaining line of the member.
R: Q
DO YOU WISH TO SAVE YOUR WORKSPACE(S) BEFORE
LEAVING ?**YES**
R: Z SAMPLE
R: /
S:
```

COMMENTARY

If you use **L** with a single line number, only this line is listed. It is good practice to display a line before modifying the text within it. Now use the **S** (Substitute) request to change text within this line.

You use the **P** (print) request to print line **L**. The **P** request differs from the **L** request in that P does not display the line number. It lists the text as it would appear for printing.

The **D** request deletes lines 1 through 6.

Listing the workspace shows that only line 7 remains. You can alter the whole line by using the **C** (change) request.

When you quit the Text Editor with the **Q** request, the editor asks if you want to save the current work. To save by overwriting a previously-created member, use the **Z** request.

2.5   OUTPUT

2.5.1   Using DPRINT

The DPRINT directive prints sequential files and library members on the system printer. All outputs are queued in the system and printed, unless there is intervention by the main operator.

The most important parameters of the DPRINT directive are:

SUBFILES INFILE	You must include either one of these parameters to indicate the objects to be printed: **SUBFILES** to specify a list of library members (maximum of 10) **INFILE** to specify the details of a sequential file.
COPIES	Specifies the number of copies to print.
CLASS	Specifies the output class to use.
LIB	Specifies the name of the library containing the library members.
NUMBER	Indicates whether or not to print line numbers on the output.

Examples:

```
DPR SF1;
```

Prints the member SF1 of the current SL library.

```
DPR (S* T*) PROJ1.LIB1 CLASS=Z COPIES=2;
```

Prints 2 copies of all members whose names begin with S or T. The output class is Z.

```
DPR INFILE=SEQFILE1;
```

Prints the sequential file SEQFILE1.

2.5.2   Using Scanner

While your output is in the queue, you can use the SCANNER utility to examine output contents on the terminal.

You can scan outputs by page, by line, or by context. SCANNER also allows you to print an output or selected pages of an output.

You can scan the following objects with SCANNER:

- job outputs;

- SL library members;

- permanent sequential files.

Note that you can only scan your own outputs and only those that are still in the queue and not already printed.

SCANNER is particularly useful during program development. It can save much unnecessary printing, since you can examine your outputs on the terminal and, if you wish, cancel what you do not need.

To call the SCANNER utility, type in SCANNER at the system prompt:

```
SCANNER;
```

The utility provides you with four commands:

SCAN (SCN)	Displays specified details of an output.
LIST (LS)	Lists the outputs associated with a job.
PRINT (PR)	Prints an output or selected pages of an output.
QUIT (Q)	Exits SCANNER.

Example:

```
S: DPRINT MEMBER1 HOLD;
S: SCANNER;
C: LIST;
   INDEX     NAME    P   C    DV     SIZE
   X7423:3                    PR     unknown
   MYPROJ.MYLIB.MEMBER1;
C: SCAN X7423:3;
R: +12;
R: L3;
This is line 13
This is line 14
This is line 15
R: Q;
CANCEL(0),HOLD(1),RELEASE(2),LOCAL
PRINT(3),SCAN(4):2
OU10 RO  1 OUTPUTS RELEASED
C: Q;
S:
```

In the above example, DPRINT is issued with the HOLD parameter to keep the output queued. This parameter is recommended if you intend to use SCANNER.

You then LIST the queued outputs and enter SCAN using the indicated RON. You move forward to line 12 and display the next 3 lines on the screen. Q exits from SCAN and, since the scanned text is as expected, you can choose the RELEASE parameter (2) to release the member from HOLD.

For further information about SCANNER, refer to the *IOF Terminal User's Reference Manual Part 3*.

3. Using IOF as a Main Operator

This section describes some of the tasks of the user acting as main operator. Users involved in job management should first read Book 2 for background material on the concept of jobs and how to manage them.

This section tells you how to:

- execute, modify, and terminate jobs;

- send a Message Of Today;

- manage outputs;

- save and restore files and disks;

- use the console log.

Commands you will use:

- for Job Management

ENTER_JOB_REQUEST   MODIFY_MOT

- for Output Management

SAVE_FILE   SAVE_DISK
RESTORE_FILE   RESTORE_DISK

3.1   JOB EXECUTION AND TERMINATION

3.1.1   Submitting Batch Jobs

A **batch job** is a job that runs without user intervention. To submit jobs for execution in batch mode, use the ENTER_JOB_REQUEST (EJR) directive.

You can submit two kinds of job in batch mode: **absentee jobs** and **conventional batch jobs**.

An **absentee job** is a job that performs batch execution of a GCOS Command Language (GCL) command. You can execute only one GCL command at a time as an absentee job.

A **conventional batch job** is a job that runs sequences of Job Control Language (JCL) statements.

Absentee job

To submit an absentee job, specify the name of the command to execute and its parameters at the S: prompt:

```
EJR PROC=CBL -
-: VALUES=(MYPG INLIB=.MINE.SL CULIB=.MINE.CU -
-: MAP PRTLIB=.MINE.LIS DEBUG XREF);
```

In this example the EJR directive requests compilation of a COBOL 85 program with the specified options and libraries.

PROC specifies the name of the GCL command.

VALUES specifies the parameters and their values for the GCL command to execute. Enclose the entire syntax within parentheses.

Conventional batch job

In a conventional batch job, you use the EJR directive to call for a sequential file or library member that contains a pre-written set of JCL statements. You may have a list of such files or library members to perform certain repetitive tasks.

To submit a conventional batch job, specify the name of the library and member or the name of the file containing the job description. The syntax for libraries, at the S: prompt, is:

```
EJR MEMBERS=<member-names>
-: LIB=<library-name> -
-: VALUES=<parameter-values>;
```

The syntax for a sequential file is:

```
EJR INFILE=<file-name> -
-: VALUES=<parameter-values>;
```

VALUES specifies the value(s) to supply for the JCL statements contained in the job description.

The ENTER_JOB_REQUEST directive has many other parameters. You can use them to define the processing of your jobs. The most commonly-used parameters are:

- USER, PROJECT, and BILLING, which allow you to submit a job for another user.

- CLASS, expressed as one or two letters in the range A to P, defining processing options such as execution priority.

- WHEN, EVERY, and WAIT, which make job execution conditional upon other events.

For a complete description of the ENTER_JOB_REQUEST directive, refer to the *IOF Terminal User's Reference Manual Part 2*.

3.1.2   Monitoring Jobs

Once you have submitted your job for execution, use the following directives to monitor and control its progress:

DISPLAY_JOB (DJ)   Displays the current status of selected jobs.

HOLD_JOB (HJ)   Prevents execution of one or more jobs.

RELEASE_JOB (RJ)   Cancels the effect of the HOLD_JOB directive.

CANCEL_JOB (CJ)   Cancels the execution of one or more jobs.

As soon as a job is submitted, you receive a message informing you of the job's Run Occurrence Number (RON):

```
Xnnnn   IN    name    .....
```

All the commands used to monitor batch jobs require this number as their first parameter. If the submitted job sends messages to your terminal, the messages are prefixed by the RON.

It is important to know which RONs were given to your jobs by the system. If you forget them, use DISPLAY_JOB (DJ) at the S: prompt to list all jobs known to the system together with their RON:

```
DJ;
```

The status of your job changes as it progresses through the system. The main statuses are:

IN   being read in.
SCH   waiting for execution.
EX   executing.
OUT   waiting for the printing of output.

A message is sent to the terminal each time a job passes to a new status. You can also execute DJ with the RON to display the current status of a specific job:

```
S: DJ Xnnnn;
```

Possible messages:

```
Xnnnn IN                  the job has been read in.
Xnnnn STARTED             execution has started.
Xnnnn COMPLETED           normal end of execution.
Xnnnn ABORTED             execution aborted.
Xnnnn OUTPUT COMPLETED    printing of outputs completed.
```

Once you receive the message OUTPUT COMPLETED, the job finishes and no longer exists under GCOS 7. Any attempt to display the status of such a job results in the message UNKNOWN JOB.

You can prevent a job from executing with HOLD_JOB (HJ):

```
HJ Xnnnn;
```

This directive is useful in circumstances where you submit a job and later discover that some other work or job must occur as a prerequisite to the job's execution. HJ allows you to hold the job and do the necessary work.

When you are ready to release the job, use the RELEASE_JOB (RJ) directive:

```
RJ Xnnnn;
```

3.1.3 Terminating Jobs

When a job terminates normally, the system displays the message:

```
hh.mm ron.ssn COMPLETED jnm usernm jclass
```

The job remains in the system until all its outputs have been processed by Output Writer. The system then displays the message:

```
hh.mm ron.ssn OUTPUT COMPLETED jnm usernm jclass
```

After a job terminates, GCOS 7 continues looking for the next job to run, unless a TERMINATE_SYSTEM command was issued.

If there are no jobs to start (that is, there are no jobs in the IN SCHEDULING state) and there are no more jobs executing, the system displays the message:

```
hh.mm GCOS : NO MORE JOBS RUNNING
```

Terminating a job before completion

In some situations you may want to terminate a job before it is completed, no matter what its status is. For example, you might suspect that there is something wrong with the job (it is consuming too much CPU time) or that it requires some resources which you know are not available.

To terminate a job in progress, use the CANCEL_JOB (CJ) directive:

```
CJ Xnnnn;
```

When a job is terminated by CJ, the system displays the message:

```
hh.mm ron.ssn KILLED jnm usernm jclass
```

Jobs aborted

A job can sometimes abort prematurely because:

- a fatal error was found in the job description during translation;

- an abnormal condition occurred during execution;

- a CANCEL_REQUEST (CR) command was used to reply to a system request for a certain device or volume.

If any of the above incidents occurs, the system displays the message:

```
hh.mm ron.ssn ABORTED jnm usernm jclass [SEV sevnb]
```

SEV sevnb specifies the severity of the termination. Table 3-1 below indicates the various abort severity levels and their causes.

Table 3-1. Abort Severities and their Causes

Severity	Status	Meaning
SEV3	10000 - 19999	User error.
SEV4	20000 - 49999	System error e.g., system overload.
SEV5	50000 - 59999	CJ (The operator did not specify STRONG in the CANCEL_JOB command).
SEV6	60000	System crash.
SEV7	"8000"X	System file error (probably I/O Error).

3.2   MESSAGE OF TODAY

You can set a **Message Of Today** (MOT) with the MODIFY_MOT (MDMOT) command. The MOT is a means for communicating important information to users. For example, it may inform users that a device is inoperable or that the system will be shut down on a specified day for maintenance.

The MOT is displayed on a user's terminal during log-on and is printed in the Job Occurrence Report (JOR) of all jobs.

To display the current value of the MOT, use the DISPLAY_MOT (DMOT) command at the S: prompt:

```
DMOT;
```

To set an MOT, use the syntax:

```
MDMOT MSG='<message>';
```

Examples:

```
S: MODIFY_MOT MSG='System Shutdown at 18H 30';
```

```
S: MDMOT MSG='PR01 inoperable';
```

To erase the current MOT, type in the MDMOT command without the MSG parameter:

```
S: MDMOT;
```

3.3 OUTPUTS

3.3.1 Output Writer

The outputs created by jobs are put into a queue and printed by the WRITER service job. The **Output Writer** controls the selection and printing of queued outputs for the WRITER service job. The use of the Output Writer allows the system to run more jobs than there are printers to print all the outputs that jobs create.

The maximum number of outputs for a job that can be known to the Output Writer is 254. A job can, however, produce more than 254 outputs while it is running, since outputs are deleted from the system immediately after printing.

GCOS 7 automatically provides names for all outputs created by a job. The output name is in one of two forms:

```
Xnnnn:<user-name>     If a user name was provided.
Xnnnn:i               The ith output of the job.
```

The system produces a **Job Occurrence Report (JOR)** for each job run. The JOR contains all system information concerning the execution of the job. The JOR always takes the output name **Xnnnn:1**.

If a step of a job produces output, then the system generates a report for this step, similar to the JOR. This report is called **JOB_OUT**. JOB_OUT always takes the output name **Xnnnn:2**.

3.3.2 Monitoring Outputs

The set of commands used to monitor outputs is similar to the set of commands used to monitor jobs:

```
DISPLAY_OUTPUT (DO)
```
Displays the current status of outputs.
```
HOLD_OUTPUT (HO)
```
Prevents the printing of outputs in the queue or suspends printing of outputs.
```
RELEASE_OUTPUT (RO)
```
Cancels the effect of a previous HOLD_OUTPUT directive.
```
CANCEL_OUTPUT (CO)
```
Cancels an output.
```
MODIFY_OUTPUT (MDO)
```
Modifies the characteristics of outputs.

To display the status of outputs, use the DISPLAY_OUTPUT (DO) directive:

```
DO;
```

Displays outputs of the current session.

```
DO X0234;
```

Displays all outputs of job X0234.

```
DO X0234 Z;
```

Displays all outputs in class Z of job X0234.

```
DO X0234:2;
```

Displays output 2 of job X0234.

To prevent an output from printing, use the HOLD_OUTPUT (HO) directive.

You may want to hold the printing of certain outputs for various reasons. For example, an output may be exceptionally long and/or of low priority, so you can hold it to allow more important listings to print first. Or you may suspect that some listings should not be printed, and you place them on HOLD until you can consult with the user.

Examples:

```
S: HO X0234;
```

Holds all outputs of job X0234.

```
S: HO X0234:1;
```

Holds output 1 of job X0234.

```
S: HO X0234 P;
```

Holds all outputs in class P of job X0234.

When you are ready to print an output that was in the HOLD state, use the RELEASE_OUTPUT (RO) directive:

```
RO X0234;
```

Releases all held outputs of job X0234.

```
RO X0234:2;
```

Releases output 2 of job X0234.

```
RO X0234 Z;
```

Releases all outputs in class Z of job X0234.

If a user does not want an output, use the CANCEL_OUTPUT (CO) directive to cancel it. You can cancel an entire output or selected portions of an output:

```
CO X0234;
```

Cancels all outputs of job X0234.

```
CO (X0234 X0567);
```

Cancels all outputs of jobs X0234 and X0567.

```
CO X0234:2;
```

Cancels output 2 of job X0234.

```
CO X0234 A;
```

Cancels all outputs of job X0234 of class A.

You can modify the characteristics of an output with the MODIFY_OUTPUT (MDO) directive. The syntax is:

```
MDO OUTPUT=<output-name>
    SELCLASS=<job-class-name>
    CLASS=<new-job-class-name>
    PRIORITY=<new-priority>
    COPIES=<number-of-copies>;
```

OUTPUT specifies the name of the output.

SELCLASS specifies a class or range of classes.

CLASS specifies a new class for the designated output.

PRIORITY specifies a new priority for the designated output.

COPIES specifies the number of copies to print.

Examples:

```
S:MDO OUTPUT=X0123 SELCLASS=C -
-:    CLASS=D PRIORITY=5;
```

Sets outputs of X0123 of class C to class D and priority 5.

```
S: MDO OUTPUT=(X0123:2 X0123:3) CLASS=C;
```

Sets two outputs of X0123 to class C.

```
S: MDO OUTPUT=X0123:2 COPIES=3;
```

Prints three copies of the named output.

3.3.3   Sending Outputs to a Sequential File

You can instruct the Output Writer to send an output to a sequential disk or tape cartridge file instead of to the system printer with the START_OUTPUT_WRITER (SOW) command. The syntax, at the S: prompt, is:

```
SOW ON=<file-name:md:dvc>
    SELCLASS=<job-class-name>
    DVC=PR/<printer-name>;
```

ON specifies the name and location of the specified file.

DVC specifies the name of the printer that the output will eventually be printed on.

You can use the DPRINT command to print the saved output later on the designated device.

Example:

```
S: SOW ON=OUTSAV:CT1:CT/M6 SELCLASS=A -
-: DVC=PR/H160;
```

In this example all outputs of class A are saved on the tape cartridge file OUTSAVE, emulating a 160-hammer printer.

Note that you can use the SOW command to regulate the Output Writer in many different ways. For a complete description of the SOW command, refer to the _System Operator's Guide._

3.3.4   Restarting an Interrupted Output

You can interrupt the printing of an output with one of the following commands.

- `TERMINATE_OUTPUT_WRITER (TOW)` with the STRONG parameter.

- `HOLD_OUTPUT (HO)` with the STRONG parameter.

- `CANCEL_REQUEST (CR),` in reply to a device incident.

When the printing of an output is interrupted by one of these commands, the system displays the question:

```
nn/OU02 dvnm OUTPUT ron :ouseqnb RESTART FROM ?
```

You type in one of the following replies. The reply determines the way in which the system will continue printing:

`BEG [HOLD] [TERM]`	Resume from the beginning of the output.
`nn [HOLD] [TERM]`	Resume nn pages before the current page.
`CUR [HOLD] [TERM]`	Resume from the beginning of the current page.
`CONT`	Resume from the next line (used only to bypass the current line, the line that caused the failure).
`FOR=n [HOLD] [TERM]`	Resume n pages after the current page. Jumps over n-1 pages without printing them.
`CAN [TERM]`	Cancel the output.

HOLD indicates that the output is to be put into the HOLD state. In this case, you must execute RELEASE_OUTPUT (RO) to resume printing. Note that if the question was displayed because of a HOLD_OUTPUT (HO) directive, then HOLD is implicit.

TERM indicates that all output classes are to be terminated on the specified device. If the question was displayed because of a TERMINATE_OUTPUT_WRITER (TOW) directive, then TERM is implicit.

Output Writer incidents

If an incident occurs on the system printer during the printing of an output, the system displays message DV12:

```
DV12 dvnm RECOVER volnm FOR ronlist
```

To reprint all or part of the output, first issue a CANCEL_REQUEST (CR) command to display the question OU02. After correcting the printer incident and switching the device to READY, reply to OU02 with the appropriate message (page 3-21 above).

If the incident is more serious, the system displays message OU27:

```
                       {HELD      }
OU27 dvnm ron :ouseqnb {          } [WRITER TERMINATED]
                       {CANCELLED }
     RC = ccccccc      siu,return code [ON SYS.OUT FILE]
```

followed by question OU02. Note that in many cases message OU27 ends after the display of HELD or CANCELLED.

After correcting the incident, output processing resumes at the point specified in your reply to question OU02.

If message OU27 displays HELD, release the output with RELEASE_OUTPUT (RO).

If the message displays CANCELLED, then the output was cancelled and is lost.

If the message displays WRITER TERMINATED, restart the Output Writer with the SOW command.

Switching output to a file

If you determine the printer cannot be reactivated, you can switch the Output Writer to a sequential file.

First issue CANCEL_REQUEST (CR) at the S: prompt with the name of the printer in the form:

```
CR PR<printer-name>;
```

For example:

```
S: CR PR01;
```

This displays the OU02 message. Reply **BEG** to restart output from the beginning.

Next terminate the Output Writer on the inoperable printer with TOW. To continue the example with PR01:

```
TOW PR01 STRONG;
```

Now start the Output Writer on the disk (or tape cartridge) sequential file. In this example the name of the file is SAVE.

For a disk:

```
SOW ON=MYPROJ.SAVE DVC=PR01;
```

For a tape cartridge:

```
SOW ON=SAVE:CT1:CT DVC=PR01
```

SYS.OUT error

Sometimes a SYS.OUT file will have errors that prevent the Output Writer from printing a line. When this happens, the system displays message OU27 with **ON SYS.OUT FILE**, followed by question OU02.

Reply **CONT** to specify that the system should bypass the line at fault and that processing should resume on the next line.

This series of messages will occur for each bad line encountered by the system. If there are many of these errors, use the HOLD option in reply to OU02 and consult the user.

3.4   SAVES AND RESTORES

An important function for the operator is to save system and user files and disks regularly and to restore damaged files and disks. The frequency of saves and restores depends upon the requirements of your particular site.

In some cases regular saves and restores are handled by stored routines. Here, the operator's job is to run these routines at the prescribed times.

From time to time, however, you may need to save or restore files or disks by using GCL commands. The commands you will use are:

SAVE_FILE (SVF)	Saves a disk file onto a UFAS disk file or onto a tape cartridge.
SAVE_DISK (SVD)	Saves a disk volume onto a UFAS disk file or tape cartridge.
RESTORE_FILE (RSTF)	Restores the contents of a disk file saved by a SAVE_FILE command or restores the contents of a disk file from a tape cartridge file created by SAVE_DISK.
RESTORE_DISK (RSTD)	Restores a native disk volume from a UFAS disk file or from a tape cartridge file created by SAVE_DISK.

Examples:

```
S: SVF FILE=MYPROJ.A OUTFILE=MYPROJ.MYSAVE;
```

Saves the cataloged file MYPROJ.A onto the tape cartridge file MYSAVE.

```
S: SVD VOL=FSA3:MS/FSA OUTFILE=DISCSAVE:CT1:CT;
```

Saves disk FSA3 onto the tape cartridge file DISCSAVE.

```
S: RSTF FILE=PF:FSA3:MS/FSA -
-: INFILE=SAVE:CT1/CT2:CT;
```

Restores the disk file called PF from the two tape cartridges called CT1 and CT2.

```
S: RSTD VOL=FSA3:MS/FSA INFILE=MYPROJ.F;
```

Restores disk FSA3 from the cataloged sequential file MYPROJ.F.

3.5 **USING THE CONSOLE LOG: SYS.LOGC**

The main operator's console dialog is logged in a system file
called SYS.LOGC. SYS.LOGC keeps a record of all console
dialog with the system and all messages sent to the main
operator.

SYS.LOGC is particularly useful on a site where there is no
hardcopy printer or for periods of time when the hardcopy printer
is inoperable.

You manage the SYS.LOGC with the commands
START_CONSOLE_LOG and TERMINATE_CONSOLE_LOG.

START_CONSOLE_LOG (SCLOG) activates the logging of the
system console. If you include the parameter CLEAR=1, the
system clears the contents of SYS.LOGC before logging starts:

```
SCLOG CLEAR=1;
```

You turn off system logging with TERMINATE_CONSOLE_LOG
(TCLOG):

```
TCLOG;
```

When the SYS.LOGC file is half-full, the system displays the
message:

```
EL01    SYS.LOGC FILE 50% FULL

* EL03  RUN PRLOGC
```

To print the contents of SYS.LOGC and thereby free up the
space in the file, use the PRLOGC utility in the EJR directive:

```
EJR PRLOGC LIB=SYS.HSLLIB;
```

There are also options available for modifying the printing of the
file contents. Refer to the *System Operator's Guide* for details.

If SYS.LOGC becomes full and PRLOGC was not run, the system displays the message:

```
EL01    SYS.LOGC FILE FULL
        DATA LOST UP TO <date>
```

This message indicates that all information in the file up to the specified date was lost.

A. Sample IOF Session

```
*************************************************************
***                                                       ***
***      WELCOME TO BC0E     OPERATOR TELEPHONE:(21)6216   ***
***                                                       ***
*************************************************************


Your default SL output library is PAY.SL-JJ

S: LMN SL
>>>13:50 LMN  50.00   150  -8
C: D FILES


        LIB     : PAY.SL-JJ                          B121


C: INLIB1 PAY.ARCHIVE
C: D FILES


        INLIB1  : PAY.ARCHIVE                         B125
        LIB     : PAY.SL-JJ                           B121


C: LIST LIB: ^* DIRINF

   DIRECTORY OF : LIB             PAY.SL-JJ
            MEDIA : B121
            NUMBER OF ALLOCATED BLOCKS :       5754
            NUMBER OF BLOCKS USED :            4693
            NUMBER OF FREE BLOCKS :            1061
            PERCENTAGE OF SPACE USED :          81


  *NO MEMBER SELECTED

C: D FILES


        INLIB1  : PAY.ARCHIVE                         B125
        LIB     : PAY.SL-JJ                           B121


C: IL1
```

```
C: D FILES

        LIB    : PAY.SL-JJ                          B121

C: PR GUIDES

    10 JJ_SWGUIDE_CONTENTS
    20 JJ_SWGUIDE_PREFACE
    30 JJ_SWGUIDE_SECTION1
    40 JJ_SWGUIDE_SECTION2
    50 JJ_SWGUIDE_SECTION3
    60 JJ_SWGUIDE_SECTION4
    70 JJ_SWGUIDE_INDEX

C: LIB PAY.ARCHIVE
C: D FILES

        LIB    : PAY.ARCHIVE                        B125

C: LS ZZTURM* NAMES

   MEMBERS

   ZZTURM_PR
   ZZTURM_PRLLSAVE
   ZZTURM_PRTEST
   ZZTURMV5_PR

   NUMBER OF MEMBERS : 4

C: LS ZZTURM*

   MEMBERS
            ---CREATED-------MODIFIED----MOD--LANG--LINES
   ZZTURM_PR
            05/26/89 08:07 05/26/89 08:07 00    DAT      2
   ZZTURM_PRLLSAVE
            05/25/89 09:19 05/26/89 08:09  1    DAT      2
   ZZTURM_PRTEST
            05/25/89 09:22 05/25/89 09:22 00    DAT      2
   ZZTURMV5_PR
            03/28/89 14:13 03/28/89 14:13 00    DAT     38

   NUMBER OF MEMBERS : 4        TOTAL NUMBER OF LINES : 44
```

Sample IOF Session

```
C: DPRINT ZZTURM* LIB=PAY.ARCHIVE CLASS=A DEST=BP3C -
-: BANINF=('PAY','ZZTURM','ROBERT','MS678') COPIES=2
C: EDIT
R: RLL_LISTALL
R: ^,$L
    10 $JOB LISTALL;
    20 OUTVAL -
    -: BANINF=('PAY.ARCHIVE','MEMBERS','MUGGLES') -
    -: DEST=BP3C;
    30 LMN SL LIB=PAY.ARCHIVE COMMAND='LIST *
    40 $ENDJOB;R: 20s/MUGGLES/ROBERT/L
    20 OUTVAL BANINF=('PAY.ARCHIVE','MEMBERS','ROBERT')
       DEST=BP3C;R: Z
C: EJR LL_LISTALL
C:
-->13:59 X8589 IN LISTALL ROBERT P SPR=7 MAIN
-->13:59 X8589 STARTED LISTALL ROBERT P
C: HO X8589
C:
C: DS
   13:59 X8584.2 EX   IOF      ROBERT       Q
         X8589.1 EX   LISTALL  ROBERT       P
C: DS X8589
   14:01 X8589.1 EX   LISTALL  ROBERT       P SPR=7 DPR=9
                    CPU=0.147 ELAPSED=1.731
C:
-->14:01 X8589.1 COMPLETED LISTALL ROBERT P
C: DO X8589
14:01 DO  STATION=BP3C
      DO  X8589:1 JOB_REP  3 C PR HOLD  LINES=44   PAGES=
      DO  X8589:3          3 C PR HOLD  LINES=4307 PAGES=79
C: CO X8589
14:02 X8589 OUTPUT COMPLETED LISTALL ROBERT
      OU10 CO    2 OUTPUTS CANCELLED
C: INLIB1 PAY.ARCHIVE
C: OUTFILE PAY.SEQFILE
C: D FILES


        INLIB1  : PAY.ARCHIVE                        B125
        OUTFILE : PAY.SEQFILE                        B125


C: SAVE IL1: LL_LISTALL
C: Q
<<<14:03
S: CBL SOURCE=MAIN-PROG LIB=COBL.SL_LIB XREF -
-: CULIB=COBL.CU_LIB
```

```
>>>14:04 COBOL V-01.7-2
   JUL 06, 1989 14:04:24 X8584.3 PROCESSING OF
   MAIN-PROG:COBL.SL_LIB
   SUMMARY FOR MAIN-PROG: NO ERRORS, CU PRODUCED
<<<14:04
S: CBL SOURCE=SECONDARY-PROG LIB=COBL.SL_LIB XREF -
-: CULIB=COBL.CU_LIB
>>>14:05 COBOL V-01.7-2
   JUL 06, 1989 14:05:07 X8584.4 PROCESSING OF
   SECONDARY-PROG:COBL.SL_LIB
   SUMMARY FOR SECONDARY-PROG: NO ERRORS, CU PRODUCED
<<<14:05
S: LINKER MAIN-PROG INLIB=COBL.CU_LIB OUTLIB=COBL.LM_LIB
>>>14:07 LINKER  90.00    17  -2
   WORKING ON : MAIN-PROG
   LK00.(90.00) SUMMARY FOR MAIN-PROG NO ERROR DETECTED
   OUTPUT MODULE PRODUCED
<<<14:07
S: EXEC_PG MAIN-PROG LIB=COBL.LM_LIB -
-: FILE1=INFILE ASG1=COBL.F1 -
-: FILE2=OUTFILE ASG2=COBL.F2
S: DS EX*
   14:08 X5465.1 EX    JTRA      JTRA        X
         X5466.1 EX    OMH       OPERATOR    Z
         X5472.1 EX    FEPS      OPERATOR    S
         X5474.1 EX    READER    OPERATOR    R
         X5481.1 EX    IOF       OPERATOR    Z
         X8579.7 EX    IOF       MUGGLES     Q
         X8578.3 EX    IOF       POULTRY     Q
         X8584.4 EX    IOF       ROBERT      Q
         X8580.3 EX    LINK      MUGGLES     D
         X8593.1 EX    MAIN-PRO  ROBERT      P

S: BYE


<<<14:11:49 ROBERT       LOGGED  OFF
<<<          CPU      0.123
<<<          ELAPSED  22.613
<<<
$$ 0200 VOUS ETES DECONNECTE: 0000
```

Glossary

Absentee job	A job that performs batch execution of a GCL command.
ACAS	Ares Console Application Software - The software that drives the system console and controls the interface to GCOS 7.
Access Control List (ACL)	A list that holds information on the access rights of all objects in the system.
Access rights	The degree of access that users have to files and other objects in the system.
Action field	The field in the IOF menu screen that is used to enter a move request.
Auxiliary Function Service	The part of the DPS 7000 that regulates the connection between the console and the main unit.
Backing store files (BKST)	Permanent disk files required by the system for extra storage.
Batch job	A job that runs without user intervention.
Billing	An accounting record.
Boolean parameter	A parameter that takes only the values 0 (NO) or 1 (YES).
Catalog	A file that records the links between a master directory and its files.
Catalog maintenance utility	The utility that contains the commands used to modify and maintain the site catalog.
CLEAN restart	The restart option that reinitializes the entire system by erasing all information from the previous session and resetting all system parameters to their default values.
CNP 7 Window	The window of the ACAS environment used to interact with the CNP 7's.
COLD restart	The restart option that erases all information from the previous session, but preserves the initialization settings.

Continuation prompt (-:)	A prompt that allows the extension of a single command line over the width of more than one terminal screen.
Control interval (CI)	The unit of transfer between main memory and a file on disk.
Control Window	The main window of the ACAS package, from which initiatives and other windows are accessed.
Conventional batch job	A job that runs sequences of Job Control Language (JCL) statements.
Corrective maintenance	The process of applying patches to system files.
CU library	Compiled Unit library - a library that holds compiled programs.
Data field	A unit of information belonging to a record.
Device class	The type of disk or tape cartridge that describes the volume.
Directive	A GCL command that is available anywhere in the system.
Directive identifier	The two- or three-character prefix (usually $$) that allows the user to execute a directive from within any GCOS 7 processor.
Directory	A group of files in a hierarchical tree structure.
Dispatching Priority (DPR)	The availability of CPU time for each executing job step.
Domain	A division of the production set that contains the system files relating to a specific set of functions under GCOS 7.
DSA Installation and Updating Facility (DIUF)	The group of commands used to install and maintain the files of the telecommunications domain.
EXECUTING state	The job state where an IN SCHEDULING job is executed.
Exit request (/)	The request used to exit from a command, processor, or Help text.
Firmware	Programmable circuitry.

Firmware Generation Facility (FGF)	The group of commands used to install and maintain the files of the firmware domain.
First in, first out (FIFO)	Priority order whereby the first job to enter is treated first by the system.
Free format	In line mode, the format in which the end of a command line requires the semicolon (;) to terminate the command.
Full path name	The complete name of a file, which includes the name(s) of the directory(ies) where it is located.
Full Screen Editor	The GCOS 7 text editor operating on full screen terminals.
GCL	GCOS 7 Command Language - The set of commands used to manage GCOS 7.
GCOS 7	General Comprehensive Operating System on the DPS 7000.
GCOS 7 Installation and Updating Facility (GIUF)	The group of commands used to install and maintain GCOS 7.
GCOS 7 Service Facility (GSF)	The group of commands used to install and maintain the files of the GSF domain.
GPOS	General Peripheral Processing Operating System.
Help request (?)	The request that calls up Help texts for a command or command parameter.
Help text	Pages of on-line text that contain information on how to use a command or command parameter.
HOLD state	The job state where a job is temporarily suspended by a HOLD command.
IDLE state	Job state for the service jobs JCL Translator and WRITER when they are not active.
IL01 options	The parameters that determine the way in which the system is loaded.

Initialization	The process of configuring and loading GCOS 7 for the first time on a DPS 7000.
Initiatives	The set of commands in ACAS that manage interactions with the operating systems of the DPS 7000.
Input library	A library used to read information from.
IN SCHEDULING state	The job state where a job is ready for execution
INTRODUCED state	The first job state, when a new job is translated into JCL.
IOF	Interactive Operation Facility - The GCOS 7 time-sharing application.
JCL	Job Control Language - the GCOS 7 language that controls the execution of jobs.
JCL Translator	The service job that translates a newly-submitted job into internal form.
Job	Work submitted to the system.
Job class	A group of jobs that perform related tasks.
Job Occurrence Report (JOR)	A printed report that contains all system information concerning the execution of a job.
JOB_OUT	A printed report that contains all system information concerning the execution of any job step that produces output.
Job run	The entire life of a job.
Job state	One phase of the life of a job.
Job step	A segment of a job.
Key	The data field(s) referred to first when the system is ordering a file.
Library	A file organized into a series of subfiles (members).
Library maintenance utility	The utility that contains the commands used to modify and maintain libraries.
Line format	In line mode, the format in which the end of the command line acts as terminator.

Line mode	A method of communicating with GCOS 7 in which the user enters the commands line by line, that is, without menus.
Log-on	The action or actions that a user performs to connect to GCOS 7.
Mailbox	A container for holding a user's messages.
Main operator	The operator who manages the daily functioning of a GCOS 7 system.
Mandatory parameter	A parameter of a command that requires a value in order for the command to execute.
Mandatory startup sequence	The startup sequence executed first at log-on.
Master directory	The first directory level in a hierarchical tree structure.
Media	The name of a volume.
Member	A library subfile.
Menu bar	The banner on a window used to activate pull-down menus.
Menu mode	A method of communicating with GCOS 7 in which the user is guided with menus.
Menu screen	In **menu mode**, a screen that displays a numbered list of commands.
Message Of Today (MOT)	A message sent by the main operator and displayed on user terminals for communicating important topical information about the site.
Mixed mode	The mode in GCOS 7 that allows the user to work in both line mode and menu mode.
Move requests	The set of characters used to move around the pages of menu screens.
Multiprogramming	The environment whereby more than one job is executing at any one time.
Multiprogramming Class Limit (MCL)	The maximum number of jobs that can be simultaneously executed or suspended in a job class.
Optional startup sequence	A startup sequence executed after the mandatory sequence.

Output library	A library used for read and write operations.
Output Priority	The order in which jobs are printed by WRITER.
OUTPUT state	The job state during which the outputs of a job are printed.
Output Writer	The program that controls the selection and printing of queued outputs for the WRITER service job.
Password	A string of up to 12 characters long, used to allow a specific user access to the system.
Patch	A modification applied to an existing system file member to correct a fault.
Permanent file	A file created for long-term use and stored on a volume.
Primary key	The key that determines the first ordering of the records of a file.
Private catalog	A catalog that holds information on the files belonging to one project only.
Processor	A group of commands that performs a specific set of functions in GCOS 7.
Production Set (P-Set)	The complete set of system files that constitutes the working GCOS 7 system.
Project	A logical group of users.
Project startup	A startup sequence that activates commands that pertain to a project.
Prompt lines	The horizontal lines at the bottom of a menu screen that allows command input in line mode.
Prompt mode	A special variation of **line mode** in which the user is prompted to enter parameter values.
Prompt Screen	In menu mode, a form used to display the parameters of a command.

P2P-Configuration	A system configuration with two production sets: P-Set and P2-Set, an alternative to the PO-Configuration (for P Only) delivered as standard with your DPS 7000.
Record	The unit of a file.
Reinitialization	The process of initializing the system again with a complete testing of all hardware.
Relative addressing	The method of addressing a line by its position relative to the current, first, or last line.
Reload	The process of loading the system again without running hardware tests.
Root	The basis of a hierarchical tree structure.
Run Occurrence Number (RON)	The number assigned to a job after it is submitted to the system.
SCANNER	The utility used to examine output contents on the terminal before they are printed.
Scheduling Priority (SPR)	The order in which jobs are queued while waiting to be run.
Scroll bar	The vertical bar in each window used by the mouse cursor to page through screens of text.
Scroll mode	The mode of operation activated by scrolling through screen text.
Secondary key	The key or keys that determine subsequent orderings of the records of a file (after the primary key).
Service job	Work carried out by a system function.
Service Window	The window of the ACAS environment used to interact with SPOS.
Simple name	Each component of the full path name of a file.
Site catalog	The catalog that holds information on the SITE files and on all other directories and their contents.
Site startup	A startup sequence that activates commands that pertain to an entire site.

SL library	Source Language library - a library that holds any kind of textual information.
Sort criteria	The basis of the ordering of file records.
SPM Window	The window of the ACAS environment used to interact with GPOS.
Spool files (SYS.SPOOL)	Files that contain procedures and environments for individual users of GCL.
SPOS	Service Processor Operating System.
Star-convention (*)	The character used to match any occurrence, including none, of characters within a name.
Startup sequence	A set of command entries that run automatically each time the user logs on.
Status line	The banner on a window that displays messages relating to current activity in the ACAS environment.
SUSPENDED state	The job state where a job is suspended while EXECUTING.
Symbolic names	A special set of names (LIB, INLIB1, INLIB2, and INLIB3) that represents the working libraries.
SYS.LOGC	The system file that keeps a record of all console dialog with the system and all messages sent to the main operator.
SYS.OUT	The standard system file for printing job outputs.
System Administrator	The person responsible for the technical management of a GCOS 7 system.
System builtins	Variables that provide various functions for displaying and manipulating information.
System catalog	The catalog that holds information on SYS files.
System Console	The microcomputer used to communicate with the DPS 7000.
System Console Window	The window of the ACAS environment used to interact with GCOS 7.
System dump file (SYS.SYSDUMP)	The file used to save main memory contents if the system fails.

System identifier   In console messages, a four-digit string, composed of two alphabetic characters and two numbers, used to identify the origin of a message.

System Installation Program (SIP)   A small operating system entirely resident in memory that monitors the execution of certain jobs when GCOS 7 is unavailable or inoperable.

System level commands   The GCL commands available at the first, or highest, level in GCOS 7.

System load   The number of jobs in execution at any one time.

System prompt (S:)   The prompt for the first level of commands in GCOS 7.

System startup   A startup sequence that activates commands that pertain to the way in which the system will load.

System variables   Variables that determine the way the user sees the system.

Technical Status   The updates to the files of a domain.

Temporary file   A file created for the duration of a job step or log-on session only.

Tenant Window   The window that dialogs with the GCOS 7 operating system. GCOS 7 must be loaded to use this window.

Text Editor   The GCOS 7 line editor used to create and modify SL library members.

Time stamp   In console messages, an identifier in the form **hh.mm** that records the time that a message was generated.

Time variance   The difference in time between GMT and the local time of the DPS 7000 site.

Title bar   The banner that contains the name of the displayed window.

TNS   Transport and Network Subsystem.

TNS job   The service job that handles communication between the system and the CNP 7.

UFAS indexed-sequential file	A disk file that contains indexes used to locate records.
User profile	The setting of system variables that pertains to an individual user.
User startup	A startup sequence that activates commands that pertain to an individual user.
Volume	A physical device (either a disk or a tape cartridge) used to hold information.
WARM restart	The restart option that brings back the system as near as possible to the state in which it was at the end of the previous session.
Working directory	The directory that contains a user's working files.
Working libraries	The input and output libraries that are active in the current session.
Workspace	The area in the Text Editor used to enter in and modify text.
WRITER	The service job that manages the printing of job outputs.

Index

The index in the Getting Started series is common to all three books. The references below indicate the book number in parentheses, followed by the section and page numbers.

D

E

F

G

Index

H

I

J

K

L

M

N

O

P

R

S

T

U

W