LXI CORP.

MMS®/*bms* - Backup and Recovery Management

for the iSeries

Software : MMS/*bms*

This version of the MMS/*bms* software is compatible with Releases V5R2 and above of the IBM OS/400Operating System. This product could be affected by future releases of OS/400.

IBM or LXI Corp. make no warranties or representations as to the quality of this code. The code is offered AS IS AND ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ARE SPECIFICALLY DISCLAIMED. In no event shall IBM or LXI Corp. be liable for special, consequential or incidental damages or lost profits, even if advised of the possibility thereof.

Address any written comments about this document to: LXI Corp., 1925 W. John Carpenter Freeway, Suite 485, Irving, TX 75063, USA. If you wish to fax information the number is (214) 260-2019. If you wish to transmit these comments electronically over the Internet, the address is main@lxicorp.com. You can also submit comments via the World Wide Web at the LXI Corp. web address, http://www.lxicorp.com. The company's main voice number is (214) 260-9002.

LXI Corp. may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligations whatsoever.

Backup & Recovery Management

1925 W. John Carpenter Freeway
Suite 485
Irving, Texas 75039
Phone 214.260.9002 • Fax 214.260.9019

Table of Contents

NOTICES

References in this publication to LXI Corp. products, programs, or services do not imply that LXI Corp. intends to make these available in all countries in which LXI Corp. operates. Any reference to any LXI Corp. product, program, or service is not intended to state or imply that only LXI Corp. product, program, or service may be used. Subject to LXI Corp.'s intellectual property or other legally protectable rights, any functionally equivalent product, program, or service may be used instead of the LXI Corp. product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by LXI Corp. are the responsibility of the user.

LXI Corp. may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to LXI Corp., 1925 W. John Carpenter Freeway, Suite 485, Irving TX 75063, U.S.A.

Licensees of this program who wish to have information about it for the purposes of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact the LXI Corp. R&D coordinator. Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

Address your questions to:

LXI Corp.

1925 W. John Carpenter Freeway

Suite 485

Irving, TX 75063 USA

This publication could contain technical inaccuracies or typographical errors.

This publication may refer to products that are announced but not currently available in your country. This publication may also refer to products that have not been announced in your country. LXI Corp. makes no commitment to make available any unannounced products referred to herein. The final decision to announce any product is based on LXI Corp.'s business and technical judgment.

This publication contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

This publication contains small programs that are furnished by LXI Corp. as simple examples to provide and illustration. These examples have not been thoroughly tested under all conditions. LXI Corp., therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. All programs contained herein are provided to you "AS IS." THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE EXPRESSLY DISCLAIMED.

Chapter 1

Introduction

Disasters come in all sizes. They range from total destruction of the company's buildings to a corrupt file or lost program. Data that cannot be recovered is lost forever. Recoverability is the primary reason that users perform backups. Data may be lost due to accidental deletion or through damage that prevents its use. It is only through backups that one can ensure business resumption if data becomes lost or unusable. The key to a successful recovery depends on what is being backed up and how often. A successful backup strategy will help ensure a successful recovery. MMS Backup and Recovery (MMS/*bms*) is a subset of the LXI Media Management System (MMS). This module is designed to help you implement a backup and recovery strategy and provide tools, which aid in recovery and help reduce recovery time.

Data needs to be protected for recovery purposes. In case of a disaster, it is the function of MMS/*bms* to provide the user with the information and tools needed to help ensure a successful recovery.

Why You Should Use MMS/*bms*

Many reasons exist for choosing MMS/*bms* for your backup management system, including its wide array of features and functions, ease of implementation, power, flexibility, and ease of use. For those of you familiar with other LXI products, the choice of MMS Backup and Recovery will be easy. MMS/*bms* can increase productivity and help eliminate errors caused by old, outdated and incomplete backup strategies. The flexibility provided allows you to customize and change your strategy as required without having to modify existing code. The strategy that you create can be reviewed at any time to help ensure that all your requirements are met.

Ease of implementation is key to using any software. MMS Backup and Recovery uses existing IBM facilities and requires no additional modifications to make it function. Since there is no need to make changes to existing code, MMS/*bms* is active and ready to use after the installation procedure has completed.

Flexibility When You Need It

Changes are the forte of any data center and MMS/*bms* is designed to adapt easily and quickly. The backup strategy that worked so well yesterday can be updated to meet today's challenges within a few minutes. Save commands that allow virtually every type of save provide solutions that can be implemented and maintained in one comprehensive software package.

How This Book Is Organized

This manual is organized to help you set up and use the software as quickly and efficiently as possible. If you are familiar with earlier versions of this product, you should scan the table of contents for new features. The MMS/*bms* manual is organized as follows:

- **System Overview**
  Chapters 2 through 4 outline the MMS/*bms* features and functions. These chapters also provide a guide to the menu system used by MMS/*bms*. Chapter 4 contains the Quick Start exercises, which illustrate the simplicity of use.
- **Description of Major Functions**
  Chapters 5 through 9 detail all MMS/*bms* functions including the creation of backups and recoveries plus details on using the Intelligent Backup.
- **Reports**
  Chapter 10 shows the reports available and how to use them in establishing a backup/recovery strategy. These reports provide you with the information necessary to ensure that your libraries are being saved.
- **Command References**
  Chapter 11 provides a list of all MMS/*bms* commands, command parameters and values allowed. For those familiar with commands and wishing to bypass the menus, the commands provide a fast means of setting up and using the product.
- **Installation Instructions**
  Chapter 12 contains the information required to successfully install this product. Information on license keys is also provided.
- **Options**
  Chapter 13 provides information on setting up MMS/*bms* to process concurrent saves and Chapter 14 shows an example of an exit program.
- **Troubleshooting Guide**
  Chapter 15 lists the most commonly asked questions regarding MMS/*bms* functionality. If MMS/*bms* does not function as expected, this appendix can provide you with valuable insight quickly.
- **Software Support**
  Chapter 16 provides instructions for accessing Electronic Software Support from the LXI technical support staff. In the event that you need a Program Temporary Fix (PTF) or online support, this chapter walks you step-by-step through the process of getting help.

Conventions Used

The conventions that are used in this manual have been established to help you learn and use the product quickly and easily.

The first time a function is referenced, it is displayed in **bold** type.

Menus, displays, and command prompts are shown as needed to help explain a function or location of a function.

Default parameters for commands are **bold** and **underlined**.

Command Key Actions

To help minimize the time required to learn MMS Backup and Recovery, IBM command key standards have been followed whenever and wherever possible. The following graph shows some of the commands and their use within this product. The command keys available and their associated functions are shown at the bottom of each menu and display.

Command Key	Function	Description
F1	Help	Displays cursor-sensitive help text.
F3	Exit	Exits the function and returns to the prior function.
F4	Prompt	Prompts the user for command parameters.
F5	Refresh	Updates the displays with current information.
F12	Cancel	Cancels the requested function.

Online Help

MMS/*bms* provides online help for all commands, menus, and displays. The help provides additional information on a function or field. To access help, position the cursor on the field or parameter in question and press the **F1** key.

MMS/*bms* error messages may also provide additional information on the cause of the error and the corrective action to take. To retrieve additional message help, place the cursor on the message and press the **F1** key. If second level help is available, it will be displayed.

Before You Install

Before installing this product, review the items below. Knowing this information from the beginning will simplify using MMS/*bms*.

Command Security

MMS/*bms* is a command driven software product. All menu and display options reference either an IBM or a MMS/*bms* command. Command authority for MMS/*bms* is achieved in the same way that authority is established for IBM commands. If a user is not authorized to use a MMS/*bms* command, the function that the restricted command performs will not be available for use and the option number will not be displayed. If the user tries to access the command directly via command line, he will receive a message from OS/400 stating that he is not authorized to use the command. Refer to the appropriate IBM manual for details on establishing or changing command authority.

System Defaults

MMS/*bms* command defaults conform to iSeries system defaults, where applicable. Overrides can come from IBM commands as well as MMS/*bms*. In areas where IBM has no matching default, MMS/*bms* uses values that cause the software to use the fewest resources and execute the fastest. If the MMS/*bms* command defaults are changed, it is the users responsibility to maintain the changes during product upgrades.

Requirements

Note the following requirements before using MMS/*bms*.

- MMS Tape Management (MMS/*tms*) must be installed and active.
- If you are using MMS/*bms* to save output queues and/or spooled files, the MMS Spooled File Management (MMS/*spl*) must be installed and active.
- If you are using MMS/*bms* to save Lotus Notes/Domino, the MMS Lotus Notes/Domino module (MMS/*lnd*) must be installed and active.
- If you are using MMS/*bms* to save clients, the MMS Server module (MMS/*svr*) must be installed and active.

Chapter 2

Features and Functions

This chapter documents some of the most important features in MMS Backup and Recovery. If you are an experienced user, browse through this chapter to find what is new and what features have been added.

Changes in MMS/*bms* are of two types: those that enhance existing features or make them easier to use, and new features that add flexibility and power to MMS/*bms*.

Intelligent Backup

MMS/*bms* eliminates the need to build and maintain Backup Lists. This feature searches for new and/or changed objects, folders and documents and integrated file system objects and creates Backup Lists for them. Security and configuration data is also included. This feature ensures that all new or changed objects exist on tape. The need to build and maintain Backup Lists is eliminated and true backup automation is achieved with the Intelligent Backup.

Intelligent Recovery

MMS/*bms* eliminates the need to build and maintain Recovery Lists. This feature searches for the latest saves and builds the necessary Recovery Lists for OS/400, security and configuration data, user and IBM libraries, folders and documents and integrated file system objects. This feature ensures that all the objects and their changes are restored. The need to build and maintain Recovery Lists is eliminated and true recovery automation is achieved with the Intelligent Recovery.

Parallel Save Support

MMS/*bms* provides support for IBM media definitions (parallel saves/restores). Media definitions provide the ability to perform load balancing and use multiple devices in parallel when performing backups.

Flexible Backup Lists

MMS/*bms* allows you to group different types of saves together to form a comprehensive backup strategy. A single backup can multiple Backup Lists that perform virtually every type of save allowed by OS/400. Parallel save support and Omit Lists add extra flexibility to the Backup Lists.

New Library Support

Automatically save new libraries without having to worry if someone added them to a Backup List. With this support, there is no need to wonder if new libraries are being saved.

Concurrent Saves

Maximize your resources by using concurrent save support. With this support, you can perform multiple saves at the same time to multiple devices, thereby maximizing your backup window.

Multiple Device Support

Maximize your devices by using multiple device support. With this support, a tape volume on one device rewinds and unloads while another device processes the next tape.

Recovery Options

Need to recover a single object? Need to recover all or some of last night's changes? Need to recover all or some libraries saved with a **SAVLIB** command? Need to recover a Domino/Lotus Notes server? Need a recovery strategy that matches your backup strategy? With MMS/*bms* recovery, these functions are easily accomplished. The flexibility of MMS/*bms* recovery makes restoring easy. Messages constantly inform you of the status and the selection criteria provides you with the ability to recover what you need.

Resume Capabilities

If the backup failed, due to media, device or power failures, and there is not enough time to restart the backup, you can continue where it left off with the MMS/*bms* resume function. The resume function will search for the last sequence saved and then continue. This helps maximize the backup window by eliminating restart time. This powerful function, which requires MMS Tape Management (MMS/*tms*), resumes all library Backup Lists.

User Exits

With user exits, additional processing can be performed before, during and after a backup. These exits can be used to perform processing such as ending or starting jobs or subsystems, sending messages, submitting user jobs, or any other functions which may need to occur before, during or after a backup. To maintain compatibility with older versions of MMS/*bms*, a single program can be used for both pre and post exit processing.

Chapter 3

Menus

MMS/*bms* is a command driven product. As such, most functions can be easily initiated from an OS/400 command line or from within a high level program. When the product is first installed, the MMS/*bms* menu system provides an easy method of learning the commands associated with a particular function. Over time, as the commands become familiar, the menu system can be bypassed and the commands can be accessed directly.

The menu system is comprised of a main menu and four related command menus. Each command menu provides access to another related command menu. Depending on the function and level of menu currently displayed, the related command menu may be an LXI menu or an IBM menu.

Menu Groups

Menus are grouped by function. The following functions have their own menu.

- Backup Commands
- Backup Definition Commands
- Omit List Commands
- Reports
- Save Commands
- Security Commands

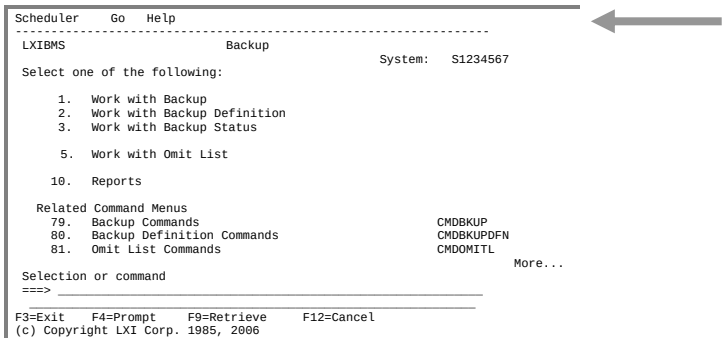
Menu Security

IBM security can be implemented for any MMS/*bms* menu or menu function. If a user is not authorized to a menu, the secured menu will not be displayed as an option from other MMS/*bms* menus. If a user is not authorized to a specific function on a menu, the option and related command will not be displayed. To change the authority of a MMS/*bms* menu or command, use the appropriate IBM command to change it.

Menu Bars

Some menus contain menu bars. Menu bars are located on the top of a menu and are assigned function names. Use the **Tab** key to position the cursor on the desired function. Once the cursor is in place, pressing the **Enter** key lists the options available. If you are using a mouse, double click on the desired function. This provides a list of the options available. Enter the desired option number in the option field provided and press **Enter**.

The following example shows the location of the menu bar on menu MMS/*bms*.



GUI Menus

All menus and displays show in the IBM Graphical User Interface (GUI) format if supported by the display device. This support means that all menus and displays will have a PC look and feel with an easy point and click interface. The function keys still work and the command line is available for use.

Chapter 4

Getting Started

In this chapter, you will learn how to implement and use the basic functions of MMS/*bms*. If you are a new user to MMS/*bms*, this chapter is important for two reasons: you will become comfortable navigating MMS/*bms*, and you will have a head start on the next MMS program you learn.

Simplicity is the key in getting started. No special commands are required. MMS/*bms* is active once the software is installed and, since MMS/*bms* uses standard IBM commands, compatibility with other products is assured.

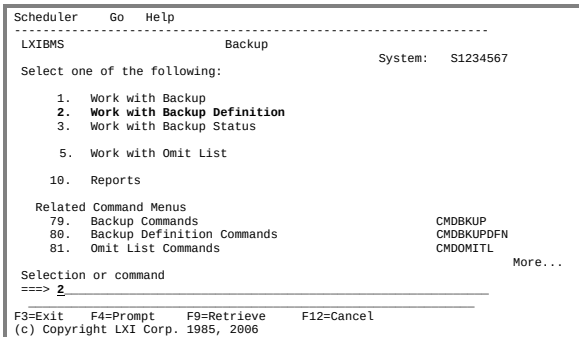
The purpose of this chapter is to:

- Create a Backup
- Create a Backup List
- Add a Backup List Entry
- Run a Backup

The remaining chapters provide additional information on other functions and options available to you.

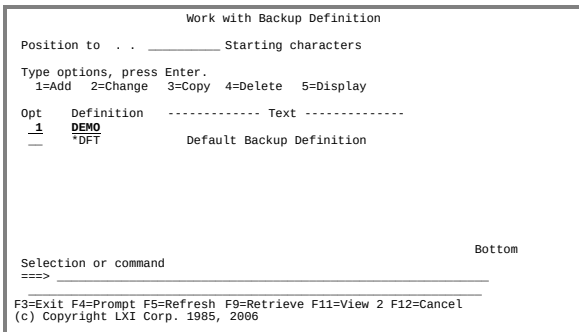
Step 1.

To access the Backup menu, enter **GO LXIBMS/LXIBMS** from any OS/400 command line. Backup Definitions must exist before a Backup is performed. To view existing Backup Definitions or create a new Backup Definition, choose **Option 2** from the Backup menu.



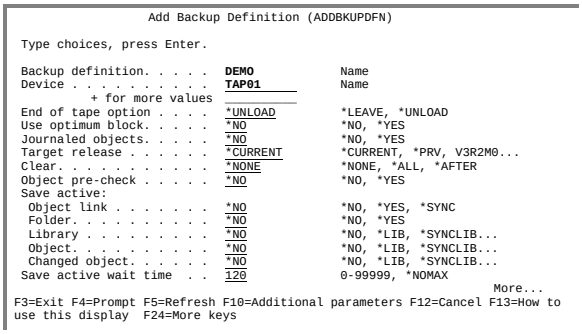
Step 2.

Using the Work with Backup Definition panel, choose **Option 1** and enter a Backup Definition name. In this example, the Backup Definition being created is named **DEMO**. Pressing **Enter** prompts the Add Backup Definition (ADDBKUPDFN) command.



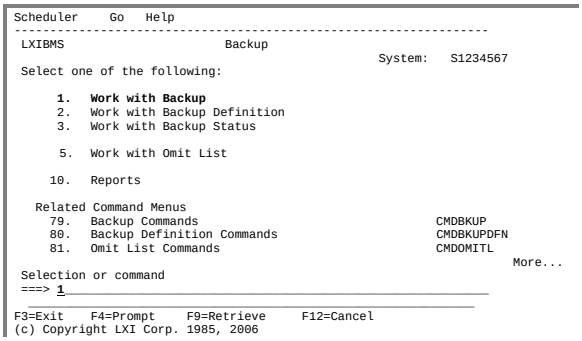
Step 3.

The Add Backup Definition (ADDBKUPDFN) command defines the save device, target release, save while active requirements and other tape specific attributes. Specify a tape device and press **Enter** twice.



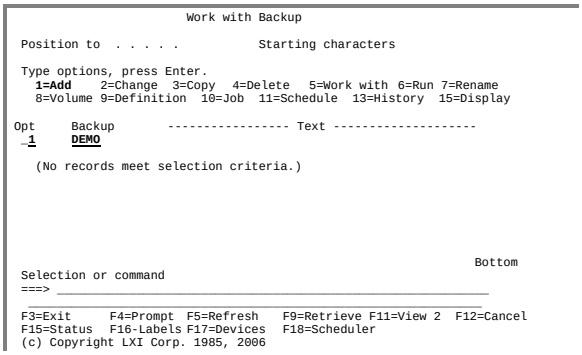
Step 4.

To view existing Backup jobs or create a new Backup job, choose **Option 1** from the Backup menu.



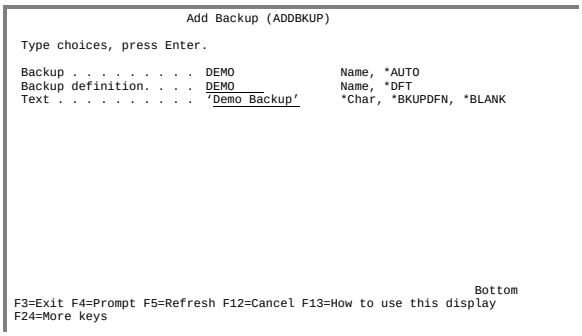
Step 5.

Using the Work with Backup panel, choose **Option 1** and type the name of the Backup job to create. In this example, the Backup job name is **DEMO**. Pressing **Enter** prompts the Add Backup (ADDBKUP) command.



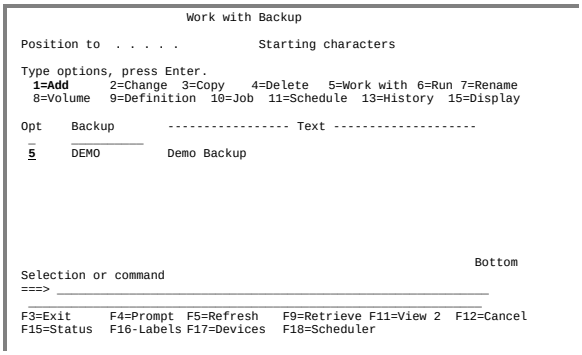
Step 6.

The Add Backup (ADDBKUP) command defines the Backup Definition to use and the text for the Backup. Specify **DEMO** as the Backup Definition. Press **Enter** to return to the Work with Backup panel.



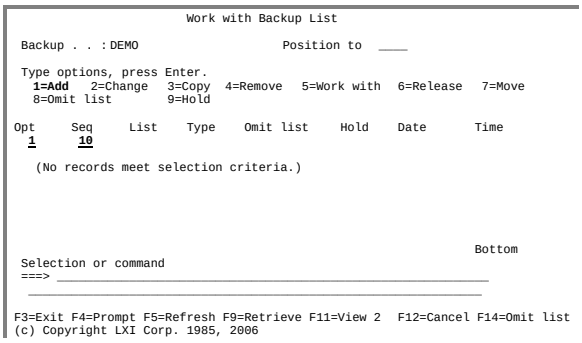
Step 7.

Select **Option 5** for the newly created Backup and press **Enter** to work with the Backup List.



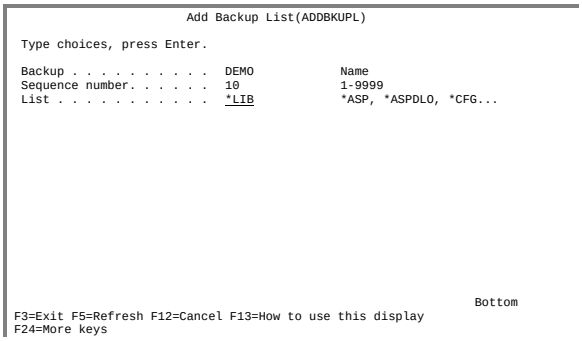
Step 8.

The Backup List displays the individual saves to perform. To add an entry to the Backup List, select **Option 1** and the sequence number of the Backup. Pressing **Enter** prompts the Add Backup List (ADDBKUPL) command.



Step 9.

The Add Backup List (ADDBKUPL) command determines the type of Backup List to create. A Backup List determines the type of save to perform. Up to 9999 Backup Lists can reside in a Backup. Specify ***LIB** as the list type and press **Enter**.



Step 10.

Based on the type of list being created, additional parameters may be prompted. Leave the default, ***FULL**, for the type of save to perform and press **Enter** to return to the Work with Backup List panel.

```
                        Add Backup List(ADDBKUPL)

 Type choices, press Enter.

 Backup . . . . . .  DEMO            Name
 Sequence number. . . . .  10        1-9999
 List . . . . . . . . . .  *LIB      *ASP, *ASPDLO, *CFG...
 Type . . . . . . . . . .  *FULL     *FULL, *INCR, *CUML
 Parallel device resources:
   Minimum resources . . .  *NONE    1-32, *NONE, *AVAIL
   Maximum resources . . .  _____    1-32, *AVAIL, *MIN
 Omit list name . . . . .  *NONE     Name, *NONE
 Error action . . . . . .  *IGNORE   *IGNORE, *CANCEL
 Text . . . . . . . . . .  'Lib Backup'  *Char, *BLANK

                                                                      Bottom
F3=Exit F5=Refresh F12=Cancel F13=How to use this display
F24=More keys
```

Step 11.

Select **Option 5** for the Backup List and press **Enter**. This displays the Work with Backup List Entries panel where the objects to save are added.

```
                          Work with Backup List

 Backup . . :DEMO                  Position to  ____

 Type options, press Enter.
   1=Add   2=Change   3=Copy  4=Remove   5=Work with  6=Release   7=Move
   8=Omit list        9=Hold

Opt     Seq     List    Type    Omit list    Hold    Date       Time
 _      ___
 5       10     *LIB    *FULL *NONE      *NO     *NONE   *NONE

                                                                      Bottom
 Selection or command
 ===> ____________________________________________________________
   ____________________________________________________________

F3=Exit F4=Prompt F5=Refresh F9=Retrieve F11=View 2  F12=Cancel F14=Omit list
(c) Copyright LXI Corp. 1985, 2006
```

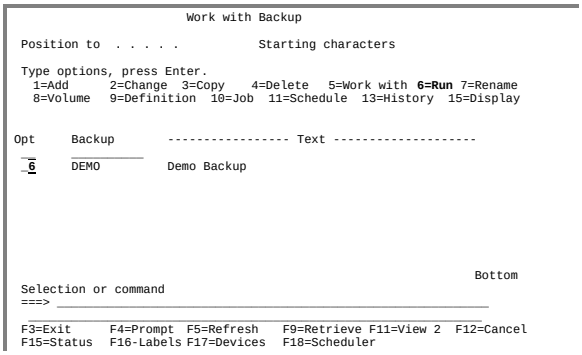
Step 12.

Using **Option 1**, enter the libraries to add to the Backup List. The **ORDER** determines the sequence in which the libraries are saved. When complete, press **Enter** until the Work with Backup panel displays. This completes the steps required to set up a Backup.

```
                       Work with Backup List Entries

 Backup . . :DEMO                  Type . . . :  *FULL
 Sequence . :10                    Omit list  :  *NONE
 List . . . :*LIB                  Position to  _________

 Type options, press Enter.
   1=Add  3=Copy  4=Remove    7=Move

 Opt    Order Library      Text
  1       10   APFILLIB

   (No records meet selection criteria.)

                                                                      Bottom
 Selection or command
 ===> ____________________________________________________________
   ____________________________________________________________

F3=Exit    F4=Prompt    F5=Refresh F9=Retrieve     F12=Cancel
F13=Select libraries
(c) Copyright LXI Corp. 1985, 2006
```

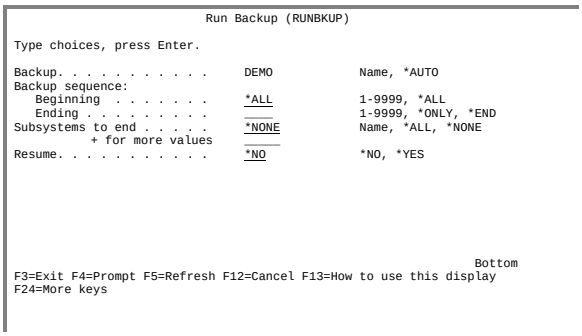
Step 13.

To run the Backup, select **Option 6** for the **DEMO** Backup from the Work with Backup panel. This prompts the Run Backup (RUNBKUP) command.



Step 14.

Pressing **Enter** prompts for additional values and then submits the Backup to batch. To run the Backup interactively, prompt the Run Backup (RUNBKUP) command from a command line, change the parameters as needed and press **Enter**.



See the following pages for detailed information. σσσ

Chapter 5

Backups

There is probably nothing worse than to trying to recover something only to find that there is no backup or that the information needed was never saved. It may take months to find out that an object was damaged or that a file is corrupt. Last nights backup seldom contains the resolution for a problem that occurred months ago and replication, mirroring or other high availability products only deal with current information. The need for backups increases as more and more information is stored. Unfortunately, it sometimes takes a disaster to appreciate a backup.

MMS/*bms* supports various types of iSeries backups. These backups can be created through menus or commands. Maintenance is simplified by eliminating the need to create or change user programs. Scheduling is accomplished through any job scheduler. Backups are initiated through user programs, submitted to batch, or executed from the OS/400 command line.

This chapter will discuss:

- Backup processing
- How to create a Backup
- How to create a Backup List

Backup Overview

Backup attributes are defined by a **Backup Definition**. The Backup Definition defines the backup devices, pre-check requirements, pre and post exit programs as well as other backup attributes. Every Backup requires a Backup Definition.

Backups consist of the following three elements:

- **Backup**, which defines the name of the backup.
- **Backup List**, which defines the type of objects to save as well as the type of save to perform.
- **Backup List entries**, which define the objects to save. If generic names are specified, the object list is determined at save time.

Since the importance of a library is not determined by the name, the user can order Backup List entries so that libraries and other objects are saved by importance to the user – not alphabetically.

Backup List Types

MMS/*bms* Backup Lists provide virtually limitless options in establishing a backup strategy. Since multiple Backup Lists can exist for a Backup, any combination of libraries, objects, links, folders, documents, output queues and spooled files can be saved in one comprehensive backup.

Backup List	Type of list
*ASP	Auxiliary storage pool list.
*ASPDLO	Document library objects in the specified ASP.
*CFG	Configuration and System Resource Management (SRM) objects.
*CLT	Client list.
*EJECT	Tape eject.
*EXIT	Command exit list.
*FLR	Document library object list.
*LIB	Library list.
*LND	Domino/Lotus Notes Server list
*LNK	Integrated file system list.
*OBJ	Object list.
*OUTQ	Output queue list.
*RCY	MMS/*tms* recovery library list.
*SAVF	Save file list.
*SEC	Security information.
*SPL	Spooled file list.
*SYS	Licensed internal code, the QSYS library, security and configuration objects.

Types of Backups

MMS/*bms* performs the following types of backups.

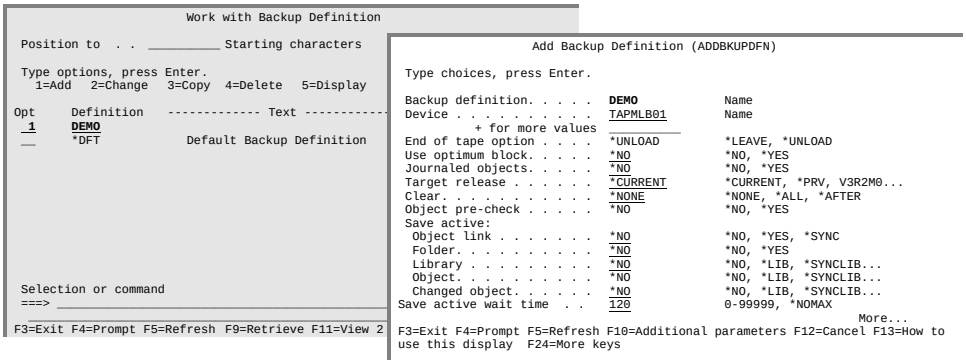
Type	Description
*FULL	A full backup is performed.
*CUML	Processes a cumulative backup. Cumulative backups save all changes since the last full backup. This value is only valid for *ASP, *ASPDLO, *FLR, *LIB, *LNK and *OBJ Backup Lists.
*INCR	Processes an incremental backup. Incremental backups save all changes since the last incremental backup. This value is only valid for *ASP, *ASPDLO, *CLT, *FLR, *LIB, *LNK and *OBJ Backup Lists.

Working with Backup Definitions

Backup definitions associate user-specified attributes to a Backup. These attributes define the processing requirements and include the save devices to use, pre-check options, save while active requirements and exit programs. To access the Work with Backup Definitions panel, select **Option 2** from the Backup menu.

Adding a Backup Definition

Using **Option 1** from the Work with Backup Definition panel, enter a Backup Definition name and press **Enter**. This prompts the Add Backup Definition (ADDBKUPDFN) command, which defines the backup attributes. Type the required values and review and optionally change the defaults. Press **Enter** when complete.



Changing a Backup Definition

To change the attributes of a Backup Definition, use **Option 2** from the Work with Backup Definition panel. This prompts the Change Backup Definition (CHGBKUPDFN) command.

Copying a Backup Definition

To copy the attributes of one Backup Definition to another, use **Option 3** from the Work with Backup Definition panel. This prompts the Copy Backup Definition (CPYBKUPDFN) command.

Deleting a Backup Definition

To delete a Backup Definition, use **Option 4** from the Work with Backup Definition panel or the Delete Backup Definition (DLTBKUPDFN) command. The default Backup Definition, ***DFT**, cannot be deleted.

Displaying a Backup Definition

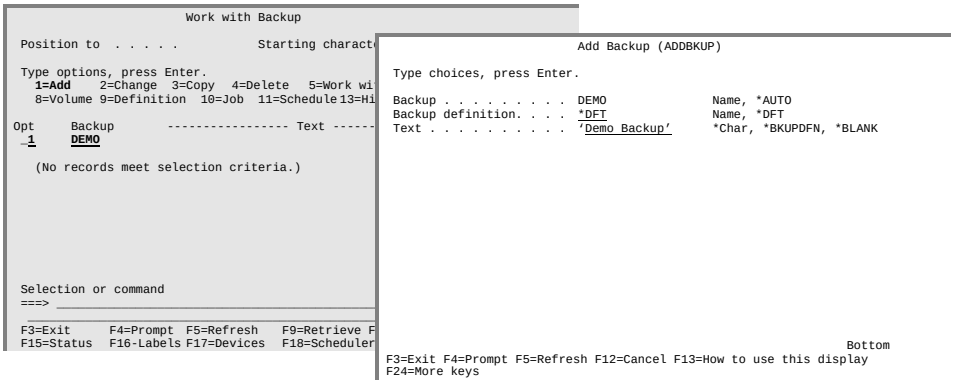
To display the Backup Definition associated with a Backup, use **Option 5** from the Work with Backup Definition panel.

Working with Backups

Backups consist of Backup Lists, which defined the attributes and Backup List entries, which define what is being backed up. Creating a Backup is the first step in defining the objects and type of save to perform. To access the Work with Backup panel, select **Option 1** from the Backup menu or use the Work with Backup (WRKBKUP) command.

Adding a Backup

Using **Option 1** from the Work with Backup panel, enter a Backup name and press **Enter**. This prompts the Add Backup (ADDBKUP) command, which associates a name and a Backup Definition to the Backup. Type the required values and press **Enter**.



Changing a Backup

To change the text and/or Backup Definition for a Backup, use **Option 2** from the Work with Backup panel. This prompts the Change Backup (CHGBKUP) command. Backups cannot be changed if they are running.

Copying a Backup

To copy the Backup Lists and Backup List entries from one Backup to another, use **Option 3** from the Work with Backup panel. This prompts the Copy Backup (CPYBKUP) command. Copying a Backup renumbers Backup Lists and Backup List entries.

Deleting a Backup

To delete a Backup and all associated Backup Lists and Backup List entries, use **Option 4** from the Work with Backup panel or the Delete Backup (DLTBKUP) command. Backups cannot be deleted if they are running.

Running a Backup

Using **Option 6** from the Work with Backup panel prompts the Run Backup (RUNBKUP) command. Pressing **Enter** will prompt for additional parameters before submitting the Backup to the QLXI subsystem. To execute the Backup interactively or call it from within a user program, call the **RUNBKUP** command. Running a Backup renumbers Backup Lists and Backup List entries. If the command is executed from the Backup menu, the Backup is submitted to subsystem **QLXI**. Subsystem **QLXI** must be active for the Backup to run.

Renaming a Backup

To rename a Backup and associated Backup Lists and Backup List entries, use **Option 7** from the Work with Backup panel. This prompts the Rename Backup (RNMBKUP) command. Renaming a Backup renumbers Backup Lists and Backup List entries. Backups cannot be renamed if they are running.

Working with Backups by Volume

To view the volumes used by a Backup and optionally change the volume attributes, use **Option 8** from the Work with Backup panel. This displays the Work with Volumes panel. This option requires that MMS Tape Management module (MMS/*tms*) be installed.

Working with the Backup Definition

To view and optionally change the Backup Definition associated with a Backup, use **Option 9** from the Work with Backup panel or the Work with Backup Definition (WRKBKUPDFN) command. This displays the Work with Backup Definition panel.

Working with the Backup Job

To view the Backup job, use **Option 10** from the Work with Backup panel. This displays the IBM Display Job panel. This option can be used to view the status of the Backup as well as the job log and other backup output.

Working with the Job Scheduler

To schedule a Backup using the LXI Job Scheduler (**LXI*sch***), use **Option 11** from the Work with Backup panel. This prompts the LXI Add Job Schedule Entry (**ADDJOBSCHE**) command. If the LXI Job Scheduler is not installed, the IBM Add Job Schedule Entry (**ADDJOBSCDE**) command is prompted.

Working with the Backup History

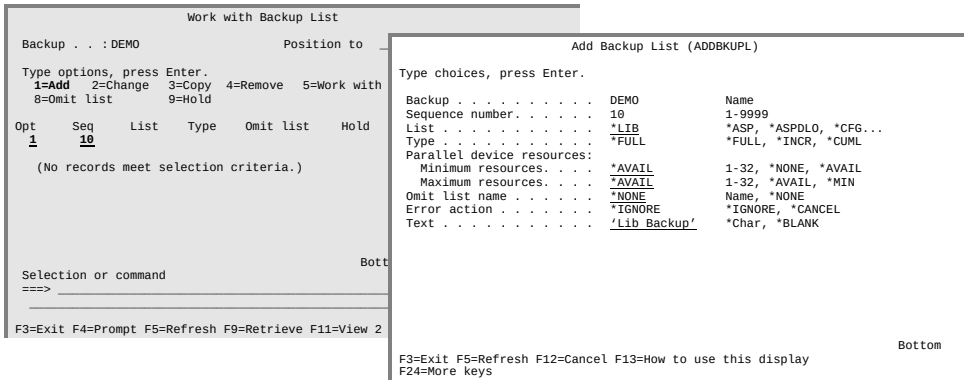
To view the history of a Backup job, use **Option 13** from the Work with Backup panel. This prompts the LXI Scan History Log (**SCNLOG**) command. The history displays all the objects saved as well as other pertinent information about the Backup job.

Working with Backup Lists

Backup Lists associate a backup type with a Backup. Backup types include the type of save to perform, such as a **SAVLIB** or **SAVCFG**. Since multiple Backup Lists can be associated with a Backup, the Backup List also determines the sequence and type of save, such as full, incremental or cumulative. To access the Work with Backup List panel, select **Option 5** for a specific Backup from the Work with Backup panel.

Adding a Backup List

Using **Option 1** from the Work with Backup List panel, enter the sequence number of the Backup List and press **Enter.** This prompts the Add Backup List (ADDBKUPL) command, which associates a backup type to the Backup. Type the required values and press **Enter**. Backup Lists cannot be added if the Backup is running.



Changing a Backup List

To change the backup type, Omit List name or text, use **Option 2** from the Work with Backup List panel. This prompts the Change Backup (CHGBKUPL) command. Backup Lists cannot be changed if the Backup is running.

Copying a Backup List

To copy a Backup List from one Backup to another, use **Option 3** from the Work with Backup List panel. This prompts the Copy Backup List (CPYBKUPL) command. Copying a Backup List renumbers the Backup List and its entries.

Removing a Backup List

To remove a Backup List and associated Backup List entries, use **Option 4** from the Work with Backup List panel or the Remove Backup List (RMVBKUPL) command. Backup Lists cannot be removed if the Backup is running.

Releasing a Backup List

Using **Option 6** from the Work with Backup List panel or the Release Backup List (RLSBKUPL) command releases a held Backup List. Held Backup Lists are not processed. Backup Lists cannot be released if the Backup is running.

Moving a Backup List

To move a Backup List from one Backup to another, use **Option 7** from the Work with Backup List panel. This prompts the Move Backup List (MOVBKUPL) command. Backup Lists cannot be moved if the Backup is running.

Omitting Entries from a Backup List

To work with the entries on the Omit List associated with the Backup List, use **Option 8** from the Work with Backup List panel or the Work with Omit List (WRKOMITL) command.

Holding a Backup List

Using **Option 9** from the Work with Backup List panel or the Hold Backup List (HLDBKUPL) command holds a Backup List. Held Backup Lists are not processed. Backup Lists cannot be held if the Backup is running.

Note: If Omit lists are specified, they must exist. Use **F14** to view, and optionally add, Omit Lists.

Note: If a Backup List receives an escape message, the Error action parameter (**ERROR**) determines if the backup continues (***IGNORE**) or stops (***CANCEL**).

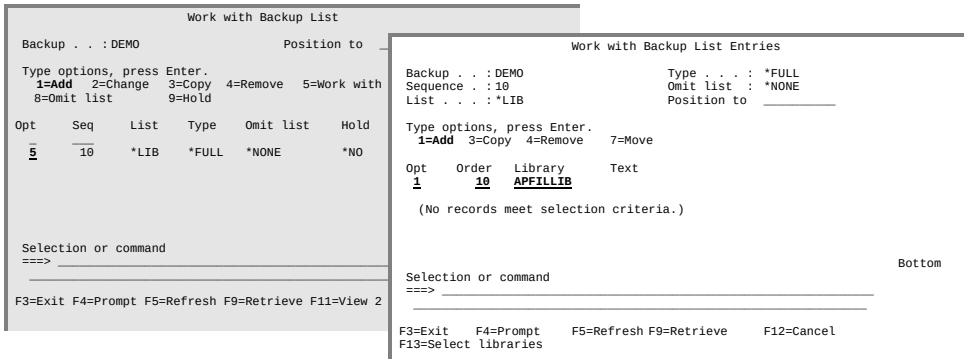
Tip: Increment the sequence numbers by five (5). This ensures that future Backup Lists can be inserted between existing Backup Lists.

Working with Backup List Entries

Backup List entries associate the objects to back up with a Backup List. The Backup List type determines the type of panel displayed. To access the Work with Backup List Entries panel, select **Option 5** for a specific Backup List from the Work with Backup List panel.

Adding a Backup List Entry

Using **Option 1** from the Work with Backup List Entries panel, enter the required fields for the Backup List entry and press **Enter.** This adds the entry to the Backup List. If additional backup attributes for the specified object are allowed, the Add Backup List Entry (ADDBKUPLE) command will be prompted. Type the required values and press **Enter**. Backup List entries cannot be added if the Backup is running. This option is not available for all Backup List types.



Copying a Backup List Entry

To copy a Backup List entry from one Backup List to another, use **Option 3** from the Work with Backup List Entry panel. This prompts the Copy Backup List Entry (CPYBKUPLE) command.

Removing a Backup List Entry

Using **Option 4** from the Work with Backup List Entry panel or the Remove Backup List Entries (RMBKUPLE) command removes the selected Backup List entry. Backup List entries cannot be removed if the Backup is running.

Moving a Backup List Entry

To move a Backup List entry from one Backup List to another, use **Option 7** from the Work with Backup List Entry panel. This prompts the Move Backup List Entry (MOVBKUPLE) command. Backup List entries cannot be moved if the Backup is running.

Adding Auxiliary Storage Pools (*ASP)

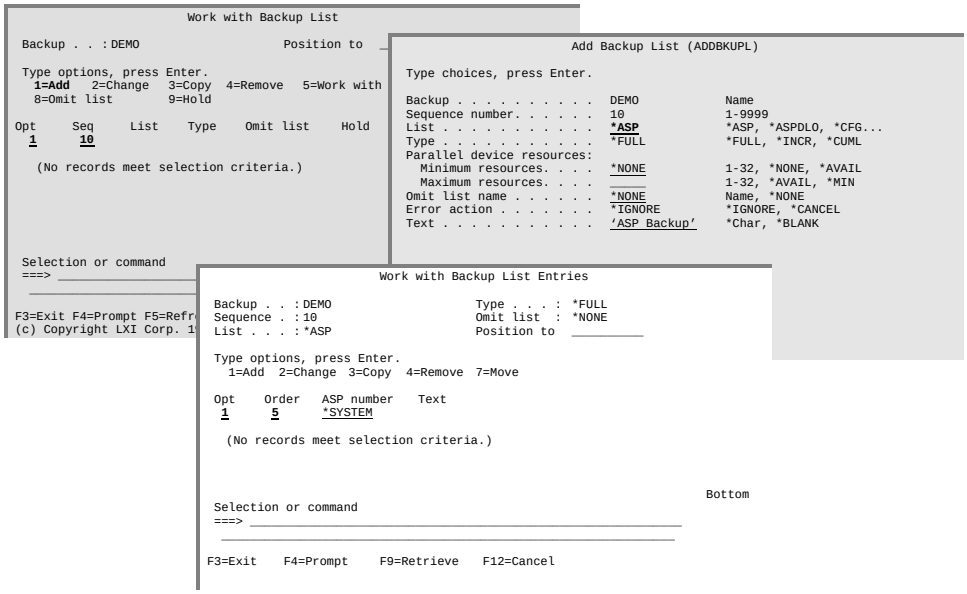
Auxiliary storage pool lists backup one or more auxiliary storage pools. Two types of auxiliary storage pool lists are supported – ***ASP** and ***ASPDLO**. The ***ASP** Backup List supports all libraries that reside on a specific auxiliary storage pool. The ***ASPDLO** Backup List supports Document Library Objects (DLO) that resides on a specific auxiliary storage pool.

To add an Auxiliary storage pool list to a Backup, create a Backup List with a list type of ***ASP** or ***ASPDLO,** if document library object storage pools are required. Once created, use **Option 5** from the Work with Backup List panel to add the auxiliary storage pools to the Backup List.

Backup List entries: *YES

Omit Lists: *YES

Parallel save: *ASP Only

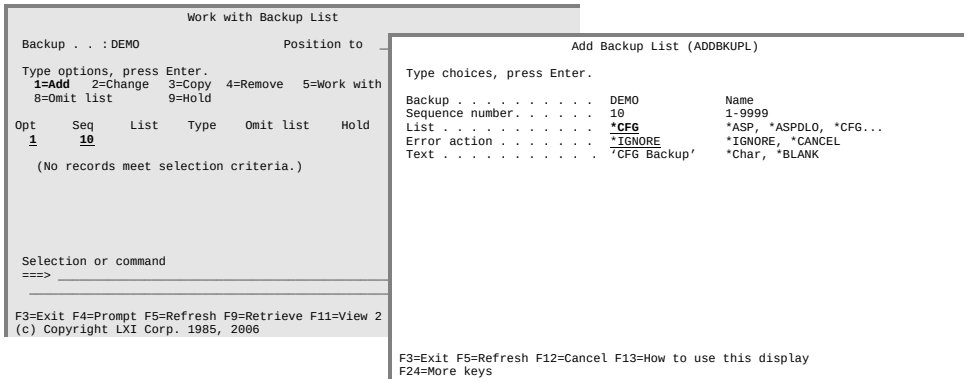


Adding Configuration Objects (*CFG)

Configuration lists backup configuration and System Resource Management (SRM) objects.

To add a Configuration list to a Backup, create a Backup List with a list type of ***CFG**.

Backup List entries: *NO
Omit Lists: *NO
Parallel save: *NO



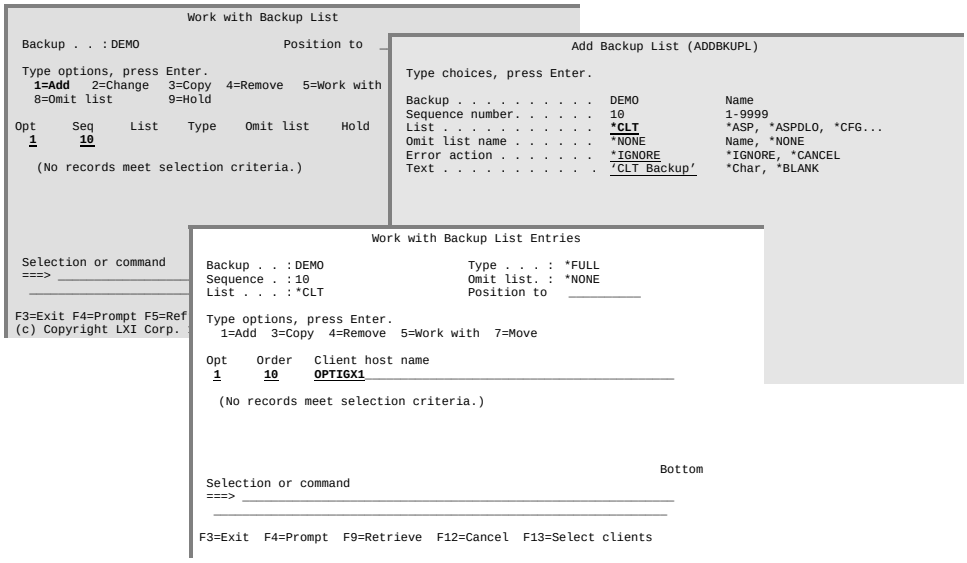
Selecting Clients (*CLT)

Client lists backup all or specific client hosts.

To add a Client list to a Backup, create a Backup List with a list type of ***CLT**. Once created, use **Option 5** from the Work with Backup List panel to add the clients to the Backup List.

Use **F13** to select from a list of available clients.

Backup List entries: *YES
Omit Lists: *YES
Parallel save: *NO



Note: This feature requires MMS Tape Management (LXI*tms*). Refer to the MMS Tape Management documentation for information on installing and using Client backup support.

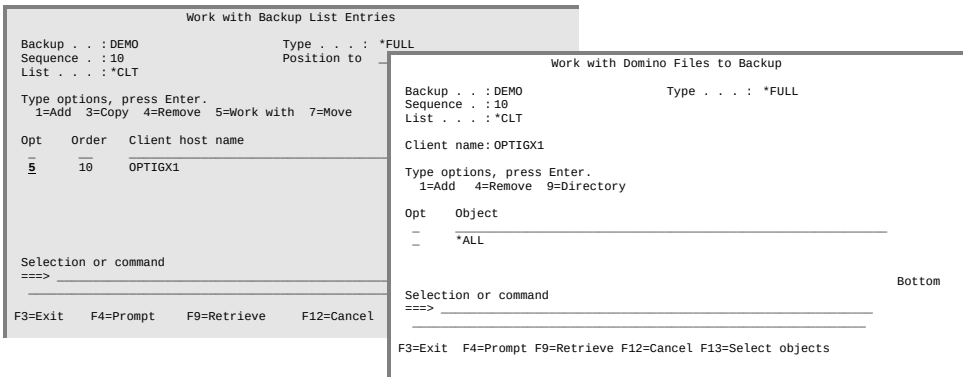
Adding Client Objects

Client objects can be added to ***CLT** lists. All or specific objects can be specified.

To add one or more client objects to an ***CLT** Backup List, use **Option 5** from the Work with Backup List Entries panel to add the objects to the Backup List entry.

Use **F13** to select from a list of available client objects.

Generic names: *YES



Adding Ejects (*EJECT)

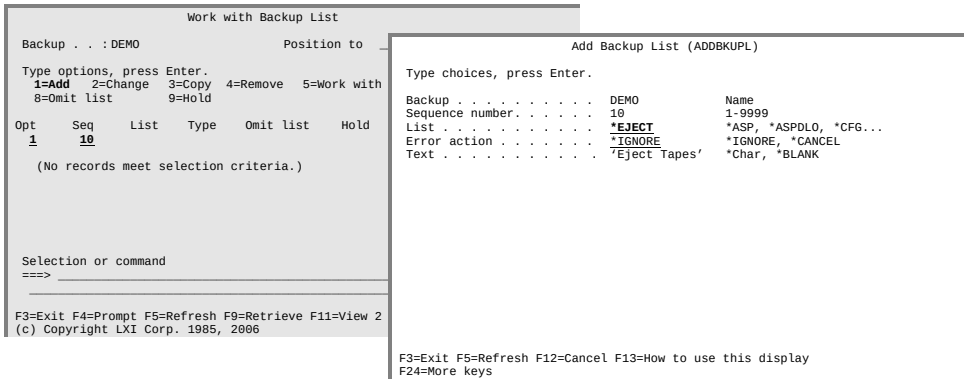
Eject lists remove the tapes associated with a backup from a tape library. When this backup list executes, all tapes associated with the Job Label currently in effect for the backup, are ejected.

To add an Eject list to a Backup, create a Backup List with a list type of ***EJECT**.

Backup List entries: *NO

Omit Lists: *NO

Parallel save: *NO

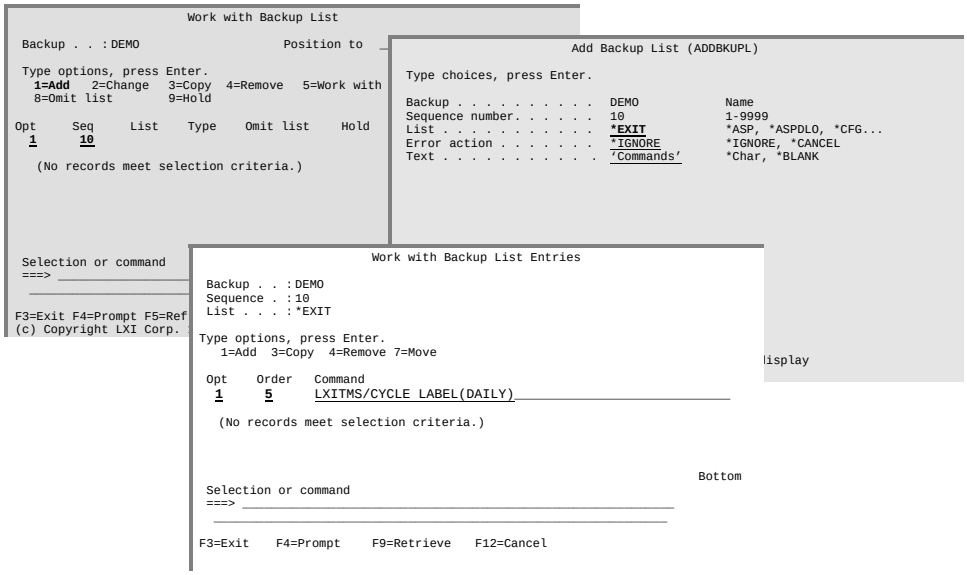


Adding Exits

Exit lists execute a command or program from within a Backup List. Multiple Exit lists can exist in a Backup List.

To add an Exit list to a Backup, create a Backup List with a list type of ***EXIT**. Once created, use **Option 5** from the Work with Backup List panel to add the commands to execute when the Backup List runs. Use **F4** to prompt the command, if needed.

To minimize potential library list problems, it is recommended that commands that do not reside in **QSYS** are qualified when entering them on the ***EXIT** Backup List entry. This ensures that the command will execute when the Backup runs.



> **Note**: Commands are validated before being added. Any command that does not exist in the jobs' library list is automatically removed after using Option 5 from the Work with Backup List panel.

> **Note**: The MMS Tape Management Cycle command should only be used as the first backup list. Using it again after the saves starts will cause another tape to be mounted without unloading the first tape.

Adding Folders (*FLR)

Folder lists backup all or specific Document Library Objects (DLO).

To add a Folder list to a Backup, create a Backup List with a list type of ***FLR**. Once created, use **Option 5** from the Work with Backup List panel to add the document library objects to the Backup List.

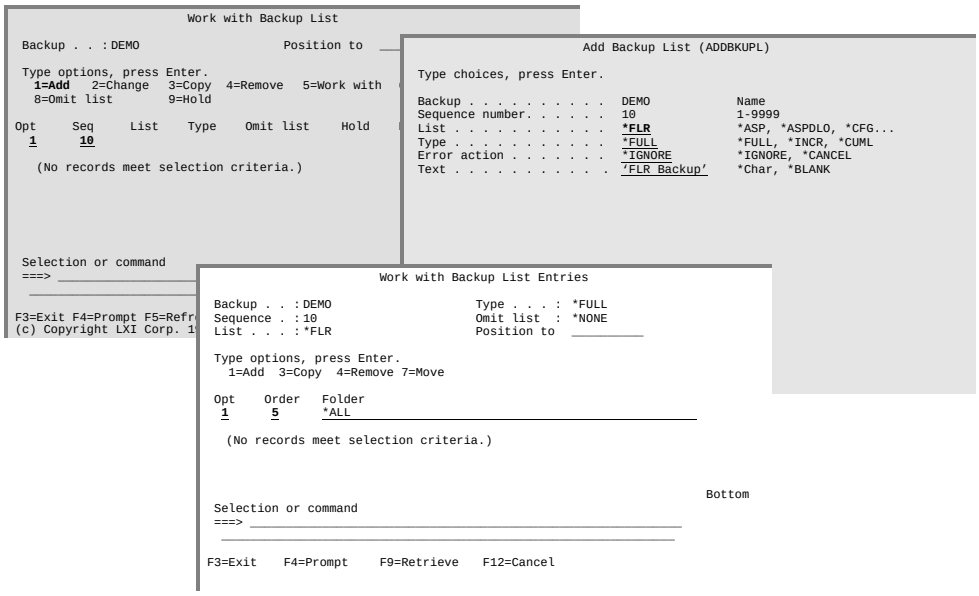
Use **F13** to select from a list of available document library objects.

Backup List entries: *YES

Generic names: *YES

Omit Lists: *YES

Parallel save: *NO



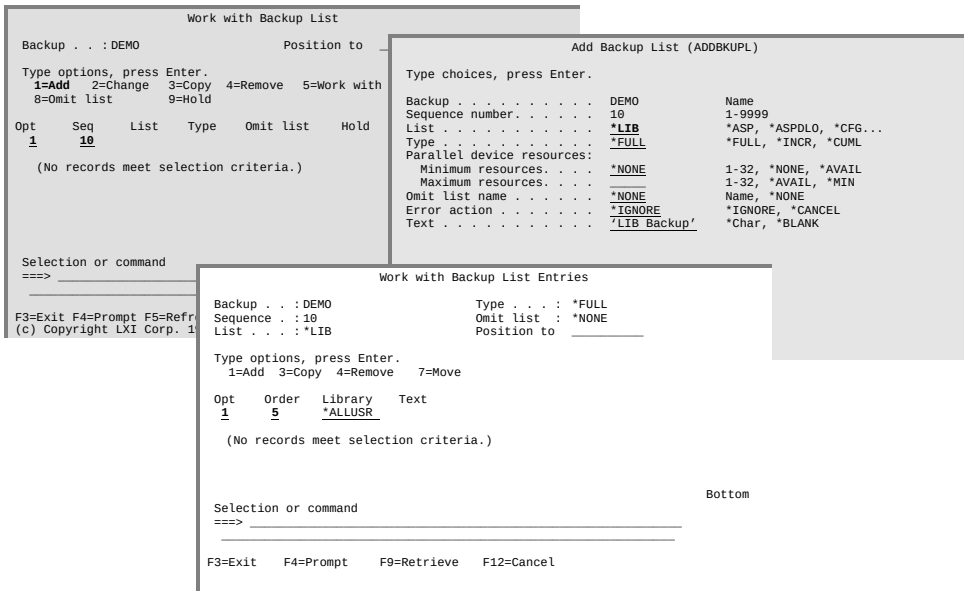
Adding Libraries (*LIB)

Library lists backup all or specific libraries.

To add a Library list to a Backup, create a Backup List with a list type of ***LIB**. Once created, use **Option 5** from the Work with Backup List panel to add the libraries to the Backup List.

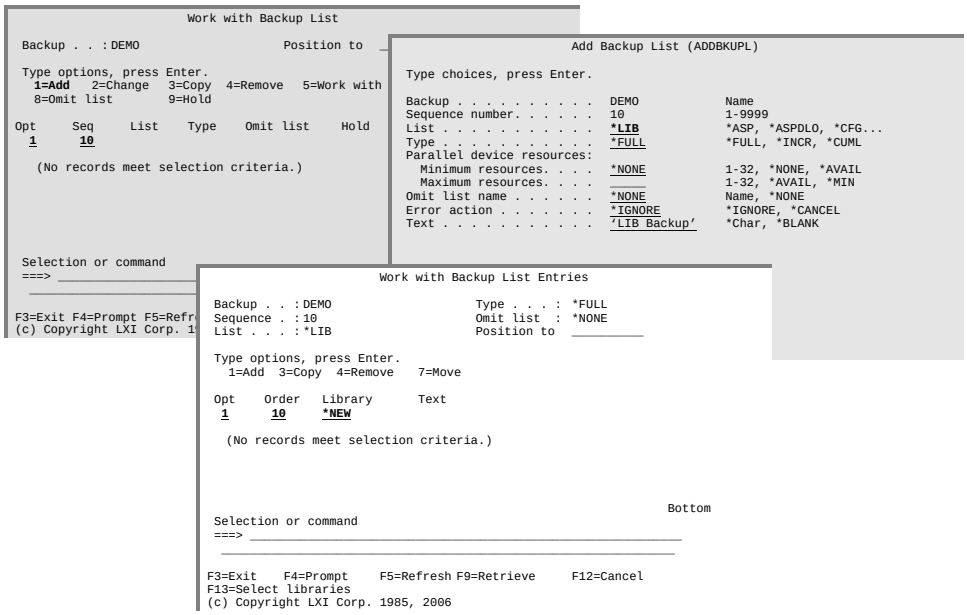
Use **F13** to select from a list of available libraries.

Backup List entries: *YES
Generic names: *YES
Omit Lists: *YES
Parallel save: *YES



Saving New Libraries

To save new libraries to a ***LIB** Backup List quickly, enter the special value *__NEW__ as the library name on the Work with Backup List Entries panel. This saves all libraries not previously saved when the Backup executes.



Selecting Servers (*LND)

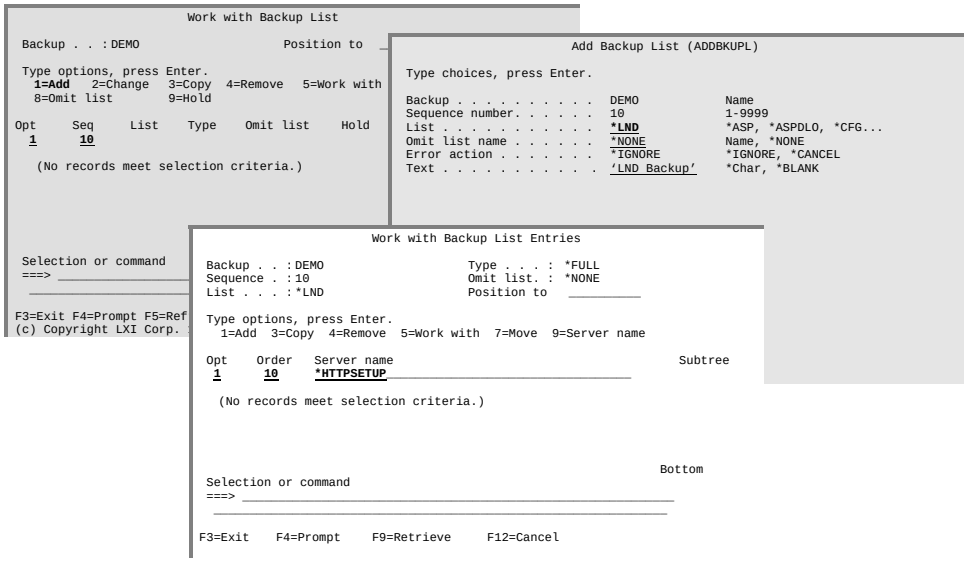
Domino server lists backup all or specific Domino servers.

To add a Domino server list to a Backup, create a Backup List with a list type of ***LND**. Once created, use **Option 5** from the Work with Backup List panel to add the servers to the Backup List.

Backup List entries: *YES

Omit Lists: *YES

Parallel save: *NO



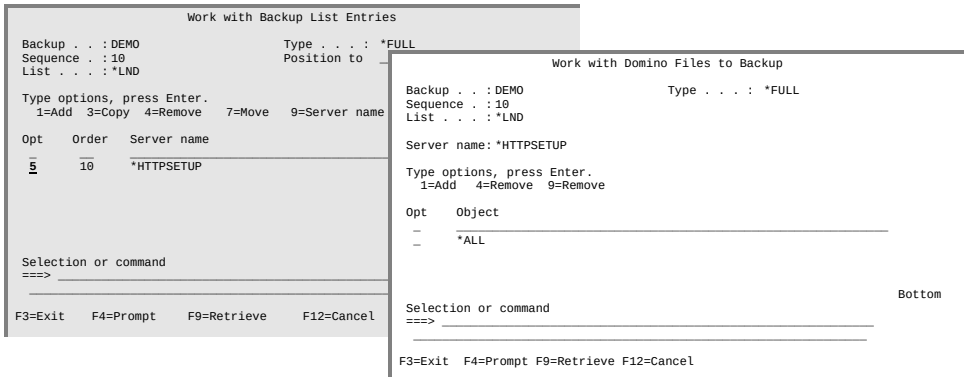
Note: This feature requires MMS Tape Management (LXI*tms*). Refer to the MMS Tape Management documentation for information on installing and using Lotus server backup support.

Adding Lotus Notes Objects

Lotus Notes objects can be added to ***LND** lists. All or specific objects can be specified.

To add one or more Lotus Notes objects to an ***LND** Backup List, use **Option 5** from the Work with Backup List Entries panel to add the objects to the Backup List entry.

Generic names: *YES



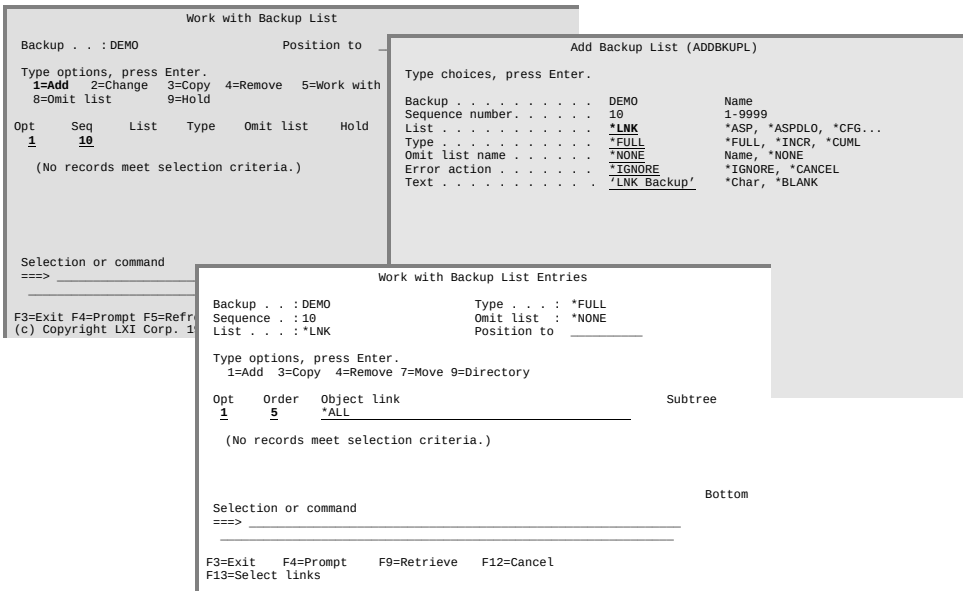
Adding Integrated File System Objects (*LNK)

Link lists backup all or specific integrated file system objects.

To add a Link list to a Backup, create a Backup List with a list type of ***LNK**. Once created, use **Option 5** from the Work with Backup List panel to add the integrated file system objects to the Backup List.

Use **F13** to select from a list of integrated file system objects.

Backup List entries: *YES
Generic names: *YES
Omit Lists: *YES
Parallel save: *NO



Adding Objects (*OBJ)

Object lists backup all or specific objects. Members can be specified for physical or logical files.

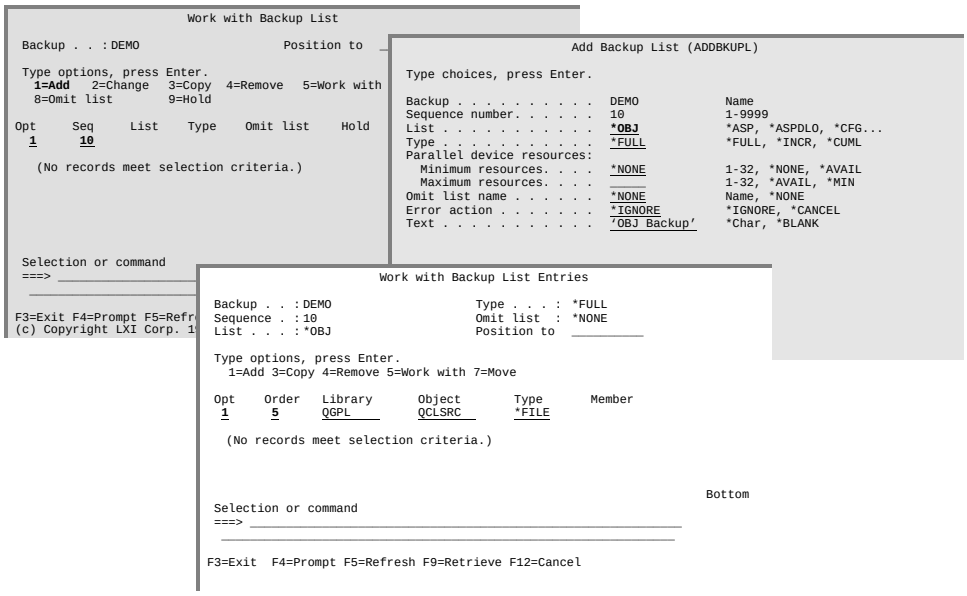
To add Object list to a Backup, create a Backup List with a list type of ***OBJ**. Once created, use **Option 5** from the Work with Backup List panel to add the objects to the Backup List.

Backup List entries: *YES

Generic names: *YES

Omit Lists: *YES

Parallel save: *YES



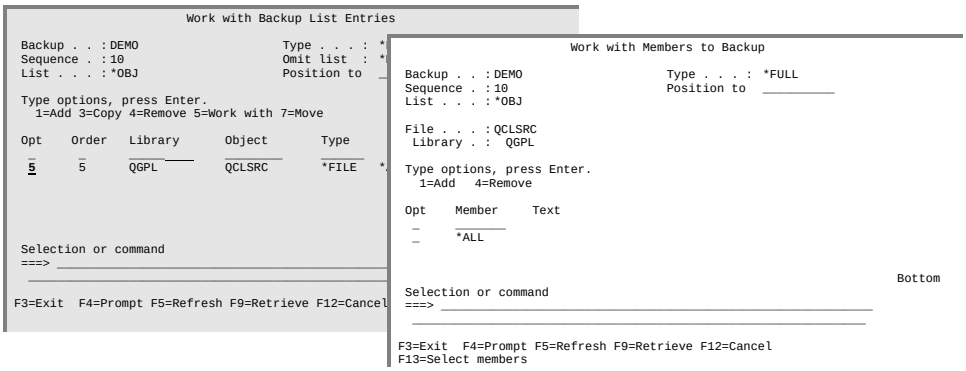
Adding Members

Members can be added to physical or logical files in ***OBJ** lists. All or specific file members can be specified.

To add one or more members to an ***OBJ** Backup List, use **Option 5** from the Work with Backup List Entries panel to add the members to the Backup List entry.

Use **F13** to select from a list of available members.

Generic names: *YES



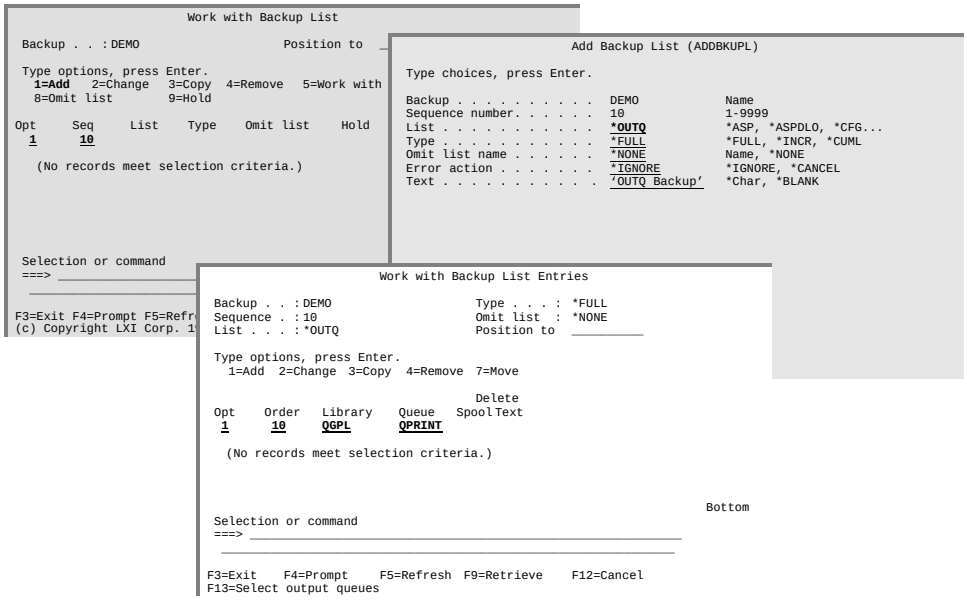
Adding Output Queues (*OUTQ)

Output queue lists backup all spooled files in all or specific output queues. Saved spooled files can optionally be deleted after the output queues is saved.

To add an Output queue list to a Backup, create a Backup List with a list type of ***OUTQ**. Once created, use **Option 5** from the Work with Backup List panel to add the output queues to the Backup List.

Use **F13** to select from a list of available output queues.

Backup List entries: *YES
Omit Lists: *YES
Parallel save: *NO



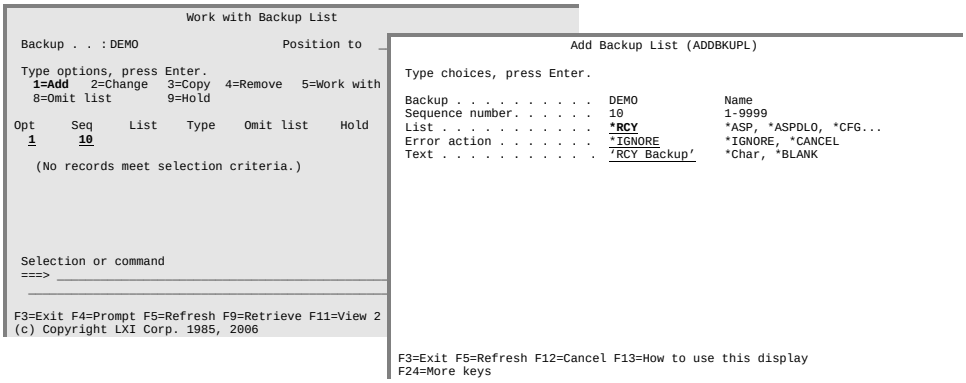
Note: This feature requires LXI Spool Management System (MMS/*spl*). Refer to the LXI Spool Management documentation for information on installing and using this product.

Adding MMS Recovery (*RCY)

Recovery Lists backup the MMS libraries needed for recovery.

To add a Recovery List to a Backup, create a Backup List with a list type of ***RCY**.

Backup List entries: *NO
Omit Lists: *NO
Parallel save: *NO



Note: This Backup List performs a full save on the following libraries:

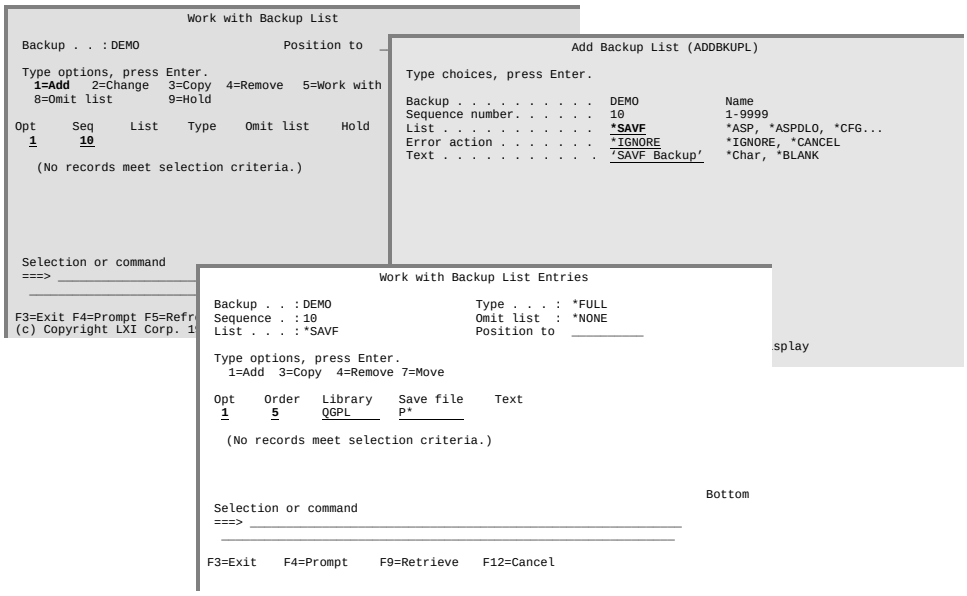
- LXI
- LXIBMS
- LXIBMS400
- LXITMS
- LXITMS400

Adding Save Files (*SAVF)

Save file lists backup one or more save files. If an empty save file is encountered during processing, it is bypassed.

To add a Save file list to a Backup, create a Backup List with a list type of ***SAVF**. Once created, use **Option 5** from the Work with Backup List panel to add the save files to the Backup List.

Backup List entries: *YES
Generic names: *YES
Omit Lists: *NO
Parallel save: *NO

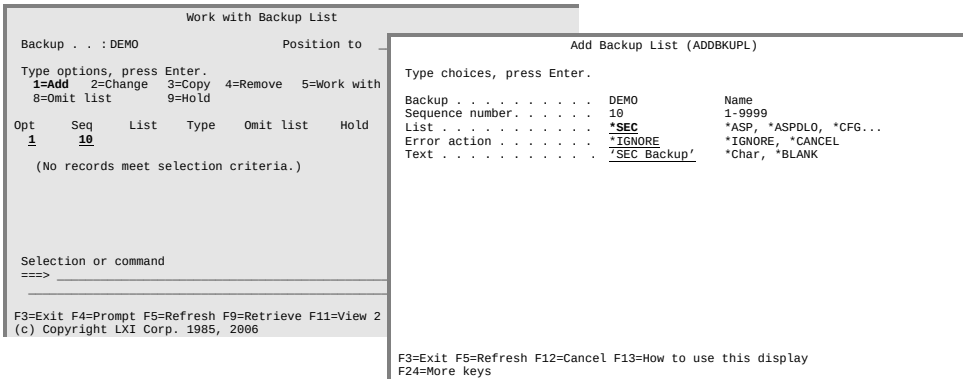


Adding Security Data (*SEC)

Security lists are used to backup system security data.

To add a Security list to a Backup, create a Backup List with a list type of ***SEC**.

Backup List entries: *NO
Omit Lists: *NO
Parallel save: *NO



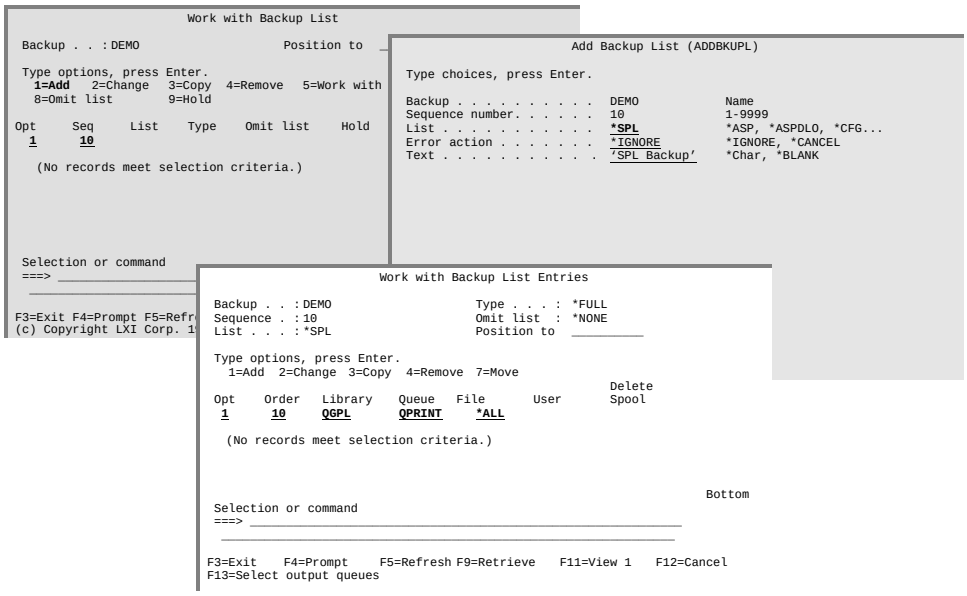
Adding Spooled Files (*SPL)

Spooled file lists are used to backup all or specific spooled files in one or more output queues. Saved spooled files can optionally be deleted after the spooled file is saved. Selection criteria includes user profile, user data and form type.

To add a spooled file list to a Backup, create a Backup List with a list type of ***SPL**. Once created, use **Option 5** from the Work with Backup List panel to add the spooled files to the Backup List.

Use **F13** to select from a list of available output queues.

Backup List entries: *YES
Generic names: *YES
Omit Lists: *NO
Parallel save: *NO



Note: This feature requires LXI Spool Management System (MMS/*spl*). Refer to the LXI Spool Management documentation for information on installing and using this product.

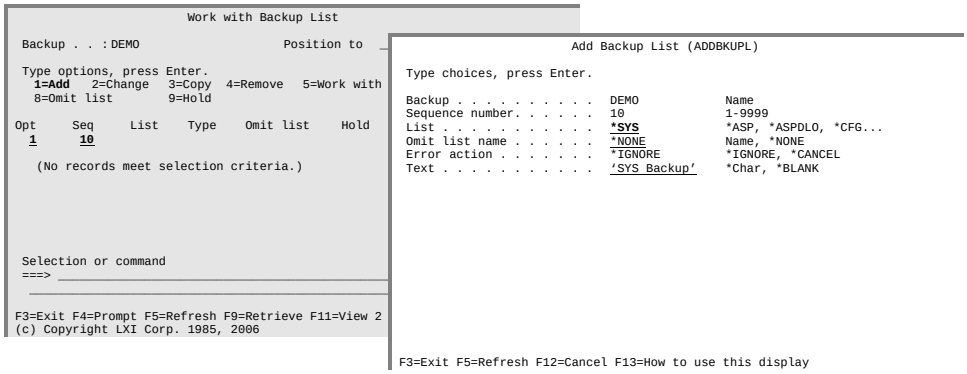
Adding System (*SYS)

System lists are used to backup licensed internal code, the QSYS library, security and configuration objects.

To add a System list to a Backup, create a Backup List with a list type of ***SYS**.

A System list must be the <u>first</u> Backup List in a Backup.

Backup List entries: *NO
Omit Lists: *YES
Parallel save: *NO



Note: Exit lists are the only list types that can precede a ***SYS** list.

Parallel Saves (Media Definitions)

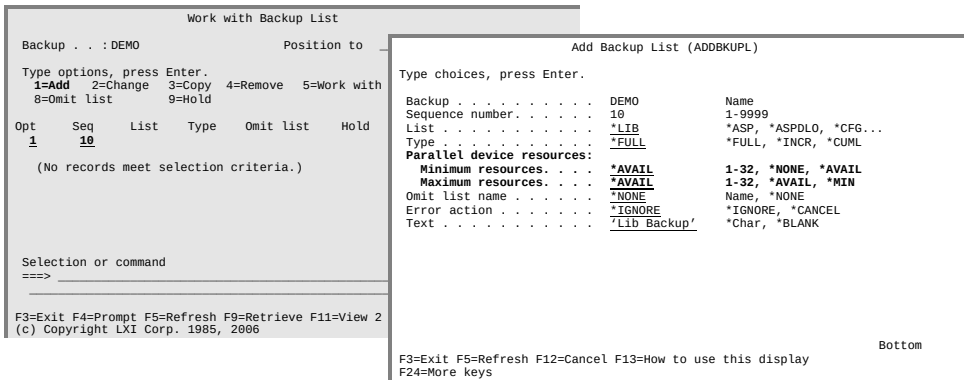
Parallel processing helps reduce the time needed to perform a backup by maximizing hardware resources. Parallel saves are available for ***ASP**, ***LIB**, ***OBJ** and ***RCY** Backup Lists.

Defining a Parallel Save

To save a library or object in parallel format, create a Backup List specifying ***ASP**, ***LIB**, ***OBJ** or ***RCY**. Pressing **Enter** prompts the parallel save parameters.

Specify the minimum and maximum number of resources to use for the Backup List. To maximize the use of hardware resources when using media definitions, type ***AVAIL** for both parameters.

When a parallel Backup List completes execution, the tapes used are automatically unloaded before the next Backup List starts.



Parallel Saves and StorageTek

Up to 32 drives can be specified in the Backup Definition for the use of saves/restores performed with Media Definitions. Unless there is a requirement for specific drives, specify one (1) drive in the Backup Definition. When the Backup executes, MMS/*bms* will allocate additional drives based on the values specified in the Drive Resource (**DRVRSC**) parameter of the Backup List. In order for this feature to work, the tape devices must be defined in MMS Tape Management as **SHARE(*YES)**. If the number of drives available is less than the minimum number of drives specified, an inquiry message will be sent to the user providing the ability to cancel the backup, continue the backup or retry the device allocation This feature is only allowed with Media Definitions and StorageTek Tape Libraries.

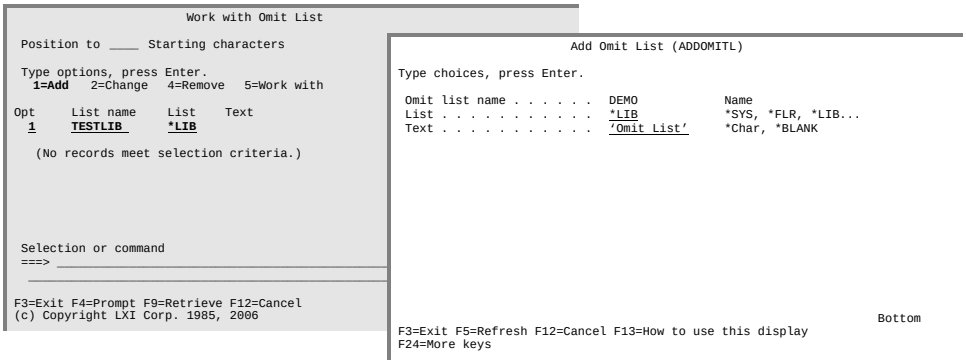
Note: Some restrictions apply when trying to restore parallel saves. Read all IBM documentation regarding parallel save/restore restrictions before using Media Definitions.

Working with Omit Lists

Some list types support Omit List capabilities. Omit Lists contain items to be omitted from a Backup or Recovery List. When an Omit List is associated with a Backup or Recovery List, MMS/*bms* verifies that the Omit List is compatible with the Backup or Recovery List. An error message is issued if the Omit List is not compatible. The Work with Omit List panel can be accessed from several backup or recovery panels or through the Work with Omit List (WRKOMITL) command.

Adding an Omit List

To add an Omit List, use **Option 1** from the Work with Omit List panel or use the Add Omit List (ADDOMITL) command.



Changing an Omit List

To change an Omit List, use **Option 2** from the Work with Omit List panel or use the Change Omit List (CHGOMITL) command.

Deleting an Omit List

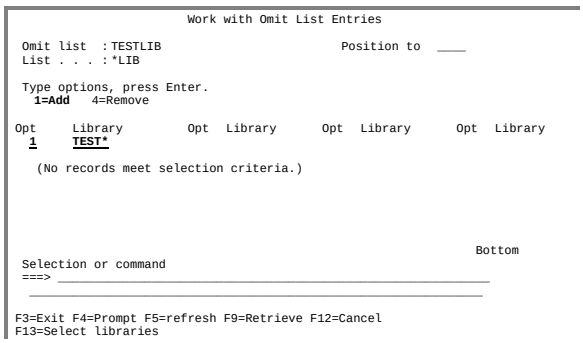
To delete an Omit List and the associated Omit List entries, use **Option 4** from the Work with Omit List panel or use the Delete Omit List (DLTOMITL) command.

Working with Omit List Entries

Omit List entries associate the objects to omit from a Backup or Recovery List. The Omit List type determines the type of panel displayed. To access the Work with Omit List Entries panel, select **Option 5** for a specific Omit List from the Work with Omit List panel or use the Work with Omit List Entries (WRKOMITLE) command.

Adding an Omit List Entry

To create an Omit List entry, use **Option 1** from the Work with Omit List Entries panel or use the Add Omit List Entry (ADDOMITLE) command.



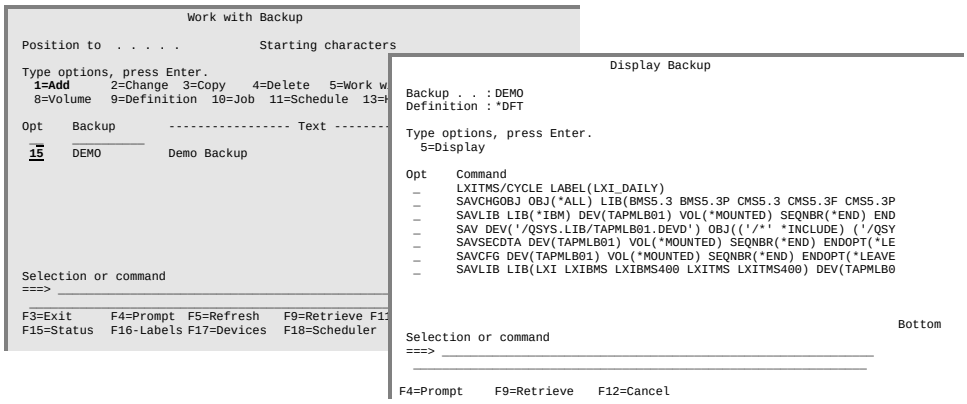
Removing an Omit List Entry

To remove an Omit List entry, use **Option 4** from the Work with Omit List Entries panel or use the Remove Omit List Entry (RMVOMITLE) command.

Displaying a Backup

Displaying a backup shows the actual commands that will be issued when the backup runs. If generic library names exist in the backup list, they will be replaced with the actual library names. If the ***NEW** value exists in a backup list and libraries that have never been saved are located, they will be displayed in the save command. If "pre" and "post" exit programs have been specified in the Backup Definition, they will also be listed in the display. The Display Backup panel provides an exact view of what will occur when the backup runs. To display a backup, select **Option 15** from the Work with Backup panel or use the (DSPBKUP) command.

If the command exceeds the length of the display, selecting **Option 5** to display a window that shows up to 15,000 characters of the command.



Exit Programs

An exit program is a user-defined program that performs processing before a Backup starts and after a Backup completes. The program could be used to start and end subsystems, hold or release job queues or submit other jobs. To maintain compatibility with older versions of MMS/*bms* exit programs, the same program can be used for pre and post processing. Exit programs are defined in the Backup Definition.

Using the Same Program

If the same program is called for pre and post exit processing, the first parameter passed to the exit program determines when it is being called. Additionally, the tape management Job Label, error status and last device used are passed. The following chart outlines the parameters. Refer to Chapter 14 for an example of an exit program.

Parameter	Length	Description	Notes
1	1	"0" = Before backup "1" = After backup	--------
2	3	Not used	--------
3	20	MMS tape management Job Label	Passed on post-exit only
4	1	"0" = No errors occurred "2" = Escape messages occurred	Passed on post-exit only
5	10	Last device used	Passed on post-exit only

Chapter 6

Using the Intelligent Backup

The MMS/*bms* Intelligent Backup automatically creates backups of all new/changed objects, document library objects and integrated file system objects that have not been saved or for which a current backup tape cannot be found. In addition, this feature also includes any object for which the current save is to a save file. This ensures that **tapes** contain the objects needed for recovery.

The Intelligent Backup scans all objects on the system and, based on user-defined attributes, builds Backup Lists that perform either a full or a cumulative save. If a user-specified threshold for changed objects has been satisfied, a full save is performed, thereby reducing the time required to continually backup all changes. The number of libraries, document library objects and integrated file system objects saved is based on the number of additions and changes and the change threshold.

To maximize hardware resources and the ever-shrinking backup window, media definitions can be used to significantly reduce the time needed to perform the backup.

The Intelligent Backup (***AUTO**) feature performs the following types of saves, if needed.

- Full save of all user libraries (*ALLUSR)
- Full save of all IBM libraries (*IBM)
- Full save of all document library objects (SAVDLO)
- Full save of all integrated files system objects (SAV)
- Saves of all changed objects in user libraries (SAVCHGOBJ)
- Saves of all changed document library objects (SAVDLO)
- Saves of all changed integrated files system objects (SAV)
- Saves of all new libraries (SAVLIB)
- Save of configuration data (SAVCFG)
- Saves of security data (SAVSECDTA)
- Any of the above for which the current backup tape cannot be found
- Any of the above for which the current save is to a save file

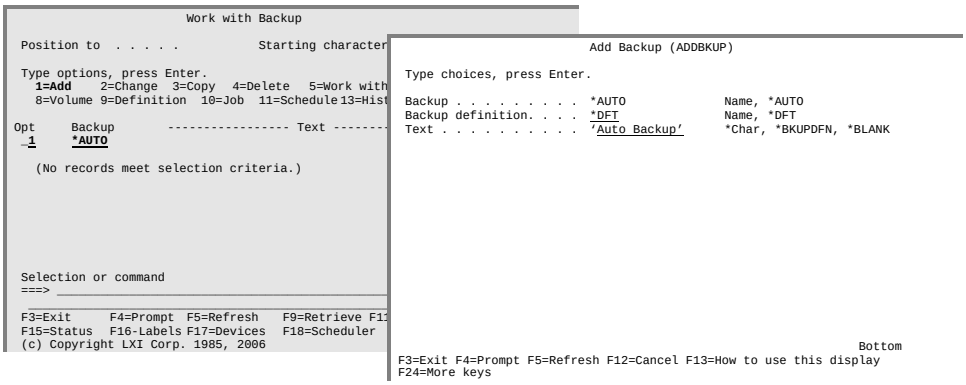
The Intelligent Backup is easy to implement and use. It saves time and eliminates incomplete backups by maintaining the Backups and Backup Lists.

Using the Auto Backup

The Intelligent Backup feature (***AUTO**) is easy to set up and use. Once created, it should be added to a job scheduler to run daily. No maintenance is required. Creating the ***AUTO** Backup is the only step required. To access the Work with Backup panel, select **Option 1** from the Backup menu or use the Work with Backup (WRKBKUP) command.

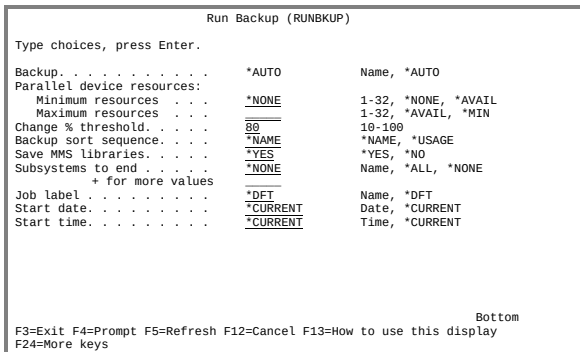
Adding the Auto Backup

Using **Option 1** from the Work with Backup panel, enter "***AUTO**" for the Backup name and press **Enter**. This prompts the Add Backup (ADDBKUP) command, which associates a Backup Definition to the Backup. Type the required values and press **Enter**.



Running the Auto Backup

To run the ***AUTO** Backup, prompt the Run Backup (RUNBKUP) command and change parameters as needed. Press **Enter** when complete. If the command is executed from the Backup menu, the Backup is submitted to subsystem **QLXI**. Subsystem **QLXI** must be active for the Backup to run.



- The ***AUTO** Backup does **not** perform the following types of backups:
  - Save system (SAVSYS)
  - Save of all non-system libraries (*NONSYS)
  - Spooled files or Output Queues
- Instead of performing a SAVLIB LIB(*NONSYS), the ***AUTO** Backup performs a SAVLIB LIB(*ALLUSR) and a SAVLIB LIB(*IBM). These two saves equal a SAVLIB LIB(*NONSYS).
- The purpose of the Intelligent Backup feature is to automatically build and maintain a backup of all objects that have been created, changed, saved to disk or overwritten. No Backup Lists, Backup List Entries or Omit Lists are allowed.
- The ***AUTO** performs a combination of full and cumulative backups. To minimize the number of tapes required for recovery, it is recommended that backups be appended to the same tape. Refer to the MMS/*tms* documentation for details on specifying ***LAST** for the Job Label being used.
- The sequence of saves performed by the ***AUTO** Backup are as follows:
  - User libraries/objects
  - IBM libraries (if needed)
  - Document library objects (if needed)
  - Integrated files system objects (if needed)
  - Configuration data (always)
  - Security data (always)
  - MMS libraries (if specified)

Chapter 7

Recovering a Backup

With backup windows getting smaller, recovery for failed backups becomes more important. It is no longer acceptable to restart a backup that uses 50 tapes and has ended abnormally while writing to tape number 49. When this situation occurs, it is common for the save to be abandoned due to time constraints. This leaves the company vulnerable for any library that was not saved.

MMS/*bms* solves this problem by providing resume capabilities. MMS/*bms* resume capabilities will continue a backup from where the error occurred without having to start from the beginning. This reduces system downtime by maximizing the recovery process.

The types of failures monitored by MMS/*bms*, are:

- Device
- Media
- Power

Backup Resumption Overview

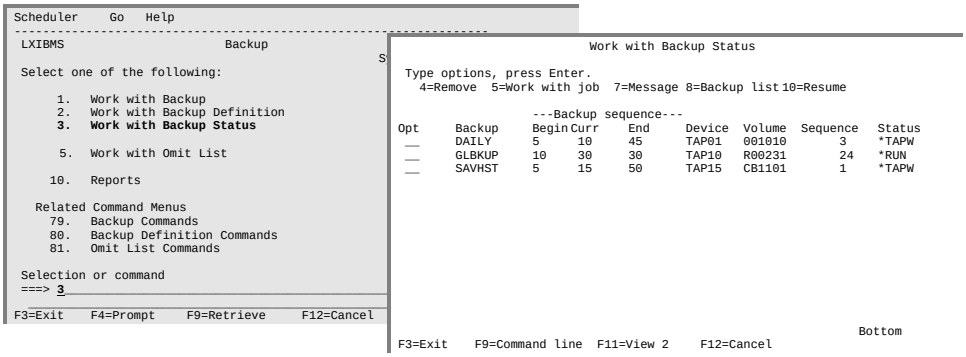
MMS/*bms* constantly monitors the status of a backup. If the backup ends abnormally, an error condition is flagged. The level of backup resumption depends on the type of save that failed. First level resumption restarts the backup from the failed Backup List sequence. If the Backup List sequence was an ***ALLUSR**, ***IBM** or ***NONSYS**, second level resumption processing continues the backup starting at the failed library.

Note: This feature requires LXI Tape Management System (MMS/*tms*). Refer to the LXI Tape Management documentation for information on installing and using this product.

Note: A resumed backup starts with the failed sequence on the same volume.

Working with Backup Status

Using **Option 3** from the Backup menu displays the Work with Backup Status panel. This panel displays active backups as well as backups that failed. The Work with Backup Status panel is also displayed by issuing the Work with Backup Status (WRKBKUPSTS) command from a command line.



Viewing Backups

The Work with Backup Status panel, **which automatically refreshes itself**, has the options needed to view the status of the backups and initiate backup resumption, if required. This panel does **not** have any entries if no backups are running or if no backups are eligible for resumption.

Removing the Status List Entry

If backup resumption for a failed backup is not required, enter **Option 4** from the Work with Backup Status panel or the Remove Backup Status (RMVBKUPSTS) command to remove the backup status list entry. The ***ERR** entry is automatically removed if the backup is run again.

Resuming Backups

If a Backup ends abnormally, select **Option 10** from the Work with Backup Status panel to resume the Backup. Specifying **Option 10** continues the Backup starting with the Backup List sequence that failed.

Resume Considerations

When a failed backup is resumed using **Option 10** from the Work with Backup Status panel, the following occurs:

1. The pre-exit program, if specified, is **not** run.
2. Subsystems specified on the **RUNBKUP** command are ended.
3. The last tape used by the failed backup is mounted.
4. The tape is positioned at the failed sequence number.
5. The backup is resumed.
6. Upon successful completion, the post-exit program, if specified, is run.
7. The subsystems that were ended by the **RUNBKUP** command are restarted.

Chapter 8

Saving the Entire System

Saving the system provides a starting point for recovery. MMS/*bms* provides two methods of saving the system. The first method provides an enhanced version of the IBM full system save (Option 21) found on the Save menu. The MMS/*bms* version adds the ability to start the function at a user-defined date and time. The second method allows the user to customize the save. With this method, the user determines what to save and when. The save can be a duplicate of the IBM function or it can perform additional saves. If the backup window is small, it can be designed to save only the most crucial objects. Both methods provide the ability to put the iSeries in restricted state.

Saving the entire system ensures that you have a complete copy of the system should a disaster occur. MMS/*bms* provides the ability to schedule and perform this backup.

Since this save requires that the iSeries be in restricted state, all jobs, except the console, must be ended. MMS/*bms* must be run from the console job. The scheduling function for this backup is performed through the Save All (**SAVE_ALL**) Backup. When the backup completes, MMS/*bms* can optionally start all subsystems and sign-off the console.

Using SAVE_ALL

MMS/*bms* provides a default Backup (**SAVE_ALL**) that saves the entire system. This Backup must be run from the iSeries console.

The default **SAVE_ALL** Backup saves the following:

- Licensed internal code
- The system library
- Security objects, including user profiles
- Device configuration objects
- All IBM supplied libraries including those containing user data
- All user libraries
- All mail
- All folders
- All documents
- All objects in directories
- MMS libraries needed for recovery

Changing the SAVE_ALL

The **SAVE_ALL** Backup can be modified as needed, however, it is recommended that the **SAVE_ALL** Backup be copied to another Backup and that the copied Backup be changed and run.

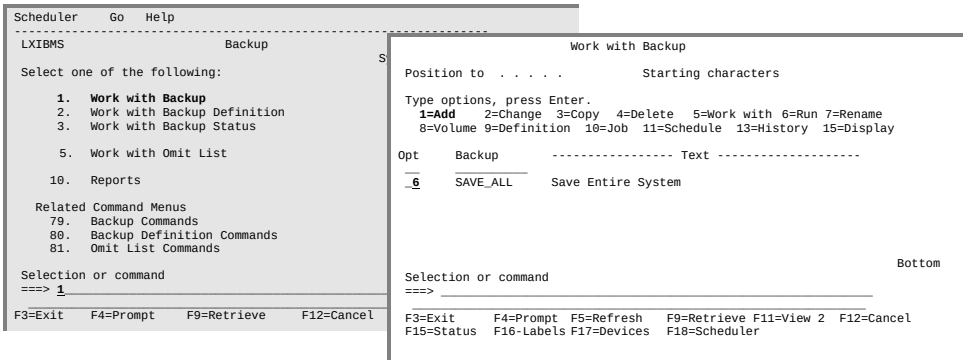
Copying the SAVE_ALL

To copy the **SAVE_ALL** Backup, Backup Lists and Backup List entries to another, use **Option 3** from the Work with Backup panel. This prompts the Copy Backup (CPYBKUP) command. Copying a Backup renumbers Backup Lists and Backup List entries.

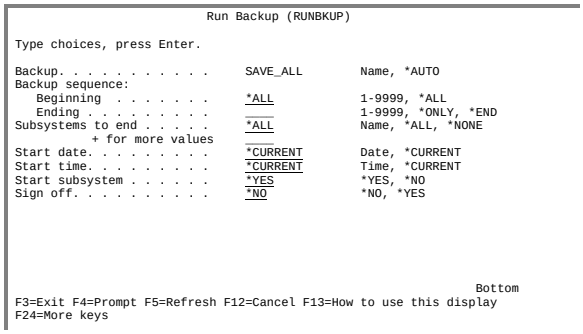
Note: The **SAVE_ALL** backup can be renamed.

Starting the SAVE_ALL

To save the entire system, choose **Option 1** from the Backup menu. From the Work with Backup panel, select **Option 6** for the **SAVE_ALL** Backup. This prompts the run Backup (RUNBKUP) command. Fill in the parameters as required and press **Enter**.



Specify ***ALL** for the "Subsystems to end" parameter and press **Enter**. After you have pressed **Enter**, MMS/*bms* verifies that you are in the controlling subsystem. Saving the entire system requires that your job be in the controlling subsystem. If you are not in the controlling subsystem, an error panel displays otherwise, the system enters restricted state and starts the backup based on the start date and time specified on the command.



When the backup starts, MMS/*bms*:

- Calls the pre-exit program
- Ends all subsystems

When the backup completes, MMS/*bms*:

- Calls the post-exit program
- Optionally starts the controlling subsystem
- Optionally signs off the console

Chapter 9

Restoring the system

Quick recovery is the key factor in determining your backup strategy. It is the ultimate test in determining if the strategy is successful or needs revising. MMS/*bms* recovery lets you recover the system the same way it was saved. If you need to restore the entire system or last night's changes, it can be done quickly and efficiently.

Recovery strategies vary based on date and time. Having some report that lists required volumes does no good if you don't know what's important to the company when a disaster strikes. MMS/*tms* solves this problem with **Recovery Lists**. With Recovery Lists, various recovery scenarios can be pre-defined so that when a disaster occurs, the recovery process becomes little more than executing the Recovery List that restores the objects that are needed the most first - automatically.

In the event of a complete system failure, restores need to occur in a specific sequence and, based on the complexity of the system involved, certain procedures will need to be followed. It is advised that you refer to the *IBM Backup and Recovery Guide* for specific details to ensure that nothing is overlooked. When the time comes to restore all user libraries from your last full system backup, MMS/*bms* will provide the ability to load them continuously without intervention. This also applies when it comes time to restore the saved changes.

Restore Overview

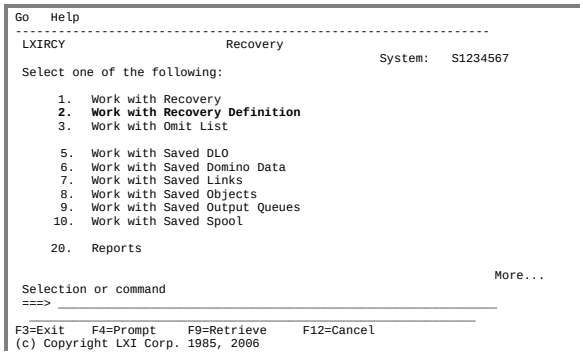
MMS/*bms* recovery is performed through Recovery Lists. Recovery Lists, like Backup Lists, identify and sequence the objects to be restored. These lists can be set up to recover an application or the system. Once the objects to restore have been identified, MMS/*bms* checks the MMS/*tms* database for the current backup of the objects. Every Recovery requires a Recovery Definition.

Backups consist of the following three elements:

- **Recovery**, which defines the name of the recovery and the Recovery Definition being used.
- **Recovery List**, which defines the type of objects to recovery as well as the type of restore to perform.
- **Recovery List entries**, which define the objects to restore. If generic names are specified, the object list is determined at recovery time.

Recovery Menus

The MMS/*bms* Recovery menu provides access to all recovery functions. To access the Recovery menu, type **GO LXI/LXIRCY** and press **Enter**. The Recovery menu can also be accessed from the **GO** option on the menu bar. The number of options displayed depends on the LXI products installed.



Recovery List Types

MMS/*bms* Recovery Lists provide virtually limitless options in establishing a recovery strategy. Since multiple Recovery Lists can exist for a recovery, any combination of libraries, objects, links, folders and documents can be specified in one comprehensive recovery.

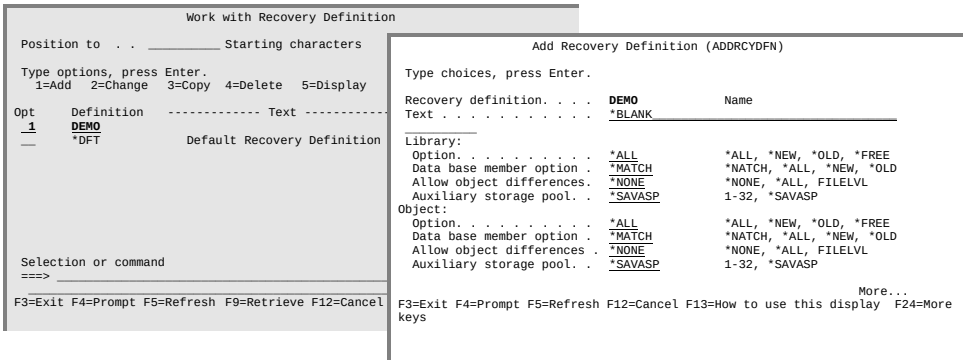
Recovery List	Type of list
*CFG	Configuration list.
*FLR	Document Library Object list.
*LIB	Library list.
*LNK	Integrated File System list.
*OBJ	Object list.
*RCY	MMS recovery libraries list.
*SEC	Security data list.
*SYS	System list.

Working with Recovery Definitions

Recovery Definitions associate user specified attributes to a Recovery. These attributes define the restore requirements and include the database member options, object differences options and auxiliary storage options. To access the Work with Recovery Definitions panel, select **Option 2** from the Recovery menu.

Adding a Recovery Definition

Using **Option 1** from the Work with Recovery Definition panel, enter a Recovery Definition name and press **Enter**. This prompts the Add Recovery Definition (ADDRCYDFN) command, which defines the Recovery attributes. Type the required values and review and optionally change the defaults. Press **Enter** when complete.



Changing a Recovery Definition

To change the attributes of a Recovery Definition, use **Option 2** from the Work with Recovery Definition panel. This prompts the Change Recovery Definition (CHGRCYDFN) command.

Copying a Recovery Definition

To copy the attributes of one Recovery Definition to another, use **Option 3** from the Work with Recovery Definition panel. This prompts the Copy Recovery Definition (CPYRCYDFN) command.

Deleting a Recovery Definition

To delete a Recovery Definition, use **Option 4** from the Work with Recovery Definition panel or the Delete Recovery Definition (DLTRCYDFN) command. The default Recovery Definition, ***DFT**, cannot be deleted.

Displaying a Recovery Definition

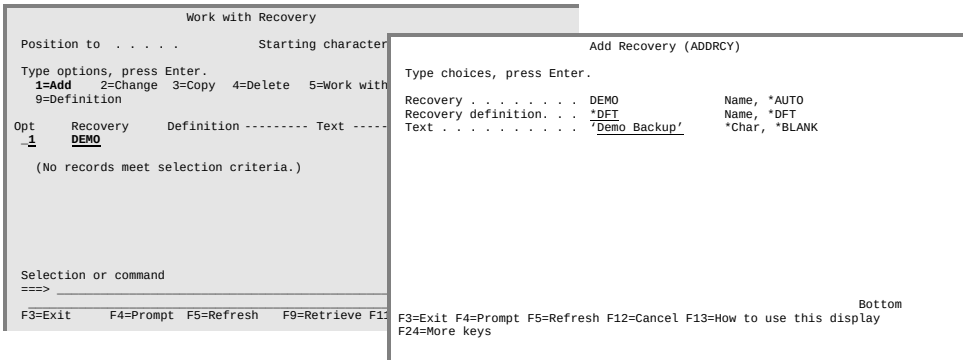
To display the Recovery Definition associated with a Recovery, use **Option 5** from the Work with Recovery Definition panel or the Display Recovery Definition (DSPRCYDFN) command.

Working with Recovery

Creating a Recovery is the first step in defining the objects and type of restore to perform. To access the Work with Recovery panel, select **Option 1** from the Recovery menu.

Adding a Recovery

Using **Option 1** from the Work with Recovery panel, enter a Recovery name and press **Enter**. This prompts the Add Recovery (ADDRCY) command, which associates a name and a Recovery Definition to the Recovery. Type the required values and press **Enter**.



Changing a Recovery

To change the text and/or Recovery Definition for a Recovery, use **Option 2** from the Work with Recovery panel. This prompts the Change Recovery (CHGRCY) command.

Copying a Recovery

To copy the Recovery Lists and Recovery List entries from one Recovery to another, use **Option 3** from the Work with Recovery panel. This prompts the Copy Recovery (CPYRCY) command. Copying a Recovery renumbers Recovery Lists and Recovery List entries.

Deleting a Recovery

To delete a Recovery and all associated Recovery Lists and Recovery List entries, use **Option 4** from the Work with Recovery panel or the Delete Recovery (DLTRCY) command.

Running a Recovery

Using **Option 6** from the Work with Recovery panel prompts the Run Recovery (RUNRCY) command. Pressing **Enter** will either print a recovery report or start the restore process. To execute the Recovery in batch or call it from within a user program, call the **RUNRCY** command. Running a Recovery renumbers Recovery Lists and Recovery List entries.

Renaming a Recovery

To rename a Recovery and associated Recovery Lists and Recovery List entries, use **Option 7** from the Work with Recovery panel. This prompts the Rename Recovery (RNMRCY) command. Renaming a Recovery renumbers Recovery Lists and Recovery List entries.

Working with the Recovery Definition

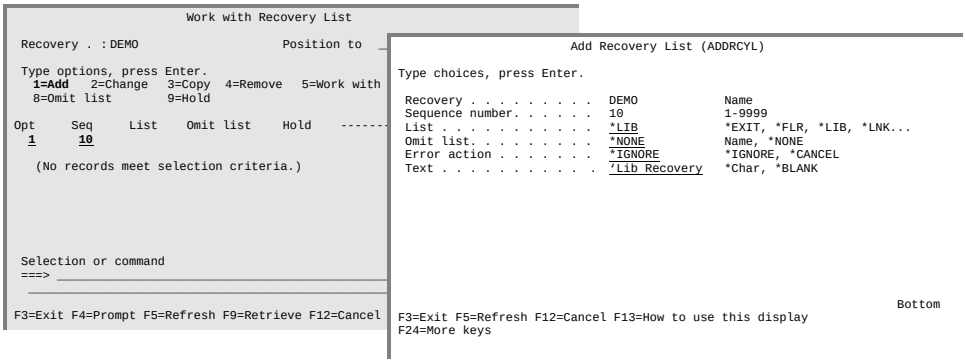
To view and optionally change the Recovery Definition associated with a Recovery, use **Option 9** from the Work with Recovery panel or the Work with Recovery Definition (WRKRCYDFN) command. This displays the Work with Recovery Definition panel.

Working with Recovery Lists

Recovery Lists associate a recovery type with a Recovery. Recovery types include the type of restore to perform, such as a **RSTLIB**, **RSTOBJ** or **RSTDLO**. Since multiple Recovery Lists can be associated with a Recovery, the Recovery List also determines the sequence and type of restore. To access the Work with Recovery List panel, select **Option 5** for a specific Recovery from the Work with Recovery panel.

Adding a Recovery List

Using **Option 1** from the Work with Recovery List panel, enter the sequence number of the Recovery List and press **Enter.** This prompts the Add Recovery List (ADDRCYL) command, which associates a recovery type to the Recovery. Type the required values and press **Enter**.



Changing a Recovery List

To change the Omit List name or text associated with a Recovery List, use **Option 2** from the Work with Recovery List panel. This prompts the Change Recovery List (CHGRCYL) command. An ***AUTO** Recovery List cannot be changed.

Copying a Recovery List

To copy a Recovery List from one Recovery to another, use **Option 3** from the Work with Recovery List panel. This prompts the Copy Recovery List (CPYRCYL) command. Copying a Recovery List renumbers the Recovery List and its entries.

Removing a Recovery List

To remove a Recovery List and associated Recovery List entries, use **Option 4** from the Work with Recovery List panel or the Remove Recovery List (RMVRCYL) command. An ***AUTO** Recovery List cannot be removed.

Releasing a Recovery List

Use **Option 6** from the Work with Recovery List panel or the Release Recovery List (RLSRCYL) command to release a held Recovery List. Held Recovery Lists are not processed.

Moving a Recovery List

To move a Recovery List from one Recovery to another, use **Option 7** from the Work with Recovery List panel. This prompts the Move Recovery List (MOVRCYL) command. An ***AUTO** Recovery List cannot be moved.

Omitting Entries from a Recovery List

To work with the entries on the Omit List associated with the Recovery List, use **Option 8** from the Work with Recovery List panel or the Work with Omit List (WRKOMITL) command. Entries cannot be omitted from an ***AUTO** Recovery List.

Holding a Recovery List

Using **Option 9** from the Work with Recovery List panel or the Hold Recovery List (HLDRCYL) command holds a Recovery List. Held Recovery Lists are not processed.

Note: If Omit lists are specified, they must exist. Use **F14** to view, and optionally add, Omit Lists.

Note: If a Recovery List receives an escape message, the Error action parameter (**ERROR**) determines if the recovery continues (***IGNORE**) or stops (***CANCEL**).

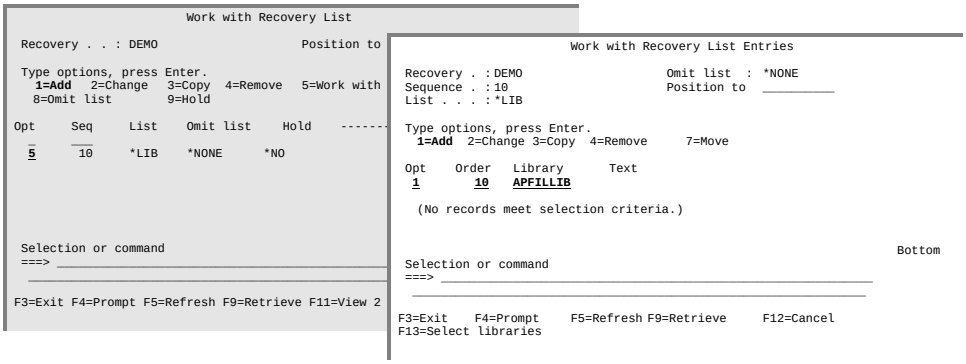
Tip: Increment the sequence numbers by five (5). This ensures that future Recovery Lists can be inserted between existing Recovery Lists.

Working with Recovery List Entries

Recovery List entries associate the objects to restore with a Recovery List. The Recovery List type determines the type of panel displayed. To access the Work with Recovery List Entries panel, select **Option 5** for a specific Recovery List from the Work with Recovery List panel.

Adding a Recovery List Entry

Using **Option 1** from the Work with Recovery List Entries panel, enter the required fields for the Recovery List entry and press **Enter.** This adds the entry to the Recovery List. If additional recovery attributes for the specified object are allowed, the Add Recovery List Entry (ADDRCYLE) command will be prompted. Type the required values and press **Enter**.



Copying a Recovery List Entry

To copy a Recovery List entry from one Recovery List to another, use **Option 3** from the Work with Recovery List Entry panel. This prompts the Copy Recovery List Entry (CPYRCYLE) command.

Removing a Recovery List Entry

Using **Option 4** from the Work with Recovery List Entry panel or the Remove Recovery List Entry (RMVRCYLE) command removes the selected Recovery List entry.

Moving a Recovery List Entry

To move a Recovery List entry from one Recovery List to another, use **Option 7** from the Work with Recovery List Entry panel. This prompts the Move Recovery List Entry (MOVRCYLE) command.

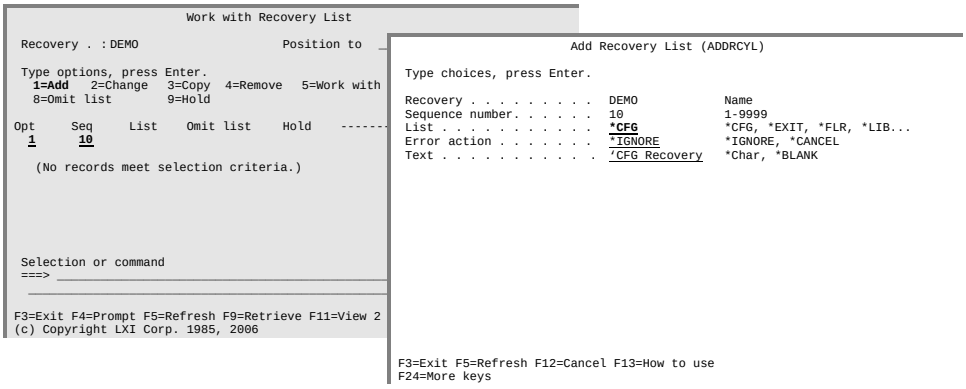
Adding Configuration Objects (*CFG)

Configuration lists recover configuration objects such as line, controller and device descriptions.

To add a Configuration list to a Recovery, create a Recovery List with a list type of ***CFG**.

Recovery List entries: *NO

Omit Lists: *NO

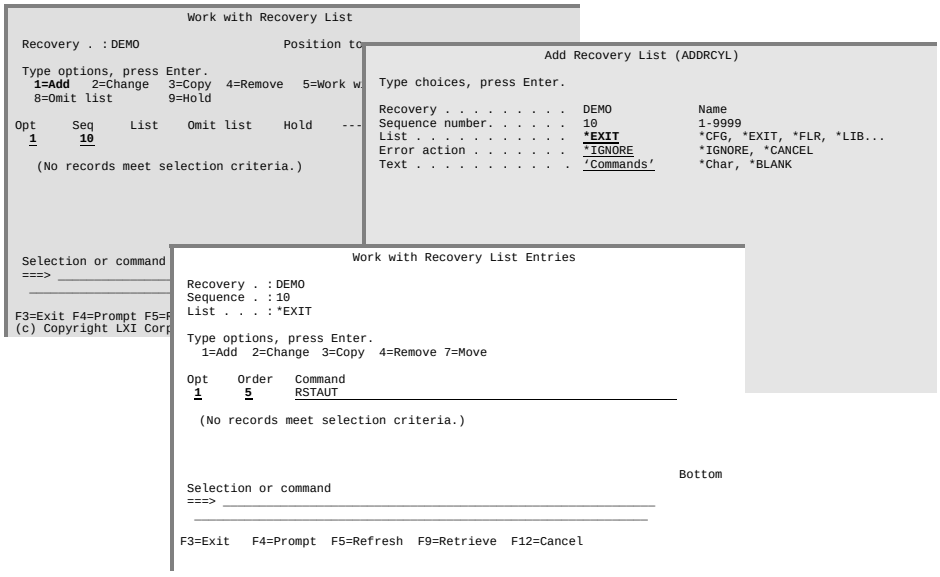


Adding Exits

Exit lists execute a command or program from within a Recovery List. Multiple Exit lists can exist in a Recovery List.

To add an Exit list to a Recovery, create a Recovery List with a list type of ***EXIT**. Once created, use **Option 5** from the Work with Recovery List panel to add the commands to execute when the Recovery List runs. Use **F4** to prompt the command, if needed.

To minimize potential library list problems, it is recommended that commands that do not reside in **QSYS** are qualified when entering them on the ***EXIT** Backup List entry. This ensures that the command will execute when the recovery runs.



Adding Folders (*FLR)

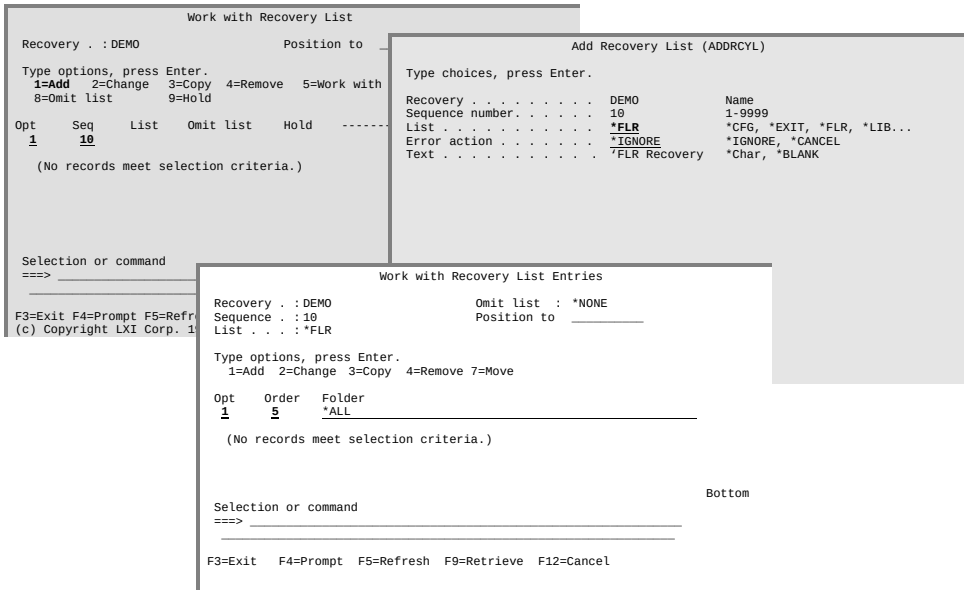
Folder lists specify which document library objects to recover.

To add a Folder list to a Recovery, create a Recovery List with a list type of ***FLR**. Once created, use **Option 5** from the Work with Recovery List panel to add the document library objects to the Recovery List.

Recovery List entries: *YES

Generic names: *YES

Omit Lists: *YES



Note: Document library objects must be tracked at detail level in order to restore specific folders or documents. Refer to the LXI Tape Management Reference Manual for details on tracking document library objects at detail level.

Adding Libraries (*LIB)

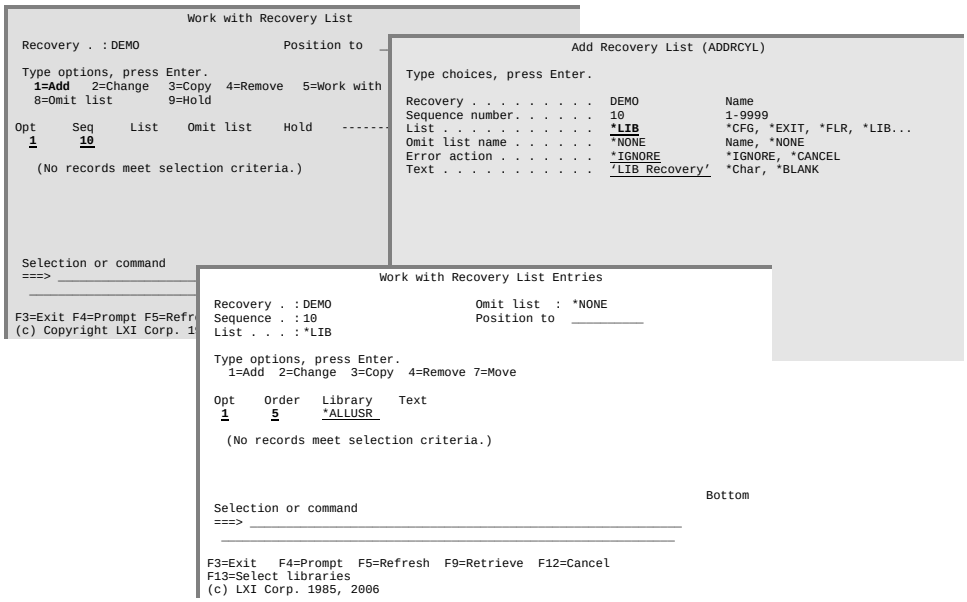
Library lists specify which libraries to recover. Special values ***ALLUSR**, ***IBM** and ***NONSYS** are allowed.

To add a Library list to a Recovery, create a Recovery List with a list type of ***LIB**. Once created, use **Option 5** from the Work with Recovery List panel to add the libraries to the Recovery List. Use **F13** to select from a list of available libraries.

Recovery List entries: *YES

Generic names: *YES

Omit Lists: *YES



Adding Integrated File System Objects (*LNK)

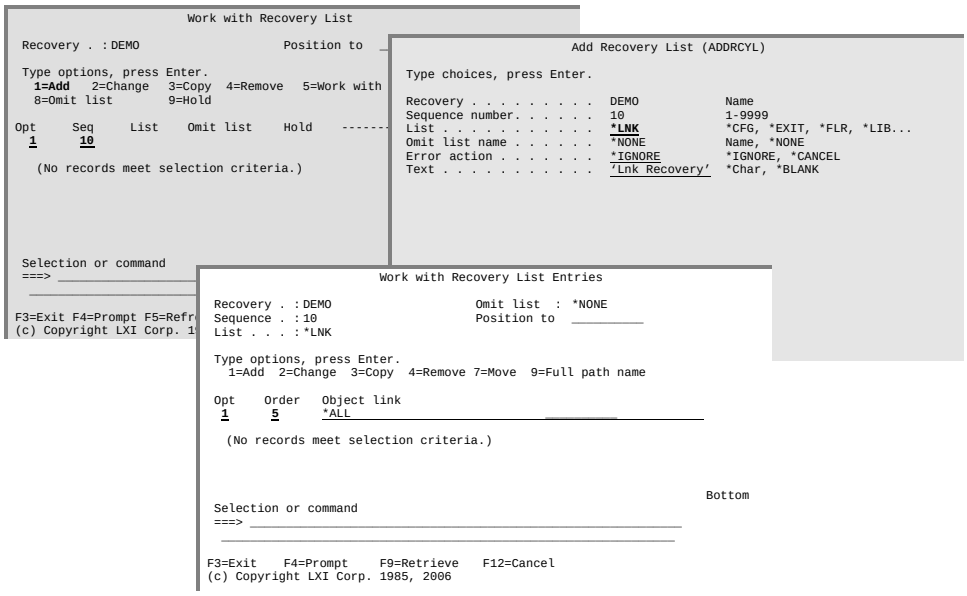
Link lists specify which integrated file system objects to recover. All or specific path names as well as generic path names can be specified.

To add a Link list to a Recovery, create a Recovery List with a list type of ***LNK**. Once created, use **Option 5** from the Work with Recovery List panel to add the integrated file system objects to the Recovery List.

Recovery List entries: *YES

Generic names: *YES

Omit Lists: *YES



Note: Integrated file system objects (object links) must be tracked at detail level in order to restore specific links. Refer to the MMS Tape Management Reference Manual for details on tracking integrated file system objects at detail level.

Adding Objects (*OBJ)

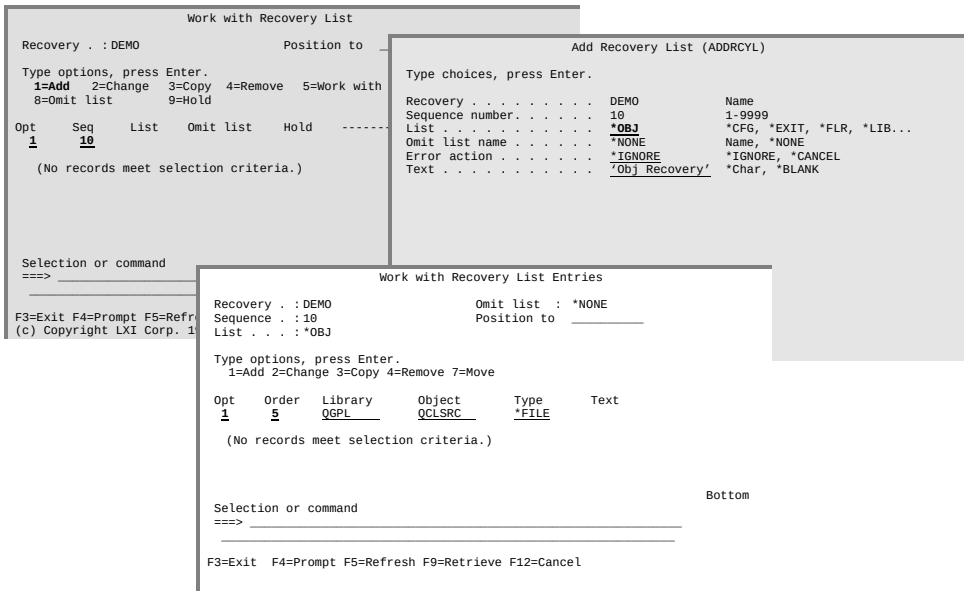
Object lists specify which objects to recover. All or specific objects as well as generic object names can be specified.

To add Object list to a Recovery, create a Recovery List with a list type of ***OBJ**. Once created, use **Option 5** from the Work with Recovery List panel to add the objects to the Recovery List.

Recovery List entries: *YES

Generic names: *YES

Omit Lists: *YES



Note: The object library is automatically created if it does not exist at restore time.

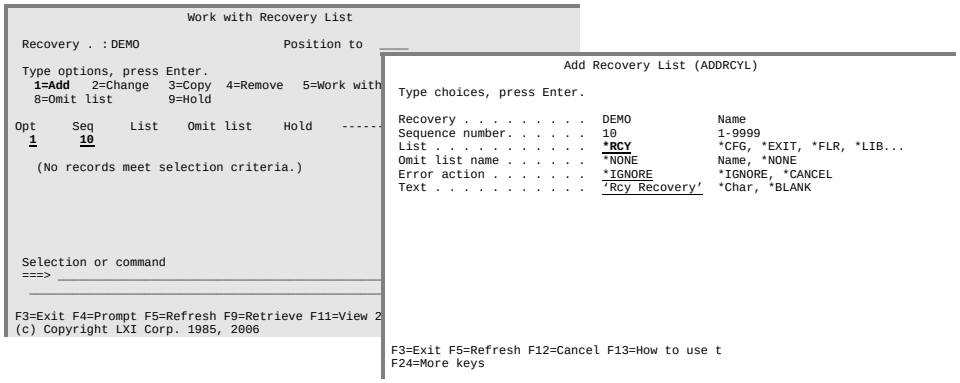
Adding MMS Recovery (*RCY)

Recovery Lists specify the MMS libraries needed for recovery. The libraries on this list are not restored as part of the recovery process. This list is for reporting purposes only.

To add a Recovery List to a Recovery, create a Recovery List with a list type of ***RCY**.

Recovery List entries: *NO

Omit Lists: *NO



Note: This Recovery List automatically adds the following libraries:

- LXI
- LXIBMS
- LXIBMS400
- LXITMS
- LXITMS400

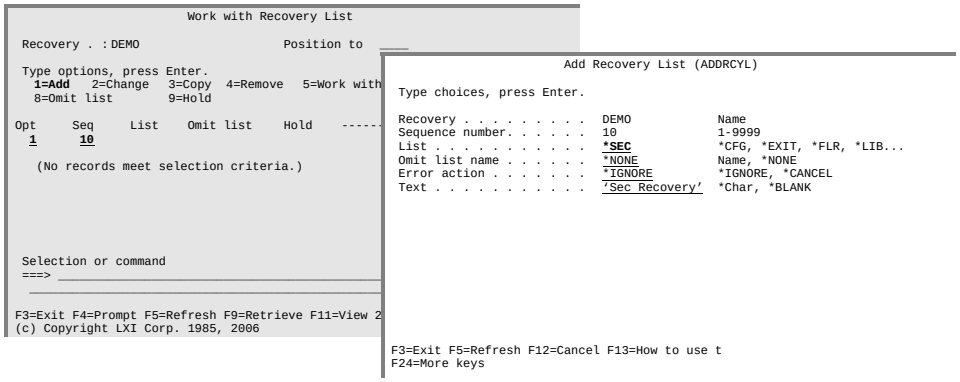
Adding Security Data (*SEC)

Security Lists specify the security data to recover. Security data includes user profiles and object authorities.

To add a Recovery List to a Recovery, create a Recovery List with a list type of ***SEC**.

Recovery List entries: *NO

Omit Lists: *NO



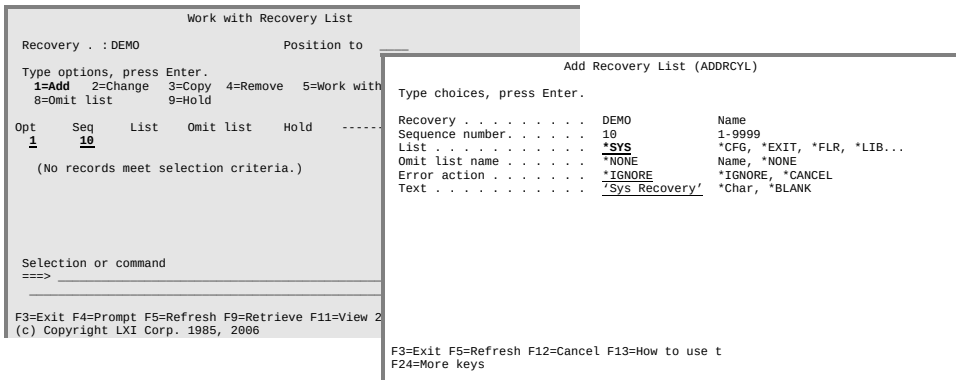
Adding System (*SYS)

System Lists specify the volumes needed to recover the operating system. This list is not restored as part of the recovery process. This list is for reporting purposes only.

To add a System List to a Recovery, create a Recovery List with a list type of ***SYS**.

Recovery List entries: *NO

Omit Lists: *NO



Using *Auto Recovery

The ***AUTO** recovery builds the lists necessary for a full system recovery. The following Recovery Lists are automatically built. The Recovery List entries are built when the Run Recovery (RUNRCY) is executed.

- *SYS - Used for reporting purposes only.
- *SEC - Used to recover security data.
- *CFG - Used to recover configuration objects.
- *RCY - Used for reporting purposes only.
- *LIB - Used to recover libraries and changed objects.
- *FLR - Used to recover document library objects and changes.
- *LNK - Used to integrated file system objects and changes.

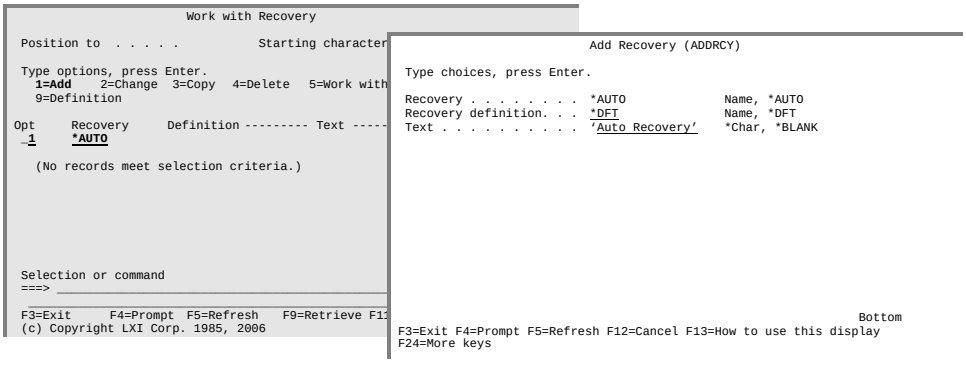
Adding a *AUTO Recovery

Using **Option 1** from the Work with Recovery panel, enter ***AUTO** for the Recovery name and press **Enter**. This prompts the Add Recovery (ADDRCY) command, which associates a name and a Recovery Definition to the Recovery. Type the required values and press **Enter**.

The Recovery List entries are built when the Run Recovery (RUNRCY) command is executed.

Recovery List entries: *NO

Omit Lists: *NO



Recovery Requirements

All MMS/*bms* recovery information is retrieved from and verified against the MMS/*tms* database. If no save information exists, the entry is bypassed, otherwise, all required full, cumulative and incremental save information is used to determine the recovery.

The MMS recovery libraries must exist in order to use MMS/*bms* recovery. The MMS libraries are saved by using the ***RCY** Backup List.

The MMS/*bms* recovery module uses the MMS/*tms* database to build the Recovery Lists. In order for MMS/*bms* to maintain a reference point from which other saves are selected, the following special saves **MUST** be performed through MMS/*bms* in order for the recovery module to function properly.

Save	List Type	List Entry
SAVDLO	*FLR	*ALL
SAV	*LNK	*ALL

Recovery and Media Definitions

MMS/*bms* automatically creates Media Definitions when a save that used Media Definitions is encountered. The Media Definition is created based on the number of resources available at the time of the restore. MMS/*bms* automatically balances the number of tapes per drive if the number of resources available is less than the number used for the save.

> **Note**: Some restrictions apply when trying to restore parallel saves. Read all IBM documentation regarding parallel save/restore restrictions before using Media Definitions.

The Recovery Report

MMS/*bms* provides two recovery reports.

- The object report lists information for the Recovery List Entries specified in the Recovery. If the ***AUTO** Recovery is specified, the report contains the following:
  - The last Save System (**SAVSYS**)
  - The last Save Security Data (**SAVSECDTA**)
  - The last Save Configuration (**SAVCFG**)
  - The MMS recovery libraries
  - The last save of the user libraries/objects being recovered
  - The last Save Document Library Objects (**SAVDLO**)
  - The last Save Integrated File System objects (**SAV**)
- The volume report, which lists the volumes needed for Recovery.

> **Tip**: Run the Recovery report(s) daily to ensure that they are available for use during recovery. A sample report can be found in Chapter 10, *Reports*.

Basic Recovery Procedure

1. Use the following steps to initiate a full system recovery. Refer to the IBM Backup and Recovery book as needed.
2. If you need to restore OS/400 from the last **SAVSYS**, use the MMS/*bms* Recovery Report to identify the volume(s) required. Refer to the IBM Backup and Recovery book for information on installing OS/400.
3. Restore the MMS recovery libraries specified on the MMS/*bms* Recovery Report. Use the Restore Library (**RSTLIB**) command to restore them.
4. View the Recovery that you are planning to run to ensure that all required Recovery Lists are released (not held).
5. Execute the required Recovery using the Run Recovery (RUNRCY) command.
6. If you have recovered the entire system, execute the Restore Authority (**RSTAUT**) command to establish object authorities and then IPL the system.

Recovery Considerations

Note the following when using the MMS/*bms* recovery module.

1. The MMS/*bms* recovery module does not install the following:

   - Save System (**SAVSYS**)
   - MMS recovery libraries

   Use the IBM Backup and Recovery manual for instructions on installing OS/400 and use the Restore Library (**RSTLIB**) command to restore the MMS Recovery libraries.

2. The recovery process automatically omits the following libraries:

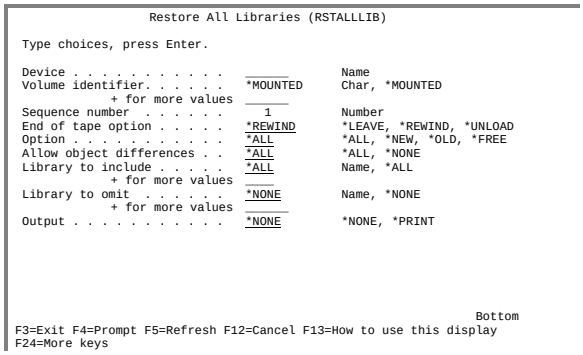
   - LXI
   - LXIBMS
   - LXIBMS400
   - LXITMS
   - LXITMS400

3. All recovery lists are executed interactively.

4. All objects not restored when recovering libraries are retried after all Recovery Lists have been processed.

Restoring all Libraries

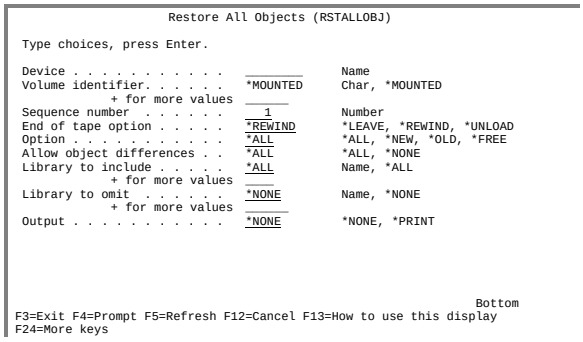
To restore some or all saved libraries, prompt the Restore All Libraries (RSTALLLIB) command. Fill in the parameters as required and press **Enter**. To ensure successful completion, no users should be accessing the libraries being restored. If the library is being accessed during the restore process, one or more objects may not restore correctly.



```
                Restore All Libraries (RSTALLLIB)

Type choices, press Enter.

Device . . . . . .            ______        Name
Volume identifier. . . . . .  *MOUNTED      Char, *MOUNTED
          + for more values   ______
Sequence number  . . . . . .    1           Number
End of tape option . . . . .  *REWIND       *LEAVE, *REWIND, *UNLOAD
Option . . . . . .            *ALL          *ALL, *NEW, *OLD, *FREE
Allow object differences . .  *ALL          *ALL, *NONE
Library to include . . . . .  *ALL          Name, *ALL
          + for more values   ____
Library to omit  . . . . . .  *NONE         Name, *NONE
          + for more values   ______
Output . . . . . .            *NONE         *NONE, *PRINT



                                                                  Bottom
F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display
F24=More keys
```

Restoring all Objects

To restore some or all saved objects, prompt the Restore All Objects (RSTALLOBJ) command. Fill in the parameters as required and press **Enter**. To ensure successful completion, no users should be accessing the libraries containing the objects being restored. If the library is being accessed during the restore process, one or more objects may not restore correctly. The library definition **must** exist on the system before restoring the objects.



```
                Restore All Objects (RSTALLOBJ)

Type choices, press Enter.

Device . . . . . .            _______       Name
Volume identifier. . . . . .  *MOUNTED      Char, *MOUNTED
          + for more values   ______
Sequence number  . . . . . .    1           Number
End of tape option . . . . .  *REWIND       *LEAVE, *REWIND, *UNLOAD
Option . . . . . .            *ALL          *ALL, *NEW, *OLD, *FREE
Allow object differences . .  *ALL          *ALL, *NONE
Library to include . . . . .  *ALL          Name, *ALL
          + for more values   ____
Library to omit  . . . . . .  *NONE         Name, *NONE
          + for more values   ______
Output . . . . . .            *NONE         *NONE, *PRINT



                                                                  Bottom
F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display
F24=More keys
```

Chapter 10

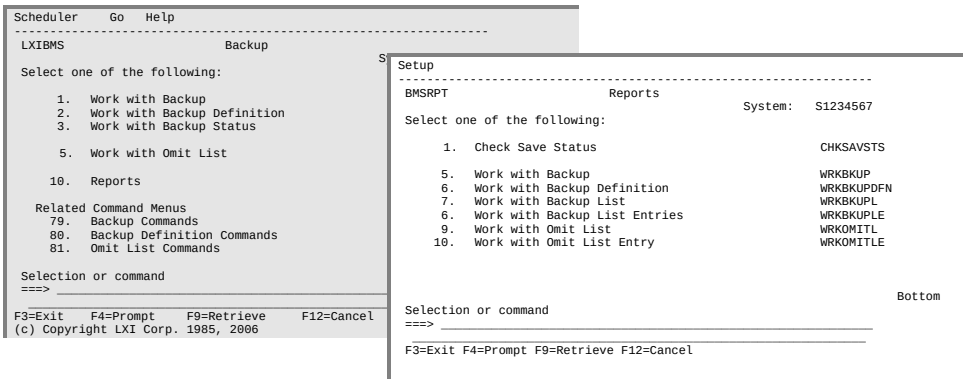
Backup Management Reports

The following is a list of all reports available from MMS/*bms*. The name of the report and the command(s) used to print them and the printer file are listed. All printer files reside in library LXIBMS400.

Report	**Command**	**Printer File**
Backup Attributes	WRKBKUP	QBMSPRT
Backup Definition	WRKBKUPDFN	QBMSPRT
Backup Lists	WRKBKUPL	QBMSPRT
Backup List Entries	WRKBKUPLE	QBMSPRT
Object Save Status	CHKSAVSTS	QBMSPRT
Omit Lists	WRKOMITL	QBMSPRT
Omit List Entries	WRKOMITLE	QBMSPRT
Recovery Report	RUNRCY	QBMSPRT
Recovery	WRKRCY	QBMSPRT
Recovery Definition	WRKRCYDFN	QBMSPRT
Recovery List	WRKRCYL	QBMSPRT
Recovery List Entry	WRKRCYLE	QBMSPRT

Accessing the Report Menus

To access the report menu, choose **Option 10** from the Backup menu.



Printing Reports

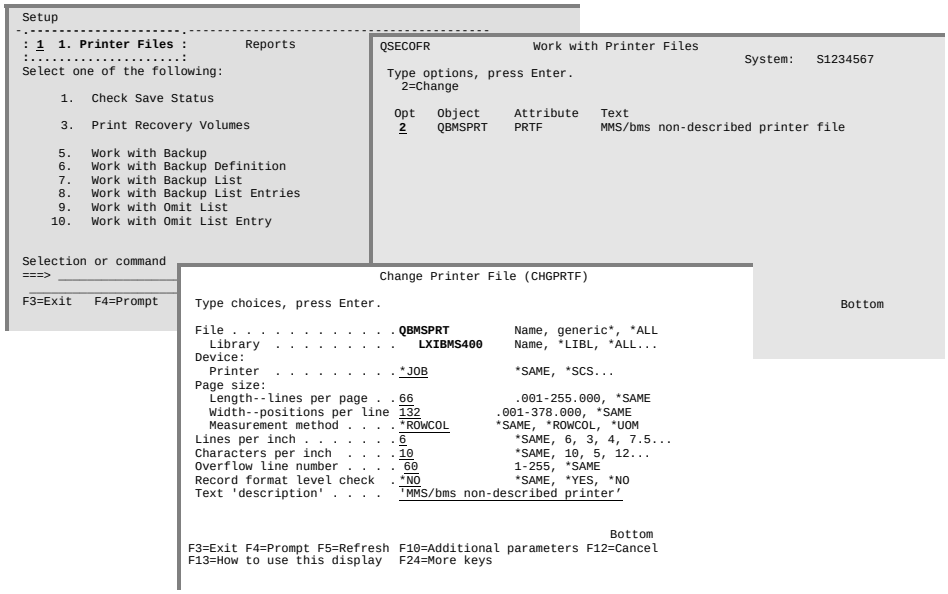
To print a report, choose the desired option number from the Reports menu. This prompts the associated command for selection criteria, if required, and spools the output to the jobs output queue or to the output queue specified in the printer file, if overridden. The reports can also be printed by prompting the commands (listed on the right hand side) and specifying ***PRINT** for the **OUTPUT** parameter.

Changing Printer File Attributes

MMS/*bms* allows you to change the attributes of the printer device file. The attributes that can be changed include the device, device type, lines per page, lines per inch, characters per inch and output queue. The ability to change the printer file is based on the user's authority to the IBM Change Printer File (**CHGPRTF**) command.

Changing Printer Attributes

To change the attributes of a MMS/*bms* printer file, select **Setup** from the Reports pull down menu. Choose **Option 1**, Printer Files. This lists the printer files available for change. Choose **Option 2** for the printer file requiring change and change the parameters as needed. The changes remain with the printer file until changed again.



Backup Report

The Work with Backup (WRKBKUP) command creates the Backup report, which shows backup information including Backup Definition, job, user, number and date/time of last save.

```
                                    Backups
                                                                              Page     1
                                                             S1234567  01/07/05  23:30:20

 Backup     Definition   Job        User      Number    Date       Time       Text
 *AUTO      TAPMLB01     BMS_DAILY  QSYSOPR   293872    01/06/05   20:15:45   Auto Backup
 DAILY      TAPMLB01     BMS_DAILY  QSYSOPR   203874    09/12/04   20:56:09   Daily Backup
 MONTHLY    TAPMLB01     BMS_MTHLY  QSYSOPR   387484    12/31/04   20:00:39   Monthly Backup

                  * * * * *   (c) Copyright LXI Corp. 1985, 2006   * * * * *
```

Backup Definition

The Work with Backup Definition (WRKBKUPDFN) command creates the Backup Definition report which shows Backup Definition information including device, save while active options and pre and post exit programs.

```
                                   Backup Definition
                                                                                Page     1
                                                                S1234567  01/07/05  23:30:20

Definition  . . . . . . . . :  *DFT
Device. . . . . . . . . . . :  TAP01
End of tape option. . . . . :  *LEAVE
Use optimum block . . . . . :  *NO
Journaled objects . . . . . :  *NO
Target release. . . . . . . :  *CURRENT
Clear . . . . . . . . . . . :   *NO
Object pre-check. . . . . . :  *NO
Save active:
   Object link. . . . . . . :  *NO
   Folder . . . . . . . . . :  *NO
   Library. . . . . . . . . :  *NO
   Object . . . . . . . . . :  *NO
   Changed object . . . . . :  *NO
Save active wait time. . . .:  120
Save active message queue. .:  *NONE
   Library. . . . . . . . . :
Save file data. . . . . . . :  *YES
Save access paths . . . . . :  *NO
Data compression. . . . . . :  *DEV
Data compaction . . . . . . :  *DEV
Output. . . . . . . . . . . :  *NONE
Pre-Program . . . . . . . . :  *NONE
   Library. . . . . . . . . :
Post-Program. . . . . . . . :  *NONE
   Library. . . . . . . . . :
Text. . . . . . . . . . . . :  Default Definition
                   * * * * *   (c) Copyright LXI Corp. 1985, 2006   * * * * *
```

Backup List

The Work with Backup List (WRKBKUPL) command creates the Backup List report, which lists the Backup Lists in a specific Backup.

```
                              Backup List
                                                                              Page     1
                                                          S1234567  01/07/05  23:40:14
Backup . . :   *AUTO

Seq  List    Type    Omit list  Hold  Date      Time        Format      Minimum    Maximum Text
  5  *EXIT   *NONE   *NONE      *NO   01/06/04  20:04:31    Serial      *NONE      *NONE   Auto Backup
 10  *LIB    *FULL   *NONE      *NO   01/06/04  20:04:31    Serial      *NONE      *NONE   Auto Backup
 15  *LIB    *CUML   *NONE      *NO   01/06/04  20:06:56    Serial      *NONE      *NONE   Auto Backup
 20  *LIB    *FULL   *NONE      *NO   01/06/04  20:07:11    Serial      *NONE      *NONE   Auto Backup
 25  *LIB    *CUML   *NONE      *NO   01/06/04  20:09:09    Serial      *NONE      *NONE   Auto Backup
 30  *LIB    *FULL   *NONE      *NO   01/06/04  20:09:35    Serial      *NONE      *NONE   Auto Backup
 35  *LIB    *CUML   *NONE      *NO   01/06/04  20:11:42    Serial      *NONE      *NONE   Auto Backup
 40  *LIB    *FULL   *NONE      *NO   01/06/04  20:12:29    Serial      *NONE      *NONE   Auto Backup
 45  *LIB    *CUML   *NONE      *NO   01/06/04  20:12:51    Serial      *NONE      *NONE   Auto Backup
 50  *LNK    *CUML   *NONE      *NO   01/06/04  20:15:33    Serial      *NONE      *NONE   Auto Backup
 55  *RCY    *FULL   *NONE      *NO   01/06/04  20:18:10    Serial      *NONE      *NONE   Auto Backup

                   * * * * *   (c) Copyright LXI Corp. 1985, 2006   * * * * *
```

Backup List Entries

The Work with Backup List Entries (WRKBKUPLE) command creates the Backup List Entries report, which lists all entries in the specified Backup List.

```
                                  Backup List Entries
                                                                              Page     1
                                                                S1234567  01/07/05  23:44:33

Backup . . :   *AUTO                                 Type . . . :   *FULL
Sequence . :   0020                                  Omit list .:   *NONE
List . . . :   *LIB

Order  Library        Text
   10  APFILLIB       A/P File Library
   15  ARFILLIB       A/R File Library
   20  GLFILLIB       G/L File Library
   25  PRFILLIB       P/R File Library
                 * * * * *   (c) Copyright LXI Corp. 1985, 2006   * * * * *
```

Object Save Status

The Check Save Status (CHKSAVSTS) command creates the Object Save Status report, which lists all changed or non-saved objects for the specified library/type.

```
                                   Object Save Status
                                                                                   Page     1
                                                                      S1234567  01/07/05  23:30:20

Library   Object     Type      Attribute  Member     ---Save date/time---   --Change date/time--
LXIBMS    BM#WL30P   *PNLGRP                         01/06/05   20:11:25   01/07/05     09:18:55
LXIBMS    BMSRPT     *MEBU     UIM                   01/06/05   20:11:25   01/07/05     15:50:09
LXIBMS    BM#RC10R   *MODULE   RPGLE                 00/00/00   00:00:00   01/07/05     12:38:06
LXIBMS    BM#RC10R   *SRVPGM   RPGLE                 00/00/00   00:00:00   01/07/05     12:43:20
LXIBMS    BM#RC30R   *MODULE   RPGLE                 01/06/05   20:11:25   01/07/05     12:50:59
LXIBMS    BM#RC30R   *SRVPGM   RPGLE                 01/06/05   20:11:25   01/07/05     12:51:43

                  * * * * *   (c) Copyright LXI Corp. 1985, 2006   * * * * *
```

Omit List

The Work with Omit List (WRKOMITL) command creates the Omit List report, which lists the specified Omit List.

```
                                   Omit List
                                                                           Page     1
                                                        S1234567  01/07/05  23:44:33

 List name  List       Text
 EXAMPLE    *OBJ       Example Omit List
 FLR        *FLR       Folder Omit List
 LIB        *LIB       Library Omit List
                * * * * *   (c) Copyright LXI Corp. 1985, 2006   * * * * *
```

Omit List Entries

The Work with Omit List Entries (WRKOMITLE) command creates the Omit List Entries report, which lists the entries in a specific Omit List.

```
                                 Omit List Entries
                                                                         Page     1
                                                           S1234567  01/07/05  23:40:14

Omit list:   LIB
List. . . :  *LIB

Library                 Library                 Library               Library
LXI                     LXITMS400
LXITMS
                  * * * * *   (c) Copyright LXI Corp. 1985, 2006   * * * * *
```

Recovery Report

The Run Recovery (RUNRCY) command creates the Recovery reports, which lists all objects and required volumes for the specified Recovery. The first Recovery report automatically includes the Save System (***SYS**) and MMS Recovery library (***MMS**) information along with the user-specified Recovery Lists. The second report lists the volumes required for the restore operation.

```
                                    Recovery Report
                                                                                   Page     1
                                                                 S1234567  01/07/05  23:30:20
Recovery . : *AUTO
Sequence . : 1
List . . . : *SYS

Order    Library      Command      Date       Time       Volume      Seq.
   1     QSYS         SAVSYS       03/22/05   19:18:55   350106       1
```

```
                                    Recovery Report
                                                                                   Page     2
                                                                 S1234567  01/07/05  23:30:20
Recovery . : *AUTO
Sequence . : 5
List . . . : *SEC

Order  Library      Command        Date       Time       Volume      Seq.
   1   QSYS         SAVSECDTA      02/27/05   20:15:40   350131      1793
```

```
                                    Recovery Report
                                                                                   Page     3
                                                                 S1234567  01/07/05  23:30:20
Recovery . : *AUTO
Sequence . : 10
List . . . : *CFG

Order  Library    Command          Date       Time       Volume      Seq.
   1   QSYS         SAVCFG         03/27/05   20:15:26   350131      1792
```

```
                                    Recovery Report
                                                                                   Page     2
                                                                 S1234567  01/07/05  23:30:20
Recovery . : *AUTO
Sequence . : 15
List . . . : *RCY

Order  Library      Command        Date       Time       Volume      Seq.
    5  LXI          SAVLIB         03/27/05   20:17:00   350131      1794
   10  LXIBMS       SAVLIB         03/27/05   20:17:03   350131      1795
   15  LXIBMS400    SAVLIB         03/27/05   20:17:07   350131      1796
   20  LXITMS       SAVLIB         03/27/05   20:17:08   350131      1797
   25  LXITMS400    SAVLIB         03/27/05   20:17:09   350131      1798
```

```
                                Recovery Report
                                                                                Page     4
                                                                 S1234567  01/07/05  23:30:20
Recovery . : *AUTO
Sequence . : 20
List . . . : *LIB

Order  Library      Command       Date      Time      Volume    Seq.
  5    *ALLUSR      SAVLIB        03/26/05  20:00:56  350131    1453
  5    *IBM         SAVLIB        03/26/05  20:39:28  350131    1680
  5    BMS5.3       SAVCHGOBJ     03/27/05  20:03:35  350131    1706
  5    LND5.3       SAVCHGOBJ     03/27/05  20:04:10  350131    1707
  5    QGPL         SAVCHGOBJ     03/27/05  20:04:26  350131    1708
  5    QUSRSYS      SAVCHGOBJ     03/27/05  20:04:48  350131    1709


                                Recovery Report
                                                                                Page     5
                                                                 S1234567  01/07/05  23:30:20
Recovery . : *AUTO
Sequence . : 25
List . . . : *FLR

Order  Folder                                  Date      Time      Volume    Seq.
  5    *ALL                                    03/22/05  20:01:18  350106    309


                                Recovery Report
                                                                                Page     6
                                                                 S1234567  01/07/05  23:30:20
Recovery . : NONSYS
Sequence . : 20
List . . . : *LNK

Order  Object link                               Date      Time      Volume    Seq.
  5    *                                         03/22/05  20:01:18  350106    309
  5    *                                         03/27/05  20:13:59  350131    1791


                                Recovery Report
                                                                                Page     1
                                                                 S1234567  01/07/05  23:30:20
Recovery . : *AUTO

Volume   Density    Location     Ctn/Slot
350106   *FMT3590H  *DFTLOC      *NONE
350131   *FMT3590H  *DFTLOC      *NONE

                  * * * * *   (c) Copyright LXI Corp. 1985, 2006   * * * * *
```

Recovery Report

The Work with Recovery (WRKRCY) command creates the Recovery report, which shows recovery information including Recovery Definition and text.

```
                                  Recovery
                                                                                   Page     1
                                                                S1234567  01/07/05  23:30:20

 Recovery   Definition     ----------------- Text ----------------------
 *AUTO      TAPMLB01       MMS/bms Auto Recovery
 *NONSYS    TAPMLB01       NonSys Recovery
 GL_APP     TAP01          General Ledger Application Recovery

                 * * * * *   (c) Copyright LXI Corp. 1985, 2006   * * * * *
```

Recovery Definition

The Work with Recovery Definition (WRKRCYDFN) command creates the Recovery Definition report which shows Recovery Definition information including device, save while active options and pre and post exit programs.

```
                                 Recovery Definition
                                                                        Page     1
                                                     S1234567  01/07/05  23:30:20

Definition  . . . . . . . . :   *DFT
Text. . . . . . . . . . . . :   Default Definition

Library:
  Option. . . . . . . . . . :   *ALL
  Data base member option . :   *MATCH
  Allow object differences. :   *NONE
  Auxiliary storage pool ID :   *SAVASP
Object:
  Option. . . . . . . . . . :   *ALL
  Data base member option . :   *MATCH
  Allow object differences. :   *NONE
  Auxiliary storage pool ID :   *SAVASP
Folder:
  Object name generation. . :   *SAME
  Allow object differences. :   *NONE
  Saved from ASP. . . . . . :   *ANY
  Restore to ASP. . . . . . :   *SAVASP
Object link:
  Option. . . . . . . . . . :   *ALL
  Allow object differences. :   *NONE
                  * * * * *   (c) Copyright LXI Corp. 1985, 2006   * * * * *
```

Recovery List

The Work with Recovery List (WRKRCYL) command creates the Recovery List report, which lists the Recovery Lists in a specific Recovery.

```
                                        Recovery List
                                                                                        Page     1
                                                                   S1234567  01/07/05  23:40:14
Recovery . . :   *AUTO

Seq    List    Omit list    Hold    -------------------- Text ---------------------
  10   *LIB    *NONE        *NO     MMS/bms Auto Recovery
  15   *LNK    *NONE        *NO     MMS/bms Auto Recovery

                  * * * * *   (c) Copyright LXI Corp. 1985, 2006   * * * * *
```

Recovery List Entries

The Work with Recovery List Entries (WRKRCYLE) command creates the Recovery List Entries report, which lists all entries in the specified Recovery List.

```
                              Recovery List Entries
                                                                          Page     1
                                                       S1234567  01/07/05  23:44:33

 Backup . . :   *AUTO                              Omit list .:   *NONE
 Sequence . :   0020
 List . . . :   *LIB

 Order  Library       Text
    10  APFILLIB      A/P File Library
    15  ARFILLIB      A/R File Library
    20  GLFILLIB      G/L File Library
    25  PRFILLIB      P/R File Library
                  * * * * *   (c) Copyright LXI Corp. 1985, 2006   * * * * *
```

Chapter 11

Backup and Recovery Commands

MMS/*bms* is a command-driven product. Even in the MMS/*bms* menus, commands are executed to perform the requested function. If desired, these commands can be used directly instead of the menus to provide faster access to MMS/*bms* functions. Not all commands can be used in the same environment. Some commands can only be used interactively (**I**), some only in batch (**B**) and others are available for all environments (**B/I**). Commands are restricted to the environment for which they were created. Before using a MMS/*bms* command, ensure that it is allowed in the environment from which you wish to execute it.

The following pages show all of the Backup & Recovery Management commands with their parameters and a brief description of each parameter's purpose.

The commands are listed in alphabetical sequence.

ADDBKUP - Add Backup

```
---------------------------------------------------------------------------------------------------
                    Add Backup (ADDBKUP)                    Environment: B/I

Backup . . . . . . . . . . . . . . . . . . . .  _________    Name, *AUTO
Backup definition . . . . . . . . . . . . . .  . *DFT        Name, *DFT
Text . . . . . . . . . . . . . . . . . . . . .  *BKUPDFN     Char, *BKUPDFN, *BLANK

---------------------------------------------------------------------------------------------------
```

Purpose

The Add Backup (ADDBKUP) command creates a Backup and its associated Backup Definition.

Parameters

BACKUP: Specifies the name of the Backup.

*AUTO	The Intelligent Backup is being added. Backup Lists, Backup List entries and Omit Lists are not allowed with this type of save.
Backup-name	Enter a name for the Backup.

BKUPDFN: Specifies the Backup Definition to associate with this Backup.

***DFT**	The default Backup Definition is selected.
Backup-definition	Enter a valid Backup Definition.

TEXT: Specifies the text that briefly describes the object.

***BKUPDFN**	The text associated with the Backup Definition is used.
*BLANK	No text is specified.
'description'	Enter no more than 50 characters of text, enclosed in apostrophes.

Examples

```
ADDBKUP BKUP(DAILY) BKUPDFN(*DFT) TEXT('Daily Backup')
```

This adds a Backup named DAILY, which uses the default Backup Definition.

ADDBKUPDFN - Add Backup Definition

```
------------------------------------------------------------------------------------------------------
                    Add Backup Definition  (ADDBKUPDFN)          Environment: B/I

Backup definition . . . . . .  _________        Name
Device . . . . . .  _________        Name
              + for more values  _________
End of tape option . . . . . .  *UNLOAD        *LEAVE, *UNLOAD
Use optimum block . . . . . .  *YES        *YES, *NO
Journaled objects . . . . . .  *NO        *NO, *YES
Target release . . . . . .  *CURRENT        *CURRENT, *PRV…
Clear. . . . . .  *NONE        *NONE, *ALL, *AFTER
Object pre-check . . . . . .  *NO        *NO, *YES
Save active:
  Object link . . . . . .  *NO        *NO, *YES, *SYNC
  Folder . . . . . .  *NO        *NO, *YES
  Library . . . . . .  *NO        *NO, *LIB, *SYNCLIB, *SYSDFN
  Object . . . . . .  *NO        *NO, *LIB, *SYNCLIB, *SYSDFN
  Changed object . . . . . .  *NO        *NO, *LIB, *SYNCLIB, *SYSDFN
Save active wait time . . . . . .  120        0-99999, *NOMAX
Save active message queue . . . .  *NONE        Name, *NONE
  Library . . . . . .  _________        Name
Save access paths . . . . . .  *NO        *NO, *YES
Save file data . . . . . .  *YES        *YES, *NO
Data compression . . . . . .  *DEV        *DEV, *NO, *YES
Data compaction . . . . . .  *DEV        *DEV, *NO
Output . . . . . .  *NONE        *NONE, *PRINT
Text . . . . . .  *BLANK        Char, *BLANK
Pre-Exit program . . . . . .  *NONE        Name, *NONE
  Library . . . . . .  _________        Name
Post-Exit program . . . . . .  *NONE        Name, *NONE
  Library . . . . . .  _________        Name

------------------------------------------------------------------------------------------------------
```

Purpose

The Add Backup Definition (ADDBKUPDFN) command creates a Backup Definition. The Backup Definition determines the attributes to use for the Backup.

Parameters

BKUPDFN: Specifies the name of the Backup Definition.

Backup-definition Enter a Backup Definition name.

DEVICE: Specifies the name of one or more devices to use for the backup process. Up to 32 devices can be specified. Only the first 4 are used for saves and restores.

The additional drives can be specified if using Media Definitions and a StorageTek Tape Library. To eliminate the need to enter up to 32 StorageTek tape devices, specify only one (1) device. MMS/bms automatically selects more drives based on the number of resources specified in the **DRVRSC** parameter of the Backup List. In order for this feature to work, the StorageTek drives must be defined to TMS as **SHARE(*YES)**.

Device-definition Enter one or more valid device names.

ENDOPT: Specifies the positioning option that is automatically done on the tape volume after the operation ends. If more than one volume is used, the parameter applies only to the last volume; all other volumes are rewound and unloaded when the end of tape is reached.

***UNLOAD**	The tape is automatically rewound and unloaded after the operation ends.
*LEAVE	The tape does not rewind or unload after the operation ends. It remains at the current position on the tape drive.

USEOPTBLK: Specifies whether the optimum block size is used for the operation.

***YES**	The optimum block size supported by the device is used for Save commands. Performance may improve, but the tape volume that is created is only compatible with a device that supports the block size used. Commands such as Duplicate Tape **(DUPTAP)** do not duplicate files unless the files are being duplicated to a device which supports the same block size that was used.
*NO	The optimum block size supported by the device is not used. Save commands use the default block size supported by all device types. The tape volume can be duplicated to any media format using the Duplicate Tape (**DUPTAP**) command.

OBJJRN: Specifies whether changes to objects currently being entered in a journal are saved.

***NO**	Journaled objects are not saved.
*YES	Journaled objects are saved.

TGTRLS: Specifies the release of the operating system on which you intend to restore and use the object.

When specifying the target-release value, the format VxRxMx is used to specify the release, where Vx is the version, Rx is the release, and Mx is the modification level. For example, **V4R5M0** is version 4, release 5, modification level 0.

To specify that an object be saved for distribution to a system at a different release level than the system on which the save operation is to occur, the procedure differs for program or non-program objects and by the release level on which a program object is created. If, for example, you are saving an object for distribution to a target system running on an earlier release, you have the following choices:

If the program object was created at a release level more current than the targeted earlier release, you must (1) create the program object again specifying the targeted earlier release, (2) save the program object specifying the targeted earlier release, and then (3) restore the program object on the target system.

If the program object was created at the same release level as the target system, you can (1) save the program object specifying the targeted earlier release and then (2) restore the program object on the target system.

For non-program objects:

You can (1) save the object specifying the targeted earlier release and then (2) restore the object on the target system.

The possible values are:

***CURRENT** — The object is to be restored to, and used on, the release of the operating system currently on this system. The object can also be restored to a system with any subsequent release of the operating system installed.

*PRV — The object is to be restored to the previous release with modification level 0 of the operating system. The object can also be restored to a system with any subsequent release of the operating system installed.

Target-release — Specify the release in the format VxRxMx. The object can be restored to a system with the specified release or with any subsequent release of the operating system installed.

Valid values depend on the current version, release, and modification level, and they change with each new release.

Note: If **LIB(*ALLUSR)** is specified, only the current release can be the target release. For release **V5R2M0**, valid values are ***CURRENT** or **V5R2M0**.

Not all objects can be targeted to another release. To find out which objects are supported, see the chart in the Backup and Recovery book, SC41-5304.

CLEAR: Specifies whether tapes or save files that contain active data and are encountered during the save operation are automatically cleared. An uncleared tape is one containing a file with an expiration date later than the date of the save operation (including files protected permanently with **EXPDATE(*PERM)**).

Note: This parameter does not control initializing tapes used to perform the save operation. Tapes should be initialized to a standard label format before the save command is issued.

You can use the Initialize Tape (**INZTAP**) command and specify a value on the **NEWVOL** parameter to initialize a tape to a standard label format.

If a tape volume that is not initialized is encountered during the save operation, an inquiry message is sent and an operator can initialize the tape volume.

***NONE** — None of the media used during the save operation are cleared. An inquiry message is sent to the system operator if active files are encountered.

*ALL — All uncleared media encountered during the save operation are cleared.

*AFTER — All uncleared tapes that are found after the first volume, and that are not already cleared, are cleared. If the operation cannot proceed because the first volume is uncleared, an inquiry message is sent to the system operator, who can end the operation or specify that the currently selected volume be cleared so the operation can continue. This value is not valid for save files.

PRECHK: Specifies whether the save operation for a library ends if any of the following are true:

1. The objects do not exist.
2. The library or objects were previously found to be damaged.
3. The library or objects are locked by another job.
4. The requester of the save does not have authority to the library or objects.

The possible values are:

***NO** — The save continues, only saving only those objects that can be saved.

*YES — If, after all specified objects are checked, one or more objects cannot be saved, the save operation for a library ends before any data is written. If multiple libraries are specified, the save operation continues with the next library. However, if **PRECHK(*YES)** and **SAVACT(*SYNCLIB)** are specified and an object in any library to be saved does not meet the preliminary check conditions, the save operation ends and no objects are saved.

SAVACT: Specifies whether different object types can be updated while being saved.

Object link: Specifies whether object links can be updated while being saved.

Note: If your system is in a restricted state, this parameter is ignored and the save operation is performed as if **SAVACT(*NO)** was specified.

***NO** Objects that are in use are not saved. Objects cannot be updated while being saved.

*YES Objects can be saved and used at the same time. The object checkpoints can occur at different times.

*SYNC Objects can be saved and used at the same time. The object checkpoints occur at the same time.

Folder: Specifies whether folders can be updated while being saved.

***NO** Document library objects in use are not saved. Document library objects cannot be updated while being saved.

*YES Document library objects can be changed during the save request.

Library: Specifies whether library objects can be updated while being saved.

Note: If your system is in a restricted state and the **SAVACT** parameter is specified, the save operation is performed as if **SAVACT(*NO)** was specified.

***NO** Objects in use are not saved. Objects cannot be updated while being saved.

*LIB Objects in a library can be saved while they are in use by another job. All of the objects in a library reach a checkpoint together and are saved in a consistent state in relationship to each other.

Note: Libraries with thousands of objects may be too large for this option.

*SYNCLIB Objects in a library can be saved while they are in use by another job. All of the objects and all of the libraries in the save operation reach a checkpoint together and are saved in a consistent state in relationship to each other.

*SYSDFN Objects in a library can be saved while they are in use by another job. Objects in a library may reach checkpoints at different times and may not be in a consistent state in relationship to each other.

Note: Specifying this value eliminates some size restrictions and may enable a library to be saved that could not be saved with **SAVACT(*LIB)**.

Object: Specifies whether changed objects can be updated while being saved.

Note: If your system is in a restricted state and the **SAVACT** parameter is specified, the save operation is performed as if **SAVACT(*NO)** was specified.

***NO** Objects in use are not saved. Objects cannot be updated while being saved.

*LIB Objects in a library can be saved while they are in use by another job. All of the objects in a library reach a checkpoint together and are saved in a consistent state in relationship to each other.

*SYNCLIB Objects in a library can be saved while they are in use by another job. All of the objects and all of the libraries in the save operation reach a checkpoint together and are saved in a consistent state in relationship to each other.

*SYSDFN Objects in a library can be saved while they are in use by another job. Objects in a library may reach checkpoints at different times and may not be in a consistent state in relationship to each other.

Changed Object: Specifies whether changed objects can be updated while being saved.

Note: If your system is in a restricted state and the **SAVACT** parameter is specified, the save operation is performed as if **SAVACT(*NO)** was specified.

***NO** Objects in use are not saved. Objects cannot be updated while being saved.

*LIB Objects in a library can be saved while they are in use by another job. All of the objects in a library reach a checkpoint together and are saved in a consistent state in relationship to each other.

*SYNCLIB Objects in a library can be saved while they are in use by another job. All of the objects and all of the libraries in the save operation reach a checkpoint together and are saved in a consistent state in relationship to each other.

*SYSDFN Objects in a library can be saved while they are in use by another job. Objects in a library may reach checkpoints at different times and may not be in a consistent state in relationship to each other.

SAVACTWAIT: Specifies the amount of time to wait for a commit boundary or a lock on an object, if it is not available, before continuing the save. If a lock is not obtained in the specified time, the object is not saved. If a commit boundary is not reached in the specified time, the save operation is ended.

120 The system waits up to 120 seconds for a commit boundary or an object lock before continuing the save operation.

*NOMAX No maximum wait time exists.

Wait-time Enter the time (in seconds) to wait for a commit boundary or an object lock before continuing the save operation. Valid values range from 0 through 99999.

SAVACTMSGQ: Specifies the message queue that the save operation uses to notify the user that the checkpoint processing for the library is complete. A separate message is sent for each library to be saved when the ***SYSDFN** or ***LIB** value is specified on the Save active prompt (**SAVACT** parameter). When the ***SYNCLIB** value is specified on the Save active prompt (**SAVACT** parameter), one message is sent for all libraries in the save operation.

This parameter can be used to save the objects at a known, consistent boundary to avoid additional recovery procedures following a restore operation. Applications can be stopped until the checkpoint processing complete message is received.

Message queue:

***NONE**	No notification message is sent.
Message-queue	Enter the name of a message queue.

Library:

Library-name	Enter a valid library name.

ACCPTH: Specifies whether the logical file access paths that are dependent on the physical files being saved are also saved. The access paths are saved only in the case of the following:

All members on which the access paths are built are included in this save operation.

The access paths are not invalid or damaged at the time of the save. The system checks to ensure the integrity of the access paths. Any discrepancies found by the system will result in the access paths being rebuilt.

Informational messages are sent indicating the number of logical file access paths saved with each physical file. All physical files on which an access path is built must be in the same library. This parameter does not save logical file objects; it only controls the saving of the access paths. More information on the restoring of saved access paths is in the Backup and Recovery book, SC41-5304.

* * * **Attention** * * *

* * * If the based-on physical files and the logical files are in different libraries, the access paths are saved.

* * * However, if the logical files and the based-on physical files are in different libraries and the logical files or physical files do not exist at restore time (such as during disaster recovery or the files were deleted) the access paths are not restored. They are rebuilt.

* * * For the fastest possible restore operation for logical files, the logical files and the based-on physical files must be in the same library and must be saved at the same time.

The possible values are:

***NO**	Only those objects specified on the command are saved. No logical access paths are saved.
*YES	The specified physical files and all eligible access paths built over them are saved. **Note**: Specifying this value does not save the logical files.

SAVFDTA: Specifies, for save file objects, whether the description of a save file, or both the description and the contents of a save file, are saved on the tape or in another save file.

***YES**	The description and the contents of the save file are saved.
*NO	Only the description of the save file is saved.

DTACPR: Specifies whether data compression is used.

***DEV** If the save is to tape and the target device supports compression, hardware compression is performed. Otherwise, no data compression is performed.

Note: If ***DEV** is specified on both the Data compression prompt (**DTACPR** parameter) and the Data compaction prompt (**COMPACT** parameter), only device data compaction is performed if device data compaction is supported on the device. Otherwise, data compression is performed.

If ***YES** is specified on the Data compression prompt (**DTACPR** parameter) and ***DEV** is specified on the Data compaction prompt (**COMPACT** parameter), both device data compaction and device data compression are performed.

*NO No data compression is performed.

*YES If the save is to tape and the target device supports compression, hardware compression is performed. If compression is not supported, or if the save data is written to a save file, software compression is performed. If the save is running while other jobs on the system are active and software compression is used, the overall system performance may be affected.

COMPACT: Specifies whether device data compaction is used.

***DEV** Device data compaction is performed if the data is saved to tape and all tape devices specified on the Device prompt (**DEV** parameter) support the compaction feature.

Note: If ***DEV** is specified on both the Data compression prompt (**DTACPR** parameter) and the Data compaction prompt (**COMPACT** parameter), only device data compaction is performed if device data compaction is supported on the device. Otherwise, data compression is performed.

If ***YES** is specified on the Data compression prompt (**DTACPR** parameter) and ***DEV** is specified on the Data compaction prompt (**COMPACT** parameter), both device data compaction and device data compression are performed.

*NO No device data compaction is performed.

OUTPUT: Specifies whether the output from the command is displayed at the requesting workstation or printed with the job's spooled output.

***NONE**	No output is created.
*PRINT	The output is printed with the job's spooled output.

TEXT: Specifies the text that briefly describes the object.

***BLANK**	No text is specified.
'description'	Enter no more than 50 characters of text, enclosed in apostrophes.

PREEXIT: Specifies the qualified name of the pre-exit program to call before the process starts.

Exit Program:

***NONE**	No pre-exit program is selected.
Exit-program	Enter the name of a pre-exit program.

Library:

Library-name	Enter a valid library name.

POSTEXIT: Specifies the qualified name of the post-exit program to call after the process completes.

Exit Program:

***NONE**	No post-exit program is selected.
Exit-program	Enter the name of a post -exit program.

Library:

Library-name	Enter a valid library name.

Examples

```
ADDBKUPDFN BKUPDFN(DAILY) DEVICE(TAPMLB01) ENDOPT(*UNLOAD) +
    TEXT('Daily Backup Definition') POSTEXIT(PRODBKUP/POSTPGM)
```

This adds a Backup Definition named DAILY, which uses a device named TAPMLB01. When the Backup completes, the tape is unloaded and the post exit program POSTPGM in library PRDBKUP is called.

ADDBKUPL - Add Backup List

```
                    Add Backup List  (ADDBKUPL)          Environment: B/I

Backup . . . . . .  _________                            Name
Sequence number . . . . . .  ____                        1-9999
List . . . . . .  _________                              *ASP, *ASPDLO, *CFG…
Type . . . . . .  *FULL                                  *FULL, *INCR, *CUML
Parallel device resources:
  Minimum resources . . . . . .  *NONE                   1-32, *NONE, *AVAIL
  Maximum resources . . . . . .  _____                   1-32, *AVAIL, *MIN
ASP Device . . . . . .  *                                Name, *, *SYSBAS...
Omit list name . . . . . .  *NONE                        Name, *NONE
Error action . . . . . .  *IGNORE                        *IGNORE, *CANCEL
Text . . . . . .  *BLANK                                 Char, *BLANK
```

Purpose

The Add Backup List (ADDBKUPL) command adds a Backup List to a Backup. Backup Lists define the type and sequence of the backup to perform.

Parameters

BACKUP: Specifies the name of the Backup that is associated with this Backup List.

Backup-name Enter a valid Backup name.

SEQNBR: Specifies the sequence number of this Backup List. The sequence number determines when to perform the Backup defined by this Backup List. The Backup List entry is not added if an entry already exists at the specified sequence number.

Sequence-number Enter a number from 1-9999.

LIST: Specifies the type of Backup List.

*ASP	Auxiliary storage pool (**ASP**) list.
*ASPDLO	Document library object auxiliary storage pool (**ASP**) list.
*CFG	Configuration list.
*CLT	Client list.
*EJECT	Tape eject.
*EXIT	Command exit.
*FLR	Document library object list.
*LIB	Library list.
*LND	Domino server list.
*LNK	Integrated file system object list.
*OBJ	Object list.
*OUTQ	Output queue list.
*RCY	MMS/*tms* recovery library list.
*SAVF	Save file list.
*SEC	Security information.
*SPL	Spooled file list.
*SYS	All Licensed Internal Code; the QSYS library; security and configuration objects.

TYPE: Specifies the type of save to process.

***FULL**	A full backup is performed.
*INCR	An incremental backup is performed. This value is valid for ***ASP**, ***ASPDLO**, ***CLT**, ***FLR**, ***OBJ**, ***LNK**, ***OUTQ** and ***SPL** Backup Lists.
*CUML	A cumulative backup is performed. This value is valid for ***ASP**, ***ASPDLO**, ***FLR**, ***OBJ** and ***LNK** Backup Lists.

DRVRSC: Specifies the minimum and maximum number of device resources to use in a parallel save. This parameter is valid for ***ASP**, ***LIB** and ***OBJ** Backup Lists.

Minimum resources:

***NONE**	No device resources are used. The save is performed as a serial save.
*AVAIL	Use any available resources up to the maximum specified. This will use any available resource but will complete using one resource if only one is available.
1-32	Enter the minimum number of resources to use.

Maximum resources:

*MIN	Uses the value specified for the minimum number of device resources.
*AVAIL	The save will use any available device resources but at minimum, the value specified in the minimum element.
1-32	Enter the minimum number of resources to use.

ASPDEV: Specifies the name of the auxiliary storage pool (ASP) device to be included in the backup operation. This parameter is valid for ***LIB**, ***OBJ** and ***RCY** Backup Lists.

*****	The operation includes the system ASP (ASP number 1), all basic user ASPs (ASP numbers 2 to 32) and, if the current thread has an ASP group, all independent ASPs in the ASP group.
*SYSBAS	The system ASP (ASP number 1) and all basic user ASPs (ASP numbers 2 to 32) are included in the backup operation.
*CURASPGRP	If the current thread has an ASP group, all independent ASPs in the ASP group are included in the backup operation.
*ALLAVL	The private authorities from the system ASP (ASP number 1), all basic user ASPs (ASP numbers 2 to 32) and all independent ASPs are saved.
ASP-device-name	Enter a valid independent ASP.

OMITL: Specifies the name of the Omit List. This parameter is valid for ***ASP**, ***ASPDLO**, ***CLT**, ***FLR**, ***LIB**, ***LND**, ***LNK**, ***OBJ**, ***OUTQ** and ***SPL** Backup Lists.

***NONE**	No Omit List is selected.
Omit-list	Enter a valid Omit List name.

ERROR: Specifies the action to take if a severe error occurs. A severe error occurs when an escape message is issued in a Backup List entry.

***IGNORE**	The process continues.
*CANCEL	The process stops.

TEXT: Specifies the text that briefly describes the object.

***BLANK**	No text is specified.

'description' Enter no more than 50 characters of text, enclosed in apostrophes.

Examples

```
ADDBKUPL BACKUP(DAILY) SEQNBR(10) LIST(*ALLUSR) TYPE(*FULL) +
    TEXT('Full Backup of all User Libraries')
```

This adds Backup List sequence number 10 to a Backup named DAILY. The Backup List does a full save of all user libraries.

ADDBKUPLE - Add Backup List Entry

```
                        Add Backup List Entry (ADDBKUPLE)   Environment: B/I

Backup . . . . . .   _________          Name
Sequence number . . . . . .   ____       1-9999
List . . . . . .   _________             *ASP, *ASPDLO, *FLR…
Order number . . . . . .   *END          *END, 1-9999
ASP device . . . . . .   __              1-16, *SYSTEM
Text . . . . . .   *BLANK                Char, *BLANK
Object . . . . . .   _________          Name, generic*, *ALL
  Library . . . . . .   _________       Name, generic*, *ALLPROD…
Object type . . . . . .   ________       *ALL, *ALRTBL, *BNDDIR…
Member . . . . . .   _________          Name, generic*, *ALL
            + for more values   _________
Client . . . . . .   ____________________ Char
Domino server . . . . . .   ____________________ Path name, *ALL, *HTTPSETUP
Path name . . . . . .   *ALL             Path name, *ALL
Directory subtree . . . . . .   *ALL      *ALL. *DIR, *NONE, *OBJ
Output queue . . . . . .   _________     Name, *ALL
  Library . . . . . .   _________       Name, *ALL
Spooled file . . . . . .   *ALL          Name, generic*, *ALL
Delete spooled files . . . . . .   *NO    *NO, *YES
User profile . . . . . .   *ALL          Name, *ALL, *CURRENT
User data . . . . . .   *ALL             User data, *ALL
Form type . . . . . .   *ALL             Form type, *ALL, *STD
Command . . . . . .   ________________   Char
```

Purpose

The Add Backup List Entry (ADDBKUPLE) command adds one or more entries to a Backup List. Backup List entries can be libraries, objects, folders, links, output queues, save file data, spooled files and Domino servers.

Parameters

BACKUP: Specifies the name of the Backup that is associated with this Backup List entry.

Backup-name Enter a valid Backup name.

SEQNBR: Specifies the sequence number of this Backup List entry.

Sequence-number Enter a number from 1-9999.

LIST: Specifies the type of Backup List.

*ASP	Auxiliary storage pool (ASP) list.
*ASPDLO	Document library objects in the specified auxiliary storage pool (ASP) list.
*CLT	Client list.
*EXIT	Command list.
*FLR	Document library object list.
*LIB	Library list.
*LND	Lotus Notes/Domino server list.
*LNK	Integrated file system list.
*OBJ	Objects list.
*OUTQ	Output queue list.
*SAVF	Save file list.
*SPL	Spooled file list.

ORDNBR: Specifies the order number of the list entry to add. If multiple entries have the same order number, the entries are sorted alphabetically within order number.

***END**	The entry is added to the end of the backup list.
Order-number	Enter a value from 1-9999.

ASP: Specifies the auxiliary storage pool identifier to add to a ***ASP** or ***ASPDLO** Backup List.

*SYSTEM	The objects reside on the system ASP.
ASP-number	Enter a number from 1-16.

TEXT: Specifies the text that briefly describes the object.

***BLANK**	No text is specified.
'description'	Enter no more than 50 characters of text, enclosed in apostrophes.

OBJ: Specifies the name of the objects to add to a ***OBJ** or ***SAVF** Backup List.

*ALL	All objects from the specified library are selected.
generic*	Enter the generic name of the objects to select. A generic name is a character string that contains one or more characters followed by an asterisk (*).
Object-name	Enter a valid object name.

LIB: Specifies the name or type of libraries to add to a ***LIB** Backup List.

*ALLPROD	All production libraries are selected.
*ALLTEST	All test libraries are selected.
*ALLUSR	All user libraries are selected.
*IBM	All IBM libraries are selected.
*NEW	All new (never saved) libraries are selected.
*NONSYS	All non-system libraries are selected.
generic*	Enter the generic name of the libraries to select. A generic name is a character string that contains one or more characters followed by an asterisk (*).
Library-name	Enter a valid library name.

OBJTYPE: Specifies which type of objects to add to a ***OBJ** Backup List.

*ALL	All object types are selected.
Object-type	Enter a valid object type.

MBR: Specifies which member(s) to add to a ***OBJ** Backup List.

*ALL	All database members are selected.
generic*	Enter the generic name of the objects to select. A generic name is a character string that contains one or more characters followed by an asterisk (*).
Member-name	Enter a valid object type.

CLIENT: Specifies the name of the client to add to a ***CLT** Backup List.

Client-host	Enter the client host name.

DOMSVR: Specifies the name of the server to add to a ***LND** Backup List.

*ALL	All Domino servers are selected.
*HTTPSETUP	The HTTPSETUP server is selected.
Domino-server	Enter the Domino server name.

PATH: Specifies the path name to add to a ***CLT**, ***FLR**, ***LND** or ***LNK** Backup List.

***ALL**	All path names associated with the Backup List type are selected.
Link-name	Enter a valid path name.

SUBTREE: Specifies whether the directory subtrees from ***LND** or ***LNK** Backup Lists are included in the save.

***ALL**	The entire subtree of each directory is included. This value is valid for ***LND** and ***LNK** lists.
*DIR	The objects in the first level of each directory that matches the object name pattern is included. This value is valid for ***LNK** lists.
*NONE	No subtrees are included. This value is valid for ***LNK** lists.
*OBJ	Only the objects that exactly match the object name pattern are included. This value is valid for ***LND** and ***LNK** lists.

OUTQ: Specifies the output queue to add to a ***OUTQ** or ***SPL** Backup List.

Output queue:

*ALL	All output queues are selected.
Output-queue	Enter the name of a valid output queue.

Library:

Library-name	Enter a valid library name.

FILE: Specifies the name of the spooled file for a ***SPL** Backup List.

***ALL**	All spooled files are selected.
generic*	Enter the generic name of the spooled files to select. A generic name is a character string that contains one or more characters followed by an asterisk (*).
Spooled-file	Enter the name of a valid spooled file name.

DLTSPLF: Specifies whether spooled files are deleted after the save completes. This parameter applies to either a ***OUTQ** or ***SPL** Backup List.

***NO**	The saved spooled files are not deleted.
*YES	The saved spooled files are deleted.

USRPRF: Specifies the name of the user profile that created the spooled files. This parameter applies to a ***SPL** Backup List.

***ALL**	Spooled files created by all users are selected.
*CURRENT	Spooled files created by the current user are selected.
User-profile	Enter a valid user profile name.

USRDTA: Specifies the user data of the spooled files. This parameter applies to a ***SPL** Backup List.

***ALL**	All user data is selected.
User-data	Enter the user data of the spooled file.

FORMTYPE: Specifies the form type of the spooled files. This parameter applies to a ***SPL** Backup List.

***ALL**	All form types are selected.
*STD	Standard form types are selected.
Form-type	Enter the form type of the spooled file.

COMMAND: Specifies the command to execute. This parameter applies to an ***EXIT** Backup List.

command	Enter the command to execute. The command is validated prior to being added.

Examples

```
ADDBKUPLE BACKUP(DAILY) SEQNBR(5) OBJ(PRODLIB/*ALL) +
    OBJTYPE(*ALL)
```

This adds a Backup List entry for a Backup named DAILY. The Backup List entry specifies that all objects from library PRODLIB are saved.

ADDOMITL - Add Omit List

```
-------------------------------------------------------------------------------------------------------
                    Add Omit List  (ADDOMITL)              Environment: B/I

Omit list name . . . . . . . . . . . . . .   _________      Name
List . . . . . . . . . . . . . . . . . . . .   _________      *CLT, *FLR, *LIB, *LND…
Text . . . . . . . . . . . . . . . . . . . .   *BLANK         Char, *BLANK

-------------------------------------------------------------------------------------------------------
```

Purpose

The Add Omit List (ADDOMITL) command creates an Omit List. Omit Lists define the name and type of object to omit from a Backup.

Parameters

OMITL: Specifies the name of the Omit List.

List-name	Enter an Omit List name.

LIST: Specifies the type of Omit List.

*CLT	Client object Omit List.
*FLR	Document library object Omit List.
*LIB	Library Omit List.
*LND	Domino server Omit List.
*LNK	Integrated file system Omit List.
*OBJ	Object Omit List.
*OUTQ	Output queue Omit List. MMS/spl must be installed to use this value.
*SYS	System data Omit List.

TEXT: Specifies the text that briefly describes the object.

***BLANK**	No text is specified.
'description'	Enter no more than 50 characters of text, enclosed in apostrophes.

Examples

```
ADDOMITL OMITL(DAILY) LIST(*OBJ) TEXT('Object Omit List')
```

This adds an Omit List named DAILY.

ADDOMITLE - Add Omit List Entry

```
------------------------------------------------------------------------------------------------------------
                    Add Omit List Entry (ADDOMITLE)          Environment: B/I

Omit list name . . . . . . . . . . . . . . .  __________      Name
List . . . . . . . . . . . . . . . . . . . .  __________      *CLT, *FLR, *LIB, *LND…
Object . . . . . . . . . . . . . . . . . . .  __________      Name, generic*, *ALL
Library . . . . . . . . . . . . . . . . . .     __________    Name, generic*
Object type . . . . . . . . . . . . . . . .  ________         *ALL, *LIB, *ALRTBL…
Object link . . . . . . . . . . . . . . . .  ____________________  Path name
Output queue . . . . . . . . . . . . . . . .  __________      Name
  Library . . . . . . . . . . . . . . . . .     __________    Name
Path name . . . . . . . . . . . . . . . . .  ____________________  Path name
System data . . . . . . . . . . . . . . . .  __________      *CFG, *SECDTA

------------------------------------------------------------------------------------------------------------
```

Purpose

The Add Omit List Entry (ADDOMITLE) command adds one or more items to an Omit List. Omit Lists are used to omit specific items from a Backup.

Parameters

OMITL: Specifies the name of the Omit List.

List-name	Enter an Omit List name.

LIST: Specifies the type of Omit List.

*CLT	The list is a client object Omit List.
*FLR	The list is a document library object Omit List.
*LIB	The list is a library Omit List.
*LND	The list is a Domino server Omit List.
*LNK	The list is a integrated file system object Omit List.
*OBJ	The list is an object Omit List.
*OUTQ	The list is an output queue Omit List.
*SYS	The list is a system data Omit List.

OBJ: Specifies the name of the objects to add to a ***OBJ** Omit List.

*ALL	All objects from the specified library are omitted.
generic*	Enter the generic name of the objects to omit. A generic name is a character string that contains one or more characters followed by an asterisk (*).
Object-name	Enter a valid object name.

LIB: Specifies the name of the libraries to add to a ***LIB** Omit List.

generic*	Enter the generic name of the libraries to omit. A generic name is a character string that contains one or more characters followed by an asterisk (*).
Library-name	Enter a valid library name.

OBJTYPE: Specifies which types of objects to add to a ***OBJ** Omit List.

*ALL	All object types are omitted.
Object-type	Enter a valid object type.

OUTQ: Specifies the output queue to add to a ***OUTQ** Omit List.

Output queue:

Output-queue	Enter the name of a valid output queue.

Library:

Library-name	Enter a valid library name.

PATH: Specifies the name of the path to add to either a ***CLT**, ***FLR**, ***LND** or ***LNK** Omit List.

Path-name	Enter a valid path name.

SYSDTA: Specifies the type of system data to add to a ***SYS** Omit List.

*CFG	Configuration data is omitted.
*SECDTA	Security data is omitted.

Examples

```
ADDOMITLE OMITL(DAILY) LIST(*OBJ) OBJ(PRODLIB/*ALL) +
    OBJTYPE(*ALL)
```

This adds an Omit List entry for a Backup named DAILY. The Omit List entry specifies that all objects from library PRODLIB are to be omitted.

ADDRCY - Add Recovery

```
---------------------------------------------------------------------------------------------------
                    Add Recovery (ADDRCY)                  Environment: B/I

Recovery . . . . . . . . . . . . . . . . . .  _________    Name, *AUTO
Recovery definition . . . . . . . . . .  *DFT              Name, *DFT
Text . . . . . . . . . . . . . . . . . . . .  *BLANK       Char, *BLANK

---------------------------------------------------------------------------------------------------
```

Purpose

The Add Recovery (ADDRCY) command creates a Recovery and associates a Recovery Definition with the Recovery.

Parameters

RCY: Specifies the name of the Recovery.

*AUTO	The Intelligent Recovery is being added. Recovery Lists, Recovery List entries and Omit Lists are not allowed with this type of restore.
Recovery-name	Enter a name for the Recovery.

RCYDFN: Specifies the Recovery Definition to associate with this Recovery.

***DFT**	The default Recovery Definition is selected.
Recovery-definition	Enter a valid Recovery Definition.

TEXT: Specifies the text that briefly describes the object.

***BLANK**	No text is specified.
'description'	Enter no more than 50 characters of text, enclosed in apostrophes.

Examples

```
ADDRCY RCY(NONSYS) RCYDFN(*DFT) TEXT('NONSYS Recovery')
```

This adds a Recovery named NONSYS, which uses the default Recovery Definition.

ADDRCYDFN - Add Recovery Definition

Add Recovery Definition (ADDRCYDFN) Environment: B/I

Recovery definition	__________	Name
Library:		
Option	*ALL	*All, *NEW, *OLD, *FREE
Data base member option	*MATCH	*MATCH, *ALL, *NEW, *OLD
Allow object differences	*NONE	*NONE, *ALL, *FILELVL
Auxiliary storage pool ID	*SAVASP	1-32, *SAVASP
Object:		
Option	*ALL	*All, *NEW, *OLD, *FREE
Data base member option	*MATCH	*MATCH, *ALL, *NEW, *OLD
Allow object differences	*NONE	*NONE, *ALL, *FILELVL
Auxiliary storage pool ID	*SAVASP	1-32, *SAVASP
Folder:		
Object name generation	*SAME	*SAME, *NEW
Allow object differences	*NONE	*NONE, *ALL
Saved from ASP.	*ANY	1-32, *ANY
Restore to ASP.	*SAVASP	1-32, *SAVASP
Object link:		
Option	*ALL	*All, *NEW, *OLD
Allow object differences	*NONE	*NONE, *ALL, *AUTL…
+ for more values	____	
Configuration:		
System Resource Management .	*ALL	*ALL, *NONE, *HDW, *TRA
Allow object differences	*NONE	*ALL, *NONE
Text 'description'	*BLANK	Char, *BLANK

Purpose

The Add Recovery Definition (ADDRCYDFN) command creates a Recovery Definition, which determines the attributes to use for the Recovery.

Parameters

RCYDFN: Specifies the name of the Recovery Definition.

Recovery-definition Enter a name for the Recovery Definition.

LIB: Specifies the restore options used when restoring libraries.

Option	Specifies how to handle restoring each object.
***ALL**	All the objects in the saved library are restored to the library. Objects in the saved library replace the current versions of the system. Objects not having a current version are added. Objects presently in the library, but not on the media, remain in the library.
*NEW	Only the objects in the saved library that do not exist in the current version of the system library are added to the library. Only objects not known to the system library are restored; known objects are not restored. This option restores objects that were deleted after they were saved or that are new to this library. If any saved objects have a version already in the system library, they are not restored, and an informational message is sent for each one, but the restore operation continues.
. *OLD	Only the objects in the library having a saved version are restored; that is, the version of each object currently in the library is replaced by the saved version. Only objects known to the library are restored. If any saved objects are no longer part of the online version of the library, they are not added to the library; an informational message is sent for each one, but the restore continues.
*FREE	The saved objects are restored only if they exist in the system library with their space freed. The saved version of each object is restored on the system in its previously freed space. This option restores objects that had their space freed when they were saved. If any saved objects are no longer part of the current version of the library, or if the space is not free for any object, the object is not restored and an informational message is sent for each one. The restore operation continues, and all of the freed objects are restored.

dB member option: Specifies, for database files that exist on the system, which members are restored. If ***MATCH** is used, the member list in the saved file must match, member for member, the current version on the system. All members are restored for files that do not exist, if the file is restored.

***MATCH** The saved members are restored if the lists of the members where they exist match, member for member, the lists of the current system version. **MBROPT(*MATCH)** is not valid when ***ALL** is specified on the Allow object differences parameter.

*ALL All members in the saved file are restored.

*NEW Only new members (members not known to the system) are restored.

*OLD Only members known to the system are restored.

Allow obj. diff: Specifies whether certain differences encountered during a restore operation are allowed. The differences include:

- **Ownership** -- the owner of the object on the system is different than the owner of the object from the save operation.
- **File creation date** -- the creation date of the database file on the system does not match the creation date of the file that was saved.
- **Member creation date** -- the creation date of the database file member on the system does not match the creation date of the member that was saved.
- **Validation value verification** -- the validation value created at the time an object was created does not match the validation value created during the restore operation of an object on a system with a **QSECURITY** level of **40** or higher.
- **Authorization list linking** -- the object is being restored to a system different from the one on which it was saved.

Note*: To use this parameter, you need* ****ALLOBJ*** *special authority.*

***NONE** None of the differences described above are allowed on the restore operation. For validation value verification failure cases, the object is restored but ownership is transferred to **QDFTOWN** and all authorities are revoked. For authorization list cases, the objects is restored, but the object is not linked to the authorization list, and public authority is set to ***EXCLUDE**. For all other cases, a diagnostic message is sent for the object, and the object is not restored.

*ALL All of the differences listed above are allowed for the restore operation. An informational message is sent, except for validation value verification and authorization list linking cases, and the object is restored. The following should be noted:

- If object differences are found, the final message for the restore operation is an escape message rather than the normal completion message.
- If the media and system owner of the object do not match, the system owner becomes the owner of the object.
- If there is a file level mismatch and ***ALL** is specified on this parameter and the Data base member option prompt, the existing version of the file is renamed and the saved version of the file is restored. If there is a member level mismatch, the existing version of the member is renamed and the saved version of the member is restored.

- If the system security level is **40**, you are restoring a program, you specify ***ALL**, and the program's validation value is missing or incorrect, the program is restored without authority changes. For programs without a validation value, specifying ***ALL** also prevents the system from attempting to translate the program again.

- If you are restoring objects to a system different from the one on which they were saved and the objects are secured by an authorization list, specifying ***ALL** automatically links the objects to the authorization list again. If the authorization list does not exist on the new system, a message that includes the name of the missing list is issued and the public authority is set to ***EXCLUDE**.

ASP ID: Specifies whether objects are restored to the auxiliary storage pool (ASP) from which they were saved or to another ASP. ASP 1 is the system ASP. Libraries and their contained objects may be restored to user ASPs (2 through 32). However, some objects cannot be restored to user ASPs.

More information about object types that can be restored to user ASPs is in the Backup and Recovery book, SC41-5304. If the library exists in, or is being restored to the system ASP, journals, journal receivers, and save files can be restored to user ASPs. All other object types will be restored to the ASP of the library.

Attention * * * System or product libraries (libraries that begin with a Q or #) must not be created in or restored to a user ASP. Doing so can cause unpredictable results. *** * ***

***SAVASP** The objects are restored to the ASP from which they were saved.

ASP-ID Specifies the ASP identifier. When the specified ASP is 1, the specified objects are restored to the system ASP, and when the specified ASP is 2 through 32, the objects are restored to the user ASP specified.

OBJ: Specifies the restore options used when restoring objects.

Option	Specifies how to handle restoring each object.
***ALL**	All the objects in the saved library are restored to the library. Objects in the saved library replace the current versions of the system. Objects not having a current version are added. Objects presently in the library, but not on the media, remain in the library.
*NEW	Only the objects in the saved library that do not exist in the current version of the system library are added to the library. Only objects not known to the system library are restored; known objects are not restored. This option restores objects that were deleted after they were saved or that are new to this library. If any saved objects have a version already in the system library, they are not restored, and an informational message is sent for each one, but the restore operation continues.
. *OLD	Only the objects in the library having a saved version are restored; that is, the version of each object currently in the library is replaced by the saved version. Only objects known to the library are restored. If any saved objects are no longer part of the online version of the library, they are not added to the library; an informational message is sent for each one, but the restore continues.
*FREE	The saved objects are restored only if they exist in the system library with their space freed. The saved version of each object is restored on the system in its previously freed space. This option restores objects that had their space freed when they were saved. If any saved objects are no longer part of the current version of the library, or if the space is not free for any object, the object is not restored and an informational message is sent for each one. The restore operation continues, and all of the freed objects are restored.

dB mbr option: Specifies, for database files that exist on the system, which members are restored. If ***MATCH** is used, the member list in the saved file must match, member for member, the current version on the system. All members are restored for files that do not exist, if the file is restored.

***MATCH** The saved members are restored if the lists of the members where they exist match, member for member, the lists of the current system version. **MBROPT(*MATCH)** is not valid when ***ALL** is specified on the Allow object differences parameter.

*ALL All members in the saved file are restored.

*NEW Only new members (members not known to the system) are restored.

*OLD Only members known to the system are restored.

Allow obj. diff: Specifies whether certain differences encountered during a restore operation are allowed. The differences include:

- **Ownership** -- the owner of the object on the system is different than the owner of the object from the save operation.
- **File creation date** -- the creation date of the database file on the system does not match the creation date of the file that was saved.
- **Member creation date** -- the creation date of the database file member on the system does not match the creation date of the member that was saved.
- **Validation value verification** -- the validation value created at the time an object was created does not match the validation value created during the restore operation of an object on a system with a **QSECURITY** level of **40** or higher.
- **Authorization list linking** -- the object is being restored to a system different from the one on which it was saved.

Note*: To use this parameter, you need* ****ALLOBJ*** *special authority.*

***NONE** None of the differences described above are allowed on the restore operation. For validation value verification failure cases, the object is restored but ownership is transferred to **QDFTOWN** and all authorities are revoked. For authorization list cases, the objects is restored, but the object is not linked to the authorization list, and public authority is set to ***EXCLUDE**. For all other cases, a diagnostic message is sent for the object, and the object is not restored.

*ALL All of the differences listed above are allowed for the restore operation. An informational message is sent, except for validation value verification and authorization list linking cases, and the object is restored. The following should be noted:

- If object differences are found, the final message for the restore operation is an escape message rather than the normal completion message.
- If the media and system owner of the object do not match, the system owner becomes the owner of the object.

- If there is a file level mismatch and ***ALL** is specified on this parameter and the Data base member option prompt, the existing version of the file is renamed and the saved version of the file is restored. If there is a member level mismatch, the existing version of the member is renamed and the saved version of the member is restored.

- If the system security level is **40**, you are restoring a program, you specify ***ALL**, and the program's validation value is missing or incorrect, the program is restored without authority changes. For programs without a validation value, specifying ***ALL** also prevents the system from attempting to translate the program again.

- If you are restoring objects to a system different from the one on which they were saved and the objects are secured by an authorization list, specifying ***ALL** automatically links the objects to the authorization list again. If the authorization list does not exist on the new system, a message that includes the name of the missing list is issued and the public authority is set to ***EXCLUDE**.

ASP ID:

Specifies whether objects are restored to the auxiliary storage pool (ASP) from which they were saved or to another ASP. ASP 1 is the system ASP. Libraries and their contained objects may be restored to user ASPs (2 through 32). However, some objects cannot be restored to user ASPs.

More information about object types that can be restored to user ASPs is in the Backup and Recovery book, SC41-5304. If the library exists in, or is being restored to the system ASP, journals, journal receivers, and save files can be restored to user ASPs. All other object types will be restored to the ASP of the library.

Attention * * *
System or product libraries (libraries that begin with a Q or #) must not be created in or restored to a user ASP. Doing so can cause unpredictable results. *** * ***

***SAVASP** — The objects are restored to the ASP from which they were saved.

ASP-ID — Specifies the ASP identifier. When the specified ASP is 1, the specified objects are restored to the system ASP, and when the specified ASP is 2 through 32, the objects are restored to the user ASP specified.

FLR: Specifies the restore options used when restoring document library objects.

Object name gen. Specifies whether a new library-assigned name and system object name are generated for the folders and documents being restored.

***SAME** The library-assigned name and the system object name do not change.

*NEW A new library-assigned name and system object name are generated for each document or folder being restored.

Allow obj. diff: Specifies whether certain differences encountered during a restore operation are allowed. The differences include:

- **Ownership** -- the owner of the object on the system is different than the owner of the object from the save operation.
- **File creation date** -- the creation date of the database file on the system does not match the creation date of the file that was saved.
- **Member creation date** -- the creation date of the database file member on the system does not match the creation date of the member that was saved.
- **Validation value verification** -- the validation value created at the time an object was created does not match the validation value created during the restore operation of an object on a system with a **QSECURITY** level of **40** or higher.
- **Authorization list linking** -- the object is being restored to a system different from the one on which it was saved.

Note*: To use this parameter, you need* ****ALLOBJ*** *special authority.*

***NONE** None of the differences described above are allowed on the restore operation. For validation value verification failure cases, the object is restored but ownership is transferred to **QDFTOWN** and all authorities are revoked. For authorization list cases, the objects is restored, but the object is not linked to the authorization list, and public authority is set to ***EXCLUDE**. For all other cases, a diagnostic message is sent for the object, and the object is not restored.

*ALL All of the differences listed above are allowed for the restore operation. An informational message is sent, and the object is restored.

Notes*:*

- *If the owners of the object do not match, the object is restored, but it will keep the ownership and authorities of the object on the system before the restore operation.*
- *If* ****ALL*** *is specified on this parameter,* ****NEW*** *cannot be specified on the Object name generation parameter.*

- *If you are restoring objects to a system different from the one on which they were saved and the objects are secured by an authorization list, specifying ***ALL** automatically links the objects to the authorization list. If the authorization list does not exist on the new system, a message that includes the name of the missing list is issued.*

Saved from ASP: Specifies the identifier (ID) of the auxiliary storage pool (ASP) on media from which saved documents and folders are to be restored.

***ANY** The documents and folders saved in any ASP are restored.

Note: *When restoring Document Library Objects (DLOs) from more than one ASP, the sequence number (**SEQNBR**) parameter must be specified.*

ASP-ID Specify a value ranging from 1 through 32, which is the ID of the ASP from which documents and folders are restored.

Restore to ASP: Specifies the identifier (ID) of the auxiliary storage pool (ASP) on media in which restored documents and folders are to be placed.

***SAVSAP** The documents and folders are placed in the same ASP from which they were saved.

ASP-ID Specify a value ranging from 1 through 32, which is the ID of the ASP in which restored documents and folders are placed.

OBJLNK: Specifies the restore options used when integrated file system objects.

Option	Specifies how to handle restoring each object.
***ALL**	All of the specified objects are restored, whether they already exist on the system or not.
*NEW	Objects are restored only if they do not already exist on the system.
. *OLD	Objects are restored only if they already exist on the system.

Allow obj. diff: Specifies whether certain differences encountered during a restore operation are allowed. The differences include:

- **Ownership**: The owner of an object on the system is different than the owner of an object from the save operation.
- **Authorization list linking**: The system on which an object with an authorization list is being restored is different from the system on which it was saved.
- **Primary Group**: The primary group of an object on the system is different than the primary group of an object from the save operation.

***NONE** No differences are allowed between the saved object and the restored object. If the owner is different, the object is not restored. If the system is different for an object with an authorization list, the object is restored, but the object is not linked to its authorization list.

*ALL All differences are allowed between the saved object and the restored object. If the owner is different, the object is restored with the owner of the system on which it is restored. If the system is different for an object with an authorization list, the object is restored and linked to its authorization list.

*OWNER The object owner can be different. If an object already exists on the system with a different owner than the saved object, the object is restored with the owner of the object on the system. If owner differences are not allowed, the object is not restored.

*AUTL The system of an object with an authorization list can be different. The new object, which is being restored to a system that is different from which it was saved, is restored and linked to its authorization list. If the system of an object with an authorization list cannot be different, the object is restored but not linked to an authorization list.

*PGP The object primary group can be different. If an object already exists on the system with a different primary group than the saved object, the object is restored with the primary group of the object on the system. If primary group differences are not allowed, the object is not restored.

CFG: Specifies the parameter values to use when restoring configurations.

Sys. Resource Mgmt: Specifies the type of system resource management (SRM) information to be restored. This parameter is valid only when ***ALL** or ***SRM** is specified on the Objects prompt (CFG parameter).

Attention * * *

Unless the system you are restoring to has exactly the same hardware configuration as the system that the original configuration was saved on, you must specify SRM(*NONE) on this command to prevent the restore of the SRM information. If the SRM information is restore on a system with a different hardware configuration, the configuration objects may become unusable.

***ALL** — All system resource management information is restored.

*NONE — No system resource management information is restored.

*HDW — All hardware information is restored.

*TRA — All token-ring adapter information is restored.

Allow obj. diff: Specifies whether certain differences encountered during a restore operation are allowed. There are two differences allowed for this command:

- The owner of the object on the system is different than the owner of the object from the save.
- The object is secured by an authorization list and is being restored to a system other than the one on which it was saved.

Note: In order to use this parameter, you need ***ALLOBJ** authority.

***NONE** None of the differences previously described are allowed on the restore operation. For an ownership difference, the object is not restored. For an authorization list difference, the object is restored, but the object is not linked to the authorization list, and public authority is set to ***EXCLUDE**.

***ALL** All of the differences previously described are allowed for the restore operation. The object is restored. The following should be noted:

- If the media and system owners of the object do not match, the system owner becomes the owner of the object and an informational message is sent.
- The informational message triggers a diagnostic message to be sent indicating that security or integrity changes occurred during the restore operation. The final message for the restore operation is an escape message, rather than the normal completion message.
- If you are restoring objects to a system different from the one on which they were saved and the objects are secured by an authorization list, specifying ***ALL** automatically links the objects to the authorization list again. If the authorization list does not exist on the new system, a message that includes the name of the missing list will be used.

TEXT: Specifies the text that briefly describes the object.

***BLANK**	No text is specified.
'description'	Enter no more than 50 characters of text, enclosed in apostrophes.

Examples

```
ADDRCYDFN RCYDFN(NONSYS) TEXT('NONSYS Recovery')
```

This creates a Recovery Definition named NONSYS, which uses the default restore values for libraries, objects, document library objects and integrated file system objects.

ADDRCYL - Add Recovery List

```
------------------------------------------------------------------------------------------------------
                    Add Recovery List  (ADDRCYL)        Environment: B/I

Recovery . . . . . .  _________        Name
Sequence number . . . . . .  ____      1-9999
List . . . . . .  _________            *CFG, *EXIT, *FLR, *LIB…
Restore to ASP device . . . . . .  *SAVASPDEV   Name, *SAVASPDEV
Omit list name . . . . . .  *NONE      Name, *NONE
Error action . . . . . .  *IGNORE      *IGNORE, *CANCEL
Text . . . . . .  *BLANK               Char, *BLANK
------------------------------------------------------------------------------------------------------
```

Purpose

The Add Recovery List (ADDRCYL) command adds a Recovery List to a Recovery. Recovery Lists define the sequence of the recoveries to perform.

Parameters

RCY: Specifies the name of the Recovery that is associated with this Recovery List.

Recovery-name	Enter a valid Recovery List name.

SEQNBR: Specifies the sequence number of this Recovery List. The sequence number determines when to perform the recovery defined by this Recovery List. The Recovery List entry is not added if an entry already exists at the specified sequence number.

Sequence-number	Enter a number from 1-9999.

LIST: Specifies the Recovery List type.

*CFG	Configuration list.
*EXIT	Command exit.
*FLR	Document library object list.
*LIB	Library list.
*LNK	Integrated file system object list.
*OBJ	Object list.
*RCY	MMS recovery libraries list.
*SEC	Security data list.
*SYS	System list.

RSTASPDEV: Specifies the name of the auxiliary storage pool (ASP) device to which the data is restored. Specify either the RSTDEVASP parameter or the RSTASP parameter, which is defined in the Recovery Definition, but not both.

***SAVASPDEV**	The data is restored to the same ASP from which it was saved.
ASP-device-name	Enter a valid ASP device name.

OMITL: Specifies the name of the Omit List. This parameter is valid for ***FLR**, ***LIB**, ***LNK** and ***OBJ** Recovery Lists.

***NONE**	No Omit List is selected.
Omit-list	Enter a valid Omit List name.

ERROR: Specifies the action to take if a severe error occurs. A severe error occurs when an escape message is issued in a Recovery List entry.

***IGNORE**	The process continues.
*CANCEL	The process stops.

TEXT: Specifies the text that briefly describes the object.

***BLANK**	No text is specified.
'description'	Enter no more than 50 characters of text, enclosed in apostrophes.

Examples

```
ADDRCYL RCY(NONSYS) SEQNBR(10) LIST(*LIB) +
    TEXT('NONSYS Recovery')
```

This adds Recovery List sequence number 10 to a Recovery named NONSYS.

ADDRCYLE - Add Recovery List Entry

```
---------------------------------------------------------------------------------------------------
                    Add Recovery List Entry (ADDRCYLE)  Environment: B/I

Backup . . . . . .  _________          Name
Sequence number . . . . . .  ____      1-9999
List . . . . . .  _________            *FLR, *LIB, *LNK…
Order number . . . . . .  *END         *END, 1-9999
Object . . . . . .  _________          Name, generic*, *ALL
Library . . . . . .  _________         Name, generic*, *ALLUSR…
Object type . . . . . .  _______       *ALRTBL, *BNDDIR…
Path name . . . . . .  *ALL            Path name, *ALL
Text . . . . . .  *BLANK               Char, *BLANK
Command . . . . . .  _____________     Char

---------------------------------------------------------------------------------------------------
```

Purpose

The Add Recovery List Entry (ADDRCYLE) command adds one or more entries to a Recovery List. Recovery List entries can be libraries, objects, folders and links.

Parameters

RCY: Specifies the name of the Recovery that is associated with this Recovery List entry.

Recovery-name	Enter a valid Recovery name.

SEQNBR: Specifies the sequence number of this Recovery List entry.

Sequence-number	Enter a number from 1-9999.

LIST: Specifies the type of Recovery List.

*FLR	Document library object list.
*EXIT	Command list.
*LIB	Library list.
*LNK	Integrated file system list.
*OBJ	Object list.

ORDNBR: Specifies the order number of the list entry to add. If multiple entries have the same order number, the entries are sorted alphabetically within order number.

***END**	The entry is added to the end of the backup list.
Order-number	Enter a value from 1-9999.

OBJ: Specifies the name of the objects to add to a ***OBJ** Recovery List.

*ALL	All objects from the specified library are selected.
generic*	Enter the generic name of the objects to select. A generic name is a character string that contains one or more characters followed by an asterisk (*).
Object-name	Enter a valid object name.

LIB: Specifies the name or type of libraries to add to a ***LIB** Recovery List.

*ALLUSR	All user libraries are selected.
*IBM	All IBM libraries are selected.
*NONSYS	All non-system libraries are selected.
generic*	Enter the generic name of the libraries to select. A generic name is a character string that contains one or more characters followed by an asterisk (*).
Library-name	Enter a valid library name.

OBJTYPE: Specifies which type of objects to add to a ***OBJ** Recovery List.

Object-type	Enter a valid object type.

PATH: Specifies the path name to add to a ***LNK** or ***FLR** Recovery List.

***ALL**	All path names for the specified Recovery List type are selected.
Link-name	Enter a valid path name.

TEXT: Specifies the text that briefly describes the object.

***BLANK**	No text is specified.
'description'	Enter no more than 50 characters of text, enclosed in apostrophes.

COMMAND: Specifies the command to execute. This parameter applies to an ***EXIT** Recovery List.

command	Enter the command to execute. The command is validated prior to being added.

Examples

```
ADDRCYLE RCY(NONSYS) SEQNBR(5) OBJ(PRODLIB/*ALL) +
    OBJTYPE(*ALL)
```

This adds a Recovery List entry for a Recovery named NONSYS. The Recovery List entry specifies that all objects from library PRODLIB are restored.

CHGBKUP - Change Backup

```
------------------------------------------------------------------------------------------------------------
                    Change Backup (CHGBKUP)        Environment: B/I

Backup . . . . . . . . . . . . . . . . . . . .  _________    Name, *AUTO
Backup definition . . . . . . . . . . . . . .   *SAME        Name, *SAME, *DFT
Text . . . . . . . . . . . . . . . . . . . . .  *SAME        Char, *SAME, *BLANK

------------------------------------------------------------------------------------------------------------
```

Purpose

The Change Backup (CHGBKUP) command changes the attributes of a Backup. The attributes that can be changed are the Backup Definition and the text.

Parameters

BACKUP: Specifies the name of the Backup.

*AUTO	The Intelligent Backup is selected.
Backup-name	Enter a name for the Backup.

BKUPDFN: Specifies the Backup Definition to associate with this Backup.

***SAME**	Retain the current value.
*DFT	The default Backup Definition is selected.
Backup-definition	Enter a valid Backup Definition.

TEXT: Specifies the text that briefly describes the object.

***SAME**	Retain the current value.
*BKUPDFN	The text associated with the Backup Definition is used.
*BLANK	No text is specified.
'description'	Enter no more than 50 characters of text, enclosed in apostrophes.

Examples

```
CHGBKUP BACKUP(DAILY) BKUPDFN(DAILY)
```

This changes the Backup Definition for a Backup named DAILY to DAILY.

CHGBKUPDFN - Change Backup Definition

```
---------------------------------------------------------------------------------------------------
                    Change Backup Definition  (CHGBKUPDFN)          Environment: B/I

Backup definition . . . . . .  _________          Name, *DFT
Device . . . . . .  . . . . .  *SAME              Name, *SAME
          + for more values    _________
End of tape option . . . . . .  *SAME             *SAME, *LEAVE, *UNLOAD
Use optimum block . . . . . .  *SAME              *SAME, *YES, *NO
Journaled objects . . . . . .  *SAME              *SAME, *NO, *YES
Target release . . . . . .  . . *SAME             *SAME, *CURRENT, *PRV…
Clear. . . . . .  . . . . . .  *SAME              *SAME, *NONE, *ALL, *AFTER
Object pre-check . . . . . .  . *SAME             *SAME, *NO, *YES
Save active:
  Object link . . . . . .  . . . *SAME            *SAME, *NO, *YES, *SYNC
  Folder . . . . . .  . . . . . . *SAME           *SAME, *NO, *YES
  Library . . . . . .  . . . . . . *SAME          *SAME, *NO, *LIB, *SYNCLIB, *SYSDFN
  Object . . . . . .  . . . . . . *SAME           *SAME, *NO, *LIB, *SYNCLIB, *SYSDFN
  Changed object . . . . . .  . . *SAME           *SAME, *NO, *LIB, *SYNCLIB, *SYSDFN
Save active wait time . . . . . .  *SAME          0-99999, *SAME, *NOMAX
Save active message queue . . . .  *SAME          Name, *SAME, *NONE
  Library . . . . . .  . . . . . .  _________     Name
Save access paths . . . . . .  . *SAME            *SAME, *NO, *YES
Save file data . . . . . .  . . . *SAME           *SAME, *YES, *NO
Data compression . . . . . .  . . *SAME           *SAME, *DEV, *NO, *YES
Data compaction . . . . . .  . . *SAME            *SAME, *DEV, *NO
Output . . . . . .  . . . . . . . *SAME           *SAME, *NONE, *PRINT
Text . . . . . .  . . . . . . . . *SAME           Char, *SAME, *BLANK
Pre-Exit program . . . . . .  . . *SAME           Name, *SAME, *NONE
  Library . . . . . .  . . . . . .  _________     Name
Post-Exit program . . . . . .  . *SAME            Name, *SAME, *NONE
  Library . . . . . .  . . . . . .  _________     Name

---------------------------------------------------------------------------------------------------
```

Purpose

The Change Backup Definition (CHGBKUPDFN) command changes the attributes of an existing Backup Definition.

Parameters

BKUPDFN: Specifies the name of the Backup Definition.

*DFT	The default Backup Definition is selected.
Backup-definition	Enter a Backup Definition name.

DEVICE: Specifies the name of one or more devices to use for the backup process. Up to 32 devices can be specified. Only the first 4 are used for saves and restores.

The additional drives can be specified if using Media Definitions and a StorageTek Tape Library. To eliminate the need to enter up to 32 StorageTek tape devices, specify only one (1) device. MMS/bms automatically selects more drives based on the number of resources specified in the **DRVRSC** parameter of the Backup List. In order for this feature to work, the StorageTek drives must be defined to TMS as **SHARE(*YES)**.

***SAME**	Retain the current value.
Device-definition	Enter one or more valid device names.

ENDOPT: Specifies the positioning option that is automatically done on the tape volume after the operation ends. If more than one volume is used, the parameter applies only to the last volume; all other volumes are rewound and unloaded when the end of tape is reached.

***SAME**	Retain the current value.
*LEAVE	The tape does not rewind or unload after the operation ends. It remains at the current position on the tape drive.
*UNLOAD	The tape is automatically rewound and unloaded after the operation ends.

USEOPTBLK: Specifies whether the optimum block size is used for the operation.

***SAME**	Retain the current value.
*NO	The optimum block size supported by the device is not used. Save commands use the default block size supported by all device types. The tape volume can be duplicated to any media format using the Duplicate Tape (**DUPTAP**) command.
*YES	The optimum block size supported by the device is used for Save commands. Performance may improve, but the tape volume that is created is only compatible with a device that supports the block size used. Commands such as Duplicate Tape **(DUPTAP)** do not duplicate files unless the files are being duplicated to a device which supports the same block size that was used.

OBJJRN: Specifies whether changes to objects currently being entered in a journal are saved.

***SAME**	Retain the current value.
*NO	Journaled objects are not saved.
*YES	Journaled objects are saved.

TGTRLS: Specifies the release of the operating system on which you intend to restore and use the object.

When specifying the target-release value, the format VxRxMx is used to specify the release, where Vx is the version, Rx is the release, and Mx is the modification level. For example, **V4R5M0** is version 4, release 5, modification level 0.

To specify that an object be saved for distribution to a system at a different release level than the system on which the save operation is to occur, the procedure differs for program or non-program objects and by the release level on which a program object is created. If, for example, you are saving an object for distribution to a target system running on an earlier release, you have the following choices:

For program objects:

If the program object was created at a release level more current than the targeted earlier release, you must (1) create the program object again specifying the targeted earlier release, (2) save the program object specifying the targeted earlier release, and then (3) restore the program object on the target system.

If the program object was created at the same release level as the target system, you can (1) save the program object specifying the targeted earlier release and then (2) restore the program object on the target system.

For non-program objects:

You can (1) save the object specifying the targeted earlier release and then (2) restore the object on the target system.

The possible values are:

***SAME**	Retain the current value.
*CURRENT	The object is to be restored to, and used on, the release of the operating system currently on this system. The object can also be restored to a system with any subsequent release of the operating system installed.
*PRV	The object is to be restored to the previous release with modification level 0 of the operating system. The object can also be restored to a system with any subsequent release of the operating system installed.
Target-release	Specify the release in the format VxRxMx. The object can be restored to a system with the specified release or with any subsequent release of the operating system installed. Valid values depend on the current version, release, and modification level, and they change with each new release.

Note: If **LIB(*ALLUSR)** is specified, only the current release can be the target release. For release **V4R5M0**, valid values are ***CURRENT** or **V4R5M0**.

Not all objects can be targeted to another release. To find out which objects are supported, see the chart in the Backup and Recovery book, SC41-5304.

CLEAR: Specifies whether tapes or save files that contain active data and are encountered during the save operation are automatically cleared. An uncleared tape is one containing a file with an expiration date later than the date of the save operation (including files protected permanently with **EXPDATE(*PERM)**).

Note: This parameter does not control initializing tapes used to perform the save operation. Tapes should be initialized to a standard label format before the save command is issued.

You can use the Initialize Tape (**INZTAP**) command and specify a value on the **NEWVOL** parameter to initialize a tape to a standard label format.

If a tape volume that is not initialized is encountered during the save operation, an inquiry message is sent and an operator can initialize the tape volume.

The possible values are:

***SAME** Retain the current value.

*NONE None of the media used during the save operation are cleared. An inquiry message is sent to the system operator if active files are encountered.

*ALL All uncleared media encountered during the save operation are cleared.

*AFTER All the uncleared tapes that are found after the first volume, and that are not already cleared, are cleared. If the operation cannot proceed because the first volume is uncleared, an inquiry message is sent to the system operator, who can end the operation or specify that the currently selected volume be cleared so the operation can continue. This value is not valid for save files.

PRECHK: Specifies whether the save operation for a library ends if any of the following are true:

1. The objects do not exist.
2. The library or objects were previously found to be damaged.
3. The library or objects are locked by another job.
4. The requester of the save does not have authority to the library or objects.

The possible values are:

***SAME** Retain the current value.

*NO The save continues, only saving only those objects that can be saved.

*YES If, after all specified objects are checked, one or more objects cannot be saved, the save operation for a library ends before any data is written. If multiple libraries are specified, the save operation continues with the next library. However, if **PRECHK(*YES)** and **SAVACT(*SYNCLIB)** are specified and an object in any library to be saved does not meet the preliminary check conditions, the save operation ends and no objects are saved.

SAVACT: Specifies whether different object types can be updated while being saved.

Object link: Specifies whether object links can be updated while being saved.

Note: If your system is in a restricted state, this parameter is ignored and the save operation is performed as if **SAVACT(*NO)** was specified.

***SAME** Retain the current value.

*NO Objects that are in use are not saved. Objects cannot be updated while being saved.

*YES Objects can be saved and used at the same time. The object checkpoints can occur at different times.

*SYNC Objects can be saved and used at the same time. The object checkpoints occur at the same time.

Folder: Specifies whether folders can be updated while being saved.

***SAME** Retain the current value.

*NO Document library objects in use are not saved. Document library objects cannot be updated while being saved.

*YES Document library objects can be changed during the save request.

Library: Specifies whether library objects can be updated while being saved.

Note: If your system is in a restricted state and the **SAVACT** parameter is specified, the save operation is performed as if **SAVACT(*NO)** was specified.

***SAME** Retain the current value.

*NO	Objects in use are not saved. Objects cannot be updated while being saved.
*LIB	Objects in a library can be saved while they are in use by another job. All of the objects in a library reach a checkpoint together and are saved in a consistent state in relationship to each other. **Note**: Libraries with thousands of objects may be too large for this option.
*SYNCLIB	Objects in a library can be saved while they are in use by another job. All of the objects and all of the libraries in the save operation reach a checkpoint together and are saved in a consistent state in relationship to each other.
*SYSDFN	Objects in a library can be saved while they are in use by another job. Objects in a library may reach checkpoints at different times and may not be in a consistent state in relationship to each other. **Note**: Specifying this value eliminates some size restrictions and may enable a library to be saved that could not be saved with **SAVACT(*LIB)**.

Object: Specifies whether objects can be updated while being saved.

Note: If your system is in a restricted state and the **SAVACT** parameter is specified, the save operation is performed as if **SAVACT(*NO)** was specified.

***SAME** Retain the current value.

*NO Objects in use are not saved. Objects cannot be updated while being saved.

*LIB Objects in a library can be saved while they are in use by another job. All of the objects in a library reach a checkpoint together and are saved in a consistent state in relationship to each other.

*SYNCLIB Objects in a library can be saved while they are in use by another job. All of the objects and all of the libraries in the save operation reach a checkpoint together and are saved in a consistent state in relationship to each other.

*SYSDFN Objects in a library can be saved while they are in use by another job. Objects in a library may reach checkpoints at different times and may not be in a consistent state in relationship to each other.

Changed Object: Specifies whether changed objects can be updated while being saved.

Note: If your system is in a restricted state and the **SAVACT** parameter is specified, the save operation is performed as if **SAVACT(*NO)** was specified.

***SAME** Retain the current value.

*NO Objects in use are not saved. Objects cannot be updated while being saved.

*LIB Objects in a library can be saved while they are in use by another job. All of the objects in a library reach a checkpoint together and are saved in a consistent state in relationship to each other.

*SYNCLIB Objects in a library can be saved while they are in use by another job. All of the objects and all of the libraries in the save operation reach a checkpoint together and are saved in a consistent state in relationship to each other.

*SYSDFN Objects in a library can be saved while they are in use by another job. Objects in a library may reach checkpoints at different times and may not be in a consistent state in relationship to each other.

SAVACTWAIT: Specifies the amount of time to wait for a commit boundary or a lock on an object, if it is not available, before continuing the save. If a lock is not obtained in the specified time, the object is not saved. If a commit boundary is not reached in the specified time, the save operation is ended.

***SAME** Retain the current value.

*NOMAX No maximum wait time exists.

Wait-time Enter the time (in seconds) to wait for a commit boundary or an object lock before continuing the save operation. Valid values range from 0 through 99999.

SAVACTMSGQ: Specifies the message queue that the save operation uses to notify the user that the checkpoint processing for the library is complete. A separate message is sent for each library to be saved when the ***SYSDFN** or ***LIB** value is specified on the Save active prompt (**SAVACT** parameter). When the ***SYNCLIB** value is specified on the Save active prompt (**SAVACT** parameter), one message is sent for all libraries in the save operation.

This parameter can be used to save the objects at a known, consistent boundary to avoid additional recovery procedures following a restore operation. Applications can be stopped until the checkpoint processing complete message is received.

Message queue:

***SAME**	Retain the current value.
*NONE	No notification message is sent.
Message-queue	Enter the name of a message queue.

Library:

Library-name	Enter a valid library name.

ACCPTH: Specifies whether the logical file access paths that are dependent on the physical files being saved are also saved. The access paths are saved only in the case of the following:

All members on which the access paths are built are included in this save operation.

The access paths are not invalid or damaged at the time of the save. The system checks to ensure the integrity of the access paths. Any discrepancies found by the system will result in the access paths being rebuilt.

Informational messages are sent indicating the number of logical file access paths saved with each physical file. All physical files on which an access path is built must be in the same library. This parameter does not save logical file objects; it only controls the saving of the access paths. More information on the restoring of saved access paths is in the Backup and Recovery book, SC41-5304.

* * * **Attention** * * *

* * * If the based-on physical files and the logical files are in different libraries, the access paths are saved.

* * * However, if the logical files and the based-on physical files are in different libraries and the logical files or physical files do not exist at restore time (such as during disaster recovery or the files were deleted) the access paths are not restored. They are rebuilt.

* * * For the fastest possible restore operation for logical files, the logical files and the based-on physical files must be in the same library and must be saved at the same time.

The possible values are:

***SAME**	Retain the current value.
*NO	Only those objects specified on the command are saved. No logical access paths are saved.
*YES	The specified physical files and all eligible logical files built over them are saved.

Note: Specifying this value does not save the logical files.

SAVFDTA: Specifies, for save file objects, whether the description of a save file, or both the description and the contents of a save file, are saved on the tape or in another save file.

***SAME**	Retain the current value.
*YES	The description and the contents of the save file are saved.
*NO	Only the description of the save file is saved.

DTACPR: Specifies whether data compression is used.

***SAME** Retain the current value.

*DEV If the save is to tape and the target device supports compression, hardware compression is performed. Otherwise, no data compression is performed.

Note: If ***DEV** is specified on both the Data compression prompt (**DTACPR** parameter) and the Data compaction prompt (**COMPACT** parameter), only device data compaction is performed if device data compaction is supported on the device. Otherwise, data compression is performed.

If ***YES** is specified on the Data compression prompt (**DTACPR** parameter) and ***DEV** is specified on the Data compaction prompt (**COMPACT** parameter), both device data compaction and device data compression are performed.

*NO No data compression is performed.

*YES If the save is to tape and the target device supports compression, hardware compression is performed. If compression is not supported, or if the save data is written to a save file, software compression is performed. If the save is running while other jobs on the system are active and software compression is used, the overall system performance may be affected.

COMPACT: Specifies whether device data compaction is used.

***SAME** Retain the current value.

*DEV Device data compaction is performed if the data is saved to tape and all tape devices specified on the Device prompt (**DEV** parameter) support the compaction feature.

Note: If ***DEV** is specified on both the Data compression prompt (**DTACPR** parameter) and the Data compaction prompt (**COMPACT** parameter), only device data compaction is performed if device data compaction is supported on the device. Otherwise, data compression is performed.

If ***YES** is specified on the Data compression prompt (**DTACPR** parameter) and ***DEV** is specified on the Data compaction prompt (**COMPACT** parameter), both device data compaction and device data compression are performed.

*NO No device data compaction is performed.

OUTPUT: Specifies whether the output from the command is displayed at the requesting workstation or printed with the job's spooled output.

***SAME**	Retain the current value.
*NONE	No output is created.
*PRINT	The output is printed with the job's spooled output.

TEXT: Specifies the text that briefly describes the object.

***SAME**	Retain the current value.
*BLANK	No text is specified.
'description'	Enter no more than 50 characters of text, enclosed in apostrophes.

PREEXIT: Specifies the qualified name of the pre-exit program to call before the process starts.

Exit Program:

***SAME**	Retain the current value.
*NONE	No pre-exit program is selected.
Exit-program	Enter the name of a pre-exit program.

Library:

Library-name	Enter a valid library name.

POSTEXIT: Specifies the qualified name of the post-exit program to call after the process completes.

Exit Program:

***SAME**	Retain the current value.
*NONE	No post-exit program is selected.
Exit-program	Enter the name of a post -exit program.

Library:

Library-name	Enter a valid library name.

Examples

```
CHGBKUPDFN BKUPDFN(DAILY) DEVICE(TAPMLB01) ENDOPT(*UNLOAD) +
    TEXT('Daily Backup Definition') POSTEXIT(PRODBKUP/POSTPGM)
```

This changes a Backup Definition named DAILY, which uses a device named TAPMLB01. When the backup completes, the tape is unloaded and the post exit program POSTPGM in library PRDBKUP is called.

CHGBKUPL - Change Backup List

```
                    Change Backup List (CHGBKUPL)          Environment: B/I

Backup . . . . . .  _________          Name
Sequence number . . . . . .  ____      1-9999
List . . . . . .  _________            *ASP, *ASPDLO, *CFG…
Type . . . . . .  *SAME                *SAME, *FULL, *INCR, *CUML
Parallel device resources:
  Minimum resources . . . . . .  *SAME 1-32, *SAME, *NONE, *AVAIL
  Maximum resources . . . . . .  _____ 1-32, *AVAIL,  *MIN
ASP Device . . . . . .  *SAME          Name, *SAME, *, *SYSBAS...
Omit list name . . . . . .  *SAME      Name, *SAME, *NONE
Error action . . . . . .  *SAME        *SAME, *IGNORE, *CANCEL
Text . . . . . .  *SAME                Char, *SAME, *BLANK
```

Purpose

The Change Backup List (CHGBKUPL) command changes a Backup List for a Backup. Backup Lists define the type and sequence of the backup to perform.

Parameters

BACKUP: Specifies the name of the Backup that is associated with this Backup List.

Backup-name	Enter a valid Backup name.

SEQNBR: Specifies the sequence number of this Backup List.

Sequence-number	Enter a number from 1-9999.

LIST: Specifies the type of Backup List.

*ASP	Auxiliary storage pool (**ASP**) list.
*ASPDLO	Document library object auxiliary storage pool (**ASP**) list.
*CFG	Configuration list.
*CLT	Client list.
*EJECT	Tape eject.
*EXIT	Command exit.
*FLR	Document library object list.
*LIB	Library list.
*LND	Domino server list.
*LNK	Integrated file system object list.
*OBJ	Object list.
*OUTQ	Output queue list.
*RCY	MMS/*tms* recovery library list.
*SAVF	Save file list.
*SEC	Security information.
*SPL	Spooled file list.
*SYS	All Licensed Internal Code; the QSYS library; security and configuration objects.

TYPE: Specifies the type of save to process.

<u>*SAME</u>	Retain the current value.
*FULL	A full backup is performed.
*INCR	An incremental backup is performed. This value is valid for ***ASP**, ***ASPDLO**, ***FLR**, ***OBJ**, ***LNK**, ***OUTQ** and ***SPL** Backup Lists.
*CUML	A cumulative backup is performed. This value is valid for ***ASP**, ***ASPDLO**, ***FLR**, ***OBJ** and ***LNK** Backup Lists.

DRVRSC: Specifies the minimum and maximum number of device resources to use in a parallel save. This parameter is valid for ***ASP**, ***LIB** and ***OBJ** Backup Lists.

<u>Minimum resources:</u>

<u>*SAME</u>	Retain the current value.
*NONE	No device resources are used. The save is performed as a serial save.
*AVAIL	Use any available resources up to the maximum specified. This will use any available resource but will complete using one resource if only one is available.
1-32	Enter the minimum number of resources to use.

<u>Maximum resources</u>:

*MIN	Uses the value specified for the minimum number of device resources.
*AVAIL	The save will use any available device resources but at minimum, the value specified in the minimum element.
1-32	Enter the minimum number of resources to use.

ASPDEV: Specifies the name of the auxiliary storage pool (ASP) device to be included in the backup operation. This parameter is valid for ***LIB**, ***OBJ** and ***RCY** Backup Lists.

***SAME**	Retain the current value.
*	The operation includes the system ASP (ASP number 1), all basic user ASPs (ASP numbers 2 to 32) and, if the current thread has an ASP group, all independent ASPs in the ASP group.
*SYSBAS	The system ASP (ASP number 1) and all basic user ASPs (ASP numbers 2 to 32) are included in the backup operation.
*CURASPGRP	If the current thread has an ASP group, all independent ASPs in the ASP group are included in the backup operation.
*ALLAVL	The private authorities from the system ASP (ASP number 1), all basic user ASPs (ASP numbers 2 to 32) and all independent ASPs are saved.
ASP-device-name	Enter a valid independent ASP.

OMITL: Specifies the name of the Omit List. This parameter is valid for ***ASP**, ***ASPDLO**, ***FLR**, ***LIB**, ***LND**, ***LNK**, ***OBJ**, ***OUTQ** and ***SPL** Backup Lists.

***SAME**	Retain the current value.
*NONE	No Omit List is used.
Omit-list	Enter the name of a valid Omit List.

ERROR: Specifies the action to take if a severe error occurs. A severe error occurs when an escape message is issued in a Backup List entry.

***SAME**	Retain the current value.
*IGNORE	The process continues.
*CANCEL	The process stops.

TEXT: Specifies the text that briefly describes the object.

***SAME**	Retain the current value.
*BLANK	No text is specified.
'description'	Enter no more than 50 characters of text, enclosed in apostrophes.

Examples

```
CHGBKUPL BACKUP(DAILY) SEQNBR(5) TYPE(*FULL) +
    TEXT('Full Backup of all User Libraries')
```

This changes a Backup List to a Backup named DAILY. The Backup List for sequence number 5 now performs a full save.

CHGBKUPLE - Change Backup List Entry

```
------------------------------------------------------------------------------------------------------------
                    Change Backup List Entry (CHGBKUPLE)        Environment: B/I

Backup . . . . . .  _________   Name
Sequence number . . . . . .  ____   1-9999
List . . . . . .  _________   *ASP, *ASPDLO, *LNK…
Order number . . . . . .  ____   1-9999
Text . . . . . .  *SAME   Char, *SAME, *BLANK
Directory subtree . . . . . .  *SAME   *SAME, *ALL, *DIR, *NONE, *OBJ
Spooled file . . . . . .  _____   Name, generic*, *ALL
Delete spooled files . . . . . .  *SAME   *SAME, *NO, *YES

------------------------------------------------------------------------------------------------------------
```

Purpose

The Change Backup List Entry (CHGBKUPLE) command changes the attributes of a Backup List entry.

Parameters

BACKUP: Specifies the name of the Backup that is associated with this Backup List entry.

Backup-name	Enter a valid Backup name.

SEQNBR: Specifies the sequence number of this Backup List entry.

Sequence-number	Enter a number from 1-9999.

LIST: Specifies the type of Backup List.

*ASP	The list is an auxiliary storage pool (**ASP**) list.
*ASPDLO	All document library objects in the specified auxiliary storage pool (**ASP**) list.
*LNK	Links in the integrated file system list are saved.
*OUTQ	Output queues in the output queue list are saved.
*SPL	Spooled files in the spooled file list are saved.

ORDNBR: Specifies the order number of the list entry to change.

Order-number	Enter a value from 1-9999.

TEXT: Specifies the text that briefly describes the object.

***SAME**	Retain the current value.
*BLANK	No text is specified.
'description'	Enter no more than 50 characters of text, enclosed in apostrophes.

SUBTREE: Specifies whether the directory subtrees are included in the save.

***SAME**	Retain the current value.
*ALL	The entire subtree of each directory is included.
*DIR	The objects in the first level of each directory that matches the object name pattern is included.
*NONE	No subtrees are included.
*OBJ	Only the objects that exactly match the object name pattern are included.

FILE: Specifies the name of the spooled file for a ***SPL** Backup List.

*ALL	All spooled files are saved.
generic*	Enter the generic name of the spooled files to save. A generic name is a character string that contains one or more characters followed by an asterisk (*).
Spooled-file	Enter the name of a valid spooled file name.

DLTSPLF: Specifies whether spooled files are deleted after the save completes. This parameter applies to ***OUTQ** and ***SPL** Backup Lists.

***SAME**	Retain the current value.
*NO	The saved spooled files are not deleted.
*YES	The saved spooled files are deleted.

Examples

```
CHGBKUPLE BACKUP(DAILY) SEQNBR(5) LIST(*SPL) ORDNBR(5) TEXT('Test Output
Queue')
```

This changes a Backup List entry for a Backup named DAILY. Saved spooled files are deleted upon being saved successfully.

CHGOMITL - Change Omit List

```
---------------------------------------------------------------------------------------------------
                    Change Omit List  (CHGOMITL)          Environment: B/I

Omit list name . . . . . . . . . . . . . .   _________     Name
List . . . . . . . . . . . . . . . . . . .   _________     *CLT, *FLR, *LIB, *LND…
Text . . . . . . . . . . . . . . . . . . .   * SAME        Char, *SAME, *BLANK

---------------------------------------------------------------------------------------------------
```

Purpose

The Change Omit List (CHGOMITL) command changes the text for an existing Omit List. Omit Lists define the name and type of object to omit from a Backup.

Parameters

OMITL: Specifies the name of the Omit List.

List-name	Enter an Omit List name.

LIST: Specifies the type of Omit List .

*CLT	Client Omit List.
*FLR	Document library object Omit List.
*LIB	Library Omit List.
*LND	Domino server Omit List.
*LNK	Integrated file system Omit List.
*OBJ	Object Omit List.
*OUTQ	Output queue Omit List.
*SYS	System data Omit List.

TEXT: Specifies the text that briefly describes the object.

***SAME**	Retain the current value.
*BLANK	No text is specified.
'description'	Enter no more than 50 characters of text, enclosed in apostrophes.

Examples

```
CHGOMITL OMITL(DAILY) TEXT('Object Omit List')
```

This changes an Omit List named DAILY.

CHGRCY - Change Recovery

```
---------------------------------------------------------------------------------------------------
                    Change Recovery (CHGRCY)              Environment: B/I

Recovery . . . . . . . . . . . . . . . . . .  _________    Name, *AUTO
Recovery definition . . . . . . . . . .  * SAME          Name, *SAME, *DFT
Text . . . . . . . . . . . . . . . . . . . .  * SAME          Char, *SAME, *BLANK

---------------------------------------------------------------------------------------------------
```

Purpose

The Change Recovery (CHGRCY) command changes a Recovery and its associated Recovery Definition or text.

Parameters

RCY: Specifies the name of the Recovery.

*AUTO	The Intelligent Recovery is selected. Recovery Lists, Recovery List entries and Omit Lists are not allowed with this type of restore.
Recovery-name	Enter a name for the Recovery.

RCYDFN: Specifies the Recovery Definition to associate with this Recovery.

***SAME**	Retain the current value.
*DFT	The default Recovery Definition is selected.
Recovery-definition	Enter a valid Recovery Definition.

TEXT: Specifies the text that briefly describes the object.

***SAME**	Retain the current value.
*BLANK	No text is specified.
'description'	Enter no more than 50 characters of text, enclosed in apostrophes.

Examples

```
CHGRCY RCY(NONSYS) RCYDFN(*DFT) TEXT('NONSYS Recovery')
```

This changes a Recovery named NONSYS, which uses the default Recovery Definition.

CHGRCYDFN - Change Recovery Definition

```
                    Change Recovery Definition (CHGRCYDFN)          Environment: B/I

Recovery definition . . . . . .  _________          Name, *DFT
Library:
  Option . . . . . . . . . . . . .  *SAME            *SAME, *All, *NEW, *OLD, *FREE
  Data base member option . . . .  *SAME            *SAME, *MATCH, *ALL, *NEW, *OLD
  Allow object differences . . . . .  *SAME        *SAME, *NONE, *ALL, *FILELVL
  Auxiliary storage pool ID . . . .  *SAME          1-32, *SAME, *SAVASP
Object:
  Option . . . . . . . . . . . . .  *SAME            *SAME, *All, *NEW, *OLD, *FREE
  Data base member option . . . .  *SAME            *SAME, *MATCH, *ALL, *NEW, *OLD
  Allow object differences . . . . .  *SAME        *SAME, *NONE, *ALL, *FILELVL
  Auxiliary storage pool ID . . . .  *SAME          1-32, *SAME, *SAVASP
Folder:
  Object name generation . . . . .  *SAME           *SAME, *NEW
  Allow object differences . . . . .  *SAME        *SAME, *NONE, *ALL
  Saved from ASP. . . . . .  *SAME                 1-32, *SAME, *ANY
  Restore  to ASP. . . . . .  *SAME                1-32, *SAME, *SAVASP
Object link:
  Option . . . . . . . . . . . . .  *SAME           *SAME, *All, *NEW, *OLD
  Allow object differences . . . . .  *SAME        *SAME, *NONE, *ALL, *AUTL...
          + for more values  ____
Configuration:
  System Resource Management .  *SAME              *SAME, *ALL, *NONE, *HDW, *TRA
  Allow object differences . . . . .  *SAME        *SAME, *ALL, *NONE
Text 'description' . . . . . .  *SAME              Char, *SAME, *BLANK
```

Purpose

The Change Recovery Definition (CHGRCYDFN) command changes a Recovery Definition, which determines the attributes to use for the Recovery.

Parameters

RCYDFN: Specifies the name of the Recovery Definition.

Recovery-definition Enter a name for the Recovery Definition.

LIB: Specifies the restore options used when restoring libraries.

Option	Specifies how to handle restoring each object.
***SAME**	Retain the current value.
*ALL	All the objects in the saved library are restored to the library. Objects in the saved library replace the current versions of the system. Objects not having a current version are added. Objects presently in the library, but not on the media, remain in the library.
*NEW	Only the objects in the saved library that do not exist in the current version of the system library are added to the library. Only objects not known to the system library are restored; known objects are not restored. This option restores objects that were deleted after they were saved or that are new to this library. If any saved objects have a version already in the system library, they are not restored, and an informational message is sent for each one, but the restore operation continues.
. *OLD	Only the objects in the library having a saved version are restored; that is, the version of each object currently in the library is replaced by the saved version. Only objects known to the library are restored. If any saved objects are no longer part of the online version of the library, they are not added to the library; an informational message is sent for each one, but the restore continues.
*FREE	The saved objects are restored only if they exist in the system library with their space freed. The saved version of each object is restored on the system in its previously freed space. This option restores objects that had their space freed when they were saved. If any saved objects are no longer part of the current version of the library, or if the space is not free for any object, the object is not restored and an informational message is sent for each one. The restore operation continues, and all of the freed objects are restored.

dB mbr option: Specifies, for database files that exist on the system, which members are restored. If ***MATCH** is used, the member list in the saved file must match, member for member, the current version on the system. All members are restored for files that do not exist, if the file is restored.

***SAME**	Retain the current value.
*MATCH	The saved members are restored if the lists of the members where they exist match, member for member, the lists of the current system version. **MBROPT(*MATCH)** is not valid when ***ALL** is specified on the Allow object differences parameter.
*ALL	All members in the saved file are restored.
*NEW	Only new members (members not known to the system) are restored.
*OLD	Only members known to the system are restored.

Allow obj. diff: Specifies whether certain differences encountered during a restore operation are allowed. The differences include:

- **Ownership** -- the owner of the object on the system is different than the owner of the object from the save operation.
- **File creation date** -- the creation date of the database file on the system does not match the creation date of the file that was saved.
- **Member creation date** -- the creation date of the database file member on the system does not match the creation date of the member that was saved.
- **Validation value verification** -- the validation value created at the time an object was created does not match the validation value created during the restore operation of an object on a system with a **QSECURITY** level of **40** or higher.
- **Authorization list linking** -- the object is being restored to a system different from the one on which it was saved.

Note*: To use this parameter, you need* ****ALLOBJ*** *special authority.*

***SAME** Retain the current value.

*NONE None of the differences described above are allowed on the restore operation. For validation value verification failure cases, the object is restored but ownership is transferred to **QDFTOWN** and all authorities are revoked. For authorization list cases, the objects is restored, but the object is not linked to the authorization list, and public authority is set to ***EXCLUDE**. For all other cases, a diagnostic message is sent for the object, and the object is not restored.

*ALL All of the differences listed above are allowed for the restore operation. An informational message is sent, except for validation value verification and authorization list linking cases, and the object is restored. The following should be noted:

- If object differences are found, the final message for the restore operation is an escape message rather than the normal completion message.
- If the media and system owner of the object do not match, the system owner becomes the owner of the object.
- If there is a file level mismatch and ***ALL** is specified on this parameter and the Data base member option prompt, the existing version of the file is renamed and the saved version of the file is restored. If there is a member level mismatch, the existing version of the member is renamed and the saved version of the member is restored.

- If the system security level is **40**, you are restoring a program, you specify ***ALL**, and the program's validation value is missing or incorrect, the program is restored without authority changes. For programs without a validation value, specifying ***ALL** also prevents the system from attempting to translate the program again.

- If you are restoring objects to a system different from the one on which they were saved and the objects are secured by an authorization list, specifying ***ALL** automatically links the objects to the authorization list again. If the authorization list does not exist on the new system, a message that includes the name of the missing list is issued and the public authority is set to ***EXCLUDE**.

ASP ID: Specifies whether objects are restored to the auxiliary storage pool (ASP) from which they were saved or to another ASP. ASP 1 is the system ASP. Libraries and their contained objects may be restored to user ASPs (2 through 32). However, some objects cannot be restored to user ASPs.

More information about object types that can be restored to user ASPs is in the Backup and Recovery book, SC41-5304. If the library exists in, or is being restored to the system ASP, journals, journal receivers, and save files can be restored to user ASPs. All other object types will be restored to the ASP of the library.

Attention * * * System or product libraries (libraries that begin with a Q or #) must not be created in or restored to a user ASP. Doing so can cause unpredictable results. *** * ***

***SAME** Retain the current value.

*SAVASP The objects are restored to the ASP from which they were saved.

ASP-ID Enter the ASP identifier. When the specified ASP is 1, the specified objects are restored to the system ASP, and when the specified ASP is 2 through 32, the objects are restored to the user ASP specified.

OBJ: Specifies the restore options used when restoring objects.

Option	Specifies how to handle restoring each object.
***SAME**	Retain the current value.
*ALL	All the objects in the saved library are restored to the library. Objects in the saved library replace the current versions of the system. Objects not having a current version are added. Objects presently in the library, but not on the media, remain in the library.
*NEW	Only the objects in the saved library that do not exist in the current version of the system library are added to the library. Only objects not known to the system library are restored; known objects are not restored. This option restores objects that were deleted after they were saved or that are new to this library. If any saved objects have a version already in the system library, they are not restored, and an informational message is sent for each one, but the restore operation continues.
. *OLD	Only the objects in the library having a saved version are restored; that is, the version of each object currently in the library is replaced by the saved version. Only objects known to the library are restored. If any saved objects are no longer part of the online version of the library, they are not added to the library; an informational message is sent for each one, but the restore continues.
*FREE	The saved objects are restored only if they exist in the system library with their space freed. The saved version of each object is restored on the system in its previously freed space. This option restores objects that had their space freed when they were saved. If any saved objects are no longer part of the current version of the library, or if the space is not free for any object, the object is not restored and an informational message is sent for each one. The restore operation continues, and all of the freed objects are restored.

dB mbr option: Specifies, for database files that exist on the system, which members are restored. If ***MATCH** is used, the member list in the saved file must match, member for member, the current version on the system. All members are restored for files that do not exist, if the file is restored.

***SAME**	Retain the current value.
*MATCH	The saved members are restored if the lists of the members where they exist match, member for member, the lists of the current system version. **MBROPT(*MATCH)** is not valid when ***ALL** is specified on the Allow object differences parameter.
*ALL	All members in the saved file are restored.
*NEW	Only new members (members not known to the system) are restored.
*OLD	Only members known to the system are restored.

Allow obj. diff: Specifies whether certain differences encountered during a restore operation are allowed. The differences include:

- **Ownership** -- the owner of the object on the system is different than the owner of the object from the save operation.
- **File creation date** -- the creation date of the database file on the system does not match the creation date of the file that was saved.
- **Member creation date** -- the creation date of the database file member on the system does not match the creation date of the member that was saved.
- **Validation value verification** -- the validation value created at the time an object was created does not match the validation value created during the restore operation of an object on a system with a **QSECURITY** level of **40** or higher.
- **Authorization list linking** -- the object is being restored to a system different from the one on which it was saved.

Note*: To use this parameter, you need* ****ALLOBJ*** *special authority.*

***SAME** Retain the current value.

*NONE None of the differences described above are allowed on the restore operation. For validation value verification failure cases, the object is restored but ownership is transferred to **QDFTOWN** and all authorities are revoked. For authorization list cases, the objects is restored, but the object is not linked to the authorization list, and public authority is set to ***EXCLUDE**. For all other cases, a diagnostic message is sent for the object, and the object is not restored.

*ALL All of the differences listed above are allowed for the restore operation. An informational message is sent, except for validation value verification and authorization list linking cases, and the object is restored. The following should be noted:

- If object differences are found, the final message for the restore operation is an escape message rather than the normal completion message.
- If the media and system owner of the object do not match, the system owner becomes the owner of the object.
- If there is a file level mismatch and ***ALL** is specified on this parameter and the Data base member option prompt, the existing version of the file is renamed and the saved version of the file is restored. If there is a member level mismatch, the existing version of the member is renamed and the saved version of the member is restored.

- If the system security level is **40**, you are restoring a program, you specify ***ALL**, and the program's validation value is missing or incorrect, the program is restored without authority changes. For programs without a validation value, specifying ***ALL** also prevents the system from attempting to translate the program again.
- If you are restoring objects to a system different from the one on which they were saved and the objects are secured by an authorization list, specifying ***ALL** automatically links the objects to the authorization list again. If the authorization list does not exist on the new system, a message that includes the name of the missing list is issued and the public authority is set to ***EXCLUDE**.

ASP ID:

Specifies whether objects are restored to the auxiliary storage pool (ASP) from which they were saved or to another ASP. ASP 1 is the system ASP. Libraries and their contained objects may be restored to user ASPs (2 through 32). However, some objects cannot be restored to user ASPs.

More information about object types that can be restored to user ASPs is in the Backup and Recovery book, SC41-5304. If the library exists in, or is being restored to the system ASP, journals, journal receivers, and save files can be restored to user ASPs. All other object types will be restored to the ASP of the library.

Attention * * *
System or product libraries (libraries that begin with a Q or #) must not be created in or restored to a user ASP. Doing so can cause unpredictable results. *** * ***

***SAME** — Retain the current value.

*SAVASP — The objects are restored to the ASP from which they were saved.

ASP-ID — Specifies the ASP identifier. When the specified ASP is 1, the specified objects are restored to the system ASP, and when the specified ASP is 2 through 32, the objects are restored to the user ASP specified.

FLR: Specifies the restore options used when restoring document library objects.

Object name gen. Specifies whether a new library-assigned name and system object name are generated for the folders and documents being restored.

***SAME** The library-assigned name and the system object name do not change.

*NEW A new library-assigned name and system object name are generated for each document or folder being restored.

Allow obj. diff: Specifies whether certain differences encountered during a restore operation are allowed. The differences include:

- **Ownership** -- the owner of the object on the system is different than the owner of the object from the save operation.
- **File creation date** -- the creation date of the database file on the system does not match the creation date of the file that was saved.

  Member creation date -- the creation date of the database file member on the system does not match the creation date of the member that was saved.
- **Validation value verification** -- the validation value created at the time an object was created does not match the validation value created during the restore operation of an object on a system with a **QSECURITY** level of **40** or higher.
- **Authorization list linking** -- the object is being restored to a system different from the one on which it was saved.

Note*: To use this parameter, you need* ****ALLOBJ*** *special authority.*

***SAME** Retain the current value.

*NONE None of the differences described above are allowed on the restore operation. For validation value verification failure cases, the object is restored but ownership is transferred to **QDFTOWN** and all authorities are revoked. For authorization list cases, the objects is restored, but the object is not linked to the authorization list, and public authority is set to ***EXCLUDE**. For all other cases, a diagnostic message is sent for the object, and the object is not restored.

*ALL All of the differences listed above are allowed for the restore operation. An informational message is sent, and the object is restored.

Notes*:*

- *If the owners of the object do not match, the object is restored, but it will keep the ownership and authorities of the object on the system before the restore operation.*
- *If* ****ALL*** *is specified on this parameter,* ****NEW*** *cannot be specified on the Object name generation parameter.*

- *If you are restoring objects to a system different from the one on which they were saved and the objects are secured by an authorization list, specifying ***ALL** automatically links the objects to the authorization list. If the authorization list does not exist on the new system, a message that includes the name of the missing list is issued.*

Saved from ASP: Specifies the identifier (ID) of the auxiliary storage pool (ASP) on media from which saved documents and folders are to be restored.

***SAME** Retain the current value.

*ANY The documents and folders saved in any ASP are restored.

Note: *When restoring Document Library Objects (DLOs) from more than one ASP, the sequence number (**SEQNBR**) parameter must be specified.*

ASP-ID Specify a value ranging from 1 through 32, which is the ID of the ASP from which documents and folders are restored.

Restore to ASP: Specifies the identifier (ID) of the auxiliary storage pool (ASP) on media in which restored documents and folders are to be placed.

***SAME** Retain the current value.

*SAVSAP The documents and folders are placed in the same ASP from which they were saved.

ASP-ID Specify a value ranging from 1 through 32, which is the ID of the ASP in which restored documents and folders are placed.

OBJLNK: Specifies the restore options used when integrated file system objects.

Option	Specifies how to handle restoring each object.
***SAME**	Retain the current value.
*ALL	All of the specified objects are restored, whether they already exist on the system or not.
*NEW	Objects are restored only if they do not already exist on the system.
. *OLD	Objects are restored only if they already exist on the system.

Allow obj. diff: Specifies whether certain differences encountered during a restore operation are allowed. The differences include:

- **Ownership**: The owner of an object on the system is different than the owner of an object from the save operation.
- **Authorization list linking**: The system on which an object with an authorization list is being restored is different from the system on which it was saved.
- **Primary Group**: The primary group of an object on the system is different than the primary group of an object from the save operation.

***SAME** Retain the current value.

*NONE No differences are allowed between the saved object and the restored object. If the owner is different, the object is not restored. If the system is different for an object with an authorization list, the object is restored, but the object is not linked to its authorization list.

*ALL All differences are allowed between the saved object and the restored object. If the owner is different, the object is restored with the owner of the system on which it is restored. If the system is different for an object with an authorization list, the object is restored and linked to its authorization list.

*OWNER The object owner can be different. If an object already exists on the system with a different owner than the saved object, the object is restored with the owner of the object on the system. If owner differences are not allowed, the object is not restored.

*AUTL The system of an object with an authorization list can be different. The new object, which is being restored to a system that is different from which it was saved, is restored and linked to its authorization list. If the system of an object with an authorization list cannot be different, the object is restored but not linked to an authorization list.

*PGP The object primary group can be different. If an object already exists on the system with a different primary group than the saved object, the object is restored with the primary group of the object on the system. If primary group differences are not allowed, the object is not restored.

CFG: Specifies the parameter values to use when restoring configurations.

Sys. Resource Mgmt: Specifies the type of system resource management (SRM) information to be restored. This parameter is valid only when ***ALL** or ***SRM** is specified on the Objects prompt (CFG parameter).

Attention * * *

Unless the system you are restoring to has exactly the same hardware configuration as the system that the original configuration was saved on, you must specify SRM(*NONE) on this command to prevent the restore of the SRM information. If the SRM information is restore on a system with a different hardware configuration, the configuration objects may become unusable.

***SAME**	Retain the current value.
*ALL	All system resource management information is restored.
*NONE	No system resource management information is restored.
*HDW	All hardware information is restored.
*TRA	All token-ring adapter information is restored.

Allow obj. diff: Specifies whether certain differences encountered during a restore operation are allowed. There are two differences allowed for this command:

- The owner of the object on the system is different than the owner of the object from the save.
- The object is secured by an authorization list and is being restored to a system other than the one on which it was saved.

Note: In order to use this parameter, you need ***ALLOBJ** authority.

***SAME** Retain the current value.

*NONE None of the differences previously described are allowed on the restore operation. For an ownership difference, the object is not restored. For an authorization list difference, the object is restored, but the object is not linked to the authorization list, and public authority is set to ***EXCLUDE**.

*ALL All of the differences previously described are allowed for the restore operation. The object is restored. The following should be noted:

- If the media and system owners of the object do not match, the system owner becomes the owner of the object and an informational message is sent.
- The informational message triggers a diagnostic message to be sent indicating that security or integrity changes occurred during the restore operation. The final message for the restore operation is an escape message, rather than the normal completion message.
- If you are restoring objects to a system different from the one on which they were saved and the objects are secured by an authorization list, specifying ***ALL** automatically links the objects to the authorization list again. If the authorization list does not exist on the new system, a message that includes the name of the missing list will be used.

TEXT: Specifies the text that briefly describes the object.

***SAME**	Retain the current value.
*BLANK	No text is specified.
'description'	Enter no more than 50 characters of text, enclosed in apostrophes.

Examples

```
CHGRCYDFN RCYDFN(NONSYS) TEXT('NONSYS Recovery')
```

This changes a Recovery Definition named NONSYS, which uses the default restore values for libraries, objects, document library objects and integrated file system objects.

CHGRCYL - Change Recovery List

```
                        Change Recovery List  (CHGRCYL)        Environment: B/I

Recovery . . . . . .  _________                Name
Sequence number . . . . . .  ____              1-9999
List . . . . . .  _________                    *CFG, *EXIT, *FLR, *LIB…
Restore to ASP device . . . . . .  *SAME       Name, *SAME, *SAVASPDEV
Omit list name . . . . . .  *SAME              Name, *SAME, *NONE
Error action . . . . . .  *SAME                *SAME, *IGNORE, *CANCEL
Text . . . . . .  *SAME                        Char, *SAME, *BLANK
```

Purpose

The Change Recovery List (CHGRCYL) command changes a Recovery List to a Recovery. Recovery Lists define the sequence of the recoveries to perform.

Parameters

RCY: Specifies the name of the Recovery that is associated with this Recovery List.

Recovery-name	Enter a valid Recovery List name.

SEQNBR: Specifies the sequence number of this Recovery List.

Sequence-number	Enter a number from 1-9999.

LIST: Specifies the type of Recovery List.

*CFG	Configuration list.
*EXIT	Command exit.
*FLR	Document library object list.
*LIB	Library list.
*LNK	Integrated file system object list.
*OBJ	Object list.
*RCY	MMS recovery libraries list.
*SEC	Security data list.
*SYS	System list.

RSTASPDEV: Specifies the name of the auxiliary storage pool (ASP) device to which the data is restored. Specify either the RSTDEVASP parameter or the RSTASP parameter, which is defined in the Recovery Definition, but not both.

***SAME**	Retain the current value.
*SAVASPDEV	The data is restored to the same ASP from which it was saved.
ASP-device-name	Enter a valid ASP device name.

OMITL: Specifies the name of the Omit List. This parameter is valid for ***FLR**, ***LIB**, ***LNK** and ***OBJ** Recovery Lists.

***SAME**	Retain the current value.
*NONE	No Omit List is selected.
Omit-list	Enter a valid Omit List name.

ERROR: Specifies the action to take if a severe error occurs. A severe error occurs when an escape message is issued in a Recovery List entry.

***SAME**	Retain the current value.
*IGNORE	The process continues.
*CANCEL	The process stops.

TEXT: Specifies the text that briefly describes the object.

***SAME**	Retain the current value.
*BLANK	No text is specified.
'description'	Enter no more than 50 characters of text, enclosed in apostrophes.

Examples

```
CHGRCYL RCY(NONSYS) SEQNBR(10) LIST(*LIB) +
    TEXT('NONSYS Recovery')
```

This changes Recovery List sequence number 10 to a Recovery named NONSYS.

CHGRCYLE - Change Recovery List Entry

```
-------------------------------------------------------------------------------------------------
                Change Recovery List Entry (CHGRCYLE)  Environment: B/I

Recovery . . . . . .  _________      Name
Sequence number . . . . . .  ____    1-9999
List . . . . . .  _________          *LIB, *OBJ
Order number . . . . . .  *SAME      1-9999, *SAME
Text 'description'. . . . . .  *BLANK  Char, *BLANK

-------------------------------------------------------------------------------------------------
```

Purpose

The Change Recovery List Entry (CHGRCYLE) command changes one or more entries to a Recovery List.

Parameters

RCY: Specifies the name of the Recovery that is associated with this Recovery List entry.

Recovery-name	Enter a valid Recovery name.

SEQNBR: Specifies the sequence number of this Recovery List entry.

Sequence-number	Enter a number from 1-9999.

LIST: Specifies the type of Recovery List.

*LIB	Library list.
*OBJ	Object list.

ORDNBR: Specifies the order number of the list entry to change.

***SAME**	Retain the current value.
Order-number	Enter a value from 1-9999.

TEXT: Specifies the text that briefly describes the object.

***BLANK**	No text is specified.
'description'	Enter no more than 50 characters of text, enclosed in apostrophes.

Examples

```
CHGRCYLE RCY(NONSYS) SEQNBR(5) LIST(*LIB) ORDNBR(10) TEXT('All Libraries')
```

This changes a Recovery List entry for a Recovery named NONSYS.

CHKSAVSTS – Check Save Status

```
---------------------------------------------------------------------------------------------------
                    Check Save Status (CHKSAVSTS)          Environment: B/I

Object . . . . . .  *ALL                  Name, generic*, *ALL
  Library . . . . . .  *ALLUSR            Name, *ALLUSR
Object type . . . . . .  *ALL             *ALL, *ALRTBL, *AUTL…

---------------------------------------------------------------------------------------------------
```

Purpose

The Check Save Status (CHKSAVSTS) command compares the objects specified change date with the last save date. If the change date is greater than the save date, a report is printed. In addition to reporting on current changes, the command also includes any objects or members that currently reside in save files or for which no current backup tape is found. It is recommended that this command be executed in a batch. This command requires that MMS or MMS/*tms* be installed.

Parameters

OBJ: Specifies the name of the objects to select for processing. If generic values are used, the library **and** object can be a generic value.

Object:

***ALL**	All objects from the specified library are selected.
generic*	Enter the generic name of the objects to select. A generic name is a character string that contains one or more characters followed by an asterisk (*).
Object-name	Enter a valid object name.

Library:

***ALLUSR**	All objects from all user libraries are selected.
generic*	Enter the generic name of the libraries to select. A generic name is a character string that contains one or more characters followed by an asterisk (*).
Library-name	Enter a valid library name.

OBJTYPE: Specifies which type of objects to select.

***ALL**	All object types are selected.
Object-type	Enter a valid object type.

Examples

```
CHKSAVSTS OBJ(*ALLUSR/*ALL) OBJTYPE(*ALL)
```

This checks all user objects on the system with the last save in the MMS/*tms* database.

CPYBKUP - Copy Backup

```
------------------------------------------------------------------------------------------------------------
                         Copy Backup (CPYBKUP)          Environment: B/I

From backup . . . . . . . . . . . . . . . .   __________          Name, *AUTO
To backup . . . . . . . . . . . . . . . . . .   __________          Name
------------------------------------------------------------------------------------------------------------
```

Purpose

The Copy Backup (CPYBKUP) command copies a Backup to another Backup. The Backup List and Backup List entries are copied. An error message is sent if the Backup being copied to already exists. Copying a Backup automatically renumbers Backup Lists and Backup List entries.

Parameters

FROMBKUP: Specifies the name of the Backup to copy.

*AUTO	The Intelligent Backup is selected.
Backup-name	Enter a valid Backup name.

TOBKUP: Specifies the name of the receiving Backup. That Backup name must not exist before being copied.

Backup-name	Enter a Backup name.

Examples

```
CPYBKUP FROMBKUP(DAILY) TOBKUP(WEEKLY)
```

This copies a Backup named DAILY to Backup named WEEKLY. Backup name WEEKLY must not exist.

CPYBKUPDFN - Copy Backup Definition

```
---------------------------------------------------------------------------------------------------
                    Copy Backup Definition (CPYBKUPDFN)          Environment: B/I

From backup definition . . . . . _____________   Name, *DFT
To backup definition. . . . . _____________   Name
---------------------------------------------------------------------------------------------------
```

Purpose

The Copy Backup Definition (CPYBKUPDFN) command copies an existing Backup Definition to another. An error message is issued if the Backup Definition being copied to already exists.

Parameters

FRMBKUPDFN: Specifies the name of the Backup Definition to copy.

*DFT	The default Backup Definition is selected.
Backup-name	Enter a valid Backup Definition name.

TOBKUPDFN: Specifies the name of the receiving Backup Definition. The receiving Backup Definition must not exist before being copied.

Backup-name	Enter the receiving Backup Definition.

Examples

```
CPYBKUPDFN FRMBKUPDFN(DAILY) TOBKUPDFN(WEEKLY)
```

This copies a Backup Definition named DAILY to Backup Definition named WEEKLY. Backup definition WEEKLY must not exist.

CPYBKUPL - Copy Backup List

```
---------------------------------------------------------------------------------------------------
                    Copy Backup List (CPYBKUPL)          Environment: B/I

From backup . . . . . .  _________   Name
From sequence . . . . . .  _________   1-9999
To backup . . . . . .  _________   Name, *FROMBKUP
To sequence . . . . . .  _________   1-9999
---------------------------------------------------------------------------------------------------
```

Purpose

The Copy Backup List (CPYBKUPL) command copies an existing Backup List to another. An error message is issued if the Backup List being copied to already exists.

Parameters

FROMBKUP: Specifies the name of the Backup that contains the Backup List to copy.

Backup-name	Enter a valid Backup name.

FROMSEQ: Specifies the sequence number of the Backup List to copy.

1-9999	Enter a value from 1 through 9999.

TOBKUP: Specifies the name of the target Backup.

*FROMBKUP	The Backup List is copied to the name specified in the **FROMBKUP** parameter.
Backup-name	Enter a Backup name.

TOSEQ: Specifies the target Backup List sequence number. An error message is issued if the sequence number already exists.

1-9999	Enter a value from 1 through 9999.

Examples

```
CPYBKUPL FROMBKUP(DAILY) FROMSEQ(1) TOBKUP(WEEKLY) TOSEQ(1)
```

This copies Backup List DAILY to Backup List WEEKLY. Backup List sequence 1 must not exist in Backup WEEKLY.

CPYBKUPLE - Copy Backup List Entry

```
-------------------------------------------------------------------------------------------------
                    Copy Backup List Entry (CPYBKUPLE)          Environment: B/I

From backup . . . . . . . . . . . . . . .  __________  Name
From sequence . . . . . . . . . . . . . .  __________  1-9999
From order number . . . . . . . . . . . .  __________  1-9999
To backup . . . . . . . . . . . . . . . .  __________  Name, *FROMBKUP
To sequence . . . . . . . . . . . . . . .  __________  1-9999
To order number . . . . . . . . . . . . .  __________  1-9999

-------------------------------------------------------------------------------------------------
```

Purpose

The Copy Backup List Entry (CPYBKUPLE) command copies one or more entries from one Backup List to another. Backup List entries can be libraries, objects, folders, links, output queues, save file data, spooled files and Domino servers.

Parameters

FROMBKUP: Specifies the name of the Backup that contains the Backup List entry to copy.

Backup-name	Enter a valid Backup name.

FROMSEQ: Specifies the sequence number of the Backup List entry to copy.

1-9999	Enter a value from 1 through 9999.

FROMORDNBR: Specifies the order number of the list entry to copy.

Order-number	Enter a value from 1-9999.

TOBKUP: Specifies the name of the target Backup.

*FROMBKUP	The Backup List is copied to the name specified in the **FROMBKUP** parameter.
Backup-name	Enter a Backup name.

TOSEQ: Specifies the target Backup List sequence number. An error message is issued if the sequence number already exists.

1-9999	Enter a value from 1 through 9999.

TOORDNBR: Specifies the order number of the list entry to copy.

Order-number	Enter a value from 1-9999.

Examples

```
CPYBKUPLE BACKUP(DAILY) FROMSEQ(5) FROMORDNBR(10) TOBKUP(WEEKLY) TOSEQ(10) +
    TOORDNBR(10)
```

This copies Backup List entry 5, order number 10 from Backup DAILY to sequence 10, order number 10 in Backup WEEKLY.

CPYRCY - Copy Recovery

```
---------------------------------------------------------------------------------------------------
                    Copy Recovery (CPYRCY)          Environment: B/I

From recovery . . . . . . . . . . . . . . .  _________    Name, *AUTO
To recovery. . . . . . . . . . . . . . . . .  _________    Name
---------------------------------------------------------------------------------------------------
```

Purpose

The Copy Recovery (CPYRCY) command copies a Recovery to another Recovery. The Recovery List and Recovery List entries are copied. An error message is sent if the Recovery being copied to already exists. Copying a Recovery automatically renumbers Recovery Lists and Recovery List entries.

Parameters

FROMRCY: Specifies the name of the Recovery to copy.

*AUTO	The Intelligent Recovery is selected.
Recovery-name	Enter a valid Recovery name.

TORCY: Specifies the name of the receiving Recovery. That Recovery name must not exist before being copied.

Recovery-name	Enter a Recovery name.

Examples

```
CPYRCY FROMRCY(NONSYS) TORCY(NONSYS2)
```

This copies a Recovery named NONSYS to Recovery named NONSYS2. Recovery name NONSYS2 must not exist.

CPYRCYDFN - Copy Recovery Definition

```
---------------------------------------------------------------------------------------------------
                    Copy Recovery Definition (CPYRCYDFN)          Environment: B/I

From recovery definition . . . . . _____________   Name, *DFT
To recovery definition. . . . . .  _____________   Name

---------------------------------------------------------------------------------------------------
```

Purpose

The Copy Recovery Definition (CPYRCYDFN) command copies an existing Recovery Definition to another. An error message is issued if the Recovery Definition being copied to already exists.

Parameters

FRMRCYDFN: Specifies the name of the Recovery Definition to copy.

*DFT	The default Recovery Definition is selected.
Recovery-definition	Enter a valid Recovery Definition name.

TORCYDFN: Specifies the name of the receiving Recovery Definition. The receiving Recovery Definition must not exist before being copied.

Recovery-definition	Enter the receiving Recovery Definition.

Examples

```
CPYRCYDFN FRMRCYDFN(TAPMLB01) TORCYDFN(TAPMLB02)
```

This copies a Recovery Definition named TAPMLB01 to Recovery Definition named TAPMLB02. Recovery definition TAPMLB02 must not exist.

CPYRCYL - Copy Recovery List

```
---------------------------------------------------------------------------------------------------
                    Copy Recovery List (CPYRCYL)      Environment: B/I

From recovery . . . . . .  _____________   Name, *AUTO
From sequence . . . . . .  _____________   1-9999
To recovery. . . . . . . .  _____________   Name, *FROMRCY
To sequence  . . . . . .  _____________   1-9999
---------------------------------------------------------------------------------------------------
```

Purpose

The Copy Recovery List (CPYRCYL) command copies an existing Recovery List to another. An error message is issued if the Recovery List being copied to already exists.

Parameters

FROMRCY: Specifies the name of the Recovery that contains the Recovery List to copy.

*AUTO	The Intelligent Recovery is selected.
Recovery-name	Enter a valid Recovery name.

FROMSEQ: Specifies the sequence number of the Recovery List to copy.

1-9999	Enter a value from 1 through 9999.

TORCY: Specifies the name of the target Recovery.

*FROMRCY	The Recovery List is copied to the name specified in the **FROMRCY** parameter.
Backup-name	Enter a Recovery name.

TOSEQ: Specifies the target Recovery List sequence number. An error message is issued if the sequence number already exists.

1-9999	Enter a value from 1 through 9999.

Examples

```
CPYRCYL FROMRCY(DAILY) FROMSEQ(1) TORCY(WEEKLY) TOSEQ(1)
```

This copies Recovery List DAILY to Recovery List WEEKLY. Recovery List sequence 1 must not exist in Backup WEEKLY.

CPYRCYLE - Copy Recovery List Entry

```
-------------------------------------------------------------------------------------------------
                    Copy Recovery List Entry (CPYRCYLE)          Environment: B/I

From recovery . . . . . . . . . . . . . .  __________  Name
From sequence . . . . . . . . . . . . . .  __________  1-9999
From order number . . . . . . . . . . . .  __________  1-9999
To recovery . . . . . . . . . . . . . . .  __________  Name, *FROMRCY
To sequence . . . . . . . . . . . . . . .  __________  1-9999
To order number . . . . . . . . . . . . .  __________  1-9999

-------------------------------------------------------------------------------------------------
```

Purpose

The Copy Recovery List Entry (CPYRCYLE) command copies one or more entries from one Recovery List to another. Recovery List entries can be libraries, objects, folders and links.

Parameters

FROMRCY: Specifies the name of the Recovery that contains the Recovery List entry to copy.

Recovery-name	Enter a valid Recovery name.

FROMSEQ: Specifies the sequence number of the Recovery List entry to move.

1-9999	Enter a value from 1 through 9999.

FROMORDNBR: Specifies the order number of the list entry to copy.

Order-number	Enter a value from 1-9999.

TORCY: Specifies the name of the target Recovery.

*FROMRCY	The Recovery List is copied to the name specified in the **FROMRCY** parameter.
Recovery-name	Enter a Recovery name.

TOSEQ: Specifies the target Recovery List sequence number. An error message is issued if the sequence number already exists.

1-9999	Enter a value from 1 through 9999.

TOORDNBR: Specifies the order number of the list entry to copy.

Order-number	Enter a value from 1-9999.

Examples

```
CPYRCYLE FROMRCY(DAILY) FROMSEQ(5) FROMORDNBR(10) TORCY(WEEKLY) TOSEQ(10)
TOORDNBR(5)
```

This copies Recovery List sequence entry 5, order number 10 from Recovery DAILY to sequence 10, order number 5 in Recovery WEEKLY.

DLTBKUP - Delete Backup

```
-------------------------------------------------------------------------------------------------
                    Delete Backup (DLTBKUP)                    Environment: B/I

Backup . . . . . .  _________          Name, *AUTO
-------------------------------------------------------------------------------------------------
```

Purpose

The Delete Backup (DLTBKUP) command deletes a Backup from the MMS Backup and Recovery. Deleting a Backup also removes any Backup Lists, Backup List entries and Omit List entries associated with the Backup.

Parameters

BACKUP: Specifies the Backup to delete.

*AUTO	The *AUTO backup is selected.
Backup-name	Enter a valid Backup name.

Examples

```
DLTBKUP BACKUP(DAILY)
```

This deletes a Backup named DAILY. If a Backup List and Backup List entries are associated with Backup DAILY, they are deleted.

DLTBKUPDFN - Delete Backup Definition

```
---------------------------------------------------------------------------------------------------
                    Delete Backup Definition (DLTBKUPDFN)                  Environment: B/I

Backup definition . . . . . .  _________          Name
---------------------------------------------------------------------------------------------------
```

Purpose

The Delete Backup Definition (DLTBKUPDFN) command deletes a Backup Definition from MMS Backup and Recovery. The default Backup Definition cannot be deleted.

Parameters

BKUPDFN: Specifies the Backup Definition to delete.

Backup-definition Enter a valid Backup Definition.

Examples

```
DLTBKUPDFN BKUPDFN(DAILY)
```

This deletes a Backup Definition named DAILY.

DLTOMITL - Delete Omit List

```
---------------------------------------------------------------------------------------------------
                    Delete Omit List  (DLTOMITL)          Environment: B/I

Omit list name . . . . . . . . . . . . . . . _____________  Name
List . . . . . . . . . . . . . . . . . . . . . _____________  *CLT, *FLR, *LIB, *LND…
---------------------------------------------------------------------------------------------------
```

Purpose

The Delete Omit List (DLTOMITL) command deletes an existing Omit List. Omit Lists define the name and type of object to omit from a Backup.

Parameters

OMITL: Specifies the name of the Omit List to delete.

List-name	Enter an Omit List name.

LIST: Specifies the type of Omit List .

*CLT	Client object Omit List.
*FLR	Document library object Omit List.
*LIB	Library Omit List.
*LND	Domino server Omit List.
*LNK	Integrated file system Omit List.
*OBJ	Object Omit List.
*OUTQ	Output queue Omit List.
*SYS	System data Omit List.

Examples

```
DLTOMITL OMITL(DAILY) LIST(*OBJ) TEXT('Object Omit List')
```

This deletes an Omit List named DAILY.

DLTRCY - Delete Recovery

```
---------------------------------------------------------------------------------------------------
                    Delete Recovery (DLTRCY)          Environment: B/I

Recovery . . . . . . . . . . . . . . . . . .  _________    Name, *AUTO
---------------------------------------------------------------------------------------------------
```

Purpose

The Delete Recovery (DLTRCY) command deletes a Recovery. Deleting a Recovery also removes any Recovery Lists, Recovery List entries and Omit List entries associated with the Recovery.

Parameters

RCY: Specifies the Recovery to delete.

*AUTO	The Intelligent Recovery is selected.
Recovery-name	Enter a valid Recovery name.

Examples

```
DLTRCY RCY(DAILY)
```

This deletes a Recovery named DAILY. If Recovery List and Recovery List entries are associated with Recovery DAILY, they are deleted.

DLTRCYDFN - Delete Recovery Definition

```
---------------------------------------------------------------------------------------------------
                    Delete Recovery Definition (DLTRCYDFN)                  Environment: B/I

Recovery definition . . . . . _________          Name
---------------------------------------------------------------------------------------------------
```

Purpose

The Delete Recovery Definition (DLTRCYDFN) command deletes a Recovery Definition. The default Recovery Definition cannot be deleted.

Parameters

RCYDFN: Specifies the Recovery Definition to delete.

Recovery-definition Enter a valid Recovery Definition.

Examples

```
DLTRCYDFN RCYDFN(TAPMLB01)
```

This deletes a Recovery Definition named TAPMLB01.

DSPBKUP - Display Backup

```
---------------------------------------------------------------------------------------------------
                    Display Backup (DSPBKUP)              Environment: B/I

Backup . . . . . .  _________                             Name
---------------------------------------------------------------------------------------------------
```

Purpose

The Display Backup (DSPBKUP) command lists the actual commands that will be executed when the backup runs.

Parameters

BACKUP: Specifies the Backup whose commands to show.

Backup-name Enter a valid Backup name.

Examples

```
DSPBKUP BACKUP(DAILY)
```

This displays the actual commands that will be executed when a backup named DAILY runs.

DSPRCY - Display Recovery

```
---------------------------------------------------------------------------------------------------
                        Display Recovery (DSPRCY)          Environment: B/I

Recovery . . . . . .  _________                            Name, *AUTO
---------------------------------------------------------------------------------------------------
```

Purpose

The Display Recovery (DSPRCY) command lists the actual commands that will be executed when the recovery runs.

Parameters

RCY: Specifies the Recovery whose commands to show.

*AUTO	The Intelligent Recovery is selected.
Recovery-name	Enter a valid Recovery name.

Examples

```
DSPRCY RCY(*AUTO)
```

This displays the actual commands that will be executed when *AUTO Recovery executes.

DSPRCYDFN - Display Recovery Definition

```
---------------------------------------------------------------------------------------------------
                    Display Recovery Definition (DSPRCYDFN)          Environment: B/I

Recovery definition . . . . . .  _________          Name, *DFT
---------------------------------------------------------------------------------------------------
```

Purpose

The Display Recovery Definition (DSPRCYDFN) command displays a Recovery Definition.

Parameters

RCYDFN:	Specifies the Recovery Definition to display.	
	*DFT	The default Recovery Definition is selected.
	Recovery-definition	Enter a valid Recovery Definition.

Examples

```
DSPRCYDFN RCYDFN(TAPMLB01)
```

This displays a Recovery Definition named TAPMLB01.

HLDBKUPL - Hold Backup List

```
--------------------------------------------------------------------------------------------------------
                    Hold Backup List (HLDBKUPL)      Environment: B/I

Backup . . . . . .  _________          Name
Sequence number . . . . . .  _________          1-9999

--------------------------------------------------------------------------------------------------------
```

Purpose

The Hold Backup List (HLDBKUPL) command holds a Backup List. Held Backup Lists are bypassed when the Backup runs. Backup Lists cannot be held if a Backup is running.

Parameters

BACKUP: Specifies the name of the Backup.

Backup-name	Enter a valid Backup name.

SEQNBR: Specifies the sequence number of the Backup List to hold.

1-9999	Enter a value from 1 through 9999.

Examples

```
HLDBKUPL BACKUP(DAILY) SEQNBR(10)
```

This holds the Backup List at sequence 10 on Backup DAILY. The held Backup List will not execute when the Backup runs.

HLDRCYL - Hold Recovery List

```
---------------------------------------------------------------------------------------------------
                    Hold Recovery List (HLDRCYL)        Environment: B/I

Recovery . . . . . . . . . . . . . . . . . .  _________      Name, *AUTO
Sequence number . . . . . . . . . . . .  _________      1-9999

---------------------------------------------------------------------------------------------------
```

Purpose

The Hold Recovery List (HLDRCYL) command holds a Recovery List. Held Recovery Lists are bypassed when the Recovery runs. Recovery Lists cannot be held if a Recovery is running.

Parameters

RCY: Specifies the name of the Recovery.

*AUTO	The Intelligent Recovery is selected.
Recovery-name	Enter a valid Recovery name.

SEQNBR: Specifies the sequence number of the Recovery List to hold.

1-9999	Enter a value from 1 through 9999.

Examples

```
HLDRCYL RCY(DAILY) SEQNBR(10)
```

This holds the Recovery List at sequence 10 on Recovery DAILY. The held Recovery List will not execute when the Recovery runs.

MOVBKUPL - Move Backup List

```
---------------------------------------------------------------------------------------------------
                    Move Backup List (MOVBKUPL)     Environment: B/I

From backup . . . . . .  _________     Name
From sequence . . . . . .  _________     1-9999
To backup . . . . . .  _________     Name, *FROMBKUP
To sequence . . . . . .  _________     1-9999
---------------------------------------------------------------------------------------------------
```

Purpose

The Move Backup List (MOVBKUPL) command moves an existing Backup List to another. An error message is issued if the Backup List being moved to already exists.

Parameters

FROMBKUP: Specifies the name of the Backup that contains the Backup List to move.

Backup-name	Enter a valid Backup name.

FROMSEQ: Specifies the sequence number of the Backup List to move.

1-9999	Enter a value from 1 through 9999.

TOBKUP: Specifies the name of the target Backup.

*FROMBKUP	The Backup List is copied to the name specified in the **FROMBKUP** parameter.
Backup-name	Enter a Backup Definition.

TOSEQ: Specifies the target Backup List sequence number. An error message is issued if the sequence number already exists.

1-9999	Enter a value from 1 through 9999.

Examples

```
MOVBKUPL FROMBKUP(DAILY) FROMSEQ(1) TOBKUP(WEEKLY) TOSEQ(1)
```

This moves Backup List DAILY to Backup List WEEKLY. Backup List sequence 1 must not exist in Backup WEEKLY.

MOVBKUPLE - Move Backup List Entry

```
-------------------------------------------------------------------------------------------------
                    Move Backup List Entry (MOVBKUPLE)          Environment: B/I

From backup . . . . . . . . . . . . . . .  __________  Name
From sequence . . . . . . . . . . . . . .  __________  1-9999
From order number . . . . . . . . . . . .  __________  1-9999
To backup . . . . . . . . . . . . . . . .  __________  Name, *FROMBKUP
To sequence . . . . . . . . . . . . . . .  __________  1-9999
To order number . . . . . . . . . . . . .  __________  1-9999

-------------------------------------------------------------------------------------------------
```

Purpose

The Move Backup List Entry (MOVBKUPLE) command moves one or more entries from one Backup List to another. Backup List entries can be libraries, objects, folders, links, output queues, save file data, spooled files and Domino servers.

Parameters

FROMBKUP: Specifies the name of the Backup that contains the Backup List entry to move.

Backup-name	Enter a valid Backup name.

FROMSEQ: Specifies the sequence number of the Backup List entry to move.

1-9999	Enter a value from 1 through 9999.

FROMORDNBR: Specifies the order number of the list entry to move.

Order-number	Enter a value from 1-9999.

TOBKUP: Specifies the name of the target Backup.

*FROMBKUP	The Backup List is copied to the name specified in the **FROMBKUP** parameter.
Backup-name	Enter a Backup Definition.

TOSEQ: Specifies the target Backup List sequence number. An error message is issued if the sequence number already exists.

1-9999	Enter a value from 1 through 9999.

TOORDNBR: Specifies the order number of the list entry to move.

Order-number	Enter a value from 1-9999.

Examples

```
MOVBKUPLE BACKUP(DAILY) FROMSEQ(5) FROMORDNBR(10) TOBKUP(WEEKLY) TOSEQ(10) +
    TOORDNBR(10)
```

This moves Backup List sequence entry 5, order number 10 from Backup DAILY to sequence 10, order 10 in Backup WEEKLY.

MOVRCYL - Move Recovery List

```
---------------------------------------------------------------------------------------------------
                    Move Recovery List (MOVRCYL)    Environment: B/I

From recovery . . . . . . . . . . . . . . .  _________   Name, *AUTO
From sequence . . . . . . . . . . . . . . .  _________   1-9999
To recovery . . . . . . . . . . . . . . . .  _________   Name, *FROMRCY
To sequence . . . . . . . . . . . . . . . .  _________   1-9999

---------------------------------------------------------------------------------------------------
```

Purpose

The Move Recovery List (MOVRCYL) command moves an existing Recovery List to another. An error message is issued if the Recovery List being moved to already exists.

Parameters

FROMRCY: Specifies the name of the Recovery that contains the Recovery List to move.

*AUTO	The Intelligent Recovery is selected.
Recovery-name	Enter a valid Recovery name.

FROMSEQ: Specifies the sequence number of the Recovery List to move.

1-9999	Enter a value from 1 through 9999.

TORCY: Specifies the name of the target Recovery.

*FROMRCY	The Recovery List is copied to the name specified in the **FROMRCY** parameter.
Recovery-name	Enter a Recovery Definition.

TOSEQ: Specifies the target Recovery List sequence number. An error message is issued if the sequence number already exists.

1-9999	Enter a value from 1 through 9999.

Examples

```
MOVRCYL FROMRCY(DAILY) FROMSEQ(1) TORCY(WEEKLY) TOSEQ(1)
```

This moves Recovery List DAILY to Recovery List WEEKLY. Recovery List sequence 1 must not exist in Recovery WEEKLY.

MOVRCYLE - Move Recovery List Entry

```
---------------------------------------------------------------------------------------------------
                    Move Recovery List Entry (MOVRCYLE)          Environment: B/I

From recovery . . . . . . . . . . . . . .  _____________  Name
From sequence . . . . . . . . . . . . . .  _____________  1-9999
From order number . . . . . . . . . . . .  _____________  1-9999
To recovery . . . . . . . . . . . . . . .  _____________  Name, *FROMRCY
To sequence . . . . . . . . . . . . . . .  _____________  1-9999
To order number . . . . . . . . . . . . .  _____________  1-9999

---------------------------------------------------------------------------------------------------
```

Purpose

The Move Recovery List Entry (MOVRCYLE) command moves one or more entries from one Recovery List to another. Recovery List entries can be libraries, objects, folders and links.

Parameters

FROMRCY: Specifies the name of the Recovery that contains the Recovery List entry to move.

Recovery-name	Enter a valid Recovery name.

FROMSEQ: Specifies the sequence number of the Recovery List entry to move.

1-9999	Enter a value from 1 through 9999.

FROMORDNBR: Specifies the order number of the list entry to move.

Order-number	Enter a value from 1-9999.

TORCY: Specifies the name of the target Recovery.

*FROMRCY	The Recovery List is copied to the name specified in the **FROMRCY** parameter.
Recovery-name	Enter a Recovery name.

TOSEQ: Specifies the target Recovery List sequence number. An error message is issued if the sequence number already exists.

1-9999	Enter a value from 1 through 9999.

TOORDNBR: Specifies the order number of the list entry to move.

Order-number	Enter a value from 1-9999.

Examples

```
MOVRCYLE FROMRCY(DAILY) FROMSEQ(5) FROMORDNBR(10)TORCY(WEEKLY) TOSEQ(10) +
    TOORDNBR(5)
```

This moves Recovery List sequence entry 5, order number 10 from Recovery DAILY to sequence 10, order number 5 in Recovery WEEKLY.

RLSBKUPL - Release Backup List

```
---------------------------------------------------------------------------------------------------
                    Release Backup List (RLSBKUPL)      Environment: B/I

Backup . . . . . .  _________          Name
Sequence number . . . . . .  _________ 1-9999
---------------------------------------------------------------------------------------------------
```

Purpose

The Release Backup List (RLSBKUPL) command releases a previously held Backup List. Held Backup Lists cannot be released if the Backup is running.

Parameters

BACKUP: Specifies the name of the Backup.

Backup-name	Enter a valid Backup name.

SEQNBR: Specifies the sequence number of the Backup List to release.

1-9999	Enter a value from 1 through 9999.

Examples

```
RLSBKUPL BACKUP(DAILY) SEQNBR(10)
```

This releases the Backup List at sequence 10 on Backup DAILY.

RLSRCYL - Release Recovery List

```
---------------------------------------------------------------------------------------------------
                    Release Recovery List (RLSRCYL)     Environment: B/I

Recovery . . . . . .  _________          Name, *AUTO
Sequence number  . . . . . .  _________  1-9999

---------------------------------------------------------------------------------------------------
```

Purpose

The Release Recovery List (RLSRCYL) command releases a previously held Recovery List. Held Recovery Lists are bypassed when the Recovery runs. Held Recovery Lists cannot be released if a Recovery is running.

Parameters

RCY: Specifies the name of the Recovery.

*AUTO	The Intelligent Recovery is selected.
Recovery-name	Enter a valid Recovery name.

SEQNBR: Specifies the sequence number of the Recovery List to release.

1-9999	Enter a value from 1 through 9999.

Examples

```
RLSRCYL RCY(DAILY) SEQNBR(10)
```

This releases the Recovery List at sequence 10 on Recovery DAILY.

RMVBKUPL - Remove Backup List

```
---------------------------------------------------------------------------------------------------
                    Remove Backup List (RMVBKUPL)          Environment: B/I

Backup . . . . . . . . . . . . . . . . . . . . .  _________   Name
Sequence number . . . . . . . . . . . . . .  ____   1-9999
---------------------------------------------------------------------------------------------------
```

Purpose

The Remove Backup List (RMVBKUPL) command removes a Backup List from the specified Backup and sequence number. All associated Backup List entries and Omit List entries are also removed.

Parameters

BACKUP: Specifies the name of the Backup.

Backup-name Enter a valid Backup name.

SEQNBR: Specifies the sequence number of this Backup List entry.

Sequence-number Enter a value from 1 to 9999.

Examples

```
RMVBKUPLE BACKUP(DAILY) SEQNBR(5)
```

This removes the Backup List at sequence number 5 from Backup DAILY.

RMVBKUPLE - Remove Backup List Entry

```
----------------------------------------------------------------------------------------------------
                    Remove Backup List Entry (RMVBKUPLE)          Environment: B/I

Backup . . . . . . . . . . . . . . . . . . . .  __________  Name
Sequence number . . . . . . . . . . . . . . .  __________  1-9999
List . . . . . . . . . . . . . . . . . . . . .  __________  *ASP, *ASPDLO, *CLT, *FLR...
Order number . . . . . . . . . . . . . . . . .  ____        1-9999

----------------------------------------------------------------------------------------------------
```

Purpose

The Remove Backup List Entry (RMVBKUPLE) command removes an entry from the specified Backup List.

Parameters

BACKUP: Specifies the name of the Backup that is associated with this Backup List entry.

Backup-name	Enter a valid Backup name.

SEQNBR: Specifies the sequence number of this Backup List entry.

Sequence-number	Enter a number from 1-9999.

LIST: Specifies the type of Backup List.

*ASP	Auxiliary storage pool (ASP) list.
*ASPDLO	Document library objects in the specified auxiliary storage pool (ASP) list.
*CLT	Client object list.
*EXIT	Command list.
*FLR	Document library object list.
*LIB	Library list.
*LND	Lotus Notes/Domino server list.
*LNK	Integrated file system list.
*OBJ	Objects list.
*OUTQ	Output queue list.
*SAVF	Save file list.
*SPL	Spooled file list.

ORDNBR: Specifies the order number of the list entry to remove.

Order-number	Enter a value from 1-9999.

Examples

```
RMVBKUPLE BACKUP(DAILY) SEQNBR(5) LIST(*LIB) ORDNBR(10)
```

This removes a Backup List entry for a Backup named DAILY.

RMVBKUPSTS - Remove Backup Status

```
---------------------------------------------------------------------------------------------------
                    Remove Backup Status (RMVBKUPSTS)          Environment: B/I

Backup . . . . . . . . . . . . . . . . . . . .  _________      Name, *AUTO
---------------------------------------------------------------------------------------------------
```

Purpose

The Remove Backup Status (RMVBKUPSTS) command removes the backup status entry for a failed Backup.

Parameters

BACKUP: Specifies the name of the Backup.

*AUTO	The Intelligent Backup is selected.
Backup-name	Enter a valid Backup name.

Examples

```
RMVBKUPSTS BACKUP(DAILY)
```

This removes backup status entry from a Backup named DAILY.

RMVOMITLE - Remove Omit List Entry

```
------------------------------------------------------------------------------------------------------
                    Remove Omit List Entry (RMVOMITLE)          Environment: B/I

Omit list name . . . . . . . . . . . . . . .   __________      Name
List . . . . . . . . . . . . . . . . . . . . .   __________      *CLT, *FLR, *LIB, *LND…
Object . . . . . . . . . . . . . . . . . . . .   __________      Name, generic*, *ALL
Library . . . . . . . . . . . . . . . . . . .   __________      Name, generic*
Object type . . . . . . . . . . . . . . . . .   ________        *ALL, *LIB, *ALRTBL…
Output queue . . . . . . . . . . . . . . . .   __________      Name
  Library . . . . . . . . . . . . . . . . . .     __________      Name
Path name . . . . . . . . . . . . . . . . . .   ____________________    Path name
System data . . . . . . . . . . . . . . . . .   __________      *CFG, *SECDTA

------------------------------------------------------------------------------------------------------
```

Purpose

The Remove Omit List Entry (RMVOMITLE) command removes one or more items from an Omit List. Omit Lists are used to omit specific items from a Backup.

Parameters

OMITL: Specifies the name of the Omit List.

List-name	Enter an Omit List name.

LIST: Specifies the type of Omit List.

*CLT	The list is a Client object Omit List.
*FLR	The list is a document library object Omit List.
*LIB	The list is a library Omit List.
*LND	The list is a Domino server Omit List.
*LNK	The list is a integrated file system object Omit List.
*OBJ	The list is an object Omit List.
*OUTQ	The list is an output queue Omit List.
*SYS	The list is a system data Omit List.

OBJ: Specifies the name of the objects to remove from an ***OBJ** Omit List.

*ALL	All objects from the specified library are selected.
generic*	Enter the generic name of the objects to select. A generic name is a character string that contains one or more characters followed by an asterisk (*).
Object-name	Enter a valid object name.

LIB: Specifies the name of the libraries to remove from ***LIB** Omit List.

generic*	Enter the generic name of the libraries to select. A generic name is a character string that contains one or more characters followed by an asterisk (*).
Library-name	Enter a valid library name.

OBJTYPE: Specifies which types of objects to remove from an ***OBJ** Omit List.

*ALL	All object types are removed.
Object-type	Enter a valid object type.

PATH: Specifies the name of the path to remove from either a ***CLT**, ***FLR**, ***LND** or ***LNK** Omit List.

Path-name	Enter a valid path name.

OUTQ: Specifies the output queue to remove from a ***OUTQ** Omit List.

Output queue:

Output-queue	Enter the name of a valid output queue.

Library:

Library-name	Enter a valid library name.

SYSDTA: Specifies the type of system data to remove from a ***SYS** Omit List.

*CFG	Configuration data is selected.
*SECDTA	Security data is selected.

Examples

```
RMVOMITLE BACKUP(DAILY) SEQNBR(5) OBJ(PRODLIB/*ALL) +
    OBJTYPE(*ALL)
```

This removes an Omit List entry for a Backup named DAILY. The Omit List entry specifies that all objects from library PRODLIB are to be omitted.

RMVRCYL - Remove Recovery List

```
-----------------------------------------------------------------------------------------------------
                          Remove Recovery List (RMVRCYL)       Environment: B/I

Recovery . . . . . . . . . . . . . . . . . . . . . .   __________                  Name, *AUTO
Sequence number . . . . . . . . . . . . . .   ____                         1-9999

-----------------------------------------------------------------------------------------------------
```

Purpose

The Remove Recovery List (RMVRCYL) command removes a Recovery List from the specified Recovery and sequence number. All associated Recovery List entries and Omit List entries are also removed.

Parameters

RCY: Specifies the name of the Recovery.

*AUTO	The Intelligent Recovery is selected.
Recovery-name	Enter a valid Recovery name.

SEQNBR: Specifies the sequence number of this Recovery List.

Sequence-number	Enter a value from 1 to 9999.

Examples

```
RMVRCYL BACKUP(DAILY) SEQNBR(5)
```

This removes the Recovery List at sequence number 5 from Backup DAILY.

RMVRCYLE - Remove Recovery List Entry

```
---------------------------------------------------------------------------------------------------
                    Remove Recovery List Entry (RMVRCYLE)    Environment: B/I

Recovery . . . . . .  _________   Name
Sequence number . . . . . .  ____   1-9999
List . . . . . .  _________   *FLR, *LIB, *LNK, *OBJ…
Order number . . . . . .  ____   1-9999

---------------------------------------------------------------------------------------------------
```

Purpose

The Remove Recovery List Entry (RMVRCYLE) command removes an entry from a Recovery List. Recovery List entries can be exits, folders, libraries, links, and objects.

Parameters

RCY: Specifies the name of the Recovery that is associated with this Recovery List entry.

Recovery-name	Enter a valid Recovery name.

SEQNBR: Specifies the sequence number of this Recovery List entry.

Sequence-number	Enter a number from 1-9999.

LIST: Specifies the type of Recovery List.

*EXIT	Command list.
*FLR	Document library object list.
*LIB	Library list.
*LNK	Integrated file system list.
*OBJ	Object list.

ORDNBR: Specifies the order number of the list entry to remove.

Order-number	Enter a value from 1-9999.

Examples

```
RMVRCYLE RCY(NONSYS) SEQNBR(5) LIST(*LIB) ORDNBR(10)
```

This removes a Recovery List entry for a Recovery named NONSYS.

RNMBKUP - Rename Backup

```
---------------------------------------------------------------------------------------------------
                    Rename Backup (RNMBKUP)              Environment: B/I

From backup . . . . . . . . . . . . . . . .  _________    Name, *AUTO
To backup . . . . . . . . . . . . . . . . .  _________    Name
---------------------------------------------------------------------------------------------------
```

Purpose

The Rename Backup (RNMBKUP) command renames a Backup to another name. The Backup List and Backup List entries are renamed. An error message is sent if the new Backup name already exists. Renaming a Backup automatically renumbers Backup Lists and Backup List entries.

Parameters

FROMBKUP: Specifies the name of the Backup to rename.

*AUTO	The Intelligent Backup is selected.
Backup-name	Enter a valid Backup name.

TOBKUP: Specifies the new Backup name. The new Backup name must not exist before being renamed.

Backup-name	Enter a Backup name.

Examples

```
RNMBKUP FROMBKUP(DAILY) TOBKUP(WEEKLY)
```

This renames a Backup named DAILY to Backup named WEEKLY. Backup name WEEKLY must not exist.

RNMRCY - Rename Recovery

```
---------------------------------------------------------------------------------------------------
                    Rename Backup (RNMRCY)          Environment: B/I

From recovery . . . . . . . . . . . . .  _________      Name
To recovery. . . . . . . . . . . . . . .  _________      Name

---------------------------------------------------------------------------------------------------
```

Purpose

The Rename Recovery (RNMRCY) command renames a Recovery to another name. The Recovery List and Recovery List entries are renamed. An error message is sent if the new Recovery name already exists. Renaming a Recovery automatically renumbers Recovery Lists and Recovery List entries.

Parameters

FROMRCY: Specifies the name of the Recovery to rename.

Recovery-name Enter a valid Recovery name.

TORCY: Specifies the new Recovery name. The new Recovery name must not exist before being renamed.

Recovery-name Enter a Recovery name.

Examples

```
RNMRCY FROMRCY(DAILY) TORCY(WEEKLY)
```

This renames a Recovery named DAILY to Recovery named WEEKLY. Recovery name WEEKLY must not exist.

RSTALLLIB - Restore All Libraries

```
-------------------------------------------------------------------------------------------------
                    Restore All Libraries (RSTALLLIB)          Environment: B/I

Device . . . . . .  _________          Name
Volume identifier . . . . . .  *MOUNTED          Char, *MOUNTED
        + for more values  _________
Sequence number  . . . . . .  __1          Number
End of tape option . . . . . .  *REWIND          *REWIND, *LEAVE, *UNLOAD
Option . . . . . .  *ALL          *ALL, *NEW, *OLD, *FREE
Allow object differences . . . . . .  *ALL          *ALL, *NONE
Library to include  . . . . . .  *ALL          Name, *ALL
        + for more values  _____
Library to omit  . . . . . .  *NONE          Name, *NONE
        + for more values  _____
Output . . . . . .  *NONE          *NONE, *PRINT

-------------------------------------------------------------------------------------------------
```

Purpose

The Restore All Libraries (RSTALLLIB) command restores one or more libraries and their objects from tape to disk. Command options determine if libraries existing on disk are bypassed or replaced. This command will restore libraries saved with a Save Library (SAVLIB) command from tape. Up to 300 libraries can be omitted from the restore process.

Parameters

DEVICE: Specifies the name of the tape device to use.

Device	Enter a valid tape device name.

VOL: Specifies the volume identifiers from which the objects are restored. The volumes must be placed in the device in the same order as they were when the objects were saved.

***MOUNTED**	The objects are restored from the volumes currently loaded in the device.
Volume-id	Enter a valid volume identifier.

SEQNBR: Specifies the starting sequence number to use for the restore process. This command will automatically increment the sequence number during the restore.

1	The restore starts from sequence number 1.
Number	Enter a value from 1 to 16777215.

ENDOPT: Specifies the tape action to take when processing is complete.

***REWIND**	The tape is rewound to the starting point.
*LEAVE	The tape is not rewound.
*UNLOAD	The tape is rewound and unloaded.

OPTION: Specifies how to handle restoring each object.

***ALL**	All the objects in the saved library are restored.
*NEW	Only objects which do not exist in the specified library are restored. If the object resides in the library on disk, the saved object is not restored.
*OLD	Only objects which exist in the library on disk and are saved on tape are restored. If the saved object on tape does not reside in the same library on disk, it is not restored.
*FREE	Only objects which exist in the library on disk, but have their storage freed, and are saved on tape are restored. If the object on disk does not have its storage freed, it is not restored.

ALWOBJDIF: Specifies whether certain differences encountered during a restore operation are allowed. The differences include:

- Ownership--the owner of the object on the system is different from the owner of the object from the save operation.
- File creation date--the creation date of the database file on the system does not match the creation date of the file that was saved.
- Member creation date--the creation date of the database file member on the system does not match the creation date of the member that was saved.
- Validation value verification--The validation value created at the time an object was created does not match the validation value created during the restore operation of an object on a system with a **QSECURITY** level of 40 or higher.
- Authorization list linking--the object is being restored to a system different from the one on which it was saved.

Note: To use this parameter, you need ***ALLOBJ** special authority.

***ALL** All of the differences listed above are allowed for the restore operation. An informational message is sent, except for validation value verification and authorization list linking cases, and the object is restored. The following should be noted:

- The informational message triggers a diagnostic message to be sent indicating that security or integrity changes occurred during the restore operation. The final message for the restore operation is an escape message, rather than the normal completion message.
- If the media and system owner of the object do not match, the system owner becomes the owner of the object.
- If there is a file level mismatch and ***ALL** is specified on this parameter and the Data base member option prompt (**MBROPT** parameter), the existing version of the file is renamed and the saved version of the file is restored. If there is a member level mismatch, the existing version of the member is renamed and the saved version of the member is restored.
- If the system security level is 40 and you are restoring a program and you specify ***ALL**, and the program's validation value is missing or incorrect, the program is restored without authority changes.
- For programs without a validation value, specifying ***ALL** also prevents the system from attempting to translate the program again.
- If you are restoring objects to a system different from the one on which they were saved and the objects are secured by an authorization list, specifying ***ALL** automatically links the objects to the authorization list again. If the authorization list does not exist on the new system, a message that includes the name of the missing list is issued and the public authority is set to ***EXCLUDE**.

*NONE None of the differences described above are allowed on the restore operation. For validation value verification failure cases, the object is restored but ownership is transferred to **QDFTOWN** and all authorities are revoked. For authorization list cases, the object is restored, but the object is not linked to the authorization list, and public authority is set to ***EXCLUDE**. For all other cases, a diagnostic message is sent for the object, and the object is not restored.

INCLLIB: Specifies up to 300 libraries to include in the restore process. If library names are specified, the **OMITLIB** parameter value must be ***NONE**.

***ALL**	No libraries are selected.
Library	Enter the library to include.

OMITLIB: Specifies up to 300 libraries to omit from the restore process.

***NONE**	No libraries are omitted.
Library	Enter the library to omit.

OUTPUT: Specifies whether a listing that shows information about the status of the objects is created and directed to an output file. The listing shows the restore information and shows all objects restored, not restored, and excluded. Information about each object's security is listed for the restored objects.

***NONE**	No spooled output is created.
*PRINT	The output is printed to the job's spooled output.

Examples

```
RSTALLLIB DEV(TAP01) OPTION(*ALL) ENDOPT(*UNLOAD)
```

This restores all libraries found on the volume mounted on TAP01 and saved with the Save Library (SAVLIB) command. If the libraries found on the tape exist on disk, they are replaced.

```
RSTALLLIB DEV(TAP01) OPTION(*NEW) ENDOPT(*UNLOAD)
```

This restores libraries found on the volume mounted on TAP01 and saved with the Save Library (SAVLIB) command that do not exist on disk. If the libraries found on the tape exist on disk, they are bypassed.

RSTALLOBJ - Restore All Objects

```
                        Restore All Objects (RSTALLOBJ)          Environment: B/I

Device . . . . . .  _________          Name
Volume identifier . . . . . .  *MOUNTED          Char, *MOUNTED
      + for more values  _________
Sequence number . . . . . .  __1          Number
End of tape option . . . . . .  *REWIND          *REWIND, *LEAVE, *UNLOAD
Option . . . . . .  *ALL          *ALL, *NEW, *OLD, *FREE
Allow object differences . . . . . .  *ALL          *ALL, *NONE
Library to include . . . . . .  *ALL          Name, *ALL
      + for more values  _____
Library to omit . . . . . .  *NONE          Name, *NONE
      + for more values  ______
Output . . . . . .  *NONE          *NONE, *PRINT
```

Purpose

The Restore All Objects (RSTALLOBJ) command restores saved objects from tape to disk. The libraries, whose objects are being restored, must exist on disk prior to executing this command. Command options determine if objects existing on disk are bypassed or replaced. This command will restore objects saved with the SAVCHGOBJ or SAVOBJ command from tape. Up to 300 libraries can be omitted from the restore process.

Parameters

DEVICE: Specifies the name of the tape device to use.

Device	Enter a valid tape device name.

VOL: Specifies the volume identifiers from which the objects are restored. The volumes must be placed in the device in the same order as they were when the objects were saved.

***MOUNTED**	The objects are restored from the volumes currently loaded in the device.
Volume-id	Enter a valid volume identifier.

SEQNBR: Specifies the starting sequence number to use for the restore process. This command will automatically increment the sequence number during the restore.

1	The restore starts from sequence number 1.
Number	Enter a value from 1 to 16777215.

ENDOPT: Specifies the tape action to take when processing is complete.

***REWIND**	The tape is rewound to the starting point.
*LEAVE	The tape is not rewound.
*UNLOAD	The tape is rewound and unloaded.

OPTION: Specifies how to handle restoring each object.

***ALL**	All the objects in the saved library are restored.
*NEW	Only objects which do not exist in the specified library are restored. If the object resides in the library on disk, the saved object is not restored.
*OLD	Only objects which exist in the library on disk and are saved on tape are restored. If the saved object on tape does not reside in the same library on disk, it is not restored.
*FREE	Only objects which exist in the library on disk, but have their storage freed, and are saved on tape are restored. If the object on disk does not have its storage freed, it is not restored.

ALWOBJDIF: Specifies whether certain differences encountered during a restore operation are allowed. The differences include:

- Ownership--the owner of the object on the system is different from the owner of the object from the save operation.
- File creation date--the creation date of the database file on the system does not match the creation date of the file that was saved.
- Member creation date--the creation date of the database file member on the system does not match the creation date of the member that was saved.
- Validation value verification--The validation value created at the time an object was created does not match the validation value created during the restore operation of an object on a system with a QSECURITY level of 40 or higher.
- Authorization list linking--the object is being restored to a system different from the one on which it was saved.

Note: To use this parameter, you need *ALLOBJ special authority.

***ALL** All of the differences listed above are allowed for the restore operation. An informational message is sent, except for validation value verification and authorization list linking cases, and the object is restored. The following should be noted:

- The informational message triggers a diagnostic message to be sent indicating that security or integrity changes occurred during the restore operation. The final message for the restore operation is an escape message, rather than the normal completion message.
- If the media and system owner of the object do not match, the system owner becomes the owner of the object.
- If there is a file level mismatch and ***ALL** is specified on this parameter and the Data base member option prompt (**MBROPT** parameter), the existing version of the file is renamed and the saved version of the file is restored. If there is a member level mismatch, the existing version of the member is renamed and the saved version of the member is restored.
- If the system security level is 40, you are restoring a program, you specify ***ALL**, and the program's validation value is missing or incorrect, the program is restored without authority changes.
- For programs without a validation value, specifying ***ALL** also prevents the system from attempting to translate the program again.
- If you are restoring objects to a system different from the one on which they were saved and the objects are secured by an authorization list, specifying ***ALL** automatically links the objects to the authorization list again. If the authorization list does not exist on the new system, a message that includes the name of the missing list is issued and the public authority is set to ***EXCLUDE**.

*NONE	None of the differences described above are allowed on the restore operation. For validation value verification failure cases, the object is restored but ownership is transferred to QDFTOWN and all authorities are revoked. For authorization list cases, the object is restored, but the object is not linked to the authorization list, and public authority is set to *EXCLUDE. For all other cases, a diagnostic message is sent for the object, and the object is not restored.

INCLLIB: Specifies up to 300 libraries to include in the restore process. If library names are specified, the **OMITLIB** parameter value must be ***NONE**.

***ALL**	No libraries are selected.
Library	Enter the library to include.

OMITLIB: Specifies up to 300 libraries to omit from the restore process.

***NONE**	No libraries are omitted.
Library	Enter the library to omit.

OUTPUT: Specifies whether a listing that shows information about the status of the objects is created and directed to an output file. The listing shows the restore information and shows all objects restored, not restored, and excluded. Information about each object's security is listed for the restored objects.

***NONE**	No spooled output is created.
*PRINT	The output is printed to the job's spooled output.

Examples

```
RSTALLOBJ DEV(TAP01) OPTION(*ALL) ENDOPT(*UNLOAD)
```

This restores all objects for all libraries found on the volume mounted on TAP01 and saved with the Save Object (SAVOBJ) or Save Changed Object (SAVCHGOBJ) command. If the objects are found in the libraries on disk, they are replaced.

```
RSTALLOBJ DEV(TAP01) OPTION(*NEW) ENDOPT(*UNLOAD)
```

This restores objects for all libraries found on the volume mounted on TAP01 and saved with the Save Object (SAVOBJ) or Save Changed Object (SAVCHGOBJ) command. If the objects are found in the libraries on disk, they are bypassed.

RUNBKUP - Run Backup

Run Backup (RUNBKUP)		Environment: B/I
Backup	__________	Name, *AUTO
Backup sequence:		
Beginning.	*ALL	1-9999, *ALL
Ending.	_____	1-9999, *ONLY, *END
Parallel device resources:		
Minimum resources	*NONE	1-32, *NONE, *AVAIL
Maximum resources	______	1-32, *AVAIL. *MIN
Change % threshold	80	10-100
Backup sort sequence	*NAME	*NAME, *USAGE
Save MMS libraries	*YES	*YES, *NO
Subsystems to end	*NONE	Name, *ALL, *NONE
+ for more values	_______	
Resume	*NO	*NO, *YES
Job label	*DFT	Name, *DFT
Start date	*CURRENT	Date, *CURRENT
Start time	*CURRENT	Time, *CURRENT
Start subsystems	*YES	*YES, *NO
Signoff	*NO	*NO, *YES

Purpose

The Run Backup (RUNBKUP) command initiates a Backup. Options provide the ability to end some or all subsystems prior to starting the Backup. Additionally, failed Backups can be resumed using the resume Backup option. If a Backup is resumed, it will start with the failing sequence number and continue until the Backup completes. Backups can be run in their entirety or specific backup sequences can be specified.

Parameters

BACKUP: Specifies the name of the Backup to process.

*AUTO — The Intelligent Backup is selected.

Backup-name — Enter a valid Backup name.

SEQNBR: Specifies the beginning and ending Backup sequence to process.

Beginning:

***ALL** — All Backup sequences are processed.

Beginning-sequence — Enter the beginning sequence.

Ending:

*END — The last Backup List entry is the ending sequence.

*ONLY — The beginning sequence is the only sequence processed.

Ending-sequence — Enter the ending sequence.

DRVRSC: Specifies the minimum and maximum number of device resources to use in a parallel save. This parameter is valid for ***ASP**, ***LIB** and ***OBJ** Backup Lists in the ***AUTO** Backup.

Minimum resources:

***NONE**	No device resources are used. The save is performed as a serial save.
*AVAIL	Use any available resources up to the maximum specified. This will use any available resource but will complete using one resource if only one is available.
1-32	Enter the minimum number of resources to use.

Maximum resources:

*MIN	Uses the value specified for the minimum number of device resources.
*AVAIL	The save will use any available device resources but at minimum, the value specified in the minimum element.
1-32	Enter the minimum number of resources to use.

CHGPCT: Specifies the number of objects that must change before a full save is performed. This parameter applies to ***LIB**, ***FLR** and ***LNK** Backup Lists in the ***AUTO** Backup.

80	A full save is performed if 80 % or more of the objects have changed.
Change-percent	Enter a value between 10 and 100. If 100 is specified, a full save is performed regardless of the number of changed objects.

SRTSEQ: Specifies the order in which libraries are saved. This parameter applies to ***LIB** Backup Lists in the ***AUTO** Backup.

***NAME**	Libraries are saved alphabetically.
*USAGE	Libraries are saved based on the total "Days used" count for all objects in the library.

SAVMMS: Specifies whether to save the MMS libraries need for recovery after the ***AUTO** Backup completes.

***YES**	The MMS recovery libraries are saved.
*NO	The MMS recovery libraries are not saved.

ENDSBS: Specifies whether to end subsystems before starting the Backup. Up to 50 subsystems can be specified. If ***ALL** is specified, the Backup is performed interactively. Any subsystem that was active and ended by this parameter is restarted after the Backup completes.

***NONE**	No subsystems are ended.
*ALL	All subsystems are ended.
Subsystem-name	Enter up to 50 subsystem names.

RESUME: Specifies whether to resume a failed Backup. A failed Backup can be viewed using the Work with Backup Status (**WRKBKUPSTS**) command. **Note**: MMS or MMS/*tms* must be installed to use this feature.

***NO**	The failed Backup is not resumed.
*YES	The failed Backup is resumed.

JOBLBL: Specifies the MMS/*tms* Job Label to use for the ***AUTO** Backup.

***DFT**	The default Job Label is used.
Job-label	Enter a valid Job Label.

STRDATE: Specifies the date to start the restricted state backup. If a date is entered, it must be in the job date format.

***CURRENT**	The current date is used.
Start-date	Enter a valid date.

STRTIME: Specifies the time to start the restricted state backup.

***CURRENT**	The current time is used.
Start-time	Enter a valid time.

STRSBS: Specifies if all subsystems are started after the restricted state backup completes.

***YES**	The subsystems are started.
*NO	The subsystems are not restarted.

SIGNOFF: Specifies if the console or workstation is signed off after the restricted state backup completes.

***NO**	The console or workstation is not signed off.
*YES	The console or workstation is signed off.

Examples

```
RUNBKUP BACKUP(WEEKLY) SEQNBR(*ALL) ENDSBS(*ALL) STRDATE(*CURRENT) +
    STRTIME(*CURRENT) STRSBS(*YES) SIGNOFF(*YES)
```

This executes a restricted state Backup named WEEKLY. All Backup Lists are processed. The backup starts as soon as the command is entered. After the backup completes, the subsystems are restarted and the console is signed off.

```
RUNBKUP BACKUP(DAILY) SEQNBR(0005 *ONLY) ENDSBS(QINTER)
```

This executes a Backup named DAILY. Only Backup List entry number 5 is processed. Subsystem QINTER is ended before the backup starts and restarted after the backup completes.

```
RUNBKUP BACKUP(DAILY) SEQNBR(0010 0015)
```

This executes a Backup named DAILY. Only Backup Lists 10 through 15 are processed.

```
RUNBKUP BACKUP(DAILY) SEQNBR(*AUTO) DRVRSC(*AVAIL *AVAIL) +
    ENDSBS(QINTER QBATCH) JOBLBL(DAILY)
```

This executes the *AUTO backup. All SAVLIB and SAVCHGOBJ commands use Media Definitions to save the objects/libraries. The number of available tape devices is determined by OS/400. Subsystems QINTER and QBATCH are ended before the backup starts and the backup uses the MMS/*tms* Job Label DAILY. When the backup completes, subsystems QINTER and QBATCH are restarted.

RUNRCY - Run Recovery

```
---------------------------------------------------------------------------------------------------
                    Run Recovery (RUNRCY)                    Environment: B/I

Recovery . . . . . . . . . . . . . . . . . .  _________       Name, *AUTO
Recovery sequence:
  Beginning. . . . . . . . . . . . . . . . .  *ALL          1-9999, *ALL
  Ending. . . . . . . . . . . . . . . . . . .  _____        1-9999, *ONLY, *END
Restore changes last . . . . . . . . . . . .  *YES         *YES, *NO
Run option . . . . . . . . . . . . . . . . .  *REPORT      *REPORT, *RECOVER

---------------------------------------------------------------------------------------------------
```

Purpose

The Run Recovery (RUNRCY) command initiates a Recovery. Options provide the ability to either print the recovery reports or start the restore process.

Parameters

RCY: Specifies the name of the Recovery to process.

*AUTO	The Intelligent Recovery is selected.
Backup-name	Enter a valid Recovery name.

SEQNBR: Specifies the beginning and ending recovery sequence to process.

Beginning:

***ALL**	All recovery sequences are processed.
Beginning-sequence	Enter the beginning sequence.

Ending:

*END	The last Recovery List entry is the ending sequence.
*ONLY	The beginning sequence is the only sequence processed.
Ending-sequence	Enter the ending sequence.

RSTCHG: Specifies whether to restore a library and its changes before processing the next library or whether to restore all libraries first and then apply any changes.

***YES**	Library changes are restored after all libraries on the Recovery List have been restored.
*NO	Library changes are restored after each library is restored.

OPTION: Specifies whether to print the recovery reports for the specified Recovery or whether to start the actual restore process.

***REPORT**	The recovery reports are printed.
*RECOVER	The restore is processed.

Examples

```
RUNRCY RCY(DAILY) SEQNBR(*ALL)
```

This processes a Recovery named NONSYS. All Recovery Lists are processed.

WRKBKUP - Work with Backup

Work with Backup (WRKBKUP)		Environment: B/I
Backup	*ALL	Name, *ALL
Output	*	*, *PRINT

Purpose

The Work with Backup (WRKBKUP) command lists all or specific Backups.

Parameters

BACKUP: Specifies the Backup to show.

***ALL** All Backups are shown.

Backup-name Enter a valid Backup name.

OUTPUT: Specifies whether the output from the command is displayed at the requesting work-station or printed with the job's spooled output.

***** The output is displayed (if requested by an interactive job) or printed with the job's spooled output (if requested by a batch job).

*PRINT The output is printed with the job's spooled output.

Examples

```
WRKBKUP BACKUP(DAILY)
```

This displays a Backup named DAILY.

WRKBKUPDFN - Work with Backup Definition

```
----------------------------------------------------------------------------------------------------
                    Work with Backup Definition (WRKBKUPDFN)          Environment: B/I

Backup definition . . . . . .  *ALL                    Name, *DFT, *ALL
Output . . . . . . . . . . . . . . . .  *              *, *PRINT
----------------------------------------------------------------------------------------------------
```

Purpose

The Work with Backup Definition (WRKBKUPDFN) command lists all or specific Backup Definitions.

Parameters

BKUPDFN: Specifies the Backup Definition to show.

***ALL**	All Backup Definitions are shown.
*DFT	The default Backup Definition is shown.
Backup-name	Enter a valid Backup Definition name.

OUTPUT: Specifies whether the output from the command is displayed at the requesting workstation or printed with the job's spooled output.

*****	The output is displayed (if requested by an interactive job) or printed with the job's spooled output (if requested by a batch job).
*PRINT	The output is printed with the job's spooled output.

Examples

```
WRKBKUPDFN BKUPDFN(DAILY)
```

This displays a Backup Definition named DAILY.

WRKBKUPL - Work with Backup List

```
                        Work with Backup List (WRKBKUPL)        Environment: B/I

Backup . . . . . .  _________          Name, *AUTO
List . . . . . .    *ALL               *ALL, *ASP, *ASPDLO, *CFG…
Output . . . . . .  *                  *, *PRINT
```

Purpose

The Work with Backup List (WRKBKUPL) command lists the libraries on a specific Backup List.

Parameters

BACKUP: Specifies the name of the Backup that is associated with this Backup List.

*AUTO	The Intelligent Backup is selected.
Backup-name	Enter a valid Backup name.

LIST: Specifies the Backup List type. This specifies the type of objects on the list.

*ASP	Auxiliary storage pool (**ASP**) list.
*ASPDLO	Document library object auxiliary storage pool (**ASP**) list.
*CFG	Configuration list.
*CLT	Client list.
*EJECT	Tape eject.
*EXIT	Command exit.
*FLR	Document library object list.
*LIB	Library list.
*LND	Domino server list.
*LNK	Integrated file system object list.
*OBJ	Object list.
*OUTQ	Output queue list.
*SAVF	Save file list.
*SEC	Security information.
*SPL	Spooled file list.
*SYS	All Licensed Internal Code; the QSYS library; security and configuration objects.

OUTPUT: Specifies whether the output from the command is displayed at the requesting workstation or printed with the job's spooled output.

*****	The output is displayed (if requested by an interactive job) or printed with the job's spooled output (if requested by a batch job).
*PRINT	The output is printed with the job's spooled output.

Examples

```
WRKBKUPL BACKUP(DAILY) LIST(*ALLUSR)
```

This displays the *ALLUSR Backup List from a Backup named DAILY.

WRKBKUPLE - Work with Backup List Entry

```
---------------------------------------------------------------------------------------------------
                    Work with Backup List Entry (WRKBKUPLE)        Environment: B/I

Backup . . . . . . . . . . . . . . . . . . . .  _________        Name, *AUTO
Sequence number . . . . . . . . . . . . . . .  ____             1-9999
Output . . . . . . . . . . . . . . . . . . . .  *                *, *PRINT

---------------------------------------------------------------------------------------------------
```

Purpose

The Work with Backup List Entry (WRKBKUPLE) command lists the Backup entries for a specific Backup/Backup List.

Parameters

BACKUP: Specifies the name of the Backup that is associated with this Backup List entry.

*AUTO	The Intelligent Backup is selected.
Backup-name	Enter a valid Backup name.

SEQNBR: Specifies the sequence number of the Backup List entry to show.

Sequence-number	Enter a number from 1-9999.

OUTPUT: Specifies whether the output from the command is displayed at the requesting workstation or printed with the job's spooled output.

*	The output is displayed (if requested by an interactive job) or printed with the job's spooled output (if requested by a batch job).
*PRINT	The output is printed with the job's spooled output.

Examples

```
WRKBKUPLE BACKUP(DAILY) SEQNBR(5)
```

This displays the Backup List entries for sequence number 5 for a Backup named DAILY.

WRKBKUPSTS – Work with Backup Status

```
---------------------------------------------------------------------------------------------------
                        WRKBKUPSTS (WRKBKUPSTS)          Environment: B/I

Wait . . . . . . . . . . . . . . . . . . . . . . 5                  1-99

---------------------------------------------------------------------------------------------------
```

Purpose

The Work with Backup Status (WRKBKUPSTS) command displays all MMS/*bms* Backups currently active or in error status. This panel refreshes itself automatically based on the wait time entered on the command. **Note**: MMS or MMS/*tms* must be installed to use this a feature.

Parameters

WAIT: Specifies the number of seconds to wait before refreshing the panel.

5	The panel is refreshed every 5 seconds.
Wait-time	Enter a value from 1 to 99.

Examples

```
WRKBKUPSTS WAIT(5)
```

This displays all Backups currently active or in error status. The panel will automatically refresh itself every 5 seconds.

WRKOMITL – Work with Omit List

```
----------------------------------------------------------------------------------------------------
                    Work with Omit List  (WRKOMITL)          Environment: B/I

Omit list name . . . . . . . . . . . . . . .   *ALL          Name, *ALL
List . . . . . . . . . . . . . . . . . . . . .   *ALL        *ALL, *CLT, *FLR, *LIB…
Output . . . . . . . . . . . . . . . . . . .   *            *, *PRINT

----------------------------------------------------------------------------------------------------
```

Purpose

The Work with Omit List (WRKOMITL) command lists one or more Omit Lists. Omit Lists define the name and type of object to omit from a Backup.

Parameters

OMITL: Specifies the name of the Omit List to display.

***ALL**	All Omit Lists.
List-name	Enter an Omit List name.

LIST: Specifies the type of Omit List to display.

***ALL**	All Omit List types.
*CLT	Client data Omit List.
*FLR	Document library object Omit List.
*LIB	Library Omit List.
*LND	Domino server Omit List.
*LNK	Integrated file system Omit List.
*OBJ	Object Omit List.
*OUTQ	Output queue Omit List. MMS/spl must be installed to use this value.
*SYS	System data Omit List.

OUTPUT: Specifies whether the output from the command is displayed at the requesting workstation or printed with the job's spooled output.

*****	The output is displayed (if requested by an interactive job) or printed with the job's spooled output (if requested by a batch job).
*PRINT	The output is printed with the job's spooled output.

Examples

```
WRKOMITL OMITL(DAILY)
```

This displays an Omit List named DAILY.

WRKOMITLE - Work with Omit List Entry

```
---------------------------------------------------------------------------------------------------
                    Work with Omit List Entry (WRKOMITLE)        Environment: B/I

Omit list name . . . . . . . . . . . . . . . _________          Name
List . . . . . . . . . . . . . . . . . . . . . _____          *CLT, *FLR, *LIB, *LND…
Output . . . . . . . . . . . . . . . . . . . . *              *, *PRINT

---------------------------------------------------------------------------------------------------
```

Purpose

The Work with Omit List Entry (WRKOMITLE) command lists the items omitted from a specific Backup/Backup List.

Parameters

OMITL: Specifies the name of the Omit List to display.

List-name	Enter an Omit List name.

LIST: Specifies the type of Omit List to display.

*CLT	Client data Omit List.
*FLR	Document library object Omit List.
*LIB	Library Omit List.
*LND	Domino server Omit List.
*LNK	Integrated file system Omit List.
*OBJ	Object Omit List.
*OUTQ	Output queue Omit List. MMS/spl must be installed to use this value.
*SYS	System data Omit List.

OUTPUT: Specifies whether the output from the command is displayed at the requesting workstation or printed with the job's spooled output.

*****	The output is displayed (if requested by an interactive job) or printed with the job's spooled output (if requested by a batch job).
*PRINT	The output is printed with the job's spooled output.

Examples

```
WRKOMITLE BACKUP(DAILY) LIST(*LIB)
```

This displays the Omit List entries for sequence number 5 for a Backup named DAILY.

WRKRCY - Work with Recovery

```
------------------------------------------------------------------------------------------------------------
                          Work with Recovery (WRKRCY)        Environment: B/I

Recovery . . . . . . . . . . . . . . . . . .     *ALL                      Name, *ALL
Output . . . . . . . . . . . . .  . . . . . . . .     *                         *, *PRINT
------------------------------------------------------------------------------------------------------------
```

Purpose

The Work with Recovery (WRKRCY) command lists all or specific recoveries.

Parameters

RCY: Specifies the Recovery to show.

***ALL**	All Recovery are selected.
Recovery-name	Enter a valid Recovery name.

OUTPUT: Specifies whether the output from the command is displayed at the requesting work-station or printed with the job's spooled output.

*****	The output is displayed (if requested by an interactive job) or printed with the job's spooled output (if requested by a batch job).
*PRINT	The output is printed with the job's spooled output.

Examples

```
WRKRCY RCY(DAILY)
```

This displays a Recovery named DAILY.

WRKRCYDFN - Work with Recovery Definition

```
----------------------------------------------------------------------------------------------------
                    Work with Recovery Definition (WRKRCYDFN)        Environment: B/I

Recovery definition . . . . . .  *ALL                    Name, *ALL, *DFT
Output . . . . . . . . . . . . .  *                      *, *PRINT
----------------------------------------------------------------------------------------------------
```

Purpose

The Work with Recovery Definition (WRKRCYDFN) command lists all or specific Recovery Definitions.

Parameters

RCYDFN: Specifies the Recovery Definition to show.

***ALL**	All Recovery Definitions are selected.
*DFT	The default Recovery Definition is selected.
Recovery-name	Enter a valid Recovery Definition name.

OUTPUT: Specifies whether the output from the command is displayed at the requesting workstation or printed with the job's spooled output.

*****	The output is displayed (if requested by an interactive job) or printed with the job's spooled output (if requested by a batch job).
*PRINT	The output is printed with the job's spooled output.

Examples

```
WRKRCYDFN RCYDFN(DAILY)
```

This displays a Recovery Definition named DAILY.

WRKRCYL - Work with Recovery List

```
-------------------------------------------------------------------------------------------------
                    Work with Recovery List (WRKRCYL)          Environment: B/I

Recovery . . . . . . . . . . . . . . . . . .  _________          Name, *AUTO
Output . . . . . . . . . . . . . . . . . . .  *                  *, *PRINT
-------------------------------------------------------------------------------------------------
```

Purpose

The Work with Recovery List (WRKRCYL) command lists the libraries on a specific Recovery List.

Parameters

RCY: Specifies the name of the Recovery that is associated with this Recovery List.

*AUTO	The Intelligent Recovery is selected.
Recovery-name	Enter a valid Backup name.

OUTPUT: Specifies whether the output from the command is displayed at the requesting workstation or printed with the job's spooled output.

*****	The output is displayed (if requested by an interactive job) or printed with the job's spooled output (if requested by a batch job).
*PRINT	The output is printed with the job's spooled output.

Examples

```
WRKRCYL RCY(DAILY)
```

This displays the Recovery List from a Recovery named DAILY.

WRKRCYLE - Work with Recovery List Entry

```
---------------------------------------------------------------------------------------------------
                    Work with Recovery List Entry (WRKRCYLE)        Environment: B/I

Backup . . . . . . . . . . . . . . . . . . . .  _________        Name, *AUTO
Sequence number . . . . . . . . . . . . . . .  ____             1-9999
Output . . . . . . . . . . . . . . . . . . . .  *                *, *PRINT

---------------------------------------------------------------------------------------------------
```

Purpose

The Work with Recovery List Entry (WRKRCYLE) command lists the entries for a specific Recovery / Recovery List.

Parameters

RCY: Specifies the name of the Recovery that is associated with this Recovery List entry.

*AUTO	The Intelligent Recovery is selected.
Recovery-name	Enter a valid Recovery name.

SEQNBR: Specifies the sequence number of the Recovery List entry to show.

Sequence-number	Enter a number from 1-9999.

OUTPUT: Specifies whether the output from the command is displayed at the requesting workstation or printed with the job's spooled output.

*	The output is displayed (if requested by an interactive job) or printed with the job's spooled output (if requested by a batch job).
*PRINT	The output is printed with the job's spooled output.

Examples

```
WRKRCYLE RCY(DAILY) SEQNBR(5)
```

This displays the Recovery List entries for sequence number 5 for a Recovery named DAILY.

WRKPRTF - Work with Printer File

```
---------------------------------------------------------------------------------------------------
                    Work with Printer File (WRKPRTF)          Environment: I

Product . . . . . .  ____                                     *BMS, *CMS, *TMS
---------------------------------------------------------------------------------------------------
```

Purpose

The Work with Printer Files (WRKPRTF) command lists all printer files associated with a specific LXI product.

Parameters

PRODUCT: Specifies the LXI product printer files to show.

*AMS	MMS Archive Management.
*BMS	MMS Backup and Recovery.
*CMS	MMS Vault Management.
*SPL	MMS Spooled File Management.
*TMS	MMS Tape Management.

Examples

If the volumes being moved reside in a tape library, they are not ejected.
This will take you to the Work with Printer Files display. The list contains an entry for all printer files in the MMS/*bms* product.

Chapter 12

Install/Uninstall Instructions

This chapter describes the install/uninstall processes. The installation of MMS Backup and Recovery is easy and only takes a few minutes to load and setup. Please read and follow these instructions carefully to avoid problems and assure trouble free product performance.

Install Process

The installation process loads/updates the product from CD to disk. To install, follow the instructions in the ***Readme*** text file provided on the CD. These instructions guide you through the installation process. If this is a first time install, the installation process creates the following libraries on the system.

- LXI Base and Support Programs
- LXIBMS Backup and Recovery Programs
- LXIBMS400 Backup and Recovery Files

Changing the iSeries

Moving this product from one iSeries to another or upgrading to a different iSeries model requires a new license key. Once the product moves to the new iSeries or the iSeries is upgraded to a different model, call LXI Corp. for a new license key. No install is required.

Uninstall Process

To remove MMS/*bms* from the system, perform the following:

Remove MMS/*bms* from the system library list.

```
DLTLICPGM LICPGM(0LX0000) OPTION(92)
```

Entering the License Key

MMS Backup and Recovery (MMS/*bms*) requires a valid license key in order to function. The license key is based on the serial number and model of the iSeries. To enter a license key, perform the following:

___ Step 1. **GO LXI/LXI**

___ Step 2. Tab to the **SETUP** option on the menu bar.

___ Step 3. Press **Enter** to view the options available.

___ Step 4. Select "**Work with License Info.**".

___ Step 5. Select **Option 1** for feature **9120**.

___ Step 6. Enter the supplied license key.

Trial Period

The trial period is valid for a period of 30 days from the time the product is *first* used. In order for MMS/*bms* to continue performing backups after the 30-day trial period, a license key must be entered.

Permanent License Key

Once the software has been purchased and payment received by LXI Corp., a ***permanent license key*** will be issued. This permanent license key must be entered into the software to ensure that the product continues without interruption.

The license key remains valid unless the iSeries serial or model number changes. When a change occurs, you should notify LXI Corp. to get another license key.

Chapter 13

Concurrent Saves

Performing concurrent saves can maximize backup windows. MMS/*bms* provides the following methods of accomplishing this.

- If you use stand-alone tape devices without MMS Tape Management (MMS/*tms*), you can create Backup Definitions for each device and use one Backup Definition per Backup.

- If you use stand-alone tape devices with MMS/*tms*, you can share devices that write the same density. This allows you to create one Backup Definition specifying the same device. When the backup(s) execute, MMS/*tms* provides the device management. Refer to the MMS Tape Management documentation for details on device sharing.

- If you are using an IBM automated tape library, you can specify the tape library name in the Backup Definition. When the backup(s) execute, the tape library provides the device management. If you are using MMS/*tms*, it manages which tapes to mount and use. Refer to the MMS Tape Management documentation for details on using automated tape libraries. If you do not have MMS/*tms*, refer to the appropriate IBM manual on using the automated tape library without a tape management system.

- If you are using non-IBM automated tape libraries with MMS/*tms*, you can share devices that write the same density. This allows you to create one Backup Definition specifying the same device. When the backup(s) execute, MMS/*tms* provides the device management. Refer to the MMS Tape Management documentation for details on device sharing.

Chapter 14

Exit Program Example

The following example is a shell for an exit program. This program is divided into two parts: the first performs pre-backup tasks and the second performs post-backup tasks. Pre-backup tasks may include ending subsystems, holding job queues and checking object locks while post-backup tasks may include starting subsystems, releasing job queues and submitting jobs.

Since backups run independent of each other, each backup can use the same or have a different exit program.

```
PGMPARM(&PARM1 &PARM2 &PARM3 &PARM4 &PARM5)
DCLVAR(&PARM1) TYPE(*CHAR) LEN(1)              /* Before/After Flag    */
DCLVAR(&PARM2) TYPE(*CHAR) LEN(3)              /* Not Used              */
DCLVAR(&PARM3) TYPE(*CHAR) LEN(20)             /* MMS/tms Job Label    */
DCLVAR(&PARM4) TYPE(*CHAR) LEN(1)              /* Backup Status         */
DCLVAR(&PARM5) TYPE(*CHAR) LEN(10)             /* Last Used Device      */

IF COND(&PARM1 = '0') THEN(DO)                 /* Before Backup         */
  .                                            /* Pre-Backup Tasks     */
  .
ENDDO

IF COND(&PARM1 = '1') THEN(DO)                 /* After Backup          */
  .                                            /* Post-Backup Tasks    */
  .
ENDDO
ENDPGM
```

Chapter 15

Troubleshooting Guide

The purpose of this guide is to list commonly asked questions regarding the use of MMS Backup and Recovery. Each question and answer is designed to resolve specific situations as quickly as possible. This guide should be the first place a user looks when encountering any type of function that does not appear to operate as expected. If, after reviewing this guide, a question still exists, contact LXI Product Support for assistance.

1. **I entered the license key and MMS/*bms* says it is invalid.**

   The license key is date sensitive. If the evaluation time has elapsed, another license key will be required. If you are entering a permanent license key, make sure that you entered the license key correctly. Additional information can be found in Chapter 12, *Install/Uninstall Instructions*. If the license key still does not work, call LXI Product Support.

2. **I am trying to use *OUTQ and *SPL to save spooled files and output queues but nothing is being saved.**

   Ensure that MMS Spooled File Management (MMS/spl) is installed and has a valid license key. Ensure that the selection criteria for selecting spooled files is correct. Ensure that valid spooled files exist.

3. **I ran a Backup interactively and when I tried to run it again, either interactively or in batch, an error message stated that the Backup was in "RUN" status.**

   When a Backup runs interactively, it is associated with the job, user and number of the job performing the backup. This job remains active or in "RUN" or "DSPW" status until the job ends, which for an interactive job, means signing off.

4. **I used Option 6 from the Work with Backup menu to run a backup and nothing happened.**

   Using Option 6 from the Work with Backup menu submits the backup to the QLXI subsystem. Ensure that this subsystem is active. To start the subsystem, enter: STRSBS LXI/QLXI

5. **Can I tell what objects need to be backed up without using the *AUTO Backup?**

   Yes, use the Check Save Status (CHKSAVSTS) command to print a report of all objects that need to be saved. Submit this command to batch as it could take some time to process.

6. **When are Backup Lists and Backup List Entries automatically renumbered?**

   They are renumbered after an entry as added or removed. All renumbering is done in increments of five (5).

7. **Can Backup Lists and Backup List Entries be changed or deleted while a Backup is running?**

   No.

8. **Can I do a restricted state save by submitting the RUNBKUP command to QBATCH.**

   Yes, but you must be on OS/400 release V5R3 or greater.

Chapter 16

Electronic Software Support

Electronic Software Support (**ESS**) is a module within all LXI Corp. products that provides LXI Technical Support staff access to your system, upon your approval, to help isolate and resolve issues. This process helps ensure that your product is working correctly and performing to design standards.

Online support allows an LXI product technician to sign on to your system for diagnostic purposes. This method is beneficial when issues cannot be resolved easily. Online support requires that you provide LXI with a user profile, password and virtual device for system access.

Setting up ESS

Before using **ESS**, some initial setup must be performed. This setup includes defining any special characters or numbers that must be dialed prior to dialing the LXI Corp. Product Support number, determining the modem type and optionally defining the resource name and line speed. This setup only needs to be performed once or if the information changes.

To access the ESS main menu, type **GO LXI/ESS** on an OS/400 command line and press **Enter**.

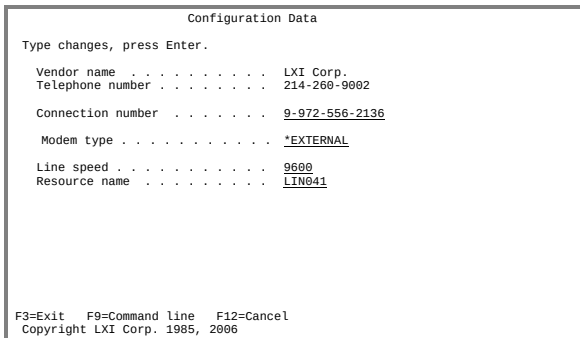
Updating the Configuration Data

Select **Option 1** from the Setup pull-down menu. This displays the Configuration Data panel. This panel specifies the LXI Product Support phone number and the modem type parameter.

Do not alter the LXI Corp. phone number unless you need to add special characters in front of it for time delay or outside line purposes.

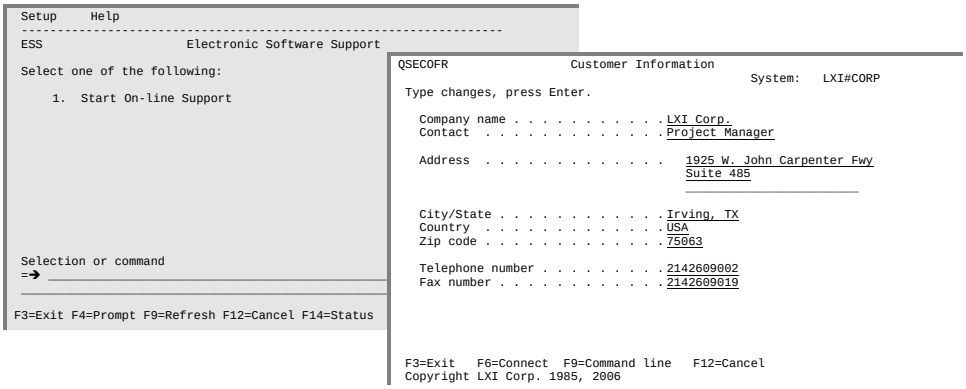
If an **external** modem is specified in the modem type parameter, press **Enter**. This displays two additional parameters which are the resource number of the line being used and the speed of the modem. Review and optionally change the information and press **Enter**.

If an **internal** modem is specified, press **Enter**.



Requesting Online Support

To start online support, enter **Option 1** from the Electronic Software Support menu. This displays the Customer Information panel. Enter the required information and press **Enter**. To start the **ESS** online support process, press **F6**.



ESS Considerations

When using an external modem, **ESS** leaves the line varied on. Due to processing restrictions, **ESS** cannot vary the line off.

Index

A

B

C

D

E

H

I

L

M

P

R