HP OpenView Storage Data Protector Command Line Interface Reference

Manufacturing Part Number: B6960-90030

Manual Edition: December 2005

Introduction to HP OpenView Storage Data Protector Command Line Interface Reference

Section 9: Introduction

Section 1: User Commands

Section 1M: Administrative Commands

Section 5: Miscellaneous

Introduction to HP OpenView Storage Data Protector Command Line Interface Reference

The *HP OpenView Storage Data Protector Command Line Interface Reference* contains the reference pages for HP OpenView Storage Data Protector A.06.00 commands.

Reference pages are available on UNIX systems as man pages. For more information about man pages, refer to the man page for `man` using the command `man man`.

The command synopsis for every command is also available using the `-help` option.

For an introduction to Data Protector A.06.00 commands, refer to the *omniintro(9)* reference page.

Reference Page Organization

The reference pages are divided in specialized sections (volumes), based on the UNIX man page organization. Each reference page belongs to a volume:

Section 1: User Commands Commands, used by the user.

Section 1M: Administrative Commands Commands, used by the administrator.

Section 5: Miscellaneous A variety of information, such as information about GUI components, and more.

Section 9: Introduction Introduction to HP OpenView Storage Data Protector.

All commands in a section are sorted by alphabetical order.

Reference pages are often referred by name and section number in the form *pagename*(*section*).

Conventions

All reference pages follow established section formats, but not all sections are present in each reference (man) page.

NAME Gives the name of the command and a brief description of the commands purpose.

SYNOPSIS Describes the syntax of the command.

The command line synopsis is formatted in the following way:

```
command -option replaceable [-option2 replaceable] ...
     {-option3 | -option4}
```

Where:

- *Italic* strings represent variables that should be replaced by the user with the appropriate value.
- Square brackets ([]) indicate that the argument is optional.
- An ellipsis (. . .) indicates that the previous argument can be repeated.
- Vertical bars (|) between several arguments indicate that only one argument from the group can be specified at once.

  Groups can be optional (inside square brackets) or required (inside curly brackets, { }).

DESCRIPTION A more detailed description of the command.

OPTIONS Detailed descriptions for all options.

NOTES Contains important notes.

EXAMPLES Provides examples on command usage.

SEE ALSO Lists man pages, containing related information.

Section 9: Introduction

omniintro (9)

NAME

omniintro – introduction to HP OpenView Storage Data Protector command utilities

DESCRIPTION

HP OpenView Storage Data Protector is an enterprise backup solution that provides reliable data protection and high accessibility for business data. Data Protector provides extensive media management, unattended backups, post-backup data management, integrations with various databases and supports various backup and other backup-dedicated devices. For information on Data Protector functionality and concepts refer to the online documentation.

COMMANDS

User Command-line Commands (1):

`omniabort` Aborts an active session.

`omniamo` Starts an automated media operation session.

`omnib` Backs up files, filesystems, very big file systems, disk images and databases. This is the only command supported with the MPE integration.

`omnicc` Handles the Data Protector licensing, reports the number of configured and available Data Protector licenses, imports and exports Data Protector clients, and manages secured clients.

`omnicellinfo` Displays configuration information of a cell.

`omniclus` Manages load balancing in a cluster environment in case of an application failover.

`omnicreatedl` Creates a Data Protector backup specification.

`omnidb` Queries the Data Protector internal database (IDB).

`omnidbeva` Sets, updates, deletes, and lists the login information for the HP StorageWorks Enterprise Virtual Array (EVA) Command View EVA (CV EVA) v3.1 or lower.

`omnidbsmis` Sets, updates, deletes, and lists the login information for the Command View EVA (CV EVA) v3.2.

`omnidbva` Queries the ZDB database (VADB) and manipulates the VA LUN Exclude File.

`omnidbxp` Queries the ZDB database (XPDB) and manipulates the XP LDEV Exclude File.

`omnidlc` Gathers or deletes Data Protector debug, log and getinfo files from the Data Protector cell or from a MoM environment.

`omnidownload` Downloads information about a backup device and a library from the Data Protector internal database (IDB).

`omniiso` Primarily serves as pre-exec script to prepare the ISO image file for One Button Disaster Recovery (OBDR), but can also be used as a standalone command to automate your backup and disaster recovery process.

`omnimcopy` Makes a copy of a Data Protector medium using Data Protector backup devices as a source and destination.

`omniminit` Initializes a Data Protector medium.

`omnimlist` Lists the contents of a Data Protector medium.

`omnimm` Provides media management for Data Protector.

`omnimnt` Responds to a Data Protector mount request for a medium.

`omnimver` Verifies data on a medium.

`omniobjcopy` Creates additional copies of objects backed up with Data Protector on a different media set.

`omniobjconsolidate` Consolidates Data Protector backup objects into synthetic full backups.

`omnir` Restores files, disk images and databases backed up using Data Protector.

`omnirpt` Generates various reports about the Data Protector environment, such as reports about backup sessions in a specific time frame, backup specifications, media, Data Protector configuration and about a single session.

`omnistat` Displays the status of active backup and restore sessions.

`omniupload` Uploads information about a backup device from an ASCII file to the Data Protector internal database (IDB).

`omniusers` Adds/removes Data Protector users to/from an existing user group.

Administrative Command-line Commands (1M):

`ob2install` Runs remote installation, uninstallation, upgrade or check of a client from the selected Installation Server. This command is supported on UNIX systems only. It is not supported on Windows systems.

`omnicheck` Performs DNS connections check within a Data Protector cell and lists Data Protector patches installed on Data Protector clients.

`omnidbcheck` Checks the consistency of the Data Protector internal database (IDB). It can be run only on the Cell Manager.

`omnidbinit` Initializes the Data Protector internal database (IDB). It can be run only on the Cell Manager.

`omnidbrestore` Restores the Data Protector internal database (IDB). It can be run only on the Cell Manager.

`omnidbupgrade` Converts filenames in the IDB to the new internal character encoding used in Data Protector A.06.00 and thus enables the correct handling of non-ASCII characters in filenames in the Data Protector GUI.

`omnidbutil` Handles various Data Protector internal database (IDB) maintenance tasks. It can be run only on the Cell Manager.

`omnidlc` Gathers or deletes Data Protector debug, log and sysinfo files from the Data Protector cell.

`omnidr` A Data Protector disaster recovery command. Based on its input, omnidr decides what type of restore is going to be performed: online restore using omnir of offline restore using omniofflr as well as how the restore is going to be performed (using or avoiding live OS features). The command can only be used on Windows systems.

`omnihealthcheck` Checks the status of Data Protector services, the consistency of the Data Protector internal database (IDB) and if at least one backup of the IDB exists. It can be run only on the Cell Manager.

`omniinstlic` Starts the HP OpenView AutoPass utility or synchronizes the Data Protector licenses between Data Protector and HP OpenView AutoPass.

`omnimigrate` Helps you migrate your existing Cell Manager from a PA-RISC architecture based HP-UX 11.x system to an HP-UX 11.23 system for the Intel Itanium 2 (IA-64) architecture.

`omniofflr` Enables restore of any type of Data Protector backup object in the absence of a working Data Protector internal database (IDB). The command can only be used on Windows systems.

`omnirsh` Returns the hostnames of the physical and virtual nodes for the specified cluster hostname, or returns the cell information, stored in the cell_info file on the specified cluster.

`omniresolve` Takes a list of files or a datafile containing a list of files, resolves them and prints the results to standard output.

`omnisetup.sh` Runs a local installation or upgrade of a Data Protector UNIX client.

`omnisrdupdate` A Data Protector disaster recovery command. Updates system recovery data (SRD). The command can only be used on Windows systems.

`omnisv` Starts, stops or displays the status of Data Protector daemons (HP-UX or Solaris Cell Manager) or services (Windows Cell Manager). It can be run only on the Cell Manager.

`omnitrig` Triggers Data Protector scheduled backups.

`sanconf` Auto-configures a library, modifies an existing library or drive configuration, or removes drives from a library configuration, within a SAN environment.

`upgrade_cfg_from_evaa` Runs the cluster nodes upgrade when upgrading from the HP StorageWorks EVA Agent (legacy) to the HP StorageWorks EVA SMI-S Agent.

`util_cmd` Sets, retrieves or lists the parameters stored in the Data Protector integrations configuration files.

Command-line Utilities (1M):

`NNMpost.ovpl` A script with no arguments that resumes the eight processes paused by NNMpre.ovpl.

`NNMpre.ovpl` Starts NNM embedded database backup.

`NNMscript.exe` Finds the location of the NNM Perl compiler and the NNMpre.ovpl and NNMpost.ovpl scripts and starts the two scripts

`uma` Controls the robotics of SCSI compliant Tape Library Units

`syb_tool` A utility used to get ISQL command needed to restore a Sybase database that was backed up by Data Protector

Return Values

Possible return values for CLI commands are:

`1`	Program failed, command syntax error.
`2`	Program failed, invalid argument.
`3`	Program failed, internal error.
`4`	Program failed, reason unknown.

GRAPHICAL USER INTERFACE APPLICATIONS ON WINDOWS

`Data Protector Manager`	GUI command panel
`Manager of Managers`	GUI used to manage Data Protector multi cell environments

GRAPHICAL USER INTERFACE COMMANDS ON UNIX

`xomni`	GUI command panel
`xomnimom`	GUI used to manage Data Protector multi cell environments

DIRECTORY STRUCTURE FOR UNIX AND LINUX CELL MANAGER

`/opt/omni` Data Protector home directory

`/etc/opt/omni/server`

Directory containing the following configuration directories:

`datalists`	backup specifications
`barlists`	database backup specifications
`devices`	templates for devices
`users`	user configuration
`cell`	cell configuration
`schedules`	backup schedules
`barschedules`	database backup specification schedules

`sessions`	data about sessions
`options`	default options
`sg`	scripts for Service Guard support
`snmp`	OpenView/SNMP trap sending configuration

`/etc/opt/omni/client`

Directory containing the client configuration directories and files

`/etc/opt/omni/server/dr`

Directory containing the following disaster recovery directories:

`p1s`	P1S files for Enhanced Automated Disaster Recovery
`srd`	SRD files
`asr`	ASR archive file

`/opt/omni`

Directory containing the following executables directories:

`bin`	user Data Protector commands
`lbin`	Disk Agent and Media Agent files and some administrative commands
`sbin`	Cell Manager and Data Protector internal database (IDB) administrative Data Protector commands

`/var/opt/omni/`

Directory containing the following directories:

`/var/opt/omni/log` and `/var/opt/omni/server/log`	log files
`/var/opt/omni/tmp`	temporary files
`/var/opt/omni/server/sessions`	data about sessions

`/var/opt/omni/server/db40`

Directory containing the following Data Protector internal database (IDB) directories:

`/var/opt/omni/server/db40/datafiles`	the IDB tablespaces
`/var/opt/omni/server/db40/dcbf`	the IDB Detail Catalog binary files (DCBF)

`/var/opt/omni/server/db40/smisdb` the ZDB database (SMISDB) when using the HP StorageWorks EVA SMI-S Agent

`/var/opt/omni/server/db40/logfiles` the IDB transaction logs and the obdrindex.dat file

`/var/opt/omni/server/db40/meta` the Serverless Integrations Binary Files (SIBF) part of the IDB

`/var/opt/omni/server/db40/msg` the Data Protector session messages

`/var/opt/omni/server/db40/vadb` the ZDB database (VADB)

`/var/opt/omni/server/db40/xpdb` the ZDB database (XPDB)

`/opt/omni/lib`

Directory containing the following directories:

`/opt/omni/lib/nls` message catalogs

`/opt/omni/lib/man` Data Protector man pages

`/opt/omni/gui/help` the Data Protector help subsystem

DIRECTORY STRUCTURE FOR WINDOWS CELL MANAGER

<Data_Protector_home> Data Protector home directory

<Data_Protector_home>`\Config\server`

Directory containing the following configuration directories:

`datalists`	backup specifications
`barlists`	database backup specifications
`devices`	templates for devices
`users`	the user configuration
`cell`	the cell configuration
`schedules`	backup schedules
`barschedules`	database backup specification schedules
`options`	default options
`sessions`	data about sessions

snmp	OpenView/SNMP trap sending configuration

<Data_Protector_home>\Config\client

Directory containing the client configuration directories and files

<Data_Protector_home>\Config\server\dr

Directory containing the following disaster recovery directories:

p1s	P1S files for Enhanced Automated Disaster Recovery
srd	SRD files
asr	ASR archive files

<Data_Protector_home>\bin

Data Protector commands, Disk Agent, Media Agent files, message catalogs and commands for Cell Manager maintenance.

<Data_Protector_home>\log and *<Data_Protector_home>*\log\server

log files

<Data_Protector_home>\tmp

temporary files

<Data_Protector_home>\db40

the Data Protector internal database (IDB)

<Data_Protector_home>\db40\datafiles

the IDB tablespaces

<Data_Protector_home>\db40\dcbf

the IDB Detail Catalog binary files (DCBF)

<Data_Protector_home>\db40\logfiles

the IDB transaction logs and the obdrindex.dat file

<Data_Protector_home>\db40\meta

the Serverless Integrations Binary Files (SIBF) part of the IDB

<Data_Protector_home>\db40\msg

the Data Protector session messages

<Data_Protector_home>`\db40\evadb`

the ZDB database (EVADB) when using the HP StorageWorks EVA Agent (legacy)

<Data_Protector_home>`\db40\smisdb`

the ZDB database (SMISDB) when using the HP StorageWorks EVA SMI-S Agent

<Data_Protector_home>`\db40\vadb`

the ZDB database (VADB)

<Data_Protector_home>`\db40\xpdb`

the ZDB database (XPDB)

<Data_Protector_home>`\help`

the Data Protector help subsystem

<Data_Protector_home>`\docs\Man`

Data Protector man pages

SEE ALSO

ob2install(1M), omniabort(1), omniamo(1), omnib(1), omnicc(1), omnicellinfo(1), omnicheck(1M), omniclus(1), omnicreatedl(1), omnidb(1), omnidbcheck(1M), omnidbeva(1), omnidbinit(1M), omnidbsmis(1), omnidbva(1), omnidbxp(1), omnidlc(1M), omnidownload(1), omnidr(1M), omnigui(5), omnihealthcheck(1M), omniinstlic(1M), omniiso(1), omnimcopy(1), omniminit(1), omnimigrate.sh(1M), omnimlist(1), omnimm(1), omnimnt(1), omnimver(1), omniobjcopy(1), omniobjconsolidate(1), omniofflr(1M), omnidbupgrade(1M), omnidbrestore(1M), omnidbutil(1M), omnir(1), omniresolve(1M), omnirpt(1), omnirsh(1M), omnisetup.sh(1M), omnisrdupdate(1M), omnistat(1), omnisv(1M), omnitrig(1M), omniupload(1),omniusers(1), sanconf(1M), syb_tool(1), uma(1M), upgrade_cfg_from_evaa(1M), util_cmd(1M), NNMpre.ovpl(1M), NNMpost.ovpl(1M), NNMScript.exe(1M)

Section 1: User Commands

omniabort (1)

NAME

omniabort – aborts an active session

SYNOPSIS

```
omniabort -version | -help
omniabort -session SessionID
```

DESCRIPTION

This command aborts an active session, identifying it by the *SessionID*. A list of all active sessions and their session IDs is available using the `omnistat` command.

OPTIONS

`-version` Displays the version of the `omniabort` command.

`-help` Displays the usage synopsis for the `omniabort` command.

`-session` *SessionID* Specifies the *SessionID* of the session to be aborted. Use the `omnistat` command to get the *SessionID* of the session.

EXAMPLES

To abort a session with the SessionID "R-1995/04/13-12" use:

```
omniabort -session R-1995/04/13-12
```

```
omniabort -sess 12
```

SEE ALSO

omnistat(1)

omniamo (1)

NAME

omniamo – starts an automated media operation session

SYNOPSIS

```
omniamo -version | -help
omniamo -amc ConfigurationName {-post_backup | -scheduled}
```

DESCRIPTION

This command starts an automated media operation session for the specified post-backup or scheduled configuration. Before starting a post-backup operation, you must export the session ID of the backup session that used the media you want to copy.

On Windows: `set SESSIONID=`*`SessionID`*

On UNIX: `export SESSIONID=`*`SessionID`*

Use this command if you want to immediately start an automated media operation. Also, if an automated media operation has failed, you can use this command to start the operation again.

OPTIONS

`-version` Displays the version of the `omniamo` command.

`-help` Displays the usage synopsis for the `omniamo` command.

`-amc` *`ConfigurationName`* `{-post_backup | -scheduled}` Starts the post-backup or scheduled automated media copy operation with the specified name.

EXAMPLES

1. To start the scheduled automated media copy operation with the configuration name "MediaCopy1", use:

   ```
   omniamo -amc MediaCopy1 -scheduled
   ```

2. To start the post-backup automated media copy operation with the configuration name "MyFiles" and session ID 2002/02/13-0001 on Windows, use:

   ```
   set SESSIONID=2002/02/13-0001
   ```

   ```
   omniamo -amc MyFiles -post_backup
   ```

3. To start the post-backup automated media copy operation with the configuration name "MyDocs" and session ID 2002/02/13-0002 on UNIX, if you are using an sh-like shell, use:

   ```
   SESSIONID=2002/02/13-0002
   ```

   ```
   export SESSIONID
   ```

   ```
   omniamo -amc MyDocs -post_backup
   ```

4. To start the post-backup automated media copy operation with the configuration name "MyBackup" and session ID 2002/02/13-0003 on UNIX, if you are using a csh-like shell, use:

   ```
   export SESSIONID=2002/02/13-0003
   ```

   ```
   omniamo -amc MyBackup -post_backup
   ```

SEE ALSO

omnidownload(1), omnimcopy(1), omniminit(1), omnimlist(1), omnimm(1), omnimnt(1), omnimver(1), omniupload(1), sanconf(1M), uma(1M)

omnib (1)

NAME

omnib – Starts a backup of filesystems (Unix or Windows), Very Big File Systems (VBFS), disk images, Data Protector internal database (IDB), MS Exchange single mailboxes, MS Exchange, MS SQL, SAP R/3, SAP DB, Oracle, Informix Server, Sybase, Lotus, IBM DB2 UDB, NetWare objects or NDMP objects.

SYNOPSIS

```
omnib -version | -help

omnib -filesystem Client:MountPoint Label -device BackupDevice
    [MIRROR_OPTIONS [MIRROR_OPTIONS] ... ] [GENERAL_OPTIONS]
    [FILESYSTEM_OPTIONS] [-public]

omnib -filesystem Client:MountPoint Label -device BackupDevice
    -ndmp Type [NDMP_OPTIONS] [-public]

omnib -winfs Client:MountPoint Label -device BackupDevice
    [MIRROR_OPTIONS [MIRROR_OPTIONS] ... ] [GENERAL_OPTIONS]
    [FILESYSTEM_OPTIONS] [WINFS_OPTIONS] [-public]

omnib -NetWare Client:MountPoint Label -device BackupDevice
    [MIRROR_OPTIONS [MIRROR_OPTIONS] ... ] [NETWARE_OPTIONS]
    [GENERAL_OPTIONS] [FILESYSTEM_OPTIONS] [-public]

omnib -host Client:/ Label -device BackupDevice [MIRROR_OPTIONS
    [MIRROR_OPTIONS] ... ] [GENERAL_OPTIONS] [FILESYSTEM_OPTIONS]
    [-public]

omnib -vbfs Client:MountPoint Label -device BackupDevice
    [GENERAL_OPTIONS] [FILESYSTEM_OPTIONS] [-public]
    [MIRROR_OPTIONS [MIRROR_OPTIONS] ... ]

omnib -rawdisk Client Label SectionList -device BackupDevice
    [MIRROR_OPTIONS [MIRROR_OPTIONS] ... ] [GENERAL_OPTIONS]
    [-public]

omnib -omnidb Client:MountPoint Label -device BackupDevice
    [MIRROR_OPTIONS [MIRROR_OPTIONS] ... ] [GENERAL_OPTIONS]

omnib -datalist Name [BACKUP_SPECIFICATION_OPTIONS]

omnib -sap_list ListName [-barmode SapMode] [LIST_OPTIONS]

omnib -sapdb_list ListName [-barmode SapdbMode] [LIST_OPTIONS]
omnib -oracle8_list ListName [-barmode Oracle8Mode]
    [LIST_OPTIONS]

omnib -sybase_list ListName [-barmode SybaseMode] [LIST_OPTIONS]

omnib -informix_list ListName [-barmode InformixMode]
    [LIST_OPTIONS]

omnib -mssql_list ListName [-barmode MSSQLMode] [LIST_OPTIONS]

omnib -msese_list ListName [-barmode MSExchangeMode]
    [LIST_OPTIONS]

omnib -mbx_list ListName [-barmode MSMailboxMode] [LIST_OPTIONS]

omnib -lotus_list ListName [-barmode LotusMode] [LIST_OPTIONS]

omnib -msvssw_list ListName [-barmode VSSMode] [LIST_OPTIONS]

omnib -db2_list ListName [-barmode DB2Mode] [LIST_OPTIONS]

omnib -restart SessionID

MIRROR_OPTIONS

 -mirror BackupDevice [-pool MediaPool -prealloc MediaList ]

GENERAL_OPTIONS

 -preview

 -pool MediaPool

 -prealloc MediaList

 -protect {none | weeks n | days n | until Date | permanent}

 -report {warning | minor | major | critical}

 -pre_exec Pathname

 -post_exec Pathname

 -compress

 -encode

 -load {low | medium | high}

 -crc

 -no_monitor

 -keepcatalog {weeks n | days n | until Date}
 -variable var_name var_value
FILESYSTEM_OPTIONS
 -trees TreeList
 -only MatchPattern
 -exclude TreeList
 -skip MatchPattern
 -lock
 -touch
 -[no_]log | -log_dirs | - log_file
 -mode {Full | Incremental[1-9]}
 -enh_incr
 -[no_]hlink
 -size FromRange ToRange
WINFS_OPTIONS
 -no_share[_info]
 -[no_]nthlinks
 -[no_]archatt
 -vss [fallback]
BACKUP_SPECIFICATION_OPTIONS
 -select SelectList
 -mode {Full | Incremental[1-9]}
 -protect {none | weeks n | days n | until Date | permanent}
 -preview
 -disk_only
 -load {low | medium | high}
 -crc
 -no_monitor
```

LIST_OPTIONS

-barcmnd *Command*

-protect {none | weeks *n* | days *n* | until *Date* | permanent}

-load {low | medium | high}

-crc

-no_monitor

-test_bar

-disk_only

NETWARE_OPTIONS

-[no_]NWuncompress

NDMP_OPTIONS

-ndmp_user *UserName*

-ndmp_passwd *Password*

-ndmp_env *FileName*

-[no_]log | -log_dirs | -log_file

-mode {full | incremental1}

-pool *MediaPool*

-prealloc *MediaList*

-protect {none | weeks *n* | days *n* | until *Date* | permanent}

-report {warning | minor | major | critical}

-variable *var_name var_value*

OTHER OPTIONS

Type = Generic | NetApp | Celerra

SapMode = full | incremental

SapdbMode = full | diff | trans

Oracle8Mode = full | incr1 | ... | incr4

SybaseMode = full | trans

InformixMode = full | inf_incr1 | inf_incr2

```
MSSQLMode = full | diff | trans

MSExchangeMode = full | incr

MSMailboxMode = full | incr | incr1

LotusMode = full | incr

VSSMode = full | copy | incr | diff

DB2Mode = full | incr | delta

Date  = [YY]YY/MM/DD (1969 < [YY]YY < 2038)
```

DESCRIPTION

The omnib command uses a backup specification (list of file or database objects) to back up data objects. The following Data Protector functionality is supported:

Session management

Controls the backup sessions. The Session Manager reads the backup specification or uses the command options to determine what to back up and how many copies of the backup objects to create (object mirroring), then initiates the disk and media agents for disks and backup devices which will be used in the session. Once the session has completed, the Session Manager updates the MMDB with the session information.

Media management

Provides easy and efficient management of large sets of media by grouping media, tracking their status, implementing a media rotation policy, supporting the barcode recognition, vaulting the media, automating the library device operations, storing the media related information in a central place and sharing this information among several Data Protector cells.

Data compression

Writes data to media in a compressed format.

Data encoding

Writes data to media in an encoded mode.

Backup monitoring

When the backup command is executed, it sends a request (specifying the backup objects) to the Session Manager. When the Session Manager (SM) accepts the request, it assigns a unique SessionID to the session. You can use this SessionID to monitor the progress of the session using the `xomnimonitor` or `omnistat` commands. You can also use the `omniabort` command to terminate a session.

OPTIONS

`-version` Displays the version of the `omnib` command

`-help` Displays the usage synopsis for the `omnib` command

`-filesystem` *`Client:MountPoint Label`* Specifies the client, mount point and label of the filesystem to be backed up.

`-winfs` *`Client:MountPoint Label`* Specifies the client, mount point and label of the Windows filesystem to be backed up.

`-NetWare` *`Client:MountPoint Label`* Specifies the client, mount point and label of the NetWare filesystem to be backed up.

`-host` *`Client:/ Label`* Specifies the client to be backed up as a set of filesystems defined at backup time. The label is used as a prefix for each of these filesystem labels. Client backup is useful for systems with filesystem configuration that often changes.

`-vbfs` *`Client:MountPoint Label`* Specifies the client, mount point and label of the vbfs (very big file system) to be backed up. This option is used to back up an OmniStorage object.

`-rawdisk` *`Client Label SectionList`* Specifies the client, sections (pathnames of disk image sections) and label of the node to be backed up.

`-omnidb` *`Client:MountPoint Label`* Specifies the client and label of the Data Protector internal database (IDB) to be backed up.

`-datalist` *`Name`* Specifies the name of the backup specification file for the backup. The backup specification contains the data objects (filesystems and disk image sections) to be backed up.

`-restart` *`SessionID`* Tries to restart a failed session, specified by its sessionID.

`-sap_list` *`ListName`* Specifies the name of the SAP R/3 backup specification file for the backup. The SAP R/3 backup specification contains the SAP R/3 objects to be backed up.

`-sapdb_list` *ListName* Specifies the name of the SAP DB backup specification file for the backup. The SAP DB backup specification contains the SAP DB objects to be backed up.

`-oracle8_list` *ListName* Specifies the name of the Oracle backup specification file for the backup. The Oracle backup specification contains the Oracle objects to be backed up.

`-sybase_list` *ListName* Specifies the name of the Sybase backup specification file for the backup. The Sybase backup specification contains the Sybase objects to be backed up.

`-informix_list` *ListName* Specifies the name of the Informix Server backup specification file for the backup. The Informix Server backup specification contains the Informix Server objects to be backed up.

`-mssql_list` *ListName* Specifies the name of the MS SQL backup specification file for the backup. The MS SQL backup specification contains the MS SQL objects to be backed up.

`-msese_list` *ListName* Specifies the name of the Microsoft Exchange Server 2000/2003 backup specification file for the backup. The Microsoft Exchange Server 2000/2003 backup specification contains the Microsoft Exchange Server 2000/2003 objects to be backed up.

`-mbx_list` *ListName* Specifies the name of the MS Exchange Single Mailbox backup specification file for the backup. The MS Exchange Single Mailbox backup specification contains single mailboxes to be backed up.

`-lotus_list` *ListName* Specifies the name of the Lotus Notes/Domino Server backup specification file for the backup. The Lotus Notes/Domino Server backup specification contains the Lotus database objects to be backed up.

`-msvssw_list` *ListName* Specifies the name of the MS Copy backup specification file for the backup. The MS Copy backup specification contains the MS Copy objects to be backed up.

`-db2_list` *ListName* Specifies the name of the IBM DB2 UDB backup specification file for the backup. The IBM DB2 UDB backup specification contains the IBM DB2 UDB objects to be backed up.

`-device` *BackupDevice* Specifies the backup device to be used for the backup.

`-public` If you use this option, you allow other users to see and restore your data. By default, only the Data Protector administrator and the user who created a backup can see and restore the data.

MIRROR_OPTIONS

`-mirror` *`BackupDevice`* Specifies one or several backup devices to be used for object mirroring. Different backup devices should be specified for the backup and for each mirror.

`-pool` *`MediaPool`* Instructs the Session Manager to use an alternate media pool for object mirroring. By default, the default media pool for the backup device is used.

`-prealloc` *`MediaList`* Specifies a list of media to be used for object mirroring. If the Media Allocation policy for the pool is set to "strict", the media in the Prealloc list are used in the sequence shown in the list. If one of these media is unavailable, a mount prompt is issued. Note: If the Media Allocation Policy is "strict", you must specify a Prealloc list.

GENERAL_OPTIONS

`-preview` Checks the backup objects, backup devices and options you selected, without performing the backup. The check includes: backup objects, status of the backup device, available media, and the approximate amount of data which will be backed up.

`-pool` *`MediaPool`* Instructs the Session Manager to use an alternate media pool for the backup. By default, the default media pool for the backup device is used.

`-prealloc` *`MediaList`* Specifies a list of media to be used for the backup. If the Media Allocation policy for the pool is set to "strict", the media in the Prealloc list are used in the sequence shown in the list. If one of these media is unavailable, a mount prompt is issued. Note: If the Media Allocation Policy is "strict", you must specify a Prealloc list.

`-protect {none | weeks` *`n`* `| days` *`n`* `| until` *`Date`* `| permanent}` Sets the level of protection for the backup session. The media containing this backup session cannot be overwritten until the protection expires. By default, the protection is permanent.

`-report {warning | minor | major| critical}` Sets the level of error notification for the session. Errors are classified (in ascending order) as: `warning`, `minor`, `major` and `critical`. When you select a level, errors of this level and higher are displayed in the Monitor window. For example, if `major` is selected, only `major` and `critical` errors are reported. By default, all errors are reported.

`-pre_exec` *`Pathname`* Instructs the Session Manager to execute this command before starting the backup session. The complete *`Pathname`* of the command should be specified. The command is executed on the Session Manager system.

`-post_exec` *`Pathname`* Instructs the Session Manager to execute this command after the backup session. The complete *`Pathname`* of the command should be specified. The command is executed on the Session Manager system.

`-compress` Instructs the General Media Agent to write data to media in the compressed format.

This option is not supported on Novell NetWare. However, it is possible to uncompress files that were compressed with this option using older versions of Data Protector.

`-encode` Instructs the General Media Agent to write data to media in encoded format.

`-load {low | medium | high}` Specifies the level of network traffic generated by a session during a time period. `High` level generates as much traffic as allowed by the network, resulting in a faster backup. `Low` level has less impact on the network performance, but results in a slower backup. By default, this option is set to `high`.

`-crc` Instructs the General Media Agent to write a CRC checksum at the end of every block on the medium. If this option is used, you can later verify the CRC checksum on the medium by using the `omniver` command.

`-no_monitor` By default, the command monitors the session and displays the status of the session during the session. If this option is used, the SessionKey is displayed and the command is disconnected from the session.

`-keepcatalog {weeks` *`n`* `| days` *`n`* `| until` *`Date`*`}` This option specifies file catalog retention time. If you do not want to save the file catalog at all, use the `-no_log` option. By default, this option is set to the same value as specified by the `protection` option.

`-variable` *`var_name var_value`* This option lets you specify a variable name and its value for proper operation of some platforms and integrations, for example, for backing up and restoring an MPE. Setting user definable variables (a variable name and its value) enables flexible operation on some platforms and integrations with

Data Protector. The list of variables and their values that are configurable with Data Protector is dynamic and comes with Data Protector patches.

FILESYSTEM_OPTIONS

`-trees` *`TreeList`* Specifies the trees to be included in the backup. If this option is not used, the filesystem is backed up from the mount point level downwards. When specifying several trees, separate each *`Tree`* with a space. *`Tree`* must start with a /. Note that when specifying trees on UNIX systems, the complete tree must be specified including the mountpoint, whereas on Windows systems trees must be specified without volumes (drives). For example: `-tree \temp` (Windows system) or `-tree /usr/temp` (UNIX system). This option is not supported with Data Protector NDMP server integration.

`-only` *`MatchPattern`* Specifies that only files that match the *`MatchPattern`* will be backed up. This option is not supported with Data Protector NDMP server integration.

`-exclude` *`TreeList`* Specifies trees not to be backed up. This option is not supported with Data Protector NDMP server integration.

`-skip` *`MatchPattern`* Specifies that files matching the *`MatchPattern`* will not be backed up. This option is not supported with the Data Protector NDMP server integration.

`-lock` Instructs the Disk Agent to lock each file before backing it up. If the file is in use (and cannot be locked), the session manager displays a warning that this file can not be locked and backs up the file anyway. This warning is also logged to the catalog database. By default, files are not locked at backup.

`-no_log` Disables the logging of backed up files to the catalog database. By default, the filename and backup history of each backed up file is written to the catalog database.

`-log` The default option. All detailed information about backed up files and directories (filenames, file versions, and attributes) are logged to the Data Protector internal database (IDB). This allows you to browse directories and files before restore and in addition look at the file attributes. Data Protector can fast position on the tape when restoring a specific file.

`-log_dirs` If this option is specified, only the directories are logged into the database. By default, the filename and backup history of each backed up file is written to the catalog database.

`-log_file` All detailed information about backed up files and directories (filenames and file versions) is logged to the Data Protector internal database (IDB). This information allows you to search for backed up files and allows Data Protector to fast position the tape. It also does not take much space since some information on file details (file attributes) is not logged to the database.

`-mode {Full | Incremental[1-9]}` Specifies the mode for the backup session. `Full` mode backs up all specified files. `Incremental[1-9]` backs up only a subset of the specified files, based on whether or not the files were modified since the last `Full` or lower-level `Incremental` backup. Default is the `Full` mode. The level of incremental backup is based on the level number which is specified. For example, an incremental level 3 backs up only those files (of the specified files) which were modified since the last incremental level 2 or lower backup.

`-touch` Updates the access date/time of a file during a backup. Besides, on UNIX, Data Protector can also use the file's change time (inode modification time) as an incremental backup criterion. As a result, files with a changed name, location, or attributes are backed up in an incremental backup.

Whenever a file is opened, read, or locked, which happens during backup, the file's access time attribute changes. By default, Data Protector resets the file's access time attribute to the value it had before backup. However, on UNIX, this change impacts the file's change time.

This option is not supported on Novell NetWare.

`-no_hlink` If this option is specified, then hardlink detection is disabled and hard links are backed up as normal files. This speeds up the first traversal of the filesystem.

`-enh_incr` This option enables enhanced incremental backup. Unlike conventional incremental backup, enhanced incremental backup reliably detects and backs up files with changes in name, location, and attributes. It is also a prerequisite for subsequent object consolidation (synthetic backup).

NOTE: After you select this option, incremental backup will run in the enhanced mode only after a full backup is performed.

`-size` *FromRange ToRange* Limits backup to those files only, of which sizes are in the specified range. The sizes are set in KB. If you set *ToRange* to 0, all files larger then *FromRange* will be backed up.

WINFS_OPTIONS

`-no_share[_info]` If this option is specified, share information for directories on Windows systems is *not* backed up. By default, if a directory was shared on the network when a backup was run, the share information for directory is backed up, unless the `-no_share[_info]` option is specified.

Backing up share information for shared directories enables you to automatically share such directories after restore.

`-[no_]nthlinks` If this option is specified then NTFS hardlink detection is disabled and NTFS hard links are backed up as normal files. This speeds up the first traversal of the filesystem.

`-[no_]archatt` If this option is specified, Data Protector ignores archive attributes and detects changed files using other criteria, such as the file's modification time.

By default, Data Protector clears the file's archive attribute after a file is backed up and also uses the archive attribute as an incremental backup criterion. The archive attribute is automatically set by the system when the file's content, properties, name, or location changes.

If archive attributes cannot be cleared, an error is reported. This affects future incremental backups, so that the files are backed up, although they have not changed. This may happen when backing up removable media with write protection. A similar problem occurs with filesystem incremental ZDB because archive attributes are cleared on the replica and this action is not reflected on the source volume. For incremental ZDB, specify this option.

`-vss [fallback]` If the `-vss` option is specified, the VSS filesystem backup is performed. If the shadow copy creation on the system where the VSS filesystem backup is running, fails, the backup also fails by default. However, you can avoid backup failure by specifying the `fallback` option. In this case, the backup will continue as the normal filesystem backup.

BACKUP_SPECIFICATION_OPTIONS

`-select` *`SelectList`* Specifies which objects (of those in the backup specification) to back up. The *`SelectList`* is the list of objects to be backed up.

`-mode {Full | Incremental[1-9]}` Specifies the Mode for the backup session. `Full` mode backs up all specified files. `Incremental[1-9]` backs up only a subset of the specified files, based on whether or not the files were modified since the last Full or lower-level `Incremental` backup. Default is the `Full` mode. The level of incremental backup is based on the level number which is specified. For example, an incremental level 3 backs up only those files (of the specified files) which were modified since the last incremental level 2 (or lower) backup. Use incremental level 1 to back up files that were changed since last full backup only. The `Incremental` without level will back up the files that changed since the last backup only (regardless whether it was full or incremental of any level).

`-preview` Checks the backup objects, backup devices and options you selected, without performing the backup. The check includes: objects due for backup, status of the backup device, available media, and approximate amount of data which will be backed up.

`-disk_only` A ZDB related option. It instructs Data Protector to perform a ZDB-to-disk session rather than a ZDB-to-tape or ZDB-to-disk+tape session. With ZDB, if the option is not specified, a ZDB-to-tape or ZDB-to-disk+tape session is performed.

`-crc` Instructs the General Media Agent to write a CRC checksum at the end of every block on the medium. If this option is used, you can later verify the CRC checksum on the medium by using the `omnimver` command.

`-no_monitor` By default, the command monitors the session and displays the status of the session during the session. If this option is used only the SessionKey is displayed and the command is disconnected from the session.

LIST OPTIONS

-`barcmnd` *`Command`* Specifies the command that will be used instead of the command specified with `exec` option in the backup specification. The command should reside in the `/opt/omni/lbin` directory.

`-barmode` *`SapMode`* For SAP R/3 objects, the possible modes are `full` and `incremental`. The default value for this option is `full`.

-barmode *SapdbMode* For SAP DB objects, the possible modes are `full`, `diff` and `trans`. The `full` option triggers a full backup of the SAP DB instance, the `diff` option triggers a differential backup, and the `trans` option triggers an archive logs backup. The default value for this option is `full`.

-barmode *InformixMode* For Informix Server objects you can specify the following modes:

`full`: full backup of dbspaces specified during the backup specification creation time,

`inf_incr1`: first incremental backup,

`inf_incr2`: second incremental backup.

The default value for this option is `full`.

-barmode *Oracle8Mode* For Oracle objects you can specify `full` for full backup or `incr1` to `incr4` for incremental backups.

-barmode *SybaseMode* For Sybase objects you can specify `full` for full database backup or `trans` for transaction backup. The default value for this option is `full`.

-barmode *MSSQLMode* For MS SQL objects you can specify `full` for full database backup, `diff` for differential database backup or `trans` for transaction log backup. The default value for this option is `full`.

-barmode *MSExchangeMode* For MS Exchange objects you can specify `full` for full database and log files backup or `incr` for incremental backup of log files. The default value for this option is `full`.

-barmode *MSMailboxMode* For MS Exchange single mailboxes, you can specify `full` for a full mailbox backup, `incr` for an incremental mailbox backup, or `incr1` for an incremental1 mailbox backup. The default value for this option is `full`.

-barmode *LotusMode* For Lotus Notes/Domino Server objects you can specify `full` for full database backup or `incr` for a full backup of selected Lotus Notes/Domino objects, if the amount of data changed from the last backup is bigger than specified by the `-need_bck` barlist option. In case that transaction logging is enabled, the full backup of all archived transaction logs is also performed. The default value for this option is `full`.

`-barmode` *`VSSMode`* The available backup modes for VSS Writer objects depend on the writer: some writers support several modes (for example `full`, `copy`, `incr`, `diff` with Microsoft Exchange Server 2003 writer), others may support only `full`. See the *HP OpenView Storage Data Protector Integration Guide for Microsoft Applications: SQL Server, Exchange Server, and Volume Shadow Copy Service*.

`-barmode` *`DB2Mode`* For IBM DB2 UDB objects you can specify `full` for full database backup, `incr` for incremental database backup or `delta` for delta database backup. The default value for this option is `full`.

`-crc` Instructs the General Media Agent to write a CRC checksum at the end of every block on the medium. If this option is used, you can later verify the CRC checksum on the medium by using the `omnimver` command.

`-no_monitor` By default, the command monitors the session and displays the status of the session during the session. If this option is used, only the SessionKey is displayed, and the command is disconnected from the session.

`-test_bar` Enables preview mode for integrations. This option is supported only for Oracle, SAP, SAPDB, Single Mailbox Restore, Lotus Notes/ Domino Server, DB2, Informix Server and Sybase. ZDB is not supported.

The option checks the backup objects, backup devices and options you selected, without doing the backup. The check includes: objects due for backup, status of the backup device, available media, and the approximate amount of data which will be backed up.

`-disk_only` A ZDB related option. It instructs Data Protector to perform a ZDB-to-disk session rather than a ZDB-to-tape or ZDB-to-disk+tape session. With ZDB, if the option is not specified, a ZDB-to-tape or ZDB-to-disk+tape session is performed.

`NETWARE_OPTION`

-`NWuncompress` By default, Data Protector backs up Novell NetWare compressed files in their compressed format. Though this approach speeds up the backup process, it makes it impossible to restore the Novell NetWare compressed files to a non-compressed Novell NetWare volume. When this option is set to `NWuncompress`, Novell NetWare compressed files are uncompressed before being backed up. Files backed up in this form can be restored to non-compressed Novell NetWare volume.

NDMP_OPTIONS

`-ndmp_user` *`UserName`* Sets the username that is used by Data Protector to establish the connection to the NDMP server.

`-ndmp_passwd` *`Password`* Sets the password for the username that is used by Data Protector to establish the connection to the NDMP server.

`-ndmp_env` *`FileName`* Specifies the filename of file with NDMP environment variables for specific NDMP implementations.

`-no_log` Disables the logging of backed up files to the catalog database. By default, the filename and backup history of each backed up file is written to the catalog database.

`-log` The default option. All detailed information about backed up files and directories (filenames, file versions, and attributes) are logged to the Data Protector internal database (IDB). This allows you to browse directories and files before restore and in addition look at the file attributes. Data Protector can fast position on the tape when restoring a specific file.

`-log_dirs` If this option is specified, only the directories are logged into the database. By default, the filename and backup history of each backed up file is written to the catalog database.

`-log_file` All detailed information about backed up files and directories (filenames and file versions) is logged to the Data Protector internal database (IDB). This information allows you to search for backed up files and allows Data Protector to fast position the tape. It also does not take much space since some information on file details (file attributes) is not logged to the database.

`-mode {Full| Incremental[1-9]}` Specifies the mode for the backup session. Full mode backs up all specified files. `Incremental[1-9]` backs up only a subset of the specified files, based on whether or not the files were modified since the last `Full` or lower-level `Incremental` backup. Default is the `Full` mode. The level of incremental backup is based on the level number which is specified. For example, an incremental level 3 backs up only those files (of the specified files) which were modified since the last incremental level 2 or lower backup.

`-pool` *`MediaPool`* Instructs the Session Manager to use an alternate media pool for the backup. By default, the default media pool for the backup device is used.

`-prealloc` *`MediaList`* Specifies a list of media to be used for the backup. If the Media Allocation policy for the pool is set to "strict", the media in the Prealloc list are used in the sequence shown in the list. If one of these media is unavailable, a mount prompt is issued. Note: If the Media Allocation Policy is "strict", you must specify a Prealloc list.

`-protect {none | weeks` *`n`* `| days` *`n`* `| until` *`Date`* `| permanent}` Sets the level of error notification for the session. Errors are classified (in ascending order) as: `warning`, `minor`, `major` and `critical`. When you select a level, errors of this level and higher are displayed in the Monitor window. For example, if `major` is selected, only `major` and `critical` errors are reported. By default, all errors are reported.

`-variable` *`var_name var_value`* This option lets you specify a variable name and its value for proper operation of some platforms and integrations, for example, for backing up and restoring an MPE. Setting user definable variables (a variable name and its value) enables flexible operation on some platforms and integrations with Data Protector. The list of variables and their values that are configurable with Data Protector is dynamic and comes with Data Protector patches.

RETURN VALUES

See the man page `omniintro` for return values.

Additional return values of the `omnib` command are:

`10` There was an error while backing up some files. All agents completed successfully.

`11` One or more agents failed, or there was a database error.

`12` None of the agents completed the operation; session was aborted by Data Protector.

`13` Session was aborted by user.

EXAMPLES

The following examples illustrate how the `omnib` command works.

1. To do a backup of a tree "/usr" of filesystem "senna" with the label "work", using the compress option, to the backup device "DAT":

   ```
   omnib -device DAT -filesystem senna:/ work -tree /usr -compress
   ```

2. To back up the Data Protector internal database (IDB) on the client "geronimo" with the label "newDB" to the backup device "ADIC3" and to create two mirrors of this backup to the backup devices "LTO1" and "LTO2":

   ```
   omnib -omnidb geronimo:/ newDB -device ADIC3 -mirror LTO1 -mirror
   LTO2
   ```

3. To perform an incremental backup using the backup specification OMNIGROUP:

   ```
   omnib -datalist OMNIGROUP -mode Incremental
   ```

4. To preview a backup of the tree "/Amt3" of the filesystem "Munich", skipping the files with the ".fin" extension:

   ```
   omnib -preview -filesystem Munich:/ -tree /Amt3 -skip "*.fin"
   ```

5. To run a disk image backup of the section "/dev/rdsk/c201d1s0" on the client "xanadu" to the backup device "Exa" and protecting the session against overwrite for 4 weeks:

   ```
   omnib -rawdisk xanadu section /dev/rdsk/c201d1s0 -dev Exa -protect
   weeks 4
   ```

6. To run a full Lotus backup using the "test2" backup specification with the high network load and permanent protection set:

   ```
   omnib -lotus_list test2 -barmode full -protect permanent -load high
   ```

7. To start a full backup using an existing IBM DB2 UDB backup specification called “TEST”, and to set data protection to 10 weeks, execute the following command:

   ```
   omnib -db2_list TEST -barmode full -protect weeks 10
   ```

8. To start a differential backup using an existing SAP DB backup specification called “test”, and write a CRC checksum at the end of every block on the medium, execute the following command:

   ```
   omnib -sapdb_list test -barmode diff -crc
   ```

SEE ALSO

omnir(1), omniobjconsolidate(1), omniobjcopy(1)

omnicc (1)

NAME

omnicc – reports the number of configured and available Data Protector licenses, installs the licenses, imports and exports systems in and out of a cell, manages access to secured clients.

SYNOPSIS

```
omnicc -version | -help

omnicc -redistribute

omnicc -import_host ClientName [-virtual]

omnicc -import_ndmp ClientName -type NdmpType -port Port -user
    UserName -passwd Password

omnicc -import_is ClientName

omnicc -update_host ClientName

omnicc -update_all [-force_cs]

omnicc -export_host ClientName

omnicc -list_authorities ClientName

omnicc -secure_client ClientName -authorities ClientName1
    [ClientName2... ]

omnicc -unsecure_client ClientName

omnicc -install_license password

omnicc -password_info

omnicc -confirm_mom_clients

omnicc -update_mom_server

omnicc -check_licenses [-detail]

omnicc [-query]
```

NdmpType

```
 Generic | NetApp | Celerra
```

DESCRIPTION

The omnicc command is used for licensing, importing and exporting clients, and for managing secured clients.

OPTIONS

-version Displays the version of the omnicc command.

-help Displays the usage synopsis for the omnicc command.

-check_licenses [-detail] Reports licensing related information from the cell.

If the -detail option is not specified, the command returns information on whether the Data Protector licensing is covered or not.

If the -detail option is specified, a detailed report is produced. The following information is returned for every license in the cell: license name, licenses installed, licenses used and licenses required.

In a MoM environment with the CMMDB configured, when producing a license report for the items that are subject to libraries and devices related licenses, such as media (including advanced file device media), backup devices, drives and slots, the omnicc command must be run on the Cell Manager with the CMMDB installed.

-query Displays information about the number of available licenses.

-redistribute Displays licensing information for multicell environments. The first part shows the number of allocated licenses and the second shows the number of licenses actually used per server.

-import_host *ClientName* Imports the specified client into a cell. This allows you to move a client between two cells without reinstalling the Data Protector modules.

When you import the next one among multiple network names (clusters, service guards), use the -virtual option. This way you keep Data Protector from assigning licenses to all the network names of the same system.

-import_is *ClientName* Imports an already installed Installation Server into the cell.

-import_ndmp *ClientName* Imports the specified NDMP server into the cell.

`-type` *`NdmpType`* Sets the NDMP data format when importing an NDMP server into a cell.

`-port` *`Port`* Sets the TCP/IP port number of the NDMP server when importing an NDMP server into a cell.

`-user` *`UserName`* Sets the username that is used by Data Protector to establish the connection to the NDMP server when importing an NDMP server into a cell.

`-passwd` *`Password`* Sets the password for the username that is used by Data Protector to establish the connection to the NDMP server when importing an NDMP server into a cell.

`-update_host` *`ClientName`* Updates the version information and installed packages information in the Cell Manager configuration file for the specified client. Useful when automatic update fails due to any reason.

`-update_all` Updates the version information and installed packages information in the Cell Manager configuration file for all clients in the cell. Useful when automatic update fails due to any reason.

`-force_cs` Checks if the clients belong to the current cell. If not, the command will change the cell to the current cell.

`-export_host` *`ClientName`* Exports the specified client from the cell. This enables you to remove a client from the cell without uninstalling its Data Protector modules.

`-list_authorities` *`ClientName`* Lists systems from which the specified client accepts requests on the Data Protector port (by default `5555`).

`-secure_client` *`ClientName`* Specifies the client to be secured.

`-authorities` *`ClientName [ClientName2...]`* Specifies systems from which the specified client accepts requests on the Data Protector port (by default `5555`). Consequently, other computers will not be able to access this client. For tasks like backup and restore, starting pre- or post-execution scripts, or importing and exporting clients, the client checks whether the computer which triggers one of these tasks via the Data Protector port is allowed to do so. This security mechanism instructs the client to accept such actions only from the systems specified by this option.

`-unsecure_client` *`ClientName`* Specifies the client from which you want to remove security. Such a client will enable access to all systems in the cell.

`-install_license` *`password`* Installs encrypted Data Protector license.

`-confirm_mom_clients` Collects cell_info files from MoM clients (`/etc/opt/omni/server/cell/mom_info` on UNIX clients or *`<Data_Protector_home>`*`\config\server\cell\mom_info` on Windows clients) and stores them into the `/etc/opt/omni/server/mom/cell_info` directory on UNIX or into *`<Data_Protector_home>`*`\config\server\mom\cell_info` directory on Windows) on the MoM Manager under client Cell Manager name. Use this command when switching MoM clients to CMMDB mode. `Omnicc` with this option specified has to be executed on the MoM Manager.

`-update_mom_server` Pushes mom_info file (located in `/etc/opt/omni/server/cell` directory on a UNIX or in *`<Data_Protector_home>`*`\config\server\cell` directory on Windows) to MoM and CMMDB server (to `/etc/opt/omni/server/mom/cell_info` directory on a UNIX MoM or into *`<Data_Protector_home>`*`\config\server\mom\cell_info` directory on a Windows MoM) under client Cell Manager name. Use this command when switching to CMMDB mode. `Omnicc` with this option specified has to be executed on the client Cell Manager.

`-password_info` Displays information about installed license passwords.

SEE ALSO

omnicheck(1M), omnisv(1M), omnicellinfo(1), omnidlc(1M), omniinstlic(1M)

omnicellinfo (1)

NAME

omnicellinfo – displays configuration information about the Data Protector cell

SYNOPSIS

```
omnicellinfo -version | -help

omnicellinfo -servers

omnicellinfo -group

omnicellinfo {-object [schedule | no_schedule] [-group Group]} |
    -db

omnicellinfo {-mm | -dev}[-detail]

omnicellinfo {-dlinfo [-group Group]} | -cell [brief] {-schinfo
    [Backup_Specification | -days NumberDays | -group Group]} |
    {-dlobj [-group Group]} | {-trees [-group Group]} | -allbdf |
    -acl
```

DESCRIPTION

The `omnicellinfo` command displays information about data objects, media pools, devices, clients, database, backup specifications and backup specification groups in the cell. It can be also used to display the cell servers in multicell environments.

Some options recognized by `omnicellinfo` are intended primarily for generating reports by shell/awk/perl scripts. Information produced is formatted in records with a newline as field separator and a blank line as record separator. Those options are: `-dlinfo`, `-schinfo`, `-dlobj`, `-trees` and `-allbdf`.

OPTIONS

`-version` Displays the version of the `omnicellinfo` command.

`-help` Displays the usage synopsis for the `omnicellinfo` command.

`-servers` Displays the list of cell servers that are included in the multicell environment.

`-group` Displays the backup specification groups that contain backup specifications. Note that the backup specification group named `Default` is not displayed.

`-object [schedule | no_schedule]` Displays information about objects (filesystems, databases and disk images) in the cell. The report shows: Object (object type, client name, and mountpoint), Label, and Next Scheduled Backup Date. When you use the `schedule` option, the report only shows those objects which are scheduled for backup. When you use the `-no_schedule` option, the report only shows those objects which are not scheduled for backup. By default, all objects (scheduled and unscheduled) are listed.

`-mm` Displays information about the media and media pools in the cell. The report shows for each pool: the Pool Name, Media Class, Media Usage Policy, Media Allocation Policy, and Amount of Free Space in the pool.

`-dev` Displays information about the backup devices in the cell. The report shows for each device: the Device Name, Client Name, Device Type and Media Pool.

`-db` Displays information about the Data Protector internal database (IDB). The database is divided in logical structures, for each of these structures the report shows: Disk Space Used, Records Used and Records Total.

`-cell` Displays information about the configured clients in the cell. The report shows for each client: client name, operating system, cell console version, Disk Agent version, Media Agent version, GUI version and all installed Data Protector integrations versions. There is also a short summary which shows the total number of clients and, if the `brief` suboption was not specified, all possible Data Protector software components, together with the total number of every software component in the cell. If the `brief` suboption was specified, only the installed Data Protector software components together with the total number of every software component in the cell is listed.

`-detail` The `-detail` option can be used in combination with the `-dev` and `-mm` options to produce a more detailed report.

`-dlinfo` Shows information about backup specifications. For each backup specifications it lists the name of the backup specification, session owner, pre-exec and post-exec script. Session owner is in format *USER.GROUP@CLIENT*.

`-schinfo [`*`Backup_Specification`*` | -days `*`NumberDays`*`]` Shows information about backup specification scheduling. If *`Backup_Specification`* and `-days` option are not specified, the command displays the next schedule time for each backup specification. If backup specification is specified the command lists all schedules in the next year for the specified backup specification. Option `-days` can be used to display schedules of all backup specifications for a specified number of days.

`-dlobj`
Shows information about all objects in backup specifications. For each object it lists object type, object name (in format *`ClientName:PathName`*), description, and the name of the backup specification. After this, the device and poolname fields are listed for each device used in the backup specification making the size of the records variable.

`-trees`
Shows information about all defined trees in backup specifications. For each tree, it lists filesystem name (in format *`ClientName:Pathname`*), tree, description, backup device, media pool and name of the backup specification.

`-acl`
Displays all Data Protector access permissions that the user running the command has.

`-group` *`Group`*
This option allows you to limit the output of the command to single backup specification group. The following options support this: `-dlinfo`, `-schinfo`, `-dlobj`, `-trees` and `-object`.

EXAMPLES

The following examples illustrate how the `omnicellinfo` command works.

1. To list detailed information about the configured devices:

   ```
   omnicellinfo -object schedule
   ```

2. To list detailed information about the configured devices:

   ```
   omnicellinfo -dev -detail
   ```

SEE ALSO

omnicc(1), omnicheck(1M), omnisv(1M), omnidlc(1M), omniinstlic(1M)

omniclus (1)

NAME

omniclus – helps to manage load balancing in the cluster environment in case of an application (Data Protector or other) failover

SYNOPSIS

```
omniclus -version | -help

omniclus -clus cluster_name -session {* | backup_specification}
    -abortsess [-abortid {== | !=} application_id]

omniclus -clus cluster_name -inhibit {* | 0 | minutes}

omniclus -clus cluster_name -session {* | backup_specification}
    -symlink {split | active}
```

NOTE: On UNIX systems replace the `*` wildcard with `'*'`

Under Windows, the `-noclus` option can be specified directly after `-clus` to prevent loading of the cluster dynamic library

DESCRIPTION

The `omniclus` command, that is common to all platforms (UNIX and Windows) allows the user to send the Data Protector Cell Manager special events that in some way control the behavior of the Cell Manager and the backup sessions in a cluster environment. `omniclus` allows balance loading by offering additional (CLI) control of the Cell Manager in the cluster environment:

- abort sessions
- temporarily disabling the Cell Manager for backups
- specify the state of EMC/Symmetrix links after an application failover

Note: that the *cluster_name* specified with the `-clus` switch must be a Cluster aware Data Protector Cell Manager.

OPTIONS

`-help` Displays the usage synopsis for the `omniclus` command.

`-clus` *cluster_name* Specifies the Cluster Cell Manager.

`-session * | backup_specification` Specifies the session(s) to which the abort message should be sent.

`-abortsess` Specifies the abort session command.

`-abortid {== | !=} application_id` Specifies the application identification.

`-inhibit {* | 0 | minutes}` Specifies the number of minutes for Cell Manager backup inactivity, where `*` means forever and `0` means activate now.

`-symlink {active | split}` Specifies the state of the EMC/Symmetrix links upon application failover if a backup is running.

NOTE

The command can only be used in the cluster environment.

EXAMPLES

Following example illustrates how the omniclus command works.

1. Abort all running sessions

   ```
   omniclus -clus cluster.domain.com -session * -abortsess
   ```

   Note: On UNIX systems replace the `*` wildcard with `'*'` .

   The utility will connect to all running sessions and will send them abort messages. The state of the sessions can be then checked with the Data Protector `omnistat` utility.

2. Abort specific running sessions

   ```
   omniclus -clus cluster.domain.com -session mybackup -abortsess
   ```

   The utility will connect to backup session managers issuing abort messages and sending them additional information - the backup specification name. Each backup session manager checks whether the command addresses it and if this is the case it aborts.

3. Abort sessions (all or specific) with application identifications

   ```
   omniclus -clus obvs.hermes.com -session * -abortsess -abortid != 10
   ```

   Note: On UNIX systems replace the `*` wildcard with `'*'`.

This way the user can define groups of sessions and abort only the ones that are actually related to the application that failed over. For example a backup session that performs a normal filesystem backup of a remote client is not aborted because an application server switches, while the application server backup can be aborted.

4. Temporarily disabling the Data Protector cell

   The following command will inhibit backup sessions for twenty minutes:

   ```
   omniclus -clus cluster.domain.com -inhibit 20
   ```

   The following command will inhibit backup sessions forever:

   ```
   omniclus -clus cluster.domain.com -inhibit *
   ```

   Note: On UNIX systems replace the * wildcard with '*'.

   The following command will re-activate backup sessions immediately:

   ```
   omniclus -clus cluster.domain.com -inhibit 0
   ```

5. EMC/Symmetrix links

   The following syntax will connect to specific (running) backup session managers and inform them to left the EMC/Symmetrix links split:

   ```
   omniclus -clus cluster.domain.com -session * -symlink split
   ```

   Note: On UNIX systems replace the * wildcard with '*'.

   The following syntax will connect to specific (running) backup session managers and inform them to left the EMC/Symmetrix links active (established):

   ```
   omniclus -clus cluster.domain.com -session * -symlink active
   ```

   Note: On UNIX systems replace the * wildcard with '*'.

SEE ALSO

omnirsh(1M)

omnicreatedl (1)

NAME

omnicreatedl – creates a filesystem backup specification file (datalist); or a HP StorageWorks Disk Array XP, HP StorageWorks Virtual Array, or HP StorageWorks Enterprise Virtual Array Microsoft Exchange Server 2000/2003 ZDB backup specification file (datalist).

SYNOPSIS

```
omnicreatedl -help | -version
```

FILESYSTEM BACKUP SPECIFICATION

```
omnicreatedl [-datalist Name] [-host HostName1 [HostName2 ...]]
    [-device BackupDevice]
```

MS EXCHANGE ZDB BACKUP SPECIFICATION

```
omnicreatedl -ex2000 -datalist Name [-device Name]
    {DISK_ARRAY_XP_OPTIONS | VIRTUAL_ARRAY_OPTIONS |
    ENTERPRISE_VIRTUAL_ARRAY_OPTIONS} EXCHANGE_2000/2003_OPTIONS
    [-force] [-virtualSrv Name]
```

DISK_ARRAY_XP_OPTIONS

1. ZDB-to-disk or ZDB-to-disk+tape session

```
 -split_mirror -sse -local app_sys bck_sys  [-mirrors MU_Numbers]
    -instant_restore [-leave_enabled_bs] [-split | -establish]
```

2. ZDB-to-tape session

```
 -split_mirror -sse -local app_sys bck_sys  [-mirrors MU_Numbers]
    [-keep_version [-leave_enabled_bs] ] [-split | -establish]

 -split_mirror -sse {-remote app_sys bck_sys | -combined app_sys
    bck_sys} [-keep_version [-leave_enabled_bs] ] [-split |
    -establish]
```

VIRTUAL_ARRAY_OPTIONS

1. ZDB-to-disk or ZDB-to-disk+tape session

```
 -snapshot -va app_sys bck_sys -instant_recovery  [-snapshots
    number] [-leave_enabled_bs] [-lun_security]
```

2. ZDB-to-tape session

```
 -snapshot -va app_sys bck_sys [-use_existing_snapshot]
    [-leave_version [-leave_enabled_bs] ] [-lun_security]
```

ENTERPRISE_VIRTUAL_ARRAY_OPTIONS

1. ZDB-to-disk session

```
 -snapshot {-eva | -smis} app_sys bck_sys -instant_recovery
    [-snapshots  number]
```

2. ZDB-to-disk+tape session

```
 -snapshot {-eva | -smis} app_sys bck_sys -instant_recovery
    [-snapshots number] [-wait_clonecopy  number]
```

3. ZDB-to-tape session

```
 -snapshot {-eva | -smis} app_sys bck_sys -snapshot_type
    {standard | vsnap | clone [-wait_clonecopy  number]}
    -snapshot_policy {strict  | loose} [-snapshots number]
```

EXCHANGE_2000/2003_OPTIONS

```
 -annotation {MIS | SRS | KMS}

 {-all_storage_groups | -storage_group Storage_Group_Name1
    [-store Store1 [Store2 ...]] [-storage_group
    Storage_Group_Name2 [-store Store1 [Store2...]] ...]}
```

DESCRIPTION

FILESYSTEM BACKUP SPECIFICATION

The `omnicreatedl` command creates a filesystem backup specification file (datalist). It searches all specified clients for local mount points and puts them in the backup specification or on the `stdout` if no backup specification name is specified. If no client is specified, all clients in the cell are searched.

MICROSOFT EXCHANGE SERVER ZDB BACKUP SPECIFICATION

The `omnicreatedl` command is also used to create an Exchange ZDB backup specification file for the following disk arrays:

HP StorageWorks Disk Array XP

HP StorageWorks Virtual Array

HP StorageWorks Enterprise Virtual Array

If you create an HP StorageWorks Enterprise Virtual Array Exchange ZDB backup specification, you run the `omnicreatedl` command with either the `-eva` or the `-smis` parameter. The choice of the parameter depends on the EVA agent - HP StorageWorks EVA Agent (legacy) or HP StorageWorks EVA SMI-S Agent - you have installed. Use the `-eva` parameter with the EVA Agent (legacy); if you have the SMI-S Agent installed, use the `-smis` parameter.

When creating an Exchange ZDB backup specification file, if the circular logging is disabled for any storage group, an Exchange ZDB transaction logs backup specification file for each such storage group specified in the Exchange ZDB backup specification file is additionally created.

An Exchange ZDB backup specification file includes the stop/ quiesce the application and restart the application scripts (`omniEx2000.exe`) sections for dismounting/ mounting backed up stores and checking their consistency. A backup specification can be edited later using the Data Protector GUI to modify backup devices, ZDB options, schedule, etc.

For a Microsoft Exchange Server ZDB, the *final* decision on whether the created backup specification will start a ZDB-to-disk, ZDB-to-disk+tape or ZDB-to-tape session depends on the Data Protector `omnib` command options selection.

OPTIONS

`-help` Displays the usage synopsis of the `omnicreatedl` command

`-version` Displays the version of the `omnicreatedl` command

FILESYSTEM BACKUP SPECIFICATION

`-datalist` *`Name`* Specifies the name of the backup specification file (datalist) for filesystem backup. The backup specification file is created in the `/etc/opt/omni/server/datalists` (HP-UX or Solaris systems) or in the *`<Data_Protector_home>`*`\config\server\datalists` (Windows systems) directory on the Cell Manager. If this option is not specified, backup specification objects are written to stdout.

`-host` *`HostName1`* `[`*`HostName2`*`]` List of all clients whose filesystems will be included in the backup specification. If this option is not specified, all clients from the cell are used.

`-device` *`BackupDevice`* Specifies the backup device to be used for backup. If this option is not used, the backup device must be specified using the Data Protector GUI.

MICROSOFT EXCHANGE SERVER ZDB BACKUP SPECIFICATION

`-ex2000` Instructs the `omnicreatedl` command to create a Disk Array XP Microsoft Exchange Server ZDB backup specification file and, if circular logging is disabled for any storage group specified, a Disk Array XP Microsoft Exchange Server ZDB transaction logs backup specification file(s) for every such storage group.

`-datalist` *`Name`* Specifies the name of the Microsoft Exchange Server ZDB backup specification file (datalist) for the Microsoft Exchange Server ZDB. The datalist is created in the *`<Data_Protector_home>`*`\config\server\datalists` directory (Windows systems) or in the `/etc/opt/omni/server/datalists` (HP-UX or Solaris systems) directory on the Cell Manager.

The corresponding datalist(s) for Microsoft Exchange Server logs for every storage group specified that has the circular logging disabled is/are also created in the same directory with the file name `<Storage_Group_Name> (LOGS) <app_sys>`.

If any of the thus created backup specification files (datalists) has a name that already exists, the `omnicreatedl` command issues a warning and, depending on whether the `-force` option is set or not, overwrites the existing backup specification files with the same name or aborts the action.

`-force` Forces overwriting of an existing backup specification file with the same name.

`-virtualSrv` *`Name`* The name of the Microsoft Exchange Server virtual server. This option is obligatory and used only in cluster configurations.

`DISK_ARRAY_XP_OPTIONS`

`-split_mirror -sse` Instructs the `omnicreatedl` command to create a Disk Array XP Microsoft Exchange Server ZDB backup specification file.

`-local` *`app_sys bck_sys`* Selects the Business Copy XP (BC) configuration, with the application system *`app_sys`* and the backup system *`bck_sys`*.

`-remote` *`app_sys bck_sys`* Selects the Continuous Access XP (CA) configuration, with the application system *`app_sys`* and the backup system *`bck_sys`*.

`-combined` *`app_sys bck_sys`* Selects the Combined (Continuous Access XP + Business Copy XP) configuration, with the application system *`app_sys`* and the backup system *`bck_sys`*.

-mirrors *MU_numbers* Specifies a specific replica or a replica set to be used in the backup session to define a replica set from which the integration, according to the replica set rotation, selects one replica to be used in the backup session. If this option is not specified, the *MU#* 0 is set.

Enter an integer number from 0 to 2, any range of integer numbers from 0 to 2, or any combination of integer numbers from 0 to 2 separated by a comma. For example:

```
1

1-2

2,0,1
```

If the sequence is specified, it does not set the order in which the replicas are used. They are used according to the replica set rotation.

If a range is entered, it must be specified in ascending order.

-instant_restore When this option is specified, the omnicreatedl command automatically sets the -keep_version option. Specify the -instant_restore option to enable ZDB to disk or ZDB to disk+tape and instant recovery from the replica. If this option is not specified, it is not possible to perform ZDB to disk or ZDB to disk+tape and instant recovery from the replica. However, this option does not influence the replica set rotation.

-keep_version If this option is specified, the pairs involved in the backup session will remain split after the backup session, enabling you to restore from the replica if an instant recovery is needed. If this option is not specified, the disks involved in the backup session are resynchronized after the backup session, only if one or no replica is set by the -mirrors option. If more than one replica is set by the -mirrors option, the disks involved in the backup session will remain split after the backup session. If this option is not specified, it is not possible to specify the -leave_enabled_bs option.

-leave_enabled_bs To specify this option, the -keep_version option has to be specified. By default, Data Protector dismounts the filesystems on the backup system after each backup. If this option is specified, the filesystems remain mounted after the backup. Thus, you can use the backup system for some data warehouse activity afterwards, but not for instant recovery.

`-split` If this option is specified, the mirrored disks in the replica selected for the current backup session are resynchronized with the P-VOLs at the start of the current backup session. If neither the `-split` nor the `-establish` option is specified, the `-establish` option is set automatically.

`-establish` If this option is specified and the replica for the next backup is not synchronized, a resync will be initiated before the next backup. If neither the `-split` nor the `-establish` option is specified, the `-establish` option is set automatically.

VIRTUAL_ARRAY_OPTIONS

`-snapshot -va` *app_sys bck_sys* Sets the application system *app_sys* and the backup system *bck_sys* and instructs the omnicreatedl command to create an HP StorageWorks Virtual Array snapshot datalist.

`-instant_recovery` Specify this option if you want to perform either a ZDB to disk or a ZDB to disk+tape and leave the replica on a disk array (after the backup session) to use it in future for instant recovery. If this option is not set, it is not possible to perform instant recovery from the replica created or reused in this backup session.

If this option is specified, you should also set the `-snapshots` option.

Note that when this option is selected, the options `-use_existing_snapshot` and `-leave_version` are automatically set by Data Protector.

`-snapshots` *number* Specify the number of replicas you want to keep on a disk array. During every backup session, Data Protector creates a new replica and leaves it on a disk array as long as the specified number is not reached. When the specified number is reached, Data Protector reuses the oldest replica.

This option sets the number of replicas in the replica set for a backup specification.

You need to specify this number, if you have selected the `-instant_recovery` option.

If the option is not specified, it is set to 1. The maximum is 1024.

`-use_existing_snapshot` By default, Data Protector automatically sets this option if the `-instant_recovery` option is specified.

If configuring a ZDB-to-tape session, select this option if you want to reuse an existing replica.

Data Protector can reuse a replica only if the following condition is met, otherwise the backup session will fail:

On a disk array, there must already exist a replica that can be reused. Only replicas that are not marked for instant recovery (are not part of the replica set) or include no snapshots that are listed in the VA LUN exclude file can be reused.

Any replica which is not marked for instant recovery or includes no snapshots which are listed in the VA LUN exclude file can be reused.

`-leave_version` By default, Data Protector automatically sets this option if the `-instant_recovery` option is specified.

If configuring a ZDB-to-tape session (the `-instant_restore` option is not specified), specify this option if you wish to keep the replica on a disk array after the ZDB-to-tape session is completed. In this case, the replica will not be available for instant recovery, but can be reused in future backup sessions using the same backup specification with the option `-use_existing_snapshot` specified.

If this option is not specified, the replica is deleted after the backup session is completed.

`-leave_enabled_bs` To specify this option, the `-leave_version` option has to be specified. By default, Data Protector dismounts the filesystems on the backup system after each backup. If this option is specified, the filesystems remain mounted after the backup is finished.

Thus, you can use the backup system for some data warehouse activity afterwards, but not for instant recovery.

By default, this option is not selected.

`-lun_security` Specify this option if you want to apply the LUN security to the child LUNs (target volumes or snapshots) that the integration creates.

If Secure Manager is activated on HP StorageWorks Virtual Array, you must specify this option and configure passwords correctly, otherwise the backup sessions will fail.

The LUN security is set using the `omnidbva` command.

By default, this option is not selected.

ENTERPRISE_VIRTUAL_ARRAY_OPTIONS

`-snapshot {-eva | -smis}` *`app_sys bck_sys`* Instructs the `omnicreatedl` command to create an HP StorageWorks Enterprise Virtual Array snapshot backup specification file and sets the application system *app_sys* and the backup system *bck_sys*. Use the `-eva` option if you have the HP StorageWorks EVA Agent (legacy), or the `-smis` option if you have the HP StorageWorks EVA SMI-S Agent installed.

`-instant_recovery` This parameter is optional. Specify this option, if you want to perform either a ZDB to disk or a ZDB to disk+tape and leave the replica on a disk array (after the backup session) to use it in future for instant recovery. If this option is not set, it is not possible to perform instant recovery from the replica created in this backup session.

Note that when this option is selected, the options `-snapshot_type clone` and `-snapshot_policy strict` are automatically set by Data Protector. If the option `-snapshots` *`number`* is not specified, it is set to `1`.

`-snapshots` *`number`* This parameter is optional. By default, Data Protector automatically sets this option to `1` if the `-instant_recovery` option is specified.

Specify this option if you wish to keep the replica on a disk array after a backup session is completed. With *number*, specify the number of replicas you want to keep on a disk array. During every backup session, Data Protector creates a new replica and leaves it on a disk array as long as the specified number is not reached. When the specified number is reached, Data Protector deletes the oldest replica and creates a new one.

The maximum number for vsnaps and standard snapshots is `7`. Data Protector does not limit the number of replicas rotated, but the session will fail if the limit is exceeded.

Note that this option sets the number of replicas in the replica set for a backup specification.

`-snapshot_type {standard | vsnap | clone}` This option instructs Data Protector to create one of the two types of HP StorageWorks Enterprise Virtual Array snapshots during the backup session.

Setting `standard` creates snapshots with the pre-allocation of disk space.

Setting `vsnap` creates snapshots without the pre-allocation of disk space.

Setting `clone` creates a clone of an original virtual disk.

`-snapshot_policy {strict | loose}` Specifies how Data Protector creates snapshots with regard to types of already existing snapshots for the same original virtual disk.

When `strict` is set, Data Protector attempts to create snapshots of the type selected by the `-snapshot_type` option. If some of the original virtual disks used in the backup session already have existing snapshots of different type, the selected type of snapshots cannot be used. Such a backup session will be aborted.

When `loose` is set, Data Protector creates snapshots of different type than specified by the `-snapshot_type` option, when this would help to make a successful session. For example, if you select standard snapshots to be created, but Data Protector detects that standard snapshots cannot be created because some vsnaps or snapclones of the source volumes already exist in a replica set, the following happens: with the loose option selected, Data Protector creates either vsnaps (if vsnaps already exist) or snapclones (if snapclones already exist) instead of standard snapshots.

Note that Data Protector can use only one type of snapshots in the backup session. In case when some of the original virtual disks used in the backup session have existing standard snapshots and some of them existing vsnaps, the backup session will be aborted.

`-wait_clonecopy` *`number`* This parameter is optional and can be selected only if the `-snapshot_type clone` option is selected.

In the case of a ZDB to tape or a ZDB to disk+tape, specify this option if you want to delay moving data to tape media until the cloning process is completed. By *`number`*, specify the maximum waiting time in minutes. After the specified number of minutes, the backup to tape will start even if the cloning process is not finished yet.

With this option, you prevent degradation of the application data access times during the phase of backup to tape.

EXCHANGE_2000/2003_OPTIONS

`-annotation {MIS | SRS | KMS}` This option specifies the possible Microsoft Exchange Server annotations: Microsoft Information Store (`MIS`), Site Replication Service (`SRS`), and Key Management Service (`KMS`). `MIS` is the default setting and does not need to be specified in case when the `MIS` will be backed up.

`-all_storage_groups` This option creates a backup specification for all databases relating to Microsoft Exchange Server Microsoft Information Store. It must be specified by the `-annotation` MIS parameter.

`-storage_group` *storage group name* This option creates a backup specification for all stores relating to the specified storage group. Multiple declarations of the `-storage_group` parameter are possible to create a backup specification for the selected storage groups.

Logical storage group names can be obtained by using the Exchange System Administrator tool, which is a part of Microsoft Exchange Server.

`-store` *Store1* `[`*Store2*`...]` When the `-store` parameter is specified, backup specification is created only for specified store(s) inside the storage group. List of stores can be specified after the `-store` parameter to create a backup specification for many stores.

Store names can be obtained by using Exchange System Administrator tool, which is a part of Microsoft Exchange Server.

EXAMPLES

The following examples show how the `omnicreatedl` command works:

1. To create a HP StorageWorks Disk Array XP Microsoft Exchange Server ZDB-to-tape backup specification file named “Exchange_example” for a Microsoft Exchange Server running on client “computer1.company.com” with the backup system “computer2.company.com”, to back up all storage groups relating to Microsoft Information Store, run:

```
omnicreatedl -ex2000 -datalist Exchange_example -all_storage_groups
-split_mirror -sse -local computer1.company.com
computer2.company.com
```

The omnicreatedl command creates the HP StorageWorks Disk Array XP Microsoft Exchange Server ZDB-to-tape backup specification file named “Exchange_example” and additional HP StorageWorks Disk Array XP Microsoft Exchange Server ZDB transaction logs backup specification files (in case they do not already exist) for each storage group with disabled circular logging option.

2. To create HP StorageWorks Disk Array XP Microsoft Exchange Server ZDB-to-tape backup specification file named “Exchange_example” for a Microsoft Exchange Server running on client “computer1.company.com” with the backup system “computer2.company.com”, to back up entire First Storage Group and Test Storage Group (both have circular logging disabled), run:

```
omnicreatedl -ex2000 -datalist Exchange_example -storage_group
"First Storage Group" -storage group "Test Storage Group"
-split_mirror -sse -local computer1.company.com
computer2.company.com
```

The omnicreatedl command creates the HP StorageWorks Disk Array XP Microsoft Exchange Server ZDB-to-tape backup specification file (datalist) named “Exchange_example” and two additional HP StorageWorks Disk Array XP Microsoft Exchange Server ZDB transaction logs backup specification files (if they do not already exist) named: “First Storage Group (LOGS) computer1.company.com” for First Storage Group log files backup and “Test Storage Group (LOGS) computer1.company.com” for Test Storage Group log files backup.

3. To create an HP StorageWorks Disk Array XP Microsoft Exchange Server ZDB-to-tape backup specification file named "Exchange_example" for a Microsoft Exchange Server running on “computer1.company.com” with the backup system “computer2.company.com”, overwriting the possible already existent backup specification files with the same name to back up First Mailbox Store, Public Folder Store, part of First Storage group and Test Mailbox Store, part of Test Storage Group, run:

```
omnicreatedl -ex2000 -datalist Exchange_example -storage_group
"First Storage Group" -store "First Mailbox Store" "Public Folder
Store" -storage group "Test Storage Group" -store "Test Mailbox
Store" -split_ mirror -sse -local computer1.company.com
computer2.company.com -force
```

The omnicreatedl command creates the HP StorageWorks Disk Array XP Microsoft Exchange Server ZDB-to-tape backup specification file (datalist) “Exchange_example” and two additional HP StorageWorks Disk Array XP Microsoft Exchange Server ZDB transaction logs backup specification files if circular logging option is disabled for a particular storage group: “First Storage

Group (LOGS) computer1.company.com” for First Storage Group log files backup and “Test Storage Group (LOGS) computer1.company.com” for Test Storage Group log files backup. Any possible already existent backup specification file with the same name is overwritten.

4. To create an HP StorageWorks Virtual Array Microsoft Exchange Server ZDB-to-tape backup specification file (datalist) "Exchange_example", to back up Site Replication Service on "dev1" device, run:

   ```
   omnicreatedl -ex2000 -datalist Exchange_example -device dev1
   -annotation SRS -snapshot -va computer1.company.com
   computer2.company.com
   ```

   The omnicreatedl command creates a HP StorageWorks Virtual Array Microsoft Exchange Server ZDB-to-tape backup specification file named “Exchange_example” and HP StorageWorks Virtual Array Microsoft Exchange Server ZDB transaction logs backup specification file in case it does not already exist: “SRS (LOGS) computer1.company.com” for Site Replication Service log files backup if the circular logging is disabled.

5. To create an HP StorageWorks Enterprise Virtual Array Microsoft Exchange Server ZDB-to-tape backup specification file (datalist) "Exchange_example", to back up Site Replication Service on "dev1" device, using the vsnap type of snapshot and the strict snapshot policy, run:

   ```
   omnicreatedl -ex2000 -datalist Exchange_example -device dev1
   -annotation SRS -snapshot -eva computer1.company.com
   computer2.company.com -snapshot_type vsnap -snapshot_policy strict
   ```

   if you have the HP StorageWorks EVA Agent (legacy) installed, or:

   ```
   omnicreatedl -ex2000 -datalist Exchange_example -device dev1
   -annotation SRS -snapshot -smis computer1.company.com
   computer2.company.com -snapshot_type vsnap -snapshot_policy strict
   ```

   if you have the HP StorageWorks EVA SMI-S Agent installed.

   The omnicreatedl command creates an HP StorageWorks Enterprise Virtual Array Microsoft Exchange Server ZDB-to-tape backup specification file named “Exchange_example” and an HP StorageWorks Enterprise Virtual Array Microsoft Exchange Server ZDB transaction logs backup specification file in case it does not already exist: “SRS (LOGS) computer1.company.com” for Site Replication Service log files backup if the circular logging is disabled. When the omnib command or Data Protector GUI is used to start the created backup specification, Data Protector tries to create the vsnap type of snapshots if they cannot be created, the session aborts.

6. To create an HP StorageWorks Enterprise Virtual Array Microsoft Exchange Server ZDB-to-disk backup specification file (datalist) “Exchange_example”, to back up Site Replication Service on the backup device “dev1”, using the replica set with “5” replicas, run:

   ```
   omnicreatedl -ex2000 -datalist Exchange_example -device dev1
   -snapshot -eva computer1.company.com computer2.company.com
   -instant_recovery -snapshots 5 -annotation SRS
   ```

   if you have the HP StorageWorks EVA Agent (legacy) installed, or:

   ```
   omnicreatedl -ex2000 -datalist Exchange_example -device dev1
   -snapshot -smis computer1.company.com computer2.company.com
   -instant_recovery -snapshots 5 -annotation SRS
   ```

   if you have the HP StorageWorks EVA SMI-S Agent installed.

   In case it does not already exist, `omnicreatedl` creates an EVA Microsoft Exchange Server transaction logs backup specification file “SRS (LOGS) computer1.company.com” for Site Replication Service log files backup (the circular logging must be disabled). When the `omnib` command or Data Protector GUI is used to start the created backup specification, you must choose the ZDB-to-disk session. Data Protector tries to create the snapclone type of snapshots; if they cannot be created, the session aborts. After the backup session, the created replica is retained on a disk array and can be used for instant recovery.

7. To create an HP StorageWorks Enterprise Virtual Array Microsoft Exchange Server ZDB-to-disk+tape backup specification file (datalist) “Exchange_example”, to back up Site Replication Service on the backup device “dev1”, using the replica set with “3” replicas and to delay the backup to tape for the maximum of “50” minutes, run:

   ```
   omnicreatedl -ex2000 -datalist Exchange_example -device dev1
   -snapshot -eva computer1.company.com computer2.company.com
   -instant_recovery -snapshots 3 -wait_clonecopy 50 -annotation SRS
   ```

   if you have the HP StorageWorks EVA Agent (legacy) installed, or:

   ```
   omnicreatedl -ex2000 -datalist Exchange_example -device dev1
   -snapshot -smis computer1.company.com computer2.company.com
   -instant_recovery -snapshots 3 -wait_clonecopy 50 -annotation SRS
   ```

   if you have the HP StorageWorks EVA SMI-S Agent installed.

   In case it does not already exist, `omnicreatedl` creates an EVA Microsoft Exchange Server transaction logs backup specification file “SRS (LOGS) computer1.company.com” for Site Replication Service log files backup (the circular logging must be disabled). When the `omnib` command or Data Protector GUI is used to start the created backup specification, you must choose the ZDB-to-disk+tape session. Data Protector tries to create the snapclone type of snapshots; if they cannot be created, the session aborts. The backup to tape will start after the snapclones are fully created or after 50 minutes. After the backup session, the created replica is retained on a disk array and can be used for instant recovery.

SEE ALSO

omnib(1), util_cmd(1M)

omnidb (1)

NAME

omnidb – queries the Data Protector internal database (IDB)

SYNOPSIS

```
omnidb -help | -version

omnidb -session [-datalist Datalist] [-type {restore | backup}]
    [-user User] {[[-since Date] [-until Date]] | [-last Number] |
    [-latest] | [-wo start duration ]} [-detail]

omnidb -filesearch [-n N] Client Directory FileName

omnidb Object [-session SessionID] [-copyid CopyID] -listdir
    Directory

omnidb -list_folders -session SessionID [-mailbox MailboxName...]

omnidb -rpt [SessionID | -latest] -detail

omnidb -rpt [-wo start duration]

omnidb -session SessionID [{-report Report | -detail | -strip |
    -purge | -change_protection Protection |
    -change_catprotection Protection | -media [-detail ]}]

omnidb -object

omnidb [-noexpand] {-filesystem | -winfs | -vbfs}
    Client:MountPoint Label [-file FileName] [-detail]

omnidb Object {[-since Date] [-until Date] | [-last
    NumberOfDays] | [-latest] } [-change_protection Protection]
    [-change_catprotection Protection]

omnidb Object {[-since Date] [-until Date] | -last NumberOfDays}
    [-latest ] [-detail]

omnidb Object -strip NumberOfDays

omnidb -strip

omnidb -change_protection Protection

omnidb -change_catprotection Protection

omnidb [-noexpand] {-filesystem | -winfs | -netware | -vbfs}
```

```
    Client:MountPoint Label -file FileName [-detail]

omnidb Object -session SessionID [-copyid CopyID] [-report
    [Report] | -catalog | -change_protection Protection |
    -change_catprotection  Protection | -strip]

omnidb Object -session SessionID [-copyid CopyID] -media
    [-detail]

omnidb Object -session SessionID -listcopies [-detail]

Object

 [-noexpand]

 { -filesystem [Client:MountPoint Label] |

 -winfs [Client:MountPoint Label] |

 -netware [Client:MountPoint Label] |

 -vbfs [Client:MountPoint Label] |

 -omnidb [Client:MountPoint Label] |

 -rawdisk [Client Label] |

 -sap [Client:Set] |

 -sapdb [Client:Set] |

 -stream [Client:Set] |

 -oracle8 [Client:Set] |

 -mssql [Client:Set] |

 -msexchange [Client:Set] |

 -mbx [Client:Set] |

 -informix [Client:Set] |

 -sybase [Client:Set] |

 -lotus [Client:Set] |

 -vss [Client:Set] |

 -db2 [Client:Set] }

 Protection  =  none | days n | weeks n | until Date | permanent

 Report  =  warning | minor | major | critical
```

```
Date  = [YY]YY/MM/DD (1969 < [YY]YY < 2038)
```

DESCRIPTION

The `omnidb` command is used to query the IDB Log database.

This command can be used to:

- list sessions and their summary reports
- list backed up objects and their details (for example: client name, mountpoint, label, object type, object status, backup type, etc.), message logs, and media location
- search for all occurrences of a pathname pattern

The `omnidb` command performs basic IDB queries.

OPTIONS

`-version` Displays the version of the `omnidb` command

`-help` Displays the usage synopsis for the `omnidb` command

`-datalist` *`IntegrationName BackupSpecificationName`* Lists the sessions resulting from backup specification backups created using this *`BackupSpecificationName`* .

NOTE: For non-filesystem backup specification (MSExchange, MSSQL, Informix Server, etc...)

`IntegrationName` must be specified in front of *`BackupSpecificationName`*. Both must be in double quotes.

`-type restore | backup` Lists either backup or restore objects. By default all objects are listed.

`-user` *`User`* Lists only the sessions belonging to the specified user.

`-since` *`Date`* Lists sessions since the given *`Date`*.

`-until` *`Date`* Lists sessions until the given *`Date`*.

`-last` *`n`* Lists sessions that occurred within the last *`n`* days.

`-latest` Lists the last active Data Protector session.

`-wo` *`start duration`* Lists the sessions that started within a specified timeframe. *`Start`* defines the start of the timeframe. *`Duration`* is the duration of the timeframe in seconds.

`-detail` Displays detailed information about the selected query.

`-session` *`SessionID`* Displays session information. If no *`SessionID`* is specified, all sessions are shown. The report shows for each session: the ID, type, status and user (UNIX login, UNIX group and client). If a *`sessionID`* is specified, then objects that are backed up within this session are shown. This information includes: client name, mountpoint, label, object type and object status.

If the `-detail` option is specified, more information is shown, such as the backup type (`full`, `incr`, ...), protection, and so on. For integration objects, also the backup ID is shown.

`-copyid` *`CopyID`* If several copies of the same object exist in one session as a result of the object copy or object mirror operation, this option is obligatory. It selects a specific copy.

`-filesearch [-n` *`N`*`]` *`Client Directory FileName`* Lists all the backed up files and directories that match the selection criteria set by the *`Client Directory FileName`* parameters. Wildcards can be used. The list can be limited to a certain number of displayed objects by setting the `-n` suboption, where *`N`* is the number of objects to be displayed. The following information is displayed about each object: object type, object name, object description, pathname.

`-listdir` *`Directory`* Lists all the backed up objects in the specified directory.

`-list_folders` MS Exchange single mailbox restore only: displays a list of all single mailbox folders (including their subfolders) backed up within a particular session.

`-mailbox` *`MailboxName`* MS Exchange single mailbox restore only: displays mailbox folders for a particular mailbox only. If the option is not specified, folders of all backed up mailboxes are listed.

`-listcopies` Lists details on all existing object or mirror copies of the specified object for the specified session. The SessionID, the CopyID, the time and the status of object copy or mirror sessions for the specified object are listed.

`-rpt` *`SessionID`* Displays session information in a form specially suited for further use of awk, grep or perl. Records are separated with blank lines and line feed is the field separator. If no *`SessionID`* is specified, all backup sessions are shown. Each record contains the following fields: the ID, backup specification name, status, start time in format *HH:MM* and duration in hours as a floating point number. For the example of how this option can be used see the script `omninotify.pl`.

`-report` *`Report`* Lists all messages (of specified report level and higher) which were generated by the specified session. Messages are classified (in ascending order) as: `warning`, `minor`, `major` and `critical`. For example, if `major` is selected, only `major` and `critical` messages are reported. By default, all messages are reported.

`-object` Displays information on all data objects. The report shows the client name, mountpoint, label, and object type.

`-filesystem [`*`Client:MountPoint Label`*`]` Displays information on all filesystem objects (displays the *`Client:MountPoint Label`* string for every filesystem object in the IDB). If a *`Client:MountPoint Label`* string is specified, the backup sessions containing the object specified by this string are listed. For each backup session, the report shows: the SessionID, start time, session duration, object status, size of object and the number of errors for the session.

`-winfs [`*`Client:MountPoint Label`*`]` Displays information on all Windows filesystem objects (displays the *`Client:MountPoint Label`* string for every Windows filesystem object in the IDB). If a *`Client:MountPoint Label`* string is specified, the backup sessions containing the object specified by this string are listed. For each backup session, the report shows: the SessionID, start time, session duration, object status, size of object and the number of errors for the session.

`-vbfs [`*`Client:MountPoint Label`*`]` Displays information on vbfs (very big file system) objects (displays the *`Client:MountPoint Label`* string for every vbfs object in the IDB). This option is used for OmniStorage data objects. If a *`Client:MountPoint Label`* string is specified, the backup sessions containing the object specified by this string are listed. For each backup session, the report shows: the SessionID, start time, session duration, object status, size of object and the number of errors for the session.

`-omnidb [`*`Client:MountPoint Label`*`]` Displays information on IDB object (displays the *`Client:MountPoint Label`* string for every IDB object in the IDB). If a *`Client:MountPoint Label`* string is specified, the backup sessions containing the object specified by this string are listed. For each backup session, the report shows: the SessionID, start time, session duration, object status, size of object and the number of errors for the session.

`-rawdisk [`*`Client Label`*`]` Displays information on disk image objects (displays the *`Client Label`* string for every rawdisk object in the IDB). If a *`Client Label`* string is specified, the backup sessions containing the object specified by this string are listed. For each backup session, the report shows: the SessionID, start time, session duration, object status, size of object and the number of errors for the session.

`-sap [`*`Client:Set`*`]` Displays information on SAP R/3 data objects (displays the *`Client:Set`* string for every SAP R/3 object in the IDB). If an ObjectName is specified, the backup sessions containing the object specified by this string are listed. For each backup session, the report shows: the SessionID, start time, session duration, object status, size of object and the number of errors for the session.

`-sapdb [`*`Client:Set`*`]` Displays information on SAP DB data objects (displays the *`Client:Set`* string for every SAP DB object in the IDB). If a *`Client:Set`* string is specified, the backup sessions containing the object specified by this string are listed. For each backup session, the report shows: the SessionID, start time, session duration, object status, size of object and the number of errors for the session.

`-stream [`*`Client:Set`*`]` Displays information on stream objects (displays the *`Client:Set`* string for every stream object in the IDB). If a *`Client:Set`* string is specified, the backup sessions containing the object specified by this string are listed. For each backup session, the report shows: the SessionID, start time, session duration, object status, size of object and the number of errors for the session.

`-oracle8 [`*`Client:Set`*`]` Displays information on Oracle objects (displays the *`Client:Set`* string for every Oracle object in the IDB). If a *`Client:Set`* string is specified, the backup sessions containing the object specified by this string are listed. For each backup session, the report shows: the status, size of object and the number of errors for the session.

`-mssql [`*`Client:Set`*`]` Displays information on MS SQL objects (displays the *`Client:Set`* string for every MS SQL object in the IDB). If a *`Client:Set`* string is specified, the backup sessions containing the object specified by this string are listed. For each backup session, the report shows: the SessionID, start time, session duration, object status, size of object and the number of errors for the session.

`-msexchange [`*`Client:Set`*`]` Displays information on MS Exchange objects (displays the *`Client:Set`* string for every MS Exchange object in the IDB). If a *`Client:Set`* string is specified, the backup sessions containing the object specified by this string are listed. For each backup session, the report shows: the SessionID, start time, session duration, object status, size of object and the number of errors for the session.

`-mbx [`*`Client:Set`*`]` Displays information on MS Exchange objects - single mailboxes (displays the *`Client:Set`* string for every MS Exchange object - single mailboxes in the IDB). If a *`Client:Set`* string is specified, the backup sessions containing the object specified by this string are listed. For each backup session, the report shows: the SessionID, start time, session duration, object status, size of object and the number of errors for the session.

`-informix [`*`Client:Set`*`]` Displays information on Informix Server objects (displays the *`Client:Set`* string for every Informix Server object in the IDB). If an *`Client:Set`* string is specified, the backup sessions containing the object specified by this string are listed. For each backup session, the report shows: the SessionID, start time, session duration, object status, size of object and the number of errors for the session.

`-sybase [`*`Client:Set`*`]` Displays information on Sybase objects (displays the *`Client:Set`* string for every Sybase object in the IDB). If a *`Client:Set`* string is specified, the backup sessions containing the object specified by this string are listed. For each backup session, the report shows: the SessionID, start time, session duration, object status, size of object and the number of errors for the session.

`-lotus [`*`Client:Set`*`]` Displays information on Lotus Notes/Domino objects (displays the *`Client:Set`* string for every Lotus Notes/Domino object in the IDB). If a *`Client:Set`* string is specified, the backup sessions containing the object specified by this string are listed. For each backup session, the report shows: the SessionID, start time, session duration, object status, size of object and the number of errors for the session.

`-vss [`*`Client:Set`*`]` Displays information on MS Copy objects (displays the *`Client:Set`* string for every MS Copy object in the IDB). If a *`Client:Set`* string is specified, the backup sessions containing the object specified by this string are listed. For each backup session, the report shows: the SessionID, start time, session duration, object status, size of object and the number of errors for the session.

`-db2 [`*`Client:Set`*`]` Displays information on IBM DB2 UDB objects (displays the *`Client:Set`* string for every IBM DB2 UDB object in the IDB). If a *`Client:Set`* string is specified, the backup sessions containing the object specified by this string are listed. For each backup session, the report shows: the SessionID, start time, session duration, object status, size of object and the number of errors for the session.

`-noexpand` Do not expand client names. Use this option if object was backed up with different client name resolution as when using this option.

`-strip` This option works in three different ways. If sessionID is specified it strips the detail catalogs for all objects of session with specified SessionID. If both SessionID and ObjectName are specified it strips the detail catalog of the object identified by ObjectName for the session with specified SessionID. If no option is specified, it strips catalogs on all data objects that are no longer protected.

`-strip` *`NumberOfDays`* This option can be used with ObjectName to strip the detail catalogs for all versions of specified object that are older than *`NumberDays`* days.

`-file` *`FileName`* Displays information on all sessions which contain the filesystem or the vbfs (very big file system - OmniStorage data object) with specified file *`FileName`*.

`-media` Shows list of the media used in the backup session. If object is also specified then it only shows list of media containing that object.

`-user_location` This option changes the output of media related reports to print out user defined location instead of physical location used by default.

`-change_protection` *`Protection`* Changes the current protection of the object versions identified by ObjectName and/or SessionID to the new protection defined as Protection. If it is specified without any other option then it changes protection for all Failed/Aborted objects. Protection can be none, permanent, until a specific date, or for a time interval. When the protection is until a specified date or for a time interval, you must specify the value. The Date form is [YY]YY/MM/DD. In the first case the value is the date until which the data is protected. In the second case the time interval is the number of days (after today) during which the data can not be overwritten.

`-change_catprotection` *`Protection`* Changes the current protection of the catalog retention time. *`Protection`* can be `none`, `same_as_data_protection`, until a specific date, or for a time interval. `same_as_data_protection` means that catalog will stay until data is overwritten/exported. When the protection is until a specified date or for a time interval, you must specify the value. The Date form is [YY]YY/MM/DD. In the first case the value is the date until which the data is protected. In the second case the time interval is the number of days (after today) during which the data can not be overwritten.

`-catalog` Displays the detail catalog of a specified object - session combination. Use an object option (for example `-filesystem`) to specify the object and use the `-session` (and *`sessionID`*) to specify the session.

`-purge` This option removes the session from the session list. All objects within the session become unprotected. It is still possible to make a restore from this session.

NOTES

With clustered objects, the *`Client`* argument must be specified as the virtual hostname.

EXAMPLES

The following examples illustrate how the `omnidb` command works.

1. To see details for the backup sessions started by user root in last three days:

   ```
   omnidb -session -user root -last 3 -type backup -detail
   ```

2. To see critical errors for the session with the sessionID "1994/07/14-17":

   ```
   omnidb -session 1994/07/14-17 -report critical
   ```

3. To see all objects of the type filesystem:

```
omnidb -filesystem
```

4. To see details for the filesystem "hpuljum.hermes.com:/ Label44" in the latest session:

```
omnidb -filesystem hpuljum.hermes.com:/ Label44 -latest -detail
```

5. To see catalog for the filesystem "bob:/" in the session "1994/07/14-6", run:

```
omnidb -filesystem bob:/ -session 1994/07/14-6 -catalog
```

6. To see details for the MSExchange backup specification “TEST” backup sessions run:

```
omnidb -session -datalist "MSExchange TEST" -details
```

7. To list all MS Exchange mailbox folders in the mailbox "User 2", backed up in the session "2004/03/16-5", run:

```
omnidb -mbx -list_folders -session 2004/03/15-6 -mailbox "User 2"
```

8. To see information on Lotus Notes/Domino objects:

```
omnidb -lotus
```

9. To see information on the SAP DB object “machine.company.com:/instance1/Config/1”:

```
omnidb -sapdb machine.company.com:/instance1/Config/1
```

10. To see detailed information on media used for the Windows filesystem object “system.company.com:/C” with the label “DTS_T” in the session “1994/07/14-17”, with CopyID “1280”:

```
omnidb -winfs system.company.com:/C "DTS_T" -session 1994/07/14-17
-copyid 1280 -media -detail
```

11. To see detailed information on all existing object or mirror copies of the Windows filesystem object “system.company.com:/D” with the label “D1” with the sessionID “2004/05/01-12”:

```
omnidb -winfs system.company.com:/D "D1" -session 2004/05/01-12
-listcopies -detail
```

SEE ALSO

omnidbcheck(1M), omnidbinit(1M), omnidbrestore(1M), omnidbupgrade(1M), omnidbutil(1M), omnidbxp(1), omnidbva(1), omnidbeva(1), omnidbsmis(1)

omnidbeva (1)

NAME

omnidbeva – sets, updates, deletes, and lists the login information for HP StorageWorks Enterprise Virtual Array (EVA) Command View EVA (CV EVA) v3.1 or lower, updates the information on EVA hardware configuration, sets the disk group pairs configuration file, queries the ZDB database (EVADB); synchronizes the EVADB with the state of the EVA storage system; deletes the source volumes in the selected replicas or selected replica set together with their entries in the EVADB.

SYNOPSIS

```
omnidbeva -version | -help

omnidbeva -empasswd {-add EM_hostname [-port  port] | -update
    EM_hostname [-port  port] | -remove EM_hostname | -list}

omnidbeva -hwrescan [EVA_name]

omnidbeva -dgrules {-put filename | -get filename | -check
    EVA_name DG_name | -init}

omnidbeva -list {-session [-ir ] | -datalist | -snapshot [-ir ] |
    -purge}

omnidbeva -show {-session sessionID | -datalist DatalistName |
    -snapshot VirtualDiskID EVA_ID}

omnidbeva -sync [-preview]

omnidbeva -delete {-session sessionID [-preview ] | -datalist
    DatalistName [-preview ]}
```

DESCRIPTION

The following tasks can be performed using the `omnidbeva` command:

SETTING, UPDATING, DELETING, AND LISTING THE LOGIN INFORMATION FOR COMMAND VIEW EVA

The `omnidbeva` command can be used to set, update, delete, and list the login information for HP StorageWorks Command View EVA (CV EVA) stored in the EVADB. The systems with the CV EVA installed are usually referred to as Management Appliance systems.

The omnidbeva options to be used for manipulating the login information for CV EVA are: -empasswd, -list, -add, -port, -remove, -update.

UPDATING THE INFORMATION ON EVA HARDWARE CONFIGURATION

The omnidbeva command can be used to trigger rescan of the EVA hardware configuration. Data Protector needs this information for EVA snapshot backup sessions. Normally, this information is automatically scanned after providing the login information for a specific Management Appliance. Using the omnidbeva command you must manually scan the hardware configuration only when you change Management Appliance for an EVA storage system or when you change at least one of the HSV EVA controllers.

The omnidbeva option to be used for scanning the EVA hardware configuration is: -hwrescan [EVA_name].

SETTING THE DISK GROUP PAIRS CONFIGURATION FILE

The omnidbeva command can be used to manipulate the EVA disk group pairs configuration file. The disk group pairs configuration file enables defining the allocation of snapclones. It is possible to allocate snapclones to a different disk group than the one used for the source volumes (original virtual disks). Thus, you can influence the application performance, since different physical disks are used for the read and write operations on the source volumes and the replica. By defining the disk group pairs configuration file, you can also allocate a replica to low-performance disks. Without setting the disk group pairs configuration file, snapclones are created in the same disk group as used for the source volumes.

The EVA disk group pairs configuration file is located on the Cell Manager as follows:

/var/opt/omni/server/db40/evadb/dgrules (HP-UX or Solaris Cell Manager) or *<Data_Protector_home>*\db40\evadb\dgrules (Windows Cell Manager).

The omnidbeva options to be used for manipulating the EVA disk group pairs configuration file are: -dgrules, -put, -get, -check, -init.

QUERYING THE INFORMATION ON BACKUP OBJECTS

The omnidbeva command can be used to query the EVADB for information about the Data Protector EVA replica set and about the source and target volumes created during the Data Protector EVA backup sessions.

Using the omnidbeva command to query the EVADB, it is possible to:

1. Get a list of all EVA ZDB sessions in the replica sets.

2. Get detailed information on a specific backup session (replica).

3. Get a list of all backup specifications with non-empty replica set.

4. Get detailed information on a specific backup specification (replica set).

5. Get a list of all target volumes in the replica sets.

6. Get detailed information on target volumes in the replica sets.

7. Get a list of source and target volumes that should be deleted (that have the purge flag).

When using the `omnidbeva` command for querying the EVADB only information on the backup sessions that have the backup option `Keep the replica after the backup` selected in the backup specification is displayed. ZDB-to-tape sessions without this backup option selected are deleted from the EVADB after each such backup.

The `omnidbeva` options to be used for querying the EVADB are: `-list`, `-session`, `-datalist`, `-ir`, `-snapshot`, `-purge`, `-show`.

SYNCHRONIZING THE EVADB

The `omnidbeva` command can be used to run the sync operation on the backup system. With the sync operation, the EVA Agent (legacy) checks if any target volume logged in the EVADB is missing on the EVA storage system. If a target volume is missing, all target volumes that were created in the backup session that created the missing target volume are deleted from the EVADB and EVA storage system. The check is performed for all replica sets.

The `omnidbeva` option to be used for synchronizing the EVADB is: `-sync`.

DELETING TARGET VOLUMES THAT ARE PART OF THE SELECTED REPLICAS OR SELECTED REPLICA SET TOGETHER WITH THEIR ENTRIES IN THE EVADB

The `omnidbeva` command can be used to delete all target volumes that are part of the selected replica (identified by the backup session ID) or in the selected replica set (identified by the backup specification name), together with their entries in the EVADB.

When the target volumes in the selected replica or in the selected replica set are deleted, it is not possible to perform instant recovery from the deleted replica or replica set.

The `omnidbeva` options to be used for deleting target volumes and their entries in the EVADB are: `-delete`, `-session`, `-datalist`.

OPTIONS

`-version` Displays the version of the `omnidbeva` command.

`-help` Displays the usage synopsis of the `omnidbeva` command.

`-port` *`port`* Specifies the port number on which CV EVA installed and running on the Management Appliance system with the hostname *`EM_hostname`* expects requests. The default port number for CV EVA v3.0 and CV EVA v3.1 is `12301`. If the port number is different, specify this option.

`-empasswd -add` *`EM_hostname`* Stores the login information for CV EVA installed and running on the Management Appliance system with the hostname *`EM_hostname`* in the EVADB. The command issues a prompt to enter the username and password for the relevant CV EVA. The password information is not shown on the screen while typing.

`-empasswd -update` *`EM_hostname`* Updates the login information for CV EVA installed and running on the Management Appliance system with the hostname *`EM_hostname`* in the EVADB. The command issues a prompt to enter the username and password for the relevant CV EVA. The password information is not shown on the screen while typing.

Together with the `-port` option, the command is also used to update the port number information (for example, after upgrading the Virtual Controller Software from 2.x to 3.0.).

`-empasswd -remove` *`EM_hostname`* Removes a Management Appliance system (together with the login and port number information) with the hostname *`EM_hostname`* from a list of systems on which CV EVAs are running from the EVADB.

`-empasswd -list` Lists all Management Appliance systems and the port numbers on which CV EVAs listen to requests.

`-hwrescan` Updates the EVADB with information on the EVA hardware configuration.

`-dgrules -put` *`filename`* Sets the EVA disk group pairs configuration file by reading the contents of the input file, checking its syntax, and uploading the file to the EVADB. If the syntax of the file is inaccurate, the file is not uploaded.

`-dgrules -get` *`filename`* Prepares the EVA disk group pairs configuration file for editing by reading the contents of the file from the EVADB and saving it under uploading *`filename`*.

`-dgrules -check` *`EVA_name DG_name`* Provides the information on the disk group that is in pair with the disk group identified by the *`EVA_name`* and *`DG_name`*. The command returns information on the `1st disk group name`, `2nd disk group name`, and `EVA name`. If there is no rule for the specified disk group, the first and the second disk groups are the same.

`-dgrules -init` Creates a template disk group pairs configuration file or overwrites an old one with the template disk group pairs configuration file. Note that only the configured disk group pair rules are overwritten.

`-ir` If `-ir` option is used with the `omnidbeva -list -session` command, only those sessions, that are marked for instant recovery (ZDB to disk and ZDB to disk+tape), are listed. If `-ir` option is used with `omnidbeva -list -snapshot` command, only those target volumes, that are marked for instant recovery, are listed.

`-list -session` Lists all replicas in the replica sets, together with the backup session IDs and the backup specification names.

`-list -datalist` Lists all backup specifications with non-empty replica set.

`-list -snapshot` Lists all target volumes created by Data Protector and stored in the replica sets. The target volumes displayed are identified by the virtual disk ID and the EVA storage system ID.

`-list -purge` Lists all virtual disks (source or target volumes) that are marked for purging in the EVADB.

`-show -session` *`sessionID`* Lists expanded details of a particular replica (identified by the backup session ID). The output of the command is information on all target volumes that were created in the specified backup session. The following is displayed:

- Name, ID, and WWN of the target volume that was created in the backup session

- Name and ID of the EVA storage system on which the target volume was created

- Type of the target volume created

- ID of the source volume used in the backup session

- The backup session ID

- Time stamp of the target volume created

- IR flag

- Name of the backup specification used in the backup session

- Names of the application and backup systems involved in the backup session

`-show -datalist` *`DatalistName`* Lists all replicas that are part of the replica set, which is identified by the backup specification name. Replicas displayed are identified by the backup session ID.

`-show -snapshot` *`VirtualDiskID EVA_ID`* Lists detailed information on a specific target volume. The following is displayed:

- Name, ID, and WWN of the target volume

- Name and ID of the EVA storage system on which the target volume resides

- Type of the target volume

- ID of its source volume

- ID of the backup session that created the target volume

- Time stamp of the target volume

- IR flag

- Name of the backup specification that created the target volume

- Names of the application and backup systems involved in the target volume creation.

`-sync` Runs the EVADB synchronizing operation. The sync operation runs the EVA Agent (legacy) on the backup system. The agent then checks if any target volume logged in the EVADB is missing on the EVA storage system. If a target volume is missing, the whole backup session that created this target volume is deleted from the EVADB. The check is performed for all replica sets.

The sync operation for one replica set is also performed automatically at the beginning of each backup session.

`-preview` Only lists the target volumes or replicas that would be deleted if `-sync` or `-delete` options were used.

`-delete -session` *`sessionID`* Deletes the target volumes created in a specific backup session (a replica), identified by the session ID, together with their entries in the EVADB.

`-delete -datalist` *`DatalistName`* Deletes all target volumes created with the backup sessions based on a specific backup specification (replica set) together with their entries in the EVADB.

EXAMPLES

1. To list all the Management Appliance systems and the port numbers in the EVADB, execute the following command:

   ```
   omnidbeva -empasswd -list
   ```

2. To remove the Management Appliance system (together with the login and port number information) with the hostname “system1” from a list of systems on which CV EVAs run from the EVADB, execute the following command:

   ```
   omnidbeva -empasswd -remove system1
   ```

3. To store the login information for CV EVA that is installed and running on the Management Appliance system with the hostname “system1” in the EVADB, execute the following command:

   ```
   omnidbeva -empasswd -add system1
   ```

   When the prompt is issued, enter the login information.

4. To update the login information for CV EVA that is installed and running on the Management Appliance system with the hostname “system2” in the EVADB, execute the following command:

   ```
   omnidbeva -empasswd -update system2
   ```

   When the prompt is issued, enter the new login information.

5. To update the port number for CV EVA v3.0 or CV v3.1 that is installed and running on the Management Appliance system with the hostname “system2” in the EVADB, execute the following command:

   ```
   omnidbeva -empasswd -update system2 -port 12301
   ```

   When the prompt is issued, enter the login information.

6. To create and set the disk group pairs configuration file or to edit it, execute the following commands on the application or backup system:

   a) To create a template disk group pairs configuration file or to overwrite an old one with the template, execute the following command:

   ```
   omnidbeva -dgrules -init
   ```

   b) To get the file for editing and to save it as “c:\tmp\dgrules.txt”, execute the following command:

   ```
   omnidbeva -dgrules -get c:\tmp\dgrules.txt
   ```

   The command reads the disk group pairs configuration file from the EVADB and saves it in the “c:\tmp” directory on local system with the name “dgrules.txt”.

   c) Edit the “dgrules.txt” file residing in the “c:\tmp” directory and save it. Note that the order of defining disk group names is ignored. This means that if an source volume is found in “disk group 1”, then its snapclone will be created in “disk group 2”, or vice versa. Note that a certain disk group can be a member of only one disk group pair.

   d) Execute the following command:

   ```
   omnidbeva -dgrules -put c:\tmp\dgrules.txt
   ```

   The command reads the contents of the file, checks its syntax, and copies the file to its location on the Cell Manager.

7. To get the information on the disk group that is the pair of a disk group named `original_disk_group` configured on EVA storage system named `EVA1`, execute the following command:

   ```
   omnidbeva -dgrules -check EVA1 original_disk_group
   ```

8. To list all target volumes created by Data Protector and stored in the EVADB, that are marked for instant recovery, execute the following command:

   ```
   omnidbeva -list -snapshot -ir
   ```

9. To find out the name, ID, WWN, type, and time stamp of the target volume(s) created in the backup session with the session ID “2003/06/13-3”, execute the following command:

   ```
   omnidbeva -show -session 2003/06/13-3
   ```

SEE ALSO

omnidbrestore(4), omnidb(1M), omnidbinit(1M), omnidbutil(1M), omnidbcheck(1M), omnidbupgrade(1M), omnidbva(1), omnidbsmis(1), omnidbxp(1)

omnidbsmis (1)

NAME

omnidbsmis – executes administrative tasks required for managing HP StorageWorks Enterprise Virtual Array used with Command View (CV) EVA v3.2.

SYNOPSIS

```
omnidbsmis -version | -help

omnidbsmis -ompasswd -add hostname [-port port] [-namespace
    namespace] [-user username] [-passwd password]

omnidbsmis -ompasswd -remove hostname [-port port] [-namespace
    namespace] [-user username] [-passwd password]

omnidbsmis -ompasswd [-list [hostname]]

omnidbsmis -dgrules {-init | -put filename | -get filename |
    -check EVA_name DG_name}

omnidbsmis [-show] {-session sessionID | -datalist DatalistName}

omnidbsmis [-list] {-session [-ir ] | -datalist}

omnidbsmis -list -purge

omnidbsmis -purge [-force] [-host hostname]

omnidbsmis -delete {-session SessionID | -datalist DatalistName}
    [-preview] [-force] [-host hostname]

omnidbsmis -sync [-preview] [-force] [-host hostname]
```

DESCRIPTION

The following tasks can be performed using the `omnidbsmis` command:

SETTING, DELETING, AND LISTING THE LOGIN INFORMATION FOR COMMAND VIEW EVA v3.2.

The `omnidbsmis` command can be used to set, delete, and list the login information for CV EVA v3.2. The systems with CV EVA installed are referred to as Management Appliance systems.

The `omnidbsmis` options used for manipulating the login information for CV EVA are: `-ompasswd`, `-add`, `-remove`, `-list`, `-port`, `-namespace`, `-user`, `-passwd`.

SETTING THE DISK GROUP PAIRS CONFIGURATION FILE

The omnidbsmis command can be used to manipulate the EVA disk group pairs configuration file.

By default, the snapclones are created in the same disk group as the source volumes. However, you can define the allocation of snapclones and allocate them to the disk group other than the one used for the source volumes.

The omnidbsmis options used for manipulating the EVA disk group pairs configuration file are: -dgrules, -init, -put, -get, -check.

QUERYING THE INFORMATION ON THE BACKUP OBJECTS

The omnidbsmis command can be used to query the SMISDB for the information on the backup sessions (the product of every successful backup session is a replica) and the backup specifications (a group of replicas, created using the same backup specification, is a replica set).

Using the omnidbsmis command to query the SMISDB, you can:

1. Get detailed information on a specific backup session (replica).

2. Get detailed information on all backup sessions created using a specified backup specification (replica set).

3. Get a list of all backup sessions created using the same backup specification.

4. Get a list of all backup sessions marked for instant recovery.

5. Get a list of all backup specifications that are part of a replica set with existing members.

6. Get a list of source volumes and replicas to be deleted (having the purge flag).

Note that session details are only displayed for the sessions that have the Keep the replica after the backup option selected in the backup specification. ZDB-to-tape sessions without this option set are deleted from the SMISDB after each such backup.

The omnidbsmis options used for querying the SMISDB are: -list, -session, -ir, -datalist, -purge, -show.

PURGING THE SMISDB

The omnidbsmis command can be used to run the purge operation that checks the SMISDB for the virtual disks with the purge flag and, in case of finding such disks, attempts to delete these objects.

The omnidbsmis options used for purging replicas and their entries in the SMISDB are: -purge, -force, -host.

DELETING REPLICAS OR REPLICA SETS TOGETHER WITH THEIR ENTRIES IN THE SMISDB

The `omnidbsmis` command can be used to delete a selected replica (identified by a backup session ID) or a selected replica set (identified by a backup specification name).

The `omnidbsmis` options to be used for deleting replicas and replica sets together with their entries in the SMISDB are: `-delete`, `-session`, `-datalist`, `-preview`, `-force`, `-host`.

SYNCHRONIZING THE SMISDB

The `omnidbsmis` command can be used to run the sync operation on the backup system. During this operation, the EVA SMI-S Agent checks if any target volumes in a replica logged in the SMISDB are missing on the EVA storage system. If a target volume is missing, the whole backup session that created the replica with a missing target volume is deleted from the SMISDB. The check is performed for all replica sets.

The `omnidbsmis` options used for synchronizing the SMISDB are: `-sync`, `-preview`, `-force`, `-host`.

OPTIONS

`-version` Displays the version of the `omnidbsmis` command.

`-help` Displays the usage synopsis of the `omnidbsmis` command.

`-ompasswd -add` *`hostname`* Stores the login information for the system with the name *hostname*, on which CV EVA v3.2 is installed, in the SMISDB.

The `-port` *`port`* option specifies the port number, on which CV EVA listens to requests. The default port number for CV EVA v3.2 is `5988`. If your CV EVA is configured to use a different port number, set it using this option.

The `-namespace` *`namespace`* parameter is used to specify the namespace that contains the CIMOM configuration for the EVA. The default namespace is `root/eva`.

The `-user` *`username`* option sets the user of CV EVA v3.2. The default user is `administrator`.

The `-passwd` *`password`* option sets the password that will be used for logging in to CV EVA v3.2. If you omit this parameter, the command will ask for a password interactively.

`-ompasswd -remove` *`hostname`* Removes a system with CV EVA installed (together with the login and port number information), which has a name *`hostname`*, from the SMISDB.

Used together with the `-port` *`port`* option, the command will only remove the entries for the specified port. Use this option if you have more than one port configured on the same system, and you want to delete only one port from the configuration.

If the `-namespace` *`namespace`* option is specified, the command will only remove the entries for the specified namespace. Use this option if you have more than one namespace configured on the same system, and you want to delete only one namespace from the configuration.

If the `-user` *`username`* option is specified, the command will only remove the entries for the specified user. Use this option if you have more than one user configured on the same system, and you want to delete only one user from the configuration.

`-ompasswd -list` *`hostname`* Lists all systems that have CV EVA installed, together with the port numbers, on which CV EVAs listen to requests. The *`hostname`* value is optional: if you enter a name of the host, only the SMI-S CIMOMs, configured for a specified host, will be displayed.

Note that you will get the same output if you run the `omnidbsmis -ompasswd` command without the `-list` parameter.

`-dgrules -init` Creates a template disk group pairs configuration file or overwrites an old one with the new template. Note that only configured disk group pair rules are overwritten.

`-dgrules -put` *`filename`* Sets the EVA disk group pairs configuration file by reading the contents of the input file, checking its syntax, and uploading the file to the SMISDB. If the syntax of the file is inaccurate, the file is not uploaded.

`-dgrules -get` *`filename`* Prepares the EVA disk group pairs configuration file for editing by reading the contents of the file from the SMISDB and saving it under *`filename`*.

`-dgrules -check` *`EVA_name DG_name`* Provides the information on the disk group that is in pair with the disk group identified by the *`EVA_name`* and *`DG_name`*. The command returns the information on the `1st` `disk group name`, `2nd disk group name`, and `EVA name`. If there is no rule for the specified disk group, the first and the second disk groups are the same.

`-show -session` *`SessionID`* Lists expanded details of a session (identified by the backup session ID). The output of the command is the information on all target volumes created in a specified backup session. The following is displayed:

- Name, ID, and WWN of the target volume created in the backup session

- Name and ID of the EVA storage system on which the target volume was created

- Type of the target volume created

- ID of the source volume used in the backup session

- The backup session ID

- Time stamp of the target volume

- IR flag

- Name of the backup specification used in the backup session

- Names of the application and backup systems involved in the backup session

Note that you will get the same output if you run the `omnidbsmis -session` *`SessionID`* command without the `-show` parameter.

`-show -datalist` *`DatalistName`* Lists all replicas that are part of the replica set, which is identified by the backup specification name. Replicas displayed are identified by their backup session IDs. Note that you will get the same output if you run the `omnidbsmis -datalist` *`DatalistName`* command without the `-show` parameter.

`-list -session` Lists all replicas that are part of a replica set, together with the backup session IDs and the backup specification names. Note that you will get the same output if you run the `omnidbsmis -session` command without the `-list` parameter.

If used together with the `-ir` option, the command lists only the sessions marked for instant recovery (ZDB-to-disk and ZDB-to-disk+tape sessions).

`-list -datalist` Lists all backup specifications that are part of a replica set with existing members. Note that you will get the same output if you run the `omnidbsmis -datalist` command without the `-list` parameter.

`-list -purge` Lists all virtual disks (source or target volumes) that are marked for purging in the SMISDB.

`-purge` Runs the SMISDB purge operation that attempts to remove the virtual disks (source or target volumes) that could not be deleted although they should be. These elements are marked for purging, and the information about them is stored in the SMISDB.

Used together with the `-force` option, the command forces removal of the elements marked for deletion even if they are presented to the hosts.

If the `-host` *`hostname`* option is specified, you can choose another location to start the SMISDB purge operation. Use this option if the systems, involved in a backup session, are no longer available, thus allowing redirection to another systems that have the SMI-S Agent installed.

`-delete -session` *`sessionID`* Deletes the replica (all target volumes in the replica) created in a specified backup session, identified by the session ID, together with its entry in the SMISDB.

`-delete -datalist` *`DatalistName`* Deletes all replicas in a replica set (i.e., all replicas created within the backup sessions based on a specified backup specification) together with their entries in the SMISDB.

`-delete -preview` If used with the `-preview` option, the command does not delete anything, but lists replicas or replica sets that will be deleted if `-delete -session` *`sessionID`* or `-delete -datalist` *`DatalistName`* commands are run.

`-delete -force` Forces deletion of the replicas even if they are presented to hosts.

`-delete -host` *`hostname`* Sets another location to start the deletion. Use this option if the systems, involved in a backup session, are no longer available, thus allowing redirection to another systems that have the SMI-S Agent installed.

`-sync` Synchronizes the persistent data in the SMISDB with the EVA storage system. If any target volume in a replica is missing on the EVA storage system, the whole backup session that created this replica is deleted from the database. Use this option if you deleted some replicas directly through the Command View EVA.

If run with the `-preview` option, the command does not delete anything, but lists the target volumes that would be deleted if `omnidbsmis -sync` is run.

Used together with the `-force` option, the command forces removal of the target volumes marked for deletion even if they are presented to hosts.

With the `-host` *`hostname`* option specified, you can choose another system to start the SMISDB synchronizing operation. Use this option if the systems, involved in a backup session, are no longer available, thus allowing redirection to another systems that have the SMI-S Agent installed.

EXAMPLES

1. To list all Management Appliance systems together with the port numbers, on which CV EVAs listen to requests, execute the following command:

   ```
   omnidbsmis -ompasswd -list
   ```

2. To remove a Management Appliance system with the hostname “system1”, together with its login and port number information, from the SMISDB, run the following command:

   ```
   omnidbsmis -ompasswd -remove system1
   ```

3. To store the login information for CV EVA v3.2, installed and running on the Management Appliance system with the hostname “system1”, in the SMISDB, execute the following command:

   ```
   omnidbsmis -ompasswd -add system1
   ```

   You can also set optional parameters, such as the port number, namespace, and username. If you omit these parameters, the command will take the default values.

4. To create and set the disk group pairs configuration file or edit it, carry out the steps provided below on the application or backup system:

   a) To create a template disk group pairs configuration file or overwrite an old one with the template, execute the following command:

   ```
   omnidbsmis -dgrules -init
   ```

   b) To get the file for editing and to save it as "c:\tmp\dgrules.txt", execute the following command:

   ```
   omnidbsmis -dgrules -get c:\tmp\dgrules.txt
   ```

   The command reads the disk group pairs configuration file from the SMISDB and saves it in the "c:\tmp" directory on a local system under "dgrules.txt".

   c) Edit the "dgrules.txt" file residing in the "c:\tmp" directory and save it. Note that the order of defining disk group names is ignored. This means that if a source volume is found in "disk group 1", its snapclone will be created in "disk group 2", and vice versa. Note that a certain disk group can be a member of only one disk group pair.

   d) Execute the following command:

   ```
   omnidbsmis -dgrules -put c:\tmp\dgrules.txt
   ```

   The command reads the contents of the file, checks its syntax, and copies the file to its location on the Cell Manager.

5. To list all existing backup sessions, together with their session IDs and backup specification names, execute the following command:

   ```
   omnidbsmis -session
   ```

6. To find out the name, ID, WWW, type, and time stamp of the target volumes created in the backup session with the session ID "2004/06/13-3", run the following command:

   ```
   omnidbsmis -session 2004/06/13-3
   ```

7. To run the SMISDB synchronizing operation and force removal of the target volumes marked for deletion, execute the following command:

   ```
   omnidbsmis -sync -force
   ```

SEE ALSO

omnidbrestore(4), omnidb(1M), omnidbinit(1M), omnidbutil(1M), omnidbcheck(1M), omnidbupgrade(1M), omnidbeva(1), omnidbva(1), omnidbxp(1)

omnidbva (1)

NAME

omnidbva – The command is used to query the ZDB database (VADB) and to administer the LUN security, instant recovery and password information

SYNOPSIS

```
omnidbva -version | -help

omnidbva -exclude  {-put filename | -get filename | -check VA_wwn
    LUN | -init | -delete}

omnidbva -init [-force]

omnidbva -delete session_id [-force]

omnidbva {-session [session_id] | -lun [LUN]}

omnidbva -vapasswd VA_wwn password

omnidbva -sampasswd SAM_server_ID user password

omnidbva -dbcheck [-force]
```

DESCRIPTION

The `omnidbva` command can be used to manipulate the VADB, which is situated on the Data Protector Cell Manager. VADB contains configuration information required for LUN security, instant recovery sessions and security passwords that affect the operation of the HP StorageWorks Virtual Array integration.

LUN security information is contained in the VA LUN exclude file. The VA LUN exclude file is used to specify LUNs, within HP StorageWorks Virtual Arrays connected to a Data Protector cell, that are to be excluded from Data Protector operations. The file contains a list of such LUNs grouped according to the world-wide-name of the Virtual Arrays concerned. LUNs specified in the VA LUN exclude file in this way cannot be accessed by Data Protector for backup and restore purposes and can safely be reserved for other purposes.

The VA LUN exclude file is located on the Cell Manager as follows:

`/var/opt/omni/server/db40/vadb/exclude` (UNIX Cell Manager) or *<Data_Protector_home>*`\db40\vadb\exclude` (Windows Cell Manager).

Other LUN and session information is contained in the VADB, which is also within VADB.

Password information is held within a series of encrypted password files within VADB.

OPTIONS

`-version` Displays the version of the `omnidbva` command.

`-help` Displays the usage synopsis of the `omnidbva` command.

`-exclude -put` *`filename`* Sets the list of excluded LUNs by reading the contents of the *`filename`*, checking its syntax and if the syntax is correct, copying the *`filename`* to its position on the Cell Manager. If the syntax is not correct, the file is not copied. This command must be used for updating the file. The VA LUN exclude file must not be edited directly.

`-exclude -get` *`filename`* Prepares the VA LUN exclude file for editing by reading the contents of the VA LUN exclude file on the Cell Manager and saving it under *`filename`*. To update the exclusion list, filename must be edited and then used to perform the update using the `-exclude -put` *`filename`* command.

`-exclude -check` *`VA_wwn LUN`* Checks whether the specified LUN, identified by its backup system virtual array world-wide-name (*`VA_wwn`*) and LUN number (*`LUN`*) is specified in the VA LUN exclude file on the Cell Manager and informs the user.

`-exclude -init` Overwrites the current VA LUN exclude file on the Cell Manager with the template VA LUN exclude file.

`-exclude -delete` Deletes the contents of the VA LUN exclude file on the Cell Manager.

`-init [-force]` Reinitializes (deletes all entries) from the VADB (but will not delete any allocated LUNs). Unless the `-force` option is given, the user will be asked for a confirmation.

`-delete` *`session_id`* `[-force]` Deletes a session from the VADB (both the disks and the database entries). If `-force` is specified, the command will try to delete the session without any consistency checks.

`-session [`*`session_id`*`]` Displays available ZDB-to-disk and ZDB-to-disk+tape sessions (instant recovery enabled) within the database. If a *`session_id`* is specified, expanded details of the session will be displayed. If required, a range of session IDs can be specified.

`-lun [`*`LUN`*`]` Displays LUN (disk) information about the available VADB objects. If a *`LUN`* is specified, detailed information for only that *`LUN`* is displayed.

`-vapasswd` *`VA_wwn password`* Sets the access password for the Virtual Array node identified by its virtual array world-wide-name (*`VA_wwn`*). If Secure Manager is set ON for the Virtual Array node concerned, the password for the Virtual Array node must be specified using this command.

`-sampasswd` *`SAM_server_ID user password`* Sets the HP Allocator access password. If a SAN network is being used, with HP Allocator installed, this password must be supplied using this command. Refer to the HP Allocator documentation for further information.

`-dbcheck [-force]` Performs a consistency check of the VADB, and offers to fix invalid entries if necessary. If `-force` is specified, the VADB will be checked/fixed without any interactive prompts.

NOTES

The command can only be used on HP-UX, Solaris and Windows clients. The information within the VADB must be edited using the supplied tools. It must not be edited directly.

EXAMPLES

1. To set or change the VA LUN exclude file:

   Use the following command on the application or backup system:

   ```
   omnidbva -exclude -get c:/tmp/filename.txt
   ```

   This command reads the VA LUN exclude file from the Cell Manager and saves it in the “c:\tmp\filename.txt” file.

   Edit the `c:\tmp\filename.txt` file and save it when you are done editing.

   Use the following command on the application or backup system:

   ```
   omnidbva -exclude -put c:\tmp\filename.txt
   ```

   This command reads the contents of the “c:\tmp\filename.txt”, checks its syntax and if the syntax is correct copies the file to its position on the Cell Manager.

2. To check whether the LUN number “123” in the virtual array with world_wide-name “Abcde12345” is specified in the VA LUN exclude file, execute the following command:

   ```
   omnidbva -exclude -check Abcde12345 123
   ```

3. To list all available ZDB-to-disk and ZDB-to-disk+tape sessions in the VADB, execute the following command:

   ```
   omnidbva -session
   ```

SEE ALSO

omnidbrestore(4), omnidb(1M), omnidbinit(1M), omnidbutil(1M), omnidbcheck(1M), omnidbupgrade(1M), omnidbxp(1), omnidbeva(1), omnidbsmis(1)

omnidbvss (1)

NAME

omnidbvss – queries VSS database; manages, browses, and lists the items of the VSS database.

SYNOPSIS

```
omnidbvss -help

omnidbvss -version

omnidbvss -perform {list | clear}  -table {metadata | clone}

omnidbvss -perform remove -table {metadata | clone} -item
    Database_Item

omnidbvss -perform get -table clone -item Database_Item

omnidbvss -perform get -table metadata -item Database_Item -out
    Pathname
```

DESCRIPTION

The omnidbvss command is used to query the VSS database.

This command can be used to:

- list sessions from the writer metadata document and backup component document from VSS sessions
- list replicas from the VSS sessions and their details
- view a document with non-VSS sessions IDs and their details
- remove specific items from the VSS database table
- clear the whole VSS database table

The omnidbvss command performs basic queries.

OPTIONS

-version	Displays the version of the omnidbvss command.
-help	Displays the usage synopsis of the omnidbvss command.

`-perform list -table {metadata |clone}` Queries the VSS database (metadata or clone table) and lists session IDs or storage IDs depending on the specified table.

`-perform clear -table {metadata |clone}` Clears the data in the metadata or replica table of the VSS database.

`-perform remove -table {metadata |clone} -item` *`Database_Item`*

Removes the specified item (session ID or replica) from the metadata or replica (clone) table of the VSS database.

`-perform get -table clone -item` *`Database_Item`* Lists detailed information on the specified replica (*`Database_Item`*) from the replica (clone) table. Use quotes ("") when specifying a *`Database_Item`*. The following is displayed: Storage ID, SnapshotSet ID, Snapshot ID, Device Path, Disk Signature, H/w Provider Name, SubSystem Name, SubSystem ID, Lun Name, Lun ID, Unmasking list, and MountPoint.

`-perform get -table metadata -item` *`Database_Item`* `-out` *`PathName`*

Creates a backup components document (`Backup Components Document.xml`) and writer metadata document (`<`*`writer name`*`>.xml`) with detailed information on the specified session from the metadata table. Use quotes ("") when specifying a *`Database_Item`* and a *`PathName`*.

`-out` *`PathName`* Specifies location for backup components document and writer metadata document. Use quotes ("") when specifying a *`PathName`*.

EXAMPLES

1. To list all replicas from the replica (clone) table of the VSS database, execute the following command:

   ```
   omnidbvss -perform:list -table:clone
   ```

2. To remove the session "2005/12/01-1" from the metadata table, execute the following command:

   ```
   omnidbvss -perform:remove -table:metadata -item:"2005/12/01-1"
   ```

3. To list detailed information from the replica (clone) table of the VSS database on database item "STORAGE#Volume#1&30a96598&0&SignatureB6893593Offset7E00Length3FBB8600#{53f5630d-b6bf-11d0-94f2-00a0c91efb8b}", execute the following command:

   ```
   omnidbvss -perform:get -table:clone
   -item:"STORAGE#Volume#1&30a96598&0&SignatureB6893593Offset7E00Lengt
   h3FBB8600#{53f5630d-b6bf-11d0-94f2-00a0c91efb8b}"
   ```

SEE ALSO

omnidbrestore(4), omnidb(1M), omnidbinit(1M), omnidbutil(1M), omnidbcheck(1M), omnidbupgrade(1M)

omnidbxp (1)

NAME

omnidbxp – Queries the ZDB database (XPDB), manipulates the XP LDEV exclude file and configures the HP StorageWorks Disk Array XP command devices usage.

SYNOPSIS

```
omnidbxp -version | -help

omnidbxp -exclude {-put filename | -get filename | -check SEQ LDEV
    | -init | -delete}

omnidbxp [-ir] -session {-list | -show sessionID}

omnidbxp [-ir] -ldev {-list | -show SEQ LDEV}

omnidbxp -cm {-add serial {CU:LDEV | LDEV} hostname [instance] |
    -update serial {CU:LDEV | LDEV} hostname [instance]}

omnidbxp -cm -remove {all | serial [{CU:LDEV | LDEV} [hostname]]}

omnidbxp -cm -list
```

DESCRIPTION

The following tasks can be performed using the `omnidbxp` command:

QUERYING THE INFORMATION ON BACKUP OBJECTS AND MANIPULATING THE LDEV EXCLUDE FILE

The `omnidbxp` command can be used to query the information stored in the ZDB database (XPDB), which stores the information about the LDEVs pairs and their mirror configurations during the Data Protector HP StorageWorks Disk Array XP backup and restore sessions. The XPDB is a set of ASCII files stored on the Cell Manager in the `/var/opt/omni/server/db40/xpdb` (UNIX Cell Manager) or in the `<Data_Protector_home>\db40\xpdb` (Windows Cell Manager) directory. The XPDB contains data about the split P-VOL - S-VOL pairs used with the Data Protector HP StorageWorks Disk Array XP integration in a set of XPDB records. The XPDB is written to whenever a pair is split using the Data Protector HP StorageWorks Disk Array XP integration. A pair is deleted from the XPDB whenever such a pair is resynchronized using the Data Protector HP StorageWorks Disk Array XP integration.

The `omnidbxp` command can also be used to manipulate the XP LDEV exclude file. The XP LDEV exclude file enables disabling of using certain LDEVs on the backup system (S-VOL LDEVs) by Data Protector. Thus, it is possible to reserve certain LDEVs for other purposes than Data Protector backup and restore. The disabled LDEVs are, if used in a Data Protector session, ignored by Data Protector and such a session fails with critical error. The list of disabled LDEVs is kept in the XP LDEV exclude file on the Cell Manager: `/var/opt/omni/server/db40/xpdb/exclude/XPexclude` (UNIX Cell Manager) or in `<`*`Data_Protector_home`*`>\db40\exclude\XPexclude` (Windows Cell Manager).

The `omnidbxp` options to be used for querying the XPDB and manipulating the XP LDEV exclude file are: `-exclude`, `-put`, `-get`, `-check`, `-init`, `-delete`, `-session`, `-list`, `-show`, `-ir`, `-ldev`, `-show`.

HP StorageWorks DISK ARRAY XP COMMAND DEVICE HANDLING

The HP StorageWorks Disk Array XP command devices are needed by any process that needs access to the HP StorageWorks Disk Array XP. The information about HP StorageWorks Disk Array XP command devices is kept in the XPDB for the purpose of eliminating duplicate instance usage and over-allocation. Data Protector provides the following mechanism to prevent duplicate instance usage and over allocation:

1. Whenever a session is started, Data Protector queries the XPDB for a list of command devices. If there is none in the XPDB (default behavior when the first session is started), Data Protector identifies command devices and generates a list of command devices in the XPDB as connected to every application and backup system in the cell.

2. Every command device is assigned an instance number (starting from 301) and the system (hostname) having access to it. If a command device can be accessed from more than one system, the hostname identifier enables Data Protector to be aware of the fact that the command device is already meant to be used by some other system; next available instance number is assigned to such a command device - hostname combination

3. When the list is created, every HP StorageWorks Disk Array XP attached to application and backup systems has a list of its command devices and systems having access to them (together with an instance number) assigned.

4. Whenever during a session an application or backup system needs access to an HP StorageWorks Disk Array XP, it uses the first assigned command device with the instance number from the list. If the command device fails, the next command device from the list assigned to a particular system is used. If all of them fail, the session fails. The successful command device is used by a particular system until the end of the session and the list of command devices is used for all consecutive sessions.

Using the `omnidbxp` command, it is possible to:

1. Specify a particular command device (identified by the HP StorageWorks Disk Array XP serial number and LDEV number) to be used by a particular system. Optionally, an instance number can be assigned too. If the instance number is not specified, Data Protector assigns the lowest not yet assigned instance number. The entire information is written in the XPDB.
2. List all command devices in the XPDB.
3. Remove a specific or all command devices from the XPDB or update the information about a specific command device in the XPDB.

The `omnidbxp` options to be used for command device handling begin with the `-cm` option and are: `-add`, `-update`, `-remove` and `-list`.

OPTIONS

`-version` Displays the version of the `omnidbxp` command

`-help` Displays the usage synopsis of the `omnidbxp` command.

`-exclude -put` *`filename`* Sets the list of excluded LDEVs by reading the contents of the *`filename`*, checking its syntax and if the syntax is correct, copying the file to its position on the Cell Manager. If the syntax is not correct, the file is not copied.

`-exclude -get` *`filename`* Prepares the XP LDEV exclude file for editing by reading the contents of the XP LDEV exclude file on the Cell Manager and saving it under the *`filename`*.

`-exclude -check` *`SEQ LDEV`* Checks whether the specified LDEV, identified by its backup system disk array serial/sequence number (*`SEQ`*) and LDEV number (*`LDEV`*) is specified in the XP LDEV exclude file on the Cell Manager. The LDEV number must be specified as the CU#:LDEV in decimal format. If the queried LDEV is specified in the XP LDEV exclude file, the command returns: `YES!` If the queried LDEV is not specified in the XP LDEV exclude file, the command returns: `NO!`

`-exclude -init` Overwrites the current XP LDEV exclude file on the Cell Manager with the template XP LDEV exclude file.

`-exclude -delete` Deletes the contents of the XP LDEV exclude file on the Cell Manager.

`-ir` Specifies that the current `omnidbxp` command is executed only for the LDEVs pairs marked for the instant recovery in the XPDB. If this option is not specified, the current command is executed for all LDEVs pairs in the XPDB.

`-session -list` Lists all available sessions in the XPDB.

`-session -show` *`sessionID`* Lists all backup system S-VOL LDEVs that were involved in the session with the *`sessionID`*.

`-ldev -list` Lists all S-VOL LDEVs in the XPDB together with their corresponding backup sessionID.

`-ldev -show` *`SEQ LDEV`* Lists all available XPDB information about the specified S-VOL *`LDEV`*. The following information is listed: sessionID, CRC, IRflag, primary XP #, primary LDEV #, primary port #, mirror type, MU#, date and time, application system and backup system hostnames.

`-cm -add` *`serial`* {*`CU:LDEV`* | *`LDEV`*} *`hostname`* [*`instance`*] Adds the command device identified by the serial number of the HP StorageWorks Disk Array XP (*`serial`*) and serial number of command device in the hexadecimal or decimal format (*`CU:LDEV`* or *`LDEV`*) to the XPDB, and assigns it the hostname of the system accessing it (*`hostname`*) and optionally the instance number (*`instance`*). If the instance number is not specified, Data Protector assigns the lowest not yet assigned instance number.

The instance number must be any number in the range between 301 and 399.

The command does not check whether the specified command device or system exist, it only checks if the optional instance number specified is within the correct range and if the command device together with the instance number is not already assigned to be used by some other system. If checks fail, the command fails with an appropriate error message.

`-cm -update` *`serial`* {*`CU:LDEV`* | *`LDEV`*} *`hostname`* [*`instance`*] Updates the XPDB information about the command device identified by the serial number of the HP StorageWorks Disk Array XP (*`serial`*), serial number of command device in the hexadecimal or decimal format (*`CU:LDEV`* or *`LDEV`*) and the specified hostname of the system accessing it (*`hostname`*), by assigning the newly specified instance number (*`instance`*) to the HP StorageWorks Disk Array XP serial number, serial number of command device and hostname combination. If the instance number is not specified, Data Protector assigns the lowest not yet assigned instance number.

The instance number must be any number in the range between 301 and 399.

The command does not check whether the specified command device or system exist, it only checks if the optional instance number specified is within the correct range and if the command device together with the instance number is not already assigned to be used by some other system. If the checks fail, the command fails with an error message.

`-cm -remove all` Removes the information about all command devices from the XPDB.

`-cm -remove` *`serial`* `[{`*`CU:LDEV`* `|` *`LDEV`*`}` `[`*`hostname`*`]]` If only the *`serial`* argument is specified, the command removes the information about command devices within a specific HP StorageWorks Disk Array XP identified by the serial number of this HP StorageWorks Disk Array XP (*`serial`*) from the XPDB.

If the *`CU:LDEV`* | *`LDEV`* and optionally *`hostname`* arguments are specified as well, the command removes the information about the command device identified by the serial number of the HP StorageWorks Disk Array XP (*`serial`*), serial number of command device in the hexadecimal or decimal format (*`CU:LDEV`* or *`LDEV`*) and optionally by the hostname of the system (*`hostname`*).

When removing the information about the command device without specifying the system (*`hostname`*), the command deletes all entries for the specified command device, regardless of the system(s) assigned to it.

`-cm -list` Lists all command devices in the XPDB.

NOTES

The command can only be used on HP-UX, Solaris and Windows clients.

EXAMPLES

1. To set or change the XP LDEV exclude file:

   a.) Use the following command on the application or backup system:

   ```
   omnidbxp -exclude -get c:\tmp\filename.txt
   ```

   This command reads the XP LDEV exclude file from the Cell Manager and saves it in the “c:\tmp\filename.txt” file.

   b.) Edit the `c:\tmp\filename.txt` file and save it when you are done editing.

   c.) Use the following command on the application or backup system:

   This command reads the contents of the “c:\tmp\filename.txt”, checks its syntax and if the syntax is correct, copies the file to its position on the Cell Manager.

   ```
   omnidbxp -exclude -put c:\tmp\filename.txt
   ```

2. To check whether the LDEV identified by the serial number “12345” and LDEV number “123” is specified in the XP LDEV exclude file, execute the following command:

   ```
   omnidbxp -exclude -check 12345 2864
   ```

3. To list all backup system LDEVs, regardless of they being marked for instant recovery or not, that were involved in the session with the sessionID “2001/09/18-22”, execute the following command:

   ```
   omnidbxp -session -show 2001/09/18-22
   ```

4. To list all command devices in the XPDB, execute the following command:

   ```
   omnidbxp -cm -list
   ```

5. To add the command device identified by the HP StorageWorks Disk Array XP serial number “00035371” and command device serial number “103” to the XPDB and assign it to be used on the “computer.company.com” system by the instance number “303”, execute the following command:

   ```
   omnidbxp -cm -add 00035371 103 computer.company.com 303
   ```

6. To remove the information about all command devices from the XPDB, execute the following command:

   ```
   omnidbxp -cm -remove all
   ```

SEE ALSO

omnidbrestore(1M), omnidb(1), omnidbinit(1M), omnidbutil(1M), omnidbcheck(1M), omnidbupgrade(1M), omnidbva(1), omnidbeva(1), omnidbsmis(1)

omnidownload (1)

NAME

omnidownload – downloads information about a backup device and a library from the Data Protector internal database (IDB)

SYNOPSIS

```
omnidownload -version | -help

omnidownload -list_devices [-detail]

omnidownload -dev_info

omnidownload -device BackupDevice [-file FileName]

omnidownload -list_libraries [-detail]

omnidownload -library Library [-file FileName]
```

DESCRIPTION

Allows the user to display information about backup devices or download the configuration of the specified backup device to an ASCII file. Used together with the `omniupload` utility, this command enables you to create and maintain backup devices using the Command-Line Interface.

OPTIONS

`-version` Displays the version of the `omnidownload` command.

`-help` Displays the usage synopsis for the `omnidownload` command.

`-device` *BackupDevice* Specifies the name of the backup device you want to download to an ASCII file.

`-library` *Library* Specifies the name of the library you want to download to an ASCII file.

`-file` *FileName* Specifies the name of the target ASCII file for the backup device. By default, the file is created in the local directory. If this option is omitted, the data is sent to the standard output.

`-list_devices` Displays information about the Data Protector backup devices. The report includes the following information for each device: device name, client, device type and pool.

-dev_info
Same as -list_devices option. Used only for compatibility with old Data Protector releases.

-list_libraries
Displays information about the Data Protector libraries. The report includes the following information for each device: library name, client and library type.

-detail
This option can be used in combination with the -list_devices and -list_libraries options to display more detailed information about the Data Protector backup devices or libraries.

EXAMPLES

This example downloads backup device "DAT1" into file "/tmp/DAT1":

```
omnidownload -device DAT1 -file /tmp/DAT1
```

SEE ALSO

omniamo(1), omnimcopy(1), omniminit(1), omnimlist(1), omnimm(1), omnimnt(1), omnimver(1), omniupload(1), sanconf(1M), uma(1M)

omniiso (1)

NAME

omniiso – a pre-exec script to prepare the disaster recovery ISO image file for disaster recovery and a standalone command to automate your backup and disaster recovery process.

SYNOPSIS

```
omniiso -version | -help

omniiso [-session SessionID] [-cd] [-isopath Path] [-srd Path]
    [-rset p1s_Path DR_Image_Path]
```

DESCRIPTION

The `omniiso` command can be used as a:

STANDALONE COMMAND

Although all functionality of the command is also available through the Disaster Recovery Wizard in the Data Protector GUI, it can also be used as a standalone command to automate your backup and disaster recovery process.

The command merges the:

- DR image (the data required for temporary DR OS installation and configuration that is created during a full client backup),
- SRD file (a file that contains all required backup and restore object information to perform the restore),
- P1S file (a file that contains information on how to format and partition all disks installed in the system)

with disaster recovery installation into a disaster recovery ISO image and saves it to disk (by default to `<Data_Protector_home>\tmp`). The ISO image can later be written to a tape/CD and used to perform disaster recovery.

Such ISO image can also be created using the OBDR Wizard in the Data Protector GUI instead of using this command (recommended).

PRE-EXEC SCRIPT

If the command is used as a pre-exec script in the OBDR Wizard in Data Protector GUI to prepare the disaster recovery ISO image, you do not have to specify any parameters, because they are obtained from the current environment.

OPTIONS

`-version` Displays the version of the `omniiso` command.

`-help` Displays the usage synopsis for the `omniiso` command.

`-session` *`SessionID`* Specifies the session ID of the session that serves as the basis for the object update. All objects, backed up in the specified session and included in the SRD file, will be included in the ISO image.

`-cd` If this option is specified, `omniiso` creates an ISO file that can be written to a CD. If this option is not specified, the command creates disaster recovery ISO file to be written on a tape.

`-isopath` *`Path`* Specifies the location where the disaster recovery ISO file is saved. If this option is not specified, the ISO file is saved into the default location (*`<Data_Protector_home>`*`\tmp`).

`-srd` *`Path`* Specifies the path to the SRD file. If the `-srd` option is not specified, the command creates a SRD file on the system, where `omniiso` is running and uses it to create the disaster recovery ISO image.

`-rset` *`p1s_Path DR_Image_Path`* Specifies the full path to the P1S file and DR Image. If this option is not specified, the command creates P1S file and DR Image on the system, where `omniiso` is running and uses them to create the disaster recovery ISO image.

NOTE

The command can only be used on Windows systems.

EXAMPLES

The following examples illustrate how the `omniiso` command works.

1. To create and save a disaster recovery ISO file for a CD in “c:\iso\dr\omnidr.iso”, containing objects backed up in the session with the session ID “2004/08/16-23”, using information stored in the SRD and P1S files stored in

"c:\iso\dr\srd\machine101.company.com" and "c:\iso\dr\p1s\machine.company.com", using DR Image stored in "c:\iso\dr\img\machine101.company.com.img", use:

```
omniiso -session 2004/08/16-23 -cd -iso c:\iso\dr\omnidr.iso -srd
c:\iso\dr\srd\machine101.company.com -rset
c:\iso\dr\p1s\machine101.company.com
c:\iso\dr\img\machine101.company.com.img
```

SEE ALSO

omniofflr(1M), omnidr(1M), omnisrdupdate(1M)

omnimcopy (1)

NAME

omnimcopy – produces a copy of a Data Protector medium.

SYNOPSIS

```
omnimcopy -help | -version

omnimcopy -copy BackupDevice [-slot Slot ...] -from BackupDevice
    [-src_slot Slot ...] [BasicOptions] [LabelOptions]

BasicOptions

 -pool PoolName

 -location Location

 -force

 -size SpecSize

 -eject

 -permanent | -until Date

 Date  = [YY]YY/MM/DD (1969 < [YY]YY < 2038)

LabelOptions

 -label UserLabel [-no_barcode_as_label] | -autolabel  |
    -[no_]barcode_as_label
```

DESCRIPTION

The `omnimcopy` copies a Data Protector medium. It reads data from the input medium and writes the data to the output medium. Note that the output medium is first initialized. During initialization, a medium is assigned a:

- Data Protector Medium Label: Depending on the selected options, the media labels can be user defined or generated automatically. By default, Data Protector automatically generates media labels from the media pool names, unless the `Use barcode as media label during initialization` option is selected in the library properties. This behavior can be changed during the initialization of media using `-barcode_as_label`, `-no_barcode_as_label` and `MediumLabel` options.
- Medium ID (system-assigned)

- Location

The physical devices used for the input and output must be the same device type and have the same block size.

This copy functionality allows the user to use multiple tapes in order to implement vaulting with Data Protector. This copy function is a separate function within Data Protector and cannot be dome automatically during back up. Main advantage of this implementation is that all devices can be used during backup (better performance).

The source and destination devices are backup devices which means they can be located everywhere in the Data Protector cell. During the copy the destination tape will be initialized before all data from the source tape is copied.

The writing destination tape will ignore the early end of tape mark and will write until the physical EOT is reached. If the space on the destination is not sufficient to keep the whole original tape the copy has to be restarted with a new tape.

After a copy operation both media are tracked in the media management database.

This enables also a listing of the copies for an original media as well as the listing of the original tape for a copy. If a mount request is issued during a restore session all tapes which contains the data will be listed (original and copies).

After the operation copy both tapes become nonappendable.

A copy of a copy is not possible.

If the original media get obsolete in the database, which means it is overwritten or it is exported from the cell, the first copy becomes automatically the original tape.

OPTIONS

`-help` Displays the usage synopsis for the `omnimcopy` command

`-version` Displays the version of the `omnimcopy` command

`-copy` *`BackupDevice`* `[-slot` *`Slot`*`...]` Specifies the output backup device - the device used to create a copy of the medium (target medium). You can specify only one slot.

`-from` *`BackupDevice`* `[-src_slot` *`Slot`*`...]` Specifies the input backup device - device which is used as a source. You can specify only one slot.

`-pool` *`PoolName`* Specify the poolname to which the copy of the medium is added. By default the medium is added to the source media poolname.

`-location` *`Location`* Specifies the location of the media, when you keep them out of the library. Used for the vaulting purposes.

`-force` Overwrites the data on the target medium even if this data is still protected by the Data Protector media management system. Note that this option must be used with an unprotected medium as well.

`-size` *`SpecSize`* This option specifies the size of the target medium.

`-eject` Ejects the target medium from the drive after the medium is copied.

`-permanent` This backup protection option provides permanent protection of backup media. This means that the data is permanently protected from being overwritten.

`-until` *`Date`* This backup protection option provides protection until a date of your choice. This means that the data on the medium cannot be overwritten until the specified date. Protection for the data stops at noon on the selected day.

`-label` *`UserLabel`* Manually specify the medium label for the copy of the medium. A description can have a maximum of 80 characters, including any keyboard character or space. If the `Use barcode as medium label on initialization` option is selected in the library properties, you have to specify also the `-no_barcode_as_label` option.

`-autolabel` If this option is specified, the medium is labeled automatically by the Data Protector media management system.

`-barcode_as_label` Data Protector uses barcode as a medium label during the initialization of the medium instead of generating media labels based on the media pools names. This option is supported only on library devices with enabled barcode support.

`-no_barcode_as_label` Data Protector does not use barcodes as a medium label during the initialization of the medium, but generates media labels based on the media pools names. This option can be used to override the `Use barcode as medium label on initialization` option (if it is selected) in the library properties in the Data Protector GUI.

SEE ALSO

omniamo(1), omnidownload(1), omniminit(1), omnimlist(1), omnimm(1), omnimnt(1), omnimver(1), omniupload(1), sanconf(1M), uma(1M)

omniminit (1)

NAME

omniminit – initializes a Data Protector medium

SYNOPSIS

```
omniminit -version | -help

omniminit -init BackupDevice [MediumLabel] [BasicOptions]
    [SlotOptions]

 [-no_barcode_as_label ]

omniminit -init BackupDevice [BasicOptions] [SlotOptions]
    [-barcode_as_label ]

omniminit -init_magazine BackupDevice [MagazineDescription]
    [BasicOptions]

omniminit -init_mag_medium BackupDevice MagazineDescription
    [BasicOptions] [SlotOptions]

omniminit -preerase BackupDevice [SlotOptions] [-eject]

BasicOptions

 -force

 -pool PoolName

 -size n

 -location OffLineLoc

 -wipe_on_init

 -eject

SlotOptions

 -slot SlotID [Side]
```

DESCRIPTION

The `omniminit` command initializes a backup medium. During initialization, a medium is assigned a:

- Data Protector Medium Label: Depending on the selected options, the media labels can be user defined or generated automatically. By default, Data Protector automatically generates media labels from the media pool names, unless the `Use barcode as media label during initialization` option is selected in the library properties. This behavior can be changed during the initialization of media using `-barcode_as_label`, `-no_barcode_as_label` and `MediumLabel` options.
- Medium ID (system-assigned)
- Location

This information is added to the Data Protector internal database (IDB) and the medium is added to a Data Protector media pool. Medium ID is its unique identifier. The Medium Label does not necessarily have to be unique, but it is recommended. The medium location is optional, and can be used to define an offline location for the medium.

OPTIONS

`-version` Displays the version of the `omniminit` command.

`-help` Displays the usage synopsis for the `omniminit` command.

`-init` *`BackupDevice`* `[`*`MediumLabel`*`]` Specifies two items: the name of the *`BackupDevice`* where the medium is mounted and the *`MediumLabel`* which is assigned to the medium by Data Protector after initialization. The *`MediumLabel`* can be up to 32 characters long. Any printable character, including spaces, can be used. The text must be enclosed in quotation marks.

`-init_magazine` *`BackupDevice`* `[`*`MagazineDescription`*`]` Specifies two items: the name of the *`BackupDevice`* where the magazine is mounted and the *`MagazineDescription`* (optional) which is assigned to the magazine. Note that the *`MagazineDescription`* must be unique for each magazine. The description is also used for assigning *`MediumLabel`* to each medium.

`-init_mag_medium` *`BackupDevice MagazineDescription`* Initializes single medium from magazine. *`BackupDevice`* specifies the device where the magazine is mounted. *`MagazineDescription`* must also be specified to identify the magazine. Note that single medium from the magazine can be initialized only if the magazine has been initialized before and therefore has a unique *MagazineDescription*.

`-preerase` *`BackupDevice`* Pre-erases the optical disk. Pre-erasing a medium enables backups which are twice as fast. This is because the pre-erase step is removed from the backup process. For best performance, optical disks should be pre-erased before each backup.

`-force` Overrides the initialization safety checks. By default, a medium containing protected data or being in a non-Data Protector format cannot be initialized.

`-pool` *`PoolName`* Specifies the name of the media pool to which this medium will be added. If no *PoolName* is specified, the medium is added to the default pool for the specified backup device.

`-slot` *`SlotID`* `[`*`Side`*`]` Specifies the *SlotID* of the exchanger backup device where the medium is mounted. This option is only valid for this backup device type, but must be given for MO devices. To specify the side of the platter in this slot, use the additional *Side* parameter. Values of *Side* are `A` or `B`.

`-size` *`n`* Specifies the medium capacity in MB. If not specified, Data Protector uses the standard capacity of the media class used with the backup device selected for initialization. The size is later used to calculate the free space remaining on the medium. (FreeSpace = Size - SpaceUsed)

`-location` *`OffSiteLoc`* Specifies the location of the medium. This information is useful if media is stored off-site. The location can have maximum 32 characters. Any printable character, including spaces, can be used. The text must be enclosed in quotation marks.

`-wipe_on_init` Wipes the data on medium after it has been initialized. This is done by overwriting the data on medium so it is impossible to restore the original data on medium after it has been wiped.

`-barcode_as_label` Data Protector uses barcode as a medium label during the initialization of the medium instead of generating media labels based on the media pools names. This option is supported only on library devices with enabled barcode support.

`-no_barcode_as_label` Data Protector does not use barcodes as a medium label during the initialization of the medium, but generates media labels based on the media pools names. This option can be used to override the `Use barcode as medium label on initialization` option (if it is selected) in the library properties in the Data Protector GUI.

EXAMPLES

The following examples illustrate how the `omniminit` command works.

1. To init slot "4" of backup device "ADIC" with medium label "Label4", in location "Backup Room" use:

```
omniminit -init ADIC Label4 -slot 4 -location "Backup Room"
```

2. To preerase slot "8" side "A" of MO tape library unit "MO_Changer":

```
omniminit -preerase MO_Changer -slot 8 A
```

SEE ALSO

omniamo(1), omnidownload(1), omnimcopy(1), omnimlist(1), omnimm(1), omnimnt(1), omnimver(1), omniupload(1), sanconf(1M), uma(1M)

omnimlist (1)

NAME

omnimlist – lists the contents of a Data Protector medium

SYNOPSIS

```
omnimlist -help  | -version

omnimlist -device BackupDevice [-slot SlotID [Side]] [-monitor]
    [-detail]

omnimlist -device BackupDevice [-slot SlotID [Side]] [-header]
    [-monitor]

omnimlist -device BackupDevice -session [-slot SlotID [Side]]
    [-monitor] [-detail]

omnimlist -device BackupDevice -session SessionID [-slot SlotID
    [Side]] [-monitor] [-detail]

omnimlist -device BackupDevice -catalog [-slot SlotID [Side]]
    [-monitor]

omnimlist -device BackupDevice -catalog DiskAgentID [-slot SlotID
    [Side]] [-monitor]
```

DESCRIPTION

The `omnimlist` command lists the contents of a Data Protector medium. The command scans the catalog (index) of the medium and shows all objects and sessions on the medium.

The command can also be used to display the Data Protector medium tape header. If used for such purpose, the command reads the first block of the tape and then displays the information.

OPTIONS

`-version` Displays the version of the `omnimlist` command.

`-help` Displays the usage synopsis for the `omnimlist` command.

`-device` *`BackupDevice`* Specifies the *`BackupDevice`* where the medium is mounted. If no other option is specified the command lists all sessions and all their objects.

`-slot` *`SlotID`* `[`*`Side`*`]` Specifies the *`SlotID`* of the library where the medium is mounted. This option is only valid for this backup device type, but must be given for MO devices. To specify the side of the platter in this slot, use the additional *`Side`* parameter. Values of *`Side`* are `A` or `B`.

`-session [`*`SessionID`*`]` Displays information about the sessions on the medium. If no *`SessionID`* is specified, all sessions are shown. This reports shows for each session: the *`SessionID`*, Session Type, Session Status. For the user who initiated the session it shows: the UNIX Login, UNIX Group, and ClientName. If a *`SessionID`* is specified, the objects for that session are shown. The session report shows for each object: the Client, Mountpoint, Object Label, Disk Agent ID and Object Status.

`-catalog [`*`DiskAgentID`*`]` Displays the detail catalog for single or multiple objects. The catalog shows file information for all the files included in the backup of the object in that session. The *`DiskAgentID`* is used to uniquely identify the backup object-session combination. If not specified all found objects are processed.

`-monitor` Displays information about the Medium (Pool, Medium ID, Medium Label, Location, and Initialization date/time), the Session (Session ID, Owner, Datalist used, and Start date/time), Objects (Type, Start date/time, Backup Mode), and Session (Client, Mountpoint, Object Label, Disk Agent ID, and Object Status).

`-header` The command first checks if the media header is in Data Protector format and if it is corrupted. If the media header is not in Data Protector format or if it is corrupted, an appropriate message is displayed. Otherwise the following information from the media header is displayed: medium ID, medium label, medium location, initialization date, last access date, last write date, last overwrite date, number of writes, number of overwrites, pool label, device information, device capacity, tape format version, medium ID from original tape (for replicated media only), medium data format type and medium data format subtype. For random access media, date and time information (last access date, last write date and last overwrite date) is updated every time the medium is accessed/ written/overwritten. For all other media, header information is not updated except when initializing the medium.

`-detail` Displays detailed information about the selected query.

NOTES

The command can be used locally on the Cell Manager. Using the `-header` option, the following limitations apply: The command displays the header information stored on the medium, ignoring possible updates in the Data Protector internal database (IDB).

EXAMPLES

The following examples show how the `omnimlist` command works.

1. To list sessions and corresponding disk agents from device "DAT2":

   ```
   omnimlist -device DAT2 -monitor
   ```

2. To list sessions on slot “43” side “B” of a tape library unit "MO_Changer":

   ```
   omnimlist -device MO_Changer -slot 43 B -session
   ```

3. To list all disk agents for the session "1994/07/13-23" on the device "Exa8500":

   ```
   omnimlist -device Exa8500 -session 1994/07/13-23
   ```

4. To list the catalog for the object-session combination with the DiskAgentID "774226832", from the medium located in slot "7" of device "Herstal2":

   ```
   omnimlist -device Herstal2 -slot 7 -catalog 774226832
   ```

5. To display media header for the medium in the backup device named “dev_1”:

   ```
   omnimlist -device dev_1 -header
   ```

SEE ALSO

omniamo(1), omnidownload(1), omnimcopy(1), omniminit(1), omnimm(1), omnimnt(1), omnimver(1), omniupload(1), sanconf(1M), uma(1M)

omnimm (1)

NAME

omnimm – manages Data Protector media

SYNOPSIS

```
omnimm -version | -help

omnimm -create_pool PoolName MediaType [Policy AgeLimit
    MaxOverWrites] [-[no_]alloc_uninit_first] [-[no_]free_pool
    [FreePoolName]] [-[no_]move_free_media]

omnimm -create_mag_pool PoolName MediaType [Policy AgeLimit
    MaxOverWrites]

omnimm -modify_pool PoolName NewPoolName [Policy AgeLimit
    MaxOverWrites] [-[no_]alloc_uninit_first] [-[no_]free_pool
    [FreePoolName]] [-[no_]move_free_media]

omnimm -modify_mag_pool PoolName MediaType [AgeLimit
    MaxOverWrites]

omnimm -create_free_pool PoolName MediaType [AgeLimit
    MaxOverWrites]

omnimm -modify_free_pool PoolName NewPoolName [AgeLimit
    MaxOverWrites]

omnimm -remove_pool PoolName

omnimm -remove_mag_pool PoolName

omnimm -show_pools [PoolName]

omnimm -move_medium Medium ToPoolName

omnimm -move_magazine MagazineDescription NewPoolName

omnimm -modify_medium Medium NewMediumLabel NewLocation

omnimm -modify_magazine MagazineDescription NewLocation
    [NewMagazineDescription]

omnimm -reset_poor_medium Medium

omnimm -list_pool [PoolName] [-detail]

omnimm -show_pool_alloc PoolName
omnimm -list_scratch_media PoolName [-detail]

omnimm -show_repository_alloc Library PoolName [-detail]

omnimm -list_media Medium [-detail]

omnimm -list_appendable_media PoolName

omnimm -list_copy Medium

omnimm -media_info Medium [-detail]

omnimm -list_magazines_of_pool PoolName [-detail]

omnimm -list_media_magazine  MagazineDescription [-detail]

omnimm -catalog Medium

omnimm -check_protection Medium

omnimm -recycle Medium

omnimm -recycle_magazine MagazineDescription

omnimm -export Medium

omnimm -export_magazine MagazineDescription

omnimm -import BackupDevice [-slot SlotID [Side]] [-no_log |
    -log_dirs | -log_file] [-pool PoolName] [-import_as_original]

omnimm -import_catalog BackupDevice [-slot SlotID [Side]]
    [-no_log | -log_dirs | -log_file]

omnimm -import_magazine BackupDevice [MagazineDescription] [-slot
    SlotID [Side]] [-no_log | -log_dirs | -log_file] [-pool
    PoolName] [-import_as_original]

omnimm -disable_lockname LockName

omnimm -enable_lockname LockName

omnimm -disable_device DeviceName

omnimm -enable_device DeviceName

omnimm -reload_serial_number DeviceName

omnimm -repository LibraryName

omnimm -repository_update DriveName [-slot SlotID [Side]]

omnimm -add_slots LibraryName {Slot ... | FromSlot-ToSlot ...}

omnimm -remove_slots LibraryName {Slot ... | FromSlot-ToSlot ...}
```

omnimm -silo_query *LibraryName* [-range *FromSlot-ToSlot*]

omnimm -silo_enter *LibraryName* -cap *CapID*

omnimm -silo_eject *LibraryNme* {*Volser* ... | *FromVolser-ToVolser* ...} -cap *CapID* [-location *Location*]

omnimm -enter *LibraryName* {*Slot* .. | *FromSlot-ToSlot* ...}

omnimm -eject *LibraryName* {*Slot*... | *FromSlot-ToSlot*... } [-location *Location*]

omnimm -group *PoolName MagazineDescription MediumLabel*...

omnimm -ungroup *MagazineDescription*

Policy =

Loose |

Strict |

App+Loose |

App+Strict |

AppIncr+Loose |

AppIncr+Strict

Basic Options =

-force |

-pool *PoolName* |

-size *n* |

-location *OffLineLoc* |

-eject

Medium =

Medium_Label |

Medium ID |

Barcode

DESCRIPTION

The main purpose of *media management* is to protect valuable user data.

To achieve this goal Data Protector provides the following functionality: protecting data from being overwritten, detecting and tracking bad or old media, utilizing and reporting space in auto changers, use of media within pools, drive cleaning, detecting standard tape and MO format. All this information is stored into the Data Protector internal database.

The `omnimm` command manages media pools, checks the protection of a medium, maintains and updates the contents of the repository in the library.

Protecting data is more than just stopping Data Protector from overwriting the tape. The detection of an old and poor media informs the administrator before data loss so that he can react before he needs to restore the data and tape will never be used for backups again. This means protection of data which are on Data Protector tapes and protection for data which is still on the system and needs to be backed up.

For the list of supported media classes, refer to the *HP OpenView Storage Data Protector Software Release Notes*.

Data Protector has the concept of *`media pools`* to manage large numbers of cartridges. Pools are logical collection of cartridges with same common media or data properties. One pool can only contain media of one type. Data Protector support several media *`pool policies`* : *`Loose`* (loose, non-appendable),When Data Protector prompts for a medium and loose policy is selected, any medium in the defined pool will be accepted.

`Strict` (strict, non-appendable); Data Protector decides which medium must be inserted for backup and only this medium will be accepted.

`App+Loose` (loose, appendable);

`App+Strict` (strict,appendable);

`AppIncr+Loose` (loose, appendable for incrementals);

`AppIncr+Strict` (strict, appendable for incrementals).

OPTIONS

`-version` Displays the version of the `omnimm` command

`-help` Displays the usage synopsis for the `omnimm` command

`-create_pool` *MediaType PoolName* [*Policy AgeLimit MaxOverWrites*]

Creates a new pool with *PoolName* for the medium of *MediaType* with the policy defined by *Policy*. For the list of supported media classes, refer to the *HP OpenView Storage Data Protector Software Release Notes*. Supported policies are: Loose, Strict, App+Loose, App+Strict, AppIncr+Loose and AppIncr+Strict. *AgeLimit* is set in months. The *MaxOverWrites* is the maximum number of times that the medium can be overwritten. The default is 250 overwrites.

`-create_mag_pool` *PoolName MediaType* [*Policy AgeLimit MaxOverWrites*] Creates pool *PoolName* with magazine support.

`-create_free_pool` *PoolName MediaType* [*AgeLimit MaxOverWrites*]

Creates a new free pool with *PoolName* for the medium of *MediaType* with the policy defined by *Policy*. The *MaxOverWrites* is the maximum number of times that the medium can be overwritten. The default is 250 overwrites.

`-modify_pool` *PoolName NewPoolName* [*Policy AgeLimit MaxOverWrites*] Renames the pool *PoolName* into *NewPoolName*. The *Policy*, *AgeLimit* and *MaxOverWrites* can also be changed. Supported policies are: Loose, Strict, App+Loose, App+Strict, AppIncr+Loose and AppIncr+Strict. *AgeLimit* is set in months. The *MaxOverWrites* is the maximum number of times that the medium can be overwritten. The default is 250 overwrites.

`-modify_mag_pool` *PoolName NewPoolName* [*AgeLimit MaxOverWrites*]

Renames the magazine pool *PoolName* into *NewPoolName*. The, *AgeLimit* and *MaxOverWrites* can also be changed. *AgeLimit* is set in months. The *MaxOverWrites* is the maximum number of times that the medium can be overwritten. The default is 250 overwrites.

`-modify_free_pool` *PoolName NewPoolName* [*AgeLimit MaxOverWrites*]

Renames the free pool *PoolName* into *NewPoolName*. The *AgeLimit* and *MaxOverWrites* can also be changed. *AgeLimit* is set in months. The *MaxOverWrites* is the maximum number of times that the medium can be overwritten. The default is 250 overwrites.

`-remove_pool` *PoolName* Removes the pool specified by *PoolName*.

`-remove_mag_pool` *PoolName* Removes the magazine pool specified by *PoolName*.

`-show_pools` *PoolName* Shows media from the specified *PoolName* pool.

`-move_medium` *`Medium ToPoolName`* Moves medium from the current pool to the pool specified by *`ToPoolName`*.

`-move_magazine` *`MagazineDescription NewPoolName`* Moves magazine *`MagazineDescription`* from the current pool to the pool specified by *`NewPoolName`*.

`-modify_medium` *`Medium NewMediumLabel NewLocation`* Modifies medium with the specified *`Medium`* . Note that you should always enter the medium label *`NewMediumLabel`* and location *`NewLocation`* in that sequence.

`-modify_magazine` *`MagDescription NewLocation`* `[`*`NewMagDescription`*`]`

Changes the location of the magazine *`MagDescription`* to *`NewLocation`*. If *`NewMagDescription`* is specified, it is assigned to the magazine as a new MagazineDescription. Note that each magazine must have a unique MagazineDescription.

`-reset_poor_medium` *`Medium`* Resets the media condition factors. Once the medium has expired (its maximum usage criteria), it is marked as `poor` and can no longer be used for backup. This option resets the medium quality status, thus enabling it to be used for backup. You have to be very cautious using this option, because a backup stored on an expired medium might not be recoverable.

`-list_pool [`*`PoolName`*`]` Displays all the media from pool *`PoolName`*. The report shows: medium label, status, location, appendability and protection. Appendability is shown under item FULL. If displayed status under FULL is "YES" then medium is unappendable, otherwise it is appendable. If *`PoolName`* is not specified, the command lists all the configured media pools. This report shows: pool name, status, media class, the number of media and free space in pool.

`-show_pool_alloc` *`PoolName`* Displays the sequence in which the media from the specified pool will be used for backup. The report shows: sequence, medium label and location.

`-list_scratch_media` *`PoolName`* Displays media from the specified pool which are not protected and can be used for backup. The report shows sequence, medium label and location.

`-show_repository_alloc` *`Library PoolName`* Displays the order in which the media in the repository of the specified *`Library`* will be used. The report shows: sequence, medium label, location and slot number.

-list_media *Medium* Displays all the objects, their type and their protection status for the medium you specified.

-list_appendable_media *PoolName* Displays all appendable media from the specified media pool.

-list_copy *Medium* List all copies of the given medium.

-media_info *Medium* Displays information on the given medium.

-list_magazines_of_pool *PoolName* Lists magazines of the pool *PoolName*.

-list_media_magazine *MagazineDescription* Lists all the media in specified magazine.

-catalog *Medium* Lists catalog for all object versions located on the specified medium. Only files located on this medium are displayed.

-[no_]alloc_uninit_first Option -noalloc_uninit_first sets/resets "Use uninitialized media first" pool policy. This option can be used with *Loose* policy only.

-[no_]free_pool [*FreePoolName*] If -free_pool is set, the pool is linked to the free pool specified with FreePoolName in order to share free media. Condition factors are inherited from the free pool. If the -no_free_pool is set, the pool is not linked. The default setting is -no_free_pool.

-[no_]move_free_media The -move_free_pool option can only be set if the -free_pool option was set. If -move_free_media is set, de-allocation of free media from a regular to a free pool is done automatically. If -no_move_free_media is set, there is no automatic de-allocation of free media. The default setting is -no_move_free_media.

-recycle *Medium* Resets the protection of data on medium. The present data can now be overwritten and medium can be used to store new data.

-recycle_magazine *MagazineDescription* Recycles all media of specified magazine.

-export *Medium* Purges from the database all data associated with the medium and the object versions it contains. This option is used when the medium will no longer be used for backup in this cell. A medium containing protected data cannot be exported.

-export_magazine *MagazineDescription* Exports all media of specified magazine.

`-import` *`BackupDevice`* Imports a medium from a different cell. The medium is put in the default pool of the specified backup device. Information about the new medium is added to the database. Slot side must be specified for MO devices.

`-import_catalog` *`BackupDevice`* Rereads the detail catalog from the specified device into the database, in case this information has been deleted. If the detail catalog for the specified medium already exists in the database, import will fail.

`-import_magazine` *`BackupDevice`* [*`MagazineDescription`*] Imports a magazine from a different cell. The magazine is put in the default pool of the specified backup device. Information about the new magazine and its media is added to the database.

`-no_log` Used with the `-import` option, this option omits the detail part of the catalog from the import.

`-log_dirs` Used with the `-import` option, this option imports only the detail part of the directories.

`-pool` *`PoolName`* Specifies the name of the pool.

`-disable_lockname` *`LockName`* Disables devices with the *`LockName`* for any operation. The *`LockName`* must be defined using the Data Protector GUI or using the `omniupload` command.

`-enable_lockname` *`LockName`* Enables devices with the *`LockName`*. The *`LockName`* must be defined using the Data Protector GUI or using the `omniupload` command.

`-disable_device` *`DeviceName`* Disables the device with the *`DeviceName`* for any operation. The *`DeviceName`* must be defined using the Data Protector GUI or using the `omniupload` command. If the device has a lockname defined, all devices with the same lockname are also disabled.

`-enable_device` *`DeviceName`* Enables the device with the *`DeviceName`*. The *`DeviceName`* must be defined using the Data Protector GUI or using the `omniupload` command. If the device has a lockname defined, all devices with the same lockname are also enabled.

`-reload_serial_number` *`DeviceName`* Reloads the device serial number and overwrites the serial number stored in the internal database. A physical device can therefore be replaced without changing the logical device properties.

`-repository` *`LibraryName`* This option is used to specify the repository backup device that you want to check. This information is then used to update the database.

`-slot` *`SlotID`* `[`*`Side`*`]` Specifies the *`SlotID`* of the library where the medium is mounted. This option is only valid for this backup device type. To specify the side of the platter in this slot, use the additional *`Side`* parameter. Slot *`SlotID`* must be specified for MO devices. Values of *`Side`* are `A` or `B` .

`-import_as_original` Imports medium copy as original if original medium does not exist in database.

`-repository_update` *`DriveName`* Updates the database by reading all the slots (loads media in drive) in the device repository. If you additionally specify the slot number of the slot that is defined for a CL cartridge, then a cleaning operation is performed on the specified drive.

`-repository_barcode_scan` *`LibraryName`* If this option is used then barcode reader is used to update the database. This option should be used only with devices that have enabled barcode reader.

`-add_slots` *`LibraryName`* {*`Slot...`* | *`FromSlot-ToSlot...`*} Adds slots to the selected library. With ADIC/GRAU DAS or StorageTek ACS libraries, this option adds volsers to the selected library.

`-remove_slots` *`LibraryName`* {*`Slot...`* | *`FromSlot-ToSlot...`*} Removes slots from the selected library.

`-enter` *`LibraryName`* {*`Slot...`* | *`FromSlot-ToSlot...`*} Moves media from the mail slots to the repository slots. This option is available only for SCSI libraries.

`-eject` *`LibraryName`* {*`Slot...`* | *`FromSlot-ToSlot...`*} Moves media from the repository slots into the mail slots. This option is available only for SCSI libraries.

`-location` *`Location`* Specifies the new location for the ejected media. Only media with barcode will be updated.

`-silo_query` *`LibraryName`* Queries ACS/DAS server for the list of currently resident volsers and updates the Data Protector repository of specified library. This option is not recommended to be used with an ACS/DAS Server when querying logical libraries configured for the same physical library. In such a case, use the `-add_slots` option to add volsers manually.

With DAS Server, however, when logical libraries are not configured using Data Protector, but using the DAS utilities, the Data Protector query operation can safely be used on such libraries instead of adding volsers manually.

`-silo_enter` *LibraryName* Moves ACS/DAS media from the CAP (ACS) or insert/eject area (DAS) to the repository.

`-silo_eject` *LibraryName* {*Volser*... | *FromVolser-ToVolser*...}

Moves media from the ACS/DAS repository into the CAP.

`-cap` *CapID* ID of Control Access Port of ACS or insert/eject area of DAS library.

`-group` *PoolName MagazineDescription MediumLabel*... Creates a magazine *MagazineDescription* out of the specified non-magazine media. Note that all specified media must be resident in the same SCSI Library at the time. The magazine is added to the pool *PoolName* which must be configured to support magazines.

`-group` *MagazineDescription* Splits the magazine *MagazineDescription* so that the magazine media become non-magazine media.

`-detail` Displays information in a more detailed format.

RETURN VALUES

See the man page `omniintro` for return values.

Additional return values of the `omnimm` command are:

1	Program failed, user error.
2	Program failed, environmental malfunction.
3	Program failed, internal malfunction.
4	Program failed, reason unknown.

EXAMPLES

The following examples illustrate how the `omnimm` command works.

1. To create pool "DDS_Pool" of the class "DDS", with policy "App+Loose". Media in the pool will be usable for 12 months or for 100 overwrites.

```
omnimm -create_pool DDS_Pool "DDS" App+Loose 12 100
```

2. To modify the medium with label "Label23" changing the label to "LABEL23" and location to "Backup Room":

   ```
   omnimm -modify_medium Label23 LABEL23 "Backup Room"
   ```

3. To list detailed information for medium "dat1":

   ```
   omnimm -list_media dat1 -detail
   ```

4. To import a medium in the backup device "Pool1" into pool "Default DDS":

   ```
   omnimm -import Pool1 -pool "Default DDS"
   ```

SEE ALSO

omniamo(1), omnidownload(1), omnimcopy(1), omniminit(1), omnimlist(1), omnimnt(1), omnimver(1), omniupload(1), sanconf(1M), uma(1M)

omnimnt (1)

NAME

omnimnt – responds to Data Protector mount requests

SYNOPSIS

```
omnimnt -help | -version
omnimnt -device BackupDevice -session SessionID [-cancel]
```

DESCRIPTION

The `omnimnt` command satisfies or aborts a Data Protector mount request. A mount request is issued by a backup device once it has filled all the available media. A mount request is a prompt to mount a new medium. Once the requested medium is inserted in the device drive, the `omnimnt` command should be used to confirm that the correct medium is inserted. The mount request can also be canceled which is done by canceling device. If you cancel device, all data objects associated with the backup device that issued the mount request will not be processed any further. To view information on currently active sessions, use the `omnistat` command.

OPTIONS

`-version` Displays the version of the `omnimnt` command

`-help` Displays the usage synopsis for the `omnimnt` command

`-cancel` Cancels the device. This will terminate processing of all objects that are associated with the backup device which issued the request.

`-device` *`BackupDevice`* References the backup device *`BackupDevice`* which issued the mount request, in order to confirm mount request or cancel the device.

`-session` *`SessionID`* Specifies the session using the backup device which issued the mount request.

EXAMPLES

The following examples illustrate how the `omnimnt` command works.

1. To satisfy a mount request issued by device "DAT1" in a session with SessionID "R-1995/05/05-275":

   ```
   omnimnt -device DAT1 -session R-1995/05/05-275
   ```

2. To cancel device for the backup device "Juke" in the session with SessionID "R-1995/05/25-3":

   ```
   omnimnt -device Juke -session R-1995/05/25-3 -cancel
   ```

SEE ALSO

omniamo(1), omnidownload(1), omnimcopy(1), omniminit(1), omnimlist(1), omnimm(1), omnimver(1), omniupload(1), sanconf(1M), uma(1M)

omnimver (1)

NAME

omnimver – verifies that data on the tape is correct

SYNOPSIS

```
omnimver -help | -version
```

```
omnimver -device BackupDevice [-slot SlotID [Side]] [-eject]
```

DESCRIPTION

The `omnimver` command is used to verify the contents of a Data Protector backup medium. It reads the data and verifies that data is written in Data Protector format. If the `-crc` option was used to backup the data it also checks the CRC for each block.

OPTIONS

`-help` Displays an extended usage synopsis for the `omnimver` command

`-version` Displays the version of the `omnimver` command

`-device` *`BackupDevice`* Specifies the backup device where medium is located.

`-slot` *`SlotID`* `[`*`Side`*`]` Specifies the *`SlotID`* of the Exchanger backup device where the medium is mounted. This option is only valid for this backup device type. To specify the side of the platter in this slot, use the additional *`Side`* parameter. Slot *`Side`* must be specified for MO devices.

`-eject` Ejects the medium from the drive after the verification.

EXAMPLES

To verify slot 32 of backup device "Spectra60":

```
omnimver -device Spectra60 -slot 32
```

SEE ALSO

omniamo(1), omnidownload(1), omnimcopy(1),omniminit(1), omnimlist(1), omnimm(1), omnimnt(1), omniupload(1), sanconf(1M), uma(1M)

omniobjconsolidate (1)

NAME

omniobjconsolidate – consolidates Data Protector backup objects into synthetic backups.

SYNOPSIS

```
omniobjconsolidate -version | -help

omniobjconsolidate -cosolidationlist
    ConsolidationSpecificationName -scheduled [GENERAL_OPTIONS]

omniobjconsolidate -consolidationlist
    ConsolidationSpecificationName -postbackup -session SessionID
    [GENERAL_OPTIONS]

omniobjconsolidate Object [Object] ... [GENERAL_OPTIONS]


Object

 {-filesystem | -winfs} Client:MountPoint Label

 -session SessionID

 [-copyid CopyID]

 [-sourcedevice BackupDevice]

 -consolidatedevice BackupDevice

 -targetdevice BackupDevice

 [-protect {none | weeks n | days n | until Date | permanent}]

 [-keepcatalog {weeks n | days n | until Date |
    same_as_data_protection}]

 [[-no_]log | -log_dirs | -log_file]

 [-recycle]

GENERAL_OPTIONS

 -no_monitor

OTHER_OPTIONS

 Date = [YY]YY/MM/DD (1969 < [YY]YY < 2038)
```

DESCRIPTION

The `omniobjconsolidate` command creates synthetic full backups from full and incremental backups on tape and disk devices.

To obtain the information about all backed up objects or sessions containing the objects you want to consolidate, use the `omnidb` command.

This command starts an interactive or automated object consolidation session.

OPTIONS

`-version` Displays the version of the `omniobjconsolidate` command.

`-help` Displays the usage synopsis of the `omniobjconsolidate` command.

`-filesystem` *`Client:MountPoint Label`* Selects the filesystem identified with *`Client:MountPoint Label`* for object consolidation.

`-winfs` *`Client:MountPoint Label`* Selects the Windows filesystem identified with *`Client:MountPoint Label`* for object consolidation.

`-consolidationlist` *`ConsolidationSpecificationName`* Specifies the name of the consolidation specification identified by *`ConsolidationSpecificationName`* for object consolidation.

`-scheduled` Immediately starts a scheduled consolidation specification.

`-postbackup` Immediately starts a post-backup consolidation specification specified by the `-session` *`SessionID`* option.

`-session` *`SessionID`* If specified with the `-postbackup` option provides the Session ID for the postbackup consolidation session.

If specified as a part of the object definition, selects the point in time for the consolidation.

`-copy` *`CopyID`* Selects the copy identified with Copy ID. If not specified, Data Protector automatically selects the most appropriate copy as the source version of the object.

`-recycle` Removes data and catalog protection of the objects on the source media. When there are no more protected objects on the media, the media can be overwritten.

IMPORTANT: If you recycle data protection of source objects, the recycled points in time will no longer be available. Unless copies of these points in time exist, you will be able to restore only to the latest (consolidated) point in time.

`-sourcedevice` *`LogicalDevice`* Specifies a logical device to be used for reading full object versions from the source media. By default (if this option is not specified), the same backup device is used for backing up and reading backed up objects from the source media.

`-consolidatedevice` *`LogicalDevice`* Specifies a logical device that will read incremental object versions and perform consolidation.

`-targetdevice` *`LogicalDevice`* Specifies a backup device that will be used for writing consolidated object versions to the target media.

`-protect {none | weeks` *`n`* `| days` *`n`* `| until` *`Date`* `| permanent}` Sets the level of protection for the consolidated object. The media containing this consolidation session cannot be overwritten until the protection expires. By default (if this option is not specified), the protection is the same as for the full backup in the restore chain that was used as a source for the synthetic backup.

`-keepcatalog {weeks` *`n`* `| days` *`n`* `| until` *`Date`* `| same_as_data_protection}` Specifies file catalog retention time. If you do not want to save the file catalog at all, use the `-no_log` option. By default (if this option is not specified), the protection is the same as for the full backup in the restore chain that was used as a source for the synthetic backup.

`-log` Specifies the logging level of the consolidation session. All detailed information about backed up files and directories (filenames, file versions, and attributes) are logged to the Data Protector internal database (IDB). This allows you to browse directories and files before restore and in addition look at the file attributes. Data Protector can fast position on the tape when restoring a specific file.

If the logging level is not specified, it is set to the same logging level as for the source object.

`-no_log` Specifies the logging level of the consolidation session. Disables the logging of backed up files to the catalog database. By default, the filename and backup history of each backed up file is written to the catalog database.

`-log_dirs` Specifies the logging level of the consolidation session. If this option is specified, only the directories are logged into the database. By default, the filename and backup history of each backed up file is written to the catalog database.

`-log_file` Specifies the logging level of the consolidation session. All detailed information about backed up files and directories (filenames and file versions) is logged to the Data Protector internal database (IDB). This information allows you to search for backed up files and allows Data Protector to fast position the tape. It also does not take much space since some information on file details (file attributes) is not logged to the database.

GeneralOptions

`-no_monitor` If this option is used, the command displays only the session ID. By default, the command monitors the session and displays all messages.

RETURN VALUES

See the man page `omniintro` for return values.

Additional return values of the `omniobjconsolidate` command are:

10	There was an error while consolidating some files. All agents completed successfully.
11	One or more agents failed, or there was a database error.
12	None of the agents completed the operation.
13	Session was aborted.

EXAMPLES

1. To start a consolidation session that consolidates the WinFS object versions for “OBJECT1” on the host “system1.company.com” to the point in time defined with the session ID “`2004/12/06-1`”, using the device “LTO3” as the source device and file device “FILEDEV1” as the consolidation device, writing to the device “LTO4”, use:

```
omniobjconsolidate -winfs system1.company.com:/C 'OBJECT1' -session
2004/12/06-1 -sourcedevice LTO3 -consolidatedevice "FILEDEV1"
-targetdevice LTO4 -recycle
```

2. To start an interactive consolidation session for the filesystem object "system1.company.com:/ 'Label42'" from the session “2005/07/01-2”, using the device “DEV1” to read the source objects and the device “DEV2” to consolidate the objects, writing them to the device “DEV3”, use:

   ```
   omniobjconsolidate -filesystem system1.company.com:/ 'Label42'
   -session 2005/07/01-2 -sourcedevice 'DEV1' -consolidatedevice
   'DEV2' -targetdevice 'DEV3'
   ```

3. To immediately start a post-backup consolidation specification named “post_BU1” for the session “2005/04/03-1”, use:

   ```
   omniobjconsolidate -consolidationlist post_BU1 -postbackup -session
   2005/04/03-1
   ```

4. To immediately start a scheduled consolidation specification named “ConsolidateSpec”, use:

   ```
   omniobjconsolidate -consolidationlist ConsolidateSpec -scheduled
   ```

SEE ALSO

omnib(1), omnir(1), omniobjcopy(1)

omniobjcopy (1)

NAME

omniobjcopy – creates additional copies of the objects backed up with Data Protector on additional media sets.

SYNOPSIS

```
omniobjcopy -version | -help

omniobjcopy -copylist CopySpecificationName -scheduled
    [GENERAL_OPTIONS]

omniobjcopy -copylist CopySpecificationName -postbackup -session
    SessionID [GENERAL_OPTIONS]

omniobjcopy Object [Object] ... [GENERAL_OPTIONS]


Object

 {-filesystem | -winfs | -vbfs | -netware | -omnidb}
    Client:MountPoint Label

 -session SessionID

 [-copyid  CopyID]

 [-sourcedevice BackupDevice]

 -targetdevice BackupDevice

 [-protect  {none | weeks n | days n | until Date | permanent}]

 [-keepcatalog  {weeks n | days n | until Date |
    same_as_data_protection}]

 [[-no_]log | -log_dirs | -log_file]

 [-recycle]

 [-full]

Object

 -rawdisk Client Label

 -session SessionID

 [-copyid  CopyID]
 [-sourcedevice BackupDevice]
 -targetdevice BackupDevice
 [-protect  {none | weeks n | days n | until Date | permanent}]
 [-recycle]
Object
 {-sap | -oracle8 | -informix | -msese | -mssql | -lotus | -mbx |
    -sapdb | -msvssw | -db2 | -sybase} Client:Set
 -session SessionID
 [-copyid  CopyID]
 [-sourcedevice BackupDevice]
 -targetdevice BackupDevice
 [-protect  {none | weeks n | days n | until Date | permanent}]
 [-recycle]
GENERAL_OPTIONS
 -no_monitor
OTHER_OPTIONS
 Date = [YY]YY/MM/DD (1969 < [YY]YY < 2038)
```

DESCRIPTION

The `omniobjcopy` command creates additional copies of objects backed up using Data Protector. You can use the `omniobjcopy` command to copy objects such as filesystems (UNIX or Windows), very big file systems, disk image sections, NetWare objects, and Data Protector internal database (IDB) to an additional media set. The command can be also used for copying the integration objects (SAP R/3, Oracle, Informix Server, MS Exchange, MS Exchange single mailboxes, MS SQL, Lotus, Sybase, DB2, MS Copy and SAP DB).

To obtain the information about all backed up objects or sessions containing the objects you want to copy, use the `omnidb` command.

This command starts an interactive or automated object copy session. Use this command to immediately start an automated (scheduled or post-backup) object copy specification.

OPTIONS

`-version` Displays the version of the `omniobjcopy` command.

`-help` Displays the usage synopsis of the `omniobjcopy` command.

`-vbfs` *`Client:MountPoint Label`* Selects the vbfs (very big file system - OmniStorage object) identified with *`Client:MountPoint Label`* for object copying.

`-filesystem` *`Client:MountPoint Label`* Selects the filesystem identified with *`Client:MountPoint Label`* for object copying.

`-winfs` *`Client:MountPoint Label`* Selects the Windows filesystem identified with *`Client:MountPoint Label`* for object copying.

`-rawdisk` *`Client Label`* Selects the disk image identified by *`Client`* and *`Label`* for object copying.

`-netware` *`Client:MountPoint Label`* Selects the Netware filesystem identified with *`Client:MountPoint Label`* for object copying.

`-omnidb` *`Client:MountPoint Label`* Selects the IDB identified by *`Client:MountPoint Label`* for object copying.

`-sap` *`Client:Set`* Selects the SAP R/3 object identified by *`Client:Set`* for object copying.

`-informix` *`Client:Set`* Selects the Informix Server object identified by *`Client:Set`* for object copying.

`-msese` *`Client:Set`* Selects the Microsoft Exchange Server 2000/2003 object identified by *`Client:Set`* for object copying.

`-mssql` *`Client:Set`* Selects the MS SQL object identified by *`Client:Set`* for object copying.

`-lotus` *`Client:Set`* Selects the Lotus Notes/Domino Server object identified by *`Client:Set`* for object copying.

`-mbx` *`Client:Set`* Selects the MS Exchange single mailbox object identified by *`Client:Set`* for object copying.

`-sapdb` *`Client:Set`* Selects the SAP DB object identified by *`Client:Set`* for object copying.

`-msvssw` *`Client:Set`* Selects the MS Copy object identified by *`Client:Set`* for object copying.

`-db2` *`Client:Set`* Selects the DB2 object identified by *`Client:Set`* for object copying.

`-sybase` *`Client:Set`* Selects the Sybase object identified by *`Client:Set`* for object copying.

`-copylist` *`CopySpecificationName`* Specifies the name of the copy specification identified by *`CopySpecificationName`* for object copying.

`-scheduled` Immediately starts a scheduled copy specification.

`-postbackup` Immediately starts a post-backup copy specification specified by the `-session` *`SessionID`* option.

`-session` *`SessionID`* Selects the Session ID for the `-postbackup` option or for the object definition.

`-full` Selects the whole restore chain of full and incremental backups for the object copy operation. This option is not supported for Data Protector application integrations.

`-recycle` Recycles the data and catalog protection of the source object after an object copy is done.

`-copyid` *`CopyID`* Selects the specified object copy as a source for object copying. By default (if this option is not specified), Data Protector selects the needed media set automatically. If several copies of the same object exist in one session as a result of the object copy or object mirror operation, this option is obligatory.

`-sourcedevice` *`BackupDevice`* Specifies a logical device different from the one used for the backup to be used for reading backed up objects from the source media. By default (if this option is not specified), the same backup device is used for backing up and reading backed up objects from the source media.

`-targetdevice` *`BackupDevice`* Specifies a backup device that will be used for writing object copies to the target media.

`-protect {none | weeks` *`n`* `| days` *`n`* `| until` *`Date`* `| permanent}` Sets the level of protection for the copy object. The media containing this copy session cannot be overwritten until the protection expires. By default (if this option is not specified), the protection is the same as for the source backed up object.

`-keepcatalog {weeks n | days n | until Date | same_as_data_protection}` Specifies file catalog retention time. If you do not want to save the file catalog at all, use the `-no_log` option. By default (if this option is not specified), the protection is the same as for the source object.

`-log` Specifies the logging level of the copy session. All detailed information about backed up files and directories (filenames, file versions, and attributes) are logged to the Data Protector internal database (IDB). This allows you to browse directories and files before restore and in addition look at the file attributes. Data Protector can fast position on the tape when restoring a specific file.

If the logging level is not specified, it is set to the same logging level as for the source object.

`-no_log` Specifies the logging level of the copy session. Disables the logging of backed up files to the catalog database. By default, the filename and backup history of each backed up file is written to the catalog database.

`-log_dirs` Specifies the logging level of the copy session. If this option is specified, only the directories are logged into the database. By default, the filename and backup history of each backed up file is written to the catalog database.

`-log_file` Specifies the logging level of the copy session. All detailed information about backed up files and directories (filenames and file versions) is logged to the Data Protector internal database (IDB). This information allows you to search for backed up files and allows Data Protector to fast position the tape. It also does not take much space since some information on file details (file attributes) is not logged to the database.

GeneralOptions

`-no_monitor` If this option is used, the command displays only the session ID. By default, the command monitors the session and displays all messages.

RETURN VALUES

See the man page `omniintro` for return values.

Additional return values of the `omniobjcopy` command are:

`10` There was an error while copying some files. All agents completed successfully.

`11` One or more agents failed, or there was a database error.

`12` None of the agents completed the operation.

`13` Session was aborted.

EXAMPLES

1. To start an interactive object copy session for copying two WinFS objects “system.company.com:/C 'Object1'“ and “system.company.com:/C 'Object1'“ from two different sessions to the device “DEV1“, so that the source object version for “Object1“ is then recycled, use:

```
omniobjcopy -winfs system.company.com:/C 'Object1' -session 2003/
09/01-3 -targetdevice 'DEV1' -recycle -winfs systems.company.com:/C
'Object2' -session 2003/09/24-2 -targetdevice 'DEV1'
```

2. To start an interactive copy session for copying the whole restore chain of full and incremental backups for the filesystem object "system1.company.com:/ 'Label42'" from the session “2004/07/01-2”, using the device “DEV1” to read the source objects and the device “DEV2” copy the objects, use:

```
omniobjcopy -filesystem system1.company.com:/ 'Label42' -session
2004/07/01-2 -sourcedevice 'DEV1' -targetdevice 'DEV2' -full
```

3. To immediately start a post-backup copy specification named “post_BU1” for the session “2004/02/03-1”, use:

```
omniobjcopy -copylist post_BU1 -postbackup -session 2004/02/03-1
```

4. To immediately start a scheduled copy specification named “CopySpec”, use:

```
omniobjcopy -copylist CopySpec -scheduled
```

SEE ALSO

omnib(1), omnir(1), omniobjconsolidate(1)

omnir (1)

NAME

omnir – command to start a restore of filesystems, Very Big File Systems (VBFS), disk images, Data Protector database (IDB), MS Exchange single mailboxes and Public Folders, MS Exchange, MS SQL, SAP R/3, SAP DB, Informix Server, Lotus, IBM DB2 UDB, NetWare objects and NDMP objects. The command is also used to start the instant recovery process. To restore a Sybase database, refer to the `syb_tool` man page.

SYNOPSIS

```
omnir -version | -help

omnir SessionOptions [-noexpand ] Object [Object... ]

 SessionOptions

 -[no_]preview

 -report {warning | minor | major | critical}


FILESYSTEM RESTORE

Object

 {-filesystem | -winfs | -vbfs | -netware} Client:MountPoint
    Label

 -session SessionID [-copyid CopyID]

 -tree TreeName...

 [DATA_OPTIONS]

 [FILESYSTEM_OPTIONS]

 [GENERAL_OPTIONS]

 [SPLIT_MIRROR_OPTIONS]

Object

 {-filesystem | -winfs | -vbfs | -netware} Client:MountPoint
    Label

 -full [-session SessionID]
 -tree TreeName...
 [DATA_OPTIONS]
 [FILESYSTEM_OPTIONS]
 [SPLIT_MIRROR_OPTIONS]
 [GENERAL_OPTIONS]
Object
 {-filesystem | -winfs | -vbfs | -netware} Client:MountPoint
    Label
 -omit_deleted_files [-session SessionID [-copyid CopyID] ]
 -overwrite
 -tree TreeName...
 [DATA_OPTIONS]
 [FILESYSTEM_OPTIONS]
 [SPLIT_MIRROR_OPTIONS]
 [GENERAL_OPTIONS]
Object
 {-filesystem | -winfs | -vbfs | -netware} Client:MountPoint
    Label
 -tree TreeName...
 MEDIUM_OPTIONS
 [DATA_OPTIONS]
 [FILESYSTEM_OPTIONS]
 [GENERAL_OPTIONS]
Object
 -host Clientname
 -session SessionID
 [-full | -omit_deleted_files | -overwrite]
 [FILESYSTEM_OPTIONS]
 [GENERAL_OPTIONS]
```

```
INTERNAL DATABASE RESTORE
Object
 -omnidb Client:MountPoint Label
 -session SessionID [-copyid CopyID]
 -tree TreeName...
 -into Pathname
 [FILESYSTEM_OPTIONS]
 [GENERAL_OPTIONS]
Object
 -omnidb Client:MountPoint Label
 -tree TreeName...
 -into Pathname
 MEDIUM_OPTIONS
 [FILESYSTEM_OPTIONS]
 [GENERAL_OPTIONS]

RAW DISK RESTORE
Object
 -rawdisk Host Label
 -session SessionID [-copyid CopyID]
 -section [ToSection=] Section
 [SPLIT_MIRROR_OPTIONS]
 [GENERAL_OPTIONS]
Object
 -rawdisk Host Label
 -section [ToSection=] Section
 MEDIUM_OPTIONS
 [GENERAL_OPTIONS]


INSTANT RECOVERY

omnir -host ClientName

 -session SessionID

 -instant_restore

 [DISK_ARRAY_XP_OPTIONS | VIRTUAL_ARRAY_OPTIONS |
   ENTERPRISE_VIRTUAL_ARRAY_OPTIONS]

 [ORACLE_SPECIFIC_OPTIONS]

 [SAP_SPECIFIC_OPTIONS]

 [VSS_SPECIFIC_OPTIONS]


 DISK_ARRAY_XP_OPTIONS

  -keep_version

  -check_config


 VIRTUAL_ARRAY_OPTIONS

  -keep_version

  -check_config


 ENTERPRISE_VIRTUAL_ARRAY_OPTIONS

  -check_config

  -force_prp_replica


 SAP_SPECIFIC_OPTIONS

  -sap

  -user UserName -group GroupName

  -recover {now | time MM/DD/YY hh:mm:ss | logseq  LogSeqNum
    thread  ThreadNum | SCN Number} [-open [-resetlogs ]]
 ORACLE_SPECIFIC_OPTIONS

  -oracle

  -user UserName -group GroupName

  -recovery {now | time MM/DD/YY hh:mm:ss | logseq  LogSeqNum
    thread  ThreadNum | SCN Number} [-open [-resetlogs ]]

  -parallelism Number


 VSS_SPECIFIC_OPTIONS

  -vss

  -barhost ClientName

  -session SessionID1 -tree TreeName1 [-tree TreeName2] ...
    [-session SessionID2 -tree TreeName3 [-tree TreeName4] ... ]
    ...


NDMP RESTORE

Object

 -filesystem Host:MountPoint Label

 -full [-session SessionID]

 -tree TreeName...

 [NDMP_DATA_OPTIONS]

 [NDMP_GENERAL_OPTIONS]

Object

 -filesystem Host:MountPoint Label

 -session sessionID [-full]

 -tree TreeName...

 [NDMP_DATA_OPTIONS]

 [NDMP_GENERAL_OPTIONS]
 NDMP_DATA_OPTIONS
  -into PathName
  -ndmp_env FileName
  -ndmp_user UserName
  -ndmp_passwd Password
 NDMP_GENERAL_OPTIONS
  -device BackupDevice
  -server ServerName
  -no_monitor
  -var[iable] var_name var_value

SAP R/3 FILE RESTORE
Object
 -sap Client:Set
 -session SessionID [-copyid CopyID]
 -tree FileName...
 [DATA_OPTIONS]
 [FILESYSTEM_OPTIONS]
 [GENERAL_OPTIONS]

SAP DB RESTORE
omnir -sapdb
 -barhost ClientName
 -instance InstanceName
 [-destination ClientName]
 [-newinstance DestinationInstanceName]
 [-session SessionID]
 [-recover [-endlogs | -time: YYYY-MM-DD.hh.mm.ss] [-from_disk] ]
 [-nochain]

INFORMIX SERVER RESTORE
omnir -informix
 -barhost ClientName
 -barcmnd PathName
 -user User:Group
 -appname ApplicationDatabaseName
 -bararg OnBarRestoreArguments
 [INFORMIX_OPTIONS]

 INFORMIX_OPTIONS
  -report {warning | minor | major | critical}
  -load {low | medium | high}
  -no_monitor

MS EXCHANGE 2000/2003 RESTORE
omnir -msese
 -barhost ClientName
 [-destination ClientName]
 -appname full_application_name
 -base DBName
 -session SessionID...
 -logpath Path
 [-mount] [-last]

MS EXCHANGE SINGLE MAILBOX RESTORE
omnir -mbx
```

```
 -barhost HostName
 [-destination HostName]
 -mailbox MailboxName -session SessionID [MAILBOX_OPTIONS] ...
 -public -session SessionID [PUBLIC_FOLDERS_OPTIONS] ...

 MAILBOX_OPTIONS
  -folder FolderName
  -exclude FolderName
  -originalfolder {-keep_msg | -overwrite_msg}
  -destMailbox DestMailboxName
  -chain
 PUBLIC_FOLDERS_OPTIONS
  -folder FolderName
  -exclude FolderName
  -originalfolder {-keep_msg | -overwrite_msg}
  -chain

MS SQL RESTORE
omnir -mssql
 -barhost ClientName
 [-destination ClientName]
 [-instance SourceInstanceName]
 [-destinstance DestinationInstanceName]
 -base DBName -session SessionID [MSSQL_OPTIONS] ...

 MSSQL_OPTIONS
 -asbase NewDBName {-file LogicalFileName1 PhysicalFileName1
    [-file LogicalFileName2 PhysicalFileName2] ...}
 -replace
 -nochain
 -recovery  {rec | norec}
 -standby File
```

LOTUS RESTORE

```
omnir -lotus
 -barhost ClientName
 [-user User:Group]
 [-destination ClientName]
 -ini path_to_notes.ini
 -domino_server srv_name
 -db db1 [-db db2] ...
 [-NSF] [-NTF] [-BOX] [-ALL]
 [-direx direx1 [-direx direx2] ...]
 [-r_dest restore_dir]
 [-recover | recovery_time yyy/mm/dd.hh:mm:ss]
 [-session SessionID]
```

VSS RESTORE

```
omnir -vss
 -barhost ClientName
 [-destination ClientName]
 -session SessionID1 {-tree TreeName1 [-tree TreeName2 ...] }
 [-session SessionID2 {-tree TreeName3 [-tree TreeName4 ...] ...}
    ]
 [-into PathName]
```

DB2 RESTORE

Object

```
omnir -db2
 -barhost ClientName
 -instance InstName
 {[-dbname DBName [-session SessionID] [-newdbname NewDBName] ...
   ] [-tsname DBName*TSName [-session SessionID] [-offline] ...]
   [-logfile DBName*LogFileName [-session SessionID] ...] }
 [DB2_OPTIONS]
 DB2_OPTIONS
 -destination ClientName
 -rollforward [-time YYYY-MM-DD.hh.mm.ss]
 -frominstance InstName

DATA_OPTIONS
 -exclude PathName ...
 -skip MatchPattern ...
 -only MatchPattern ...
 -as Pathname
 -into Pathname
MEDIUM_OPTIONS
 -device BackupDevice
 -medium MediumID
 -id DiskAgentID
 [-slot SlotID [Side]]
FILESYSTEM_OPTIONS
 -touch
 -lock
 -no_protection
 -[no_]overwrite | -merge
 -catalog
```

-sparse

-move_busy

-vsr_only

-trustee

-no_share[_info]

GENERAL_OPTIONS

-device *BackupDevice*

-server *ServerName*

-target *Client*

-profile

-load {low | medium | high}

-pre_exec *PathName*

-post_exec *PathName*

-variable *var_name var_value*

-no_monitor

SPLIT_MIRROR_OPTIONS

-sse | -symmetrix

-remote *Host Host* | -local *Host Host* | -combined *Host Host*

-quiesce *cmd*

-restart *cmd*

-mirrors *list*

[-discovery]

[-re_establish_links_before_restore]

-disable_disks

-restore_links_after_restore

DESCRIPTION

The `omnir` command restores objects backed up using Data Protector. You can use the `omnir` command to restore filesystems (UNIX, Windows), very big file systems, disk image sections, NetWare objects, NDMP objects and Data Protector internal database (IDB) to their original (or a new) location. The command can be also used for restoring Integration objects (SAP R/3, MS Exchange, MS Exchange single mailboxes, MS SQL, Lotus, Informix Server, DB2 and SAP DB) or to start the instant recovery process. To restore a Sybase database, refer to the `syb_tool` man pages.

If several copies of the same object version exist, you can let Data Protector select which media set will be used for the restore. You can also specify the media set from which you want to restore the data, except when restoring integration objects. It is not possible to specify the media set created as a result of the media copy operation.

The `omnir` command also supports parallel restore. You can achieve this by specifying more than one object using the command line options. It is not possible to use the `-medium` option when performing a parallel restore. The number of objects for parallel restore is limited by the global option `MaxSessions`, which can be set in the `/etc/opt/omni/server/options/global` (UNIX Cell Manager) or in the *`<Data_Protector_home>`*`\config\server\options\global` (Windows Cell Manager) file.

NOTE: It is not allowed to specify the same object more then once within the same `omnir` command. To differentiate options for the same object (for example, the `-tree` option) specify these options for the same object as many times as needed.

Information about all backed up objects can be obtained from the IDB by using `omnidb` command or, in the case of the instant recovery, from the ZDB database by using the `omnidbxp`, `omnidbva`, `omnidbeva` or `omnidbsmis` command. (See the related man pages for more information). For most restore actions you need to specify the *`SessionID`* of the session containing the object you want to restore, which can be obtained by the `omnidb` command.

NOTE: When restoring integration objects, provide the *`SessionID`* of the backup session. In case of object copies, do not use the copy session ID, but the object's *`BackupID`*, which equals the object's backup session ID.

To restore objects from a medium that is not in IDB, use the `-medium` *`MediumID`* option, instead of the *`SessionID`* .

NOTE: The `-medium` option is not possible when performing a parallel restore.

To get the *MediumID* and *DiskAgentID* from the medium, use the `omnimlist` command to read the medium. See the `omnimlist` man page for more information on this command.

OPTIONS

`-version` Displays the version of the `omnir` command

`-help` Displays the usage synopsis of the `omnir` command

`-vbfs` *`Client:MountPoint Label`* Specifies the vbfs (very big file system - OmniStorage object) identified with *`Client:MountPoint Label`* for restore.

`-filesystem` *`Client:MountPoint Label`* Selects the filesystem identified with *`Client:MountPoint Label`* for restore.

`-winfs` *`Client:MountPoint Label`* Selects the Windows filesystem identified with *`Client:MountPoint Label`* for restore.

`-netware` *`Client:MountPoint Label`* Selects the Netware filesystem identified with *`Client:MountPoint Label`* for restore.

`-rawdisk` *`Client Label`* Selects the disk image identified by *`Client`* and *`Label`* for restore.

`-omnidb` *`Client:MountPoint Label`* Selects the IDB identified by *`Client:MountPoint Label`* for restore

`-instant_restore` Selects instant recovery for ZDB integrations.

`-sap` *`Client:Set`* Selects the SAP R/3 object identified by *`Client:Set`* for restore

`-sapdb` Selects the SAP DB object for restore

`-informix` Selects the Informix Server object for restore

`-msese` Selects the Microsoft Exchange Server 2000/2003 object for restore

`-mbx` Selects MS Exchange single mailboxes and Public Folders for restore

`-mssql` Selects the MS SQL object, identified with *`DBName`* for restore

`-lotus` Selects the Lotus Notes/Domino Server object for restore

`-vss` Selects the VSS object for restore

`-db2` Selects the IBM DB2 UDB object to restore

`-session` *`SessionID`* Specifies the session to be used for restore.

When restoring integration objects that have copies (except in case of SAP R/3 integration objects), do not use the copy session ID, but the object's *BackupID*, which equals the object's backup session ID.

With SAP DB, if this option is not specified, the last backup session is restored regardless of the `-endlogs` or the `-time` option selection.

With Microsoft Exchange Server 2000/2003, this option must be set for every `-base` option specified.

`-copyid` *CopyID* If several copies of the same object exist in one session as a result of the object copy or object mirror operation, this option identifies the specific object copy (object mirror or object copy) to be used for restore. By default (if this option is not specified), Data Protector selects the media set to restore from automatically. When using this option, it is necessary to specify both the object and the session.

`-section [`*ToSection*`=]`*Section* Specifies the disk image section to be restored. To restore the section to a new section, include both the source and destination section.

`-full` Specifies that the selected object will be restored from the last full backup and all incremental backups related to this full backup.

`-tree` *TreeName* Specifies the file, component, or tree to restore. Note that when specifying trees on UNIX systems, the complete tree must be specified including the mount point, whereas on Windows systems trees must be specified without volumes (drives). For example: `-tree /temp` (Windows system) or `-tree /usr/temp` (UNIX system).

`-host` *ClientName* Restores all objects of the specified client that were backed up in the specified session. This option is only valid for the filesystem restore. If any other type of object (for example, the Data Protector internal database) was a part of the specified session, the restore will abort.

`-omit_deleted_files` This option can be only used in combination with the `-overwrite` option.

If this option is specified, Data Protector attempts to recreate the state of the restored directory tree as it was when the last incremental backup was run, while preserving files that were created or modified after the last incremental backup. However, if the directory contains files that did not exist there at the time of the last incremental backup, but their modification time is older than the time of the incremental backup, Data Protector will delete these files as well.

When this option is used in combination with the `-as` or `-into` option, be careful when specifying the new location to prevent accidental deletion of existing files.

If this option is not specified, when restoring a directory from which files were deleted between a full and an incremental backup, these files are also restored.

The time on the Cell Manager and clients must be synchronized for this option to function properly.

`-barhost` *`ClientName`* Specifies the system where the Data Protector application (Informix Server, MS Exchange, MS SQL, Lotus, MS VSS, DB2 or SAP DB) client that *was backed up* is installed.

`-barcmnd` *`PathName`* For Informix Server restore: The value of the `barcmnd` option has to be set to `ob2onbar.exe`. The command should reside in `/opt/omni/bin` directory on HPUX systems and in *`<Data_Protector_home>`*`\bin` directory on Windows systems.

`-bararg` *`OnBarRestoreArguments`* Pathname of an application restore command script. For Informix Server restore: Specifies the onbar restore arguments. Each onbar restore argument has to be put in double quotes.

`-user` *`UserName:GroupName`* Specifies *`Username`* and *`GroupName`* that started the script specified by the `-barcmnd` option.

`-appname ApplicationDatabaseName` This option is used for Informix Server restore. For Informix Server restore, the database server name of Informix Server to be restored is specified.

`-destination` *`ClientName`* Specifies the target client for MS Exchange, MS Exchange single mailboxes, MS SQL, Lotus Notes/Domino Server, MS VSS, IBM DB2 UDB or SAP DB restore. This option is to be used only when a restore to some other instance than the one that was backed up is to be performed.

`-appname` *`full_application_name`* Specifies a Microsoft Exchange Server 2000/2003 Information Store, Site Replication Service or Key Management Service for the restore. The name of the Store/Service (*`full_application_name`*) must be provided in double quotes as follows:

- For the Information Store: `Microsoft Exchange Server (Microsoft Information Store)`
- For the Site Replication Service: `Microsoft Exchange Server (Microsoft Site Replication Service)`
- For the Key Management Service: `Microsoft Exchange Server (Microsoft Key Management Service)`

`-mailbox` *`MailboxName`* Specifies the MS Exchange single mailboxes for restore.

`-public` Specifies the MS Exchange Public Folders for restore (as part of the MS Exchange Single Mailbox restore).

`-stop` The MS Exchange database is stopped before the restore is started.

`-start` The MS Exchange database is started after the restore session.

`-base` *`DBName`* Specifies the MS SQL database or Microsoft Exchange Server 2000/2003 store or logs for restore for restore. Note that for MS SQL the name of the database is case sensitive.

`-logpath` *`path`* Specifying this option, you set the temporary directory for the Microsoft Exchange Server 2000/2003 log files. Data Protector restores the log files to this directory. Using this directory, the Microsoft Exchange Server 2000/2003 then recovers the database - this operation is referred to as hard recovery.

`-mount` The restored Microsoft Exchange Server 2000/2003 databases will be automatically mounted after the soft or hard recovery.

`-last` Hard recovery is performed after the restore of the Microsoft Exchange Server 2000/2003 object. Use this option if you are restoring the last set of files. If you do not set this option, you have to start the recovery manually by running the `eseutil /cc /t` utility from the directory for temporary log files. If this option is not specified, soft recovery is performed after the restore.

`-ini` *`path_to_notes.ini`* Lotus Notes/Domino Server only: Sets the absolute path to the `notes.ini` file located on the Lotus Notes/Domino Server which you want to restore.

`-domino_server` *`srv_name`* Lotus Notes/Domino Server only: Sets the name of the Lotus Notes/Domino Server which you want to restore.

`-db` *`db`* Lotus Notes/Domino Server only: Sets the name of the Lotus Notes/ Domino Server database that you want to restore.

`-NSF` Lotus Notes/Domino Server only: Sets the restore of all NSF (Notes Storage Facility) databases.

`-NTF` Lotus Notes/Domino Server only: Sets the restore of all NTF (Notes Templates Facility) files.

`-BOX` Lotus Notes/Domino Server only: Sets the restore of all BOX files.

`-ALL` Lotus Notes/Domino Server only: Sets the restore of all objects, NSF databases, NTF files and BOX files.

`-dir` *`dir`* Lotus Notes/Domino Server only: Sets the Lotus Notes/Domino data directories that you want to include in the restore. Enter their relative pathnames to the Lotus Notes/Domino data directory.

`-direx` *`direx`* Lotus Notes/Domino Server only: Sets the Lotus Notes/Domino data directories that you want to exclude from the restore. Enter their relative pathname to the Lotus Notes/Domino data directory.

`-r_dest` *`restore_dir`* Lotus Notes/Domino Server only: Sets the relative pathname to the restored database directory.

`-recover` Lotus Notes/Domino Server only: Specify this option to perform the recovery of the restored database to the last possible consistent state.

`-recovery_time` *`yyyy/mm/dd.hh:mm:ss`* Lotus Notes/Domino Server only: Sets a point in time to which you want the database to be recovered.

`-instance` *`InstName`* IBM DB2 and SAP DB: Sets the name of the database instance that was backed up.

`-frominstance` *`InstName`* IBM DB2 UDB only: Sets the name of the DB2 instance from which you want to restore the data.

`-dbname` *`DBName`* Sets the name of the DB2 database that you want to restore.

`-newdbname` *`NewDBName`* IBM DB2 UDB only: Specify this option if you want to restore the whole DB2 database into a new database.

`-tsname` *`DBName*TSName`* IBM DB2 UDB only: Sets the name of the DB2 table space that you want to restore. To specify the table space you would like to restore, write the name of the database, then the “*” character and finally the name of the table space (without spaces).

`-logfile` *`DBName*LogFileName`* IBM DB2 UDB only: Sets the name of the DB2 Log file that you want to restore. It should not be used with the `-rollforward` option. To specify the Log file you would like to restore, write the name of the database, then the "*" character and finally the name of the Log file(without spaces).

`-rollforward [time:` *`YYYY-MM-DD.hh.mm.ss`*`]` IBM DB2 UDB only: Specify the point in time when you want a rollforward to be performed to. The rollforward point in time *must* be entered in local time (as it is set on the DB2 target server) and not in coordinated universal time (UTC). If you specify a `rollforward` option without `time` argument, a rollforward will be performed to the and of the logs.

`-offline` IBM DB2 UDB only: Specify this option if you want to restore a table space offline.

`-newinstance` *`DestinationInstanceName`* SAP DB only: Performs a restore to the SAP DB instance with the instance name *`DestinationInstanceName`*. This option is to be used only when a restore to an instance other than the one that was backed up is to be performed. Note that the specified instance must already exist and must be configured for use with Data Protector. This option does not create a new instance.

`-recover [-endlogs | -time:` *`YYYY-MM-DD.hh.mm.ss`*`]` SAP DB only: Specify this option to recover the restored SAP DB database by applying the restored (if the `-from_disk` option is not specified) or client-resident logs (if the `-from_disk` option is specified) to the last available log (the default behavior, or if the `-endlogs` option is specified), or to the specified point in time (if the `-time:` option is specified).

Make sure that the backup session selected by the `-session` option will restore enough data for the integration to apply the redo logs until the last available log or until the specified point in time.

When this option is not specified, the following happens after the restore:

- If archive logs are not restored (if restore from a full backup session is performed), the database remains in the `Admin` mode after the restore.

- If archive logs are restored, the database is, if the restored archive logs allow it, switched to the `Online` mode. If the database, however, cannot be switched to the `Online` mode (because the restored archive logs do not allow it), it remains in the `Admin` mode.

`-endlogs` SAP DB only: Specify this option to recover the database until the last log. This is the default option.

`-time:` *`YYYY-MM-DD.hh.mm.ss`* SAP DB only: Specify the `-time:` option to recover the database until the point specified by the *`YYYY-MM-DD.hh.mm.ss`* argument.

Note that the specified time is the system time on the system running the Data Protector CLI. If the system to be recovered is not in the same time zone as the system running the Data Protector CLI, the point of recovery is adjusted to the local time setting on the system to be restored.

`-from_disk` SAP DB only: Specify this option to apply the existing archive logs on the SAP DB Server to SAP DB Server redo logs.

If this option is not specified, the backed up archive logs on backup media are applied to the redo logs (if trans backup session is restored), or the redo logs are left intact together with the existing archive logs on the SAP DB Server (if full or diff backup session is restored).

When a transactional backup session is selected for restore or when it is a part of the needed restore chain, and the this option is specified at the same time, the archive logs from Data Protector media are applied to the redo logs. Thereafter, the archive logs on the SAP DB Server are applied to redo logs.

This option is ignored in case of SAP DB migration, thus allowing only for the restore of redo logs from the backed up archive logs on backup media (if trans backup session is restored).

`-nochain` SAP DB only: This option instructs the command to restore only the selected or last backup session; the integration does not restore the whole restore chain of full, differential, and transactional backups.

`-destinstance` *`DestinationInstanceName`* MS SQL only. Specify this option to determine an MS SQL instance into which the data will be restored. `Omnir` takes the (DEFAULT) instance by default.

`-instance` *`SourceInstanceName`* MS SQL only. Sets the name of the MS SQL instance to be restored. `Omnir` takes the (`DEFAULT`) instance by default.

`Mailbox/Public_Folders_Options`

`-folder` *`FolderName`* MS Exchange Single Mailbox restore only: Specifies folders to be restored. Note that the subfolders are also restored. If this option is not specified, all backed up folders are restored.

`-exclude` *`FolderName`* MS Exchange Single Mailbox restore only: Specifies the folders to be excluded from restore.

`-originalfolder {-keep_msg | -overwrite_msg}` MS Exchange Single Mailbox restore only: If this option is selected, Data Protector restores Exchange items to the same folders in which they were when the backup was performed.

If `-keep_msg` is selected, the messages in the mailbox or Public Folders are not restored, even if they are different from their backed up version.

If `-overwrite_msg` is selected, all messages are restored, replacing their current versions (if they exist). If different versions of the same message exist in the mailbox or Public Folders (for example, if you have a copy of the message), only one is replaced with the backed up version and all other versions remain intact.

The messages in the mailbox that were not backed up in the specified backup session (or the restore chain of backup sessions) always remain intact.

If `-originalfolder` is not specified, Data Protector creates a new folder in the root of the mailbox or in the root of `All Public Folders` and restores Exchange items into it. For a mailbox restore, the folder is named `Data Protector` *`BackupDate BackupTime`*, and for a Public Folders restore, it is named `Data Protector` *`BackupDate BackupTime`* `- public folder`. If you restore a mailbox or Public Folders from the same backup several times, a number is appended to the folder name. For example, in the second restore session of a mailbox, the folder `Data Protector` *`BackupDate BackupTime`* `(1)` is created.

`-destMailbox` *`DestMailboxName`* MS Exchange Single Mailbox restore only: Specifies the destination mailbox, into which data will be restored. The destination mailbox must exist on the target MS Exchange Server. If this option is not specified, data is restored to the original mailbox.

`-chain` MS Exchange Single Mailbox restore only: If this option is specified, data is restored not only from the specified backup session, but also from the latest full, the latest incremental1 (if exists), and all incremental backups from the last incremental1 up to the specified version.

`MSSQL_Options`

`-asbase` *`NewDBName`* {`-file` *`LogicalFileName1 PhysicalFileName1`* [`-file` *`LogicalFileName2 PhysicalFileName2`*]...}

This option enables you to restore the MS SQL database under a new name or to restore files to a new location. If the `-asbase` option is used, all logical and physical filenames have to be specified with the `-file` option.

`-replace` If the specified MS SQL database already exists on the specified client, it is overwritten.

`-nochain` MS SQL only: Restores only the session identified by the `-session` *`sessionID`* option. If no *`sessionID`* is specified, only the latest session is restored.

`-recovery {rec | norec}` Specifies the state (recovered, nonrecovered) of the MS SQL database after the restore. The default value for this option is `rec` .

`-standby` *`File`* Specifies the standby state of the MS SQL database after the restore.

`Oracle/SAP_Specific_Options` (instant recovery only)

`-oracle` Selects the Oracle options for instant recovery.

`-sap` Selects the SAP R/3 options for instant recovery.

`-recover {now | time` *`Time`* `| logseq` *`LogSeqNum`* `thread` *`ThreadNum`* `|` `SCN` *`Number`*`}`

Selects the point in time to which the database is recovered. The following options are available:

- `now`

All existing archive logs are applied.

- `time` *`MM/DD/YY hh:mm:ss`*

  Specifies an incomplete recovery. Archive logs are applied only to a specific point in time.

- `logseq` *`LogSeqNum`* `thread` *`ThreadNum`*

  Specifies an incomplete recovery. Archive logs are applied only to the specified redo log sequence and thread number.

- `SCN` *`Number`*

  Specifies an incomplete recovery. The archive logs are applied only to the specified SCN number.

`-open` Opens the database after recovery.

`-resetlogs` Resets the logs after the database is opened. Available only if the `-open` option is specified. This option is not available if the `-recovery` option is set to `now`.

The following are recommendations on when to reset the logs. *Always* reset the logs:

- After an incomplete recovery, that is if not all archive redo logs will be applied.
- If a backup of a control file is used in recovery.

Do not reset the logs:

- After a complete recovery where a backup of a control file is not used in recovery.
- If the archive logs are used for a standby database. If you must reset the archive logs, then you have to recreate the standby database.

`-user` *`UserName`* `-group` *`GroupName`* Selects the username and group name of the account under which Data Protector starts instant recovery.

`-parallelism` *`Number`* Oracle recovery only. Selects the parallelism for the restore of archive logs and restore from incremental backups.

`DataOptions`

`-exclude` *`TreeName`* Excludes the specified tree from the restore. This option is not supported with the Data Protector NDMP server integration.

`-skip` *`MatchPattern`* Excludes files matching *`MatchPattern`* from restore. This option is not supported with Data Protector NDMP server integration.

`-only` *`MatchPattern`* Restores only files that match the given *`MatchPattern`*. This option is not supported with Data Protector NDMP server integration.

`-as` *`Pathname`* Restores the selected fileset as the specified tree.

`-into` *`Pathname`* Restores the selected fileset into the given directory.

`-ndmp_user` *`UserName`* Sets the username that is used by Data Protector to establish the connection to the NDMP server.

`-ndmp_passwd` *`Password`* Sets the password for the username that is used by Data Protector to establish the connection to the NDMP server.

`-ndmp_env` *`FileName`* Specifies the filename of file with NDMP environment variables for specific NDMP implementations.

`SessionOptions`

`-preview` Checks the restore parameters without performing the actual restore.

`-report {warning | minor | major | critical}` Sets the level of error notification for the session. Errors are classified (in ascending order) as: `warning`, `minor`, `major` and `critical`. When you select a level, errors of this level and higher are displayed in the Monitor window. For example, if `major` is selected, only `major` and `critical` errors are reported. By default, all errors are reported.

`MediumOptions`

`-device` *`BackupDevice`* Specifies the backup device where the backup medium is mounted.

`-medium` *`MediumID`* Specifies the medium from which data will be restored.

This option is not possible when performing a parallel restore.

`-slot` *`SlotID`* `[`*`Side`*`]` Specifies the *`SlotID`* of the tape library unit where the medium is mounted. This option is only valid for this backup device type. To specify the side of the platter in this slot, use the additional *`Side`* parameter. Slot *`Side`* must be specified for MO devices. Values for side are `A` or `B`.

`-id` *`DiskAgentID`* Specifies the ID of the disk agent which should be used for restore.

FilesystemOptions

`-touch` Updates the access date/time of the file during the restore. By default the access date/time of the backup version is used.

This option is not supported on Novell NetWare.

`-lock` When performing a restore of a file, the disk agent tries to lock the file. By default the file is not locked.

`-no_protection` Do not restore protection of the backed up files, instead use the default protection settings.

`-overwrite` Overwrites files with the same name in the specified fileset on the disk.

`-no_overwrite` Does not overwrite existing files with the same name.

`-merge` This option merges files from the backup medium to the target directory and replaces older versions that exist in the directory with newer (if they exist on the medium) files. Existing files are overwritten if the version on the medium is newer than version on disk. No existing directory is deleted.

If a directory or file doesn't exist on disk (but is on the backup medium) it is restored (created).

`-catalog` Displays the restored files and directories.

`-sparse` Restores sparse files in their original form.

`-move_busy` This option is useful only in case the option `-overwrite` is specified. A problem can occur if, for example, a file to be overwritten cannot be deleted because it is currently in use. Setting this option causes busy files to be moved to a filename starting with #. The original file can thus be deleted as the lock is transferred to the corresponding file starting with # sign. For example, `/tmp/DIR1/DIR2/FILE` would be moved to `/tmp/DIR1/DIR2/#FILE`.

`-vsr_only` A Novell NetWare specific option, allowing a restore of volume space restrictions on NetWare or NSS volume without restoring any other data. This option works only if volume object is selected for restore. If volume object is not selected for restore, the `-vsr_only` option does not affect the restore process.

`-trustee` A Novell Netware specific option, allowing a restore of inheritance filters and ownership information of the selected objects only. If enabled, Conflict Handling options are also enabled.

`-no_share[_info]` If this option is specified, share information for directories on Windows is not restored. If a directory was shared on the network when a backup was run with the `Backup share information for directories` option set (by default), it will be automatically shared after restore, unless this option is selected for restore.

GeneralOptions

-device *BackupDevice* Specifies the backup device where the backup medium is mounted.

-server *ServerName* Selects the Cell Manager with the client name *ServerName* as the Cell Manager. Use this option to perform a restore to a client that is not in the current Data Protector cell.

`-target` *Client* Restores the selected fileset to the specified client.

`-profile` Displays restore statistics.

`-load {low | medium | high}` Specifies the level of network traffic generated by a session during a time period. `High` level generates as much traffic as allowed by the network, resulting in a faster restore. A `low` level has less impact on network performance, but results in a slower restore. By default, this option is set to `high` .

`-pre_exec` *PathName* Instructs the Disk Agent to execute this command before restoring the data object. The complete pathname of the command should be specified.

`-post_exec` *PathName* Instructs the Disk Agent to execute this command after restoring the data object. The complete pathname of the command should be specified.

`-variable` *var_name var_value* This option lets you specify a variable name and its value for proper operation of some platforms and integrations, for example, for backing up and restoring an MPE. Setting user definable variables (a variable name and its value) enables flexible operation on some platforms and integrations with Data Protector. The list of variables and their values that are configurable with Data Protector is dynamic and comes with Data Protector patches.

`-no_monitor` By default the command monitors the session and displays all messages. If this option is used, the command displays only the sessionID.

SplitMirrorOptions

`-sse` Selects the HP StorageWorks Disk Array XP split mirror restore.

`-symmetrix` Selects the EMC Symmetrix split mirror restore.

`-remote` *AplicationClient BackupClient* If the `-symmetrix` option was specified, this option selects the EMC Symmetrix Remote Data Facility (SRDF) split mirror configuration. If the `-sse` option was specified, this option selects the HP StorageWorks Continuous Access (CA) split mirror configuration.

`-local` *AplicationClient BackupClient* If the `-symmetrix` option was specified, this option selects the EMC Symmetrix Time Finder split mirror configuration. If the `-sse` option was specified, this option selects the HP StorageWorks Business Copy (BC) split mirror configuration.

`-combined` *AplicationClient BackupClient* If the `-symmetrix` option was specified, this option selects the EMC Symmetrix combined (SRDF & Time Finder) split mirror configuration. If the `-sse` option was specified, this option selects the HP StorageWorks combined (CA+BC) split mirror configuration.

`-mirrors` *list* Specify a specific first level mirror to be used in the restore session, or a range or a sequence of first level mirrors to define a replica set from which the integration, according to the replica set rotation, selects one mirror to be used in the restore session If this option is not specified, the MU# 0 is set.

`-quiesce` *cmd* Sets the command/script to be run before links are split. You must create the command/script in the `/opt/omni/lbin` (HP-UX or Solaris systems) or in the *<Data_Protector_home>*`\bin` (Windows systems) directory of the application system. The command/script is used for stopping the application and unmounting file systems that are not to be restored in the active session and to prepare volume groups for deactivation.

`-restart` *cmd* Sets the optional restart application command/script. Create this command/script in the `/opt/omni/lbin` (HP-UX or Solaris) or in the *<Data_Protector_home>*`\bin` (Windows) directory of the application system. The restart application command/script is executed on the application system immediately after the links are resynchronized. It can be used, for example, to restart the application or to mount a filesystem.

`-discovery` This option is possible only for the EMC Symmetrix split mirror restore. The option builds or re-builds the Data Protector Symmetrix database on both the application and backup systems. Its functionality is the same as that of the Data Protector `Syma -init` command. (Refer to the *HP OpenView Storage Data Protector Zero Downtime Backup Administrator's Guide*).

`-re-establish_links_before_restore` This option is used to synchronize split disks, that is, to move data to backup disks. This is necessary to prepare the disks for restore and to enable accurate restores. If the mirrored disks were split before the restore, and only some files need to be restored, then this option can be used to update the backup system. This will ensure that the correct data is resynchronized to the application system. This option is not set by default.

`-disable_disks` Disks on the application system are disabled, that is, the filesystems are dismounted and volume groups are deactivated. This is performed before the split. The disks are enabled after the links are restored. Note that only filesystems selected for restore are dismounted. If other filesystems exist in the volume or disk group, the stop/quiesce application command/script and restart application command/script must be created to dismount these filesystems. You must always select this option for restore when you want to move data from the backup to the application system, that is, to incrementally restore links. The application system disks have to be disabled to provide data integrity after the links are restored, that is, data is moved.

`-restore_links_after_restore` With this option enabled, the HP StorageWorks Disk Array XP agent incrementally restores links for LDEVs that were successfully restored by Data Protector to the backup system. The HP StorageWorks Disk Array XP agent also incrementally re-establishes links for LDEVs that were not successfully restored by Data Protector to the backup system.

DiskArrayXPOptions

`-keep_version` If this option is set, the LDEV pairs involved in the current instant recovery session are split after the restore is finished.

`-check_config` If this option is set, the current volume group configuration of the volume groups involved in the instant recovery session is compared with the volume group configuration during the split mirror backup session kept in the XPDB.

If the volume group configuration has changed since the split mirror backup session, the session is aborted.

When instant recovery is performed in an MC/ServiceGuard cluster to some other node than the one that was backed up, the current volume group configuration on the node to which instant recovery is to be performed is different from the volume group configuration kept in the XPDB. In such a case the XPDB volume group configuration data is replaced by the current volume group configuration data on the node to which instant recovery is to be performed and the session is not aborted.

When performing instant recovery in an MC/ServiceGuard cluster to some other node than the one that was backed up, select this option.

The CRC check information for the selected LDEV stored in the XPDB database is compared to the current CRC check information. If the items compared do not match, the session is aborted.

A RAID Manager/LIB XP flag, which is set whenever the selected mirror LDEV is accessed/changed by any (including non-Data Protector) process is checked. If the flag is set, the session fails with an appropriate warning.

VirtualArrayOptions

`-keep_version` If this option is set, the replica from which the data was restored is left on the disk array after the restore is finished.

`-check_config` If this option is set, the current volume group configuration of the volume groups involved in the instant recovery session is compared with the volume group configuration during the ZDB-to-disk or ZDB-to-disk+tape session kept in the ZDB database (VADB or SMISDB). If the volume group configuration has changed since the ZDB-to-disk or ZDB-to-disk+tape session, the restore is aborted.

When instant recovery is performed in an MC/ServiceGuard cluster to some other node than the one that was backed up, the current volume group configuration on the node to which instant recovery is to be performed is different from the volume group configuration kept in the VADB. In such a case the VADB volume group configuration data is replaced by the current volume group configuration data on the node to which instant recovery is to be performed and the session is not aborted.

When performing instant recovery in an MC/ServiceGuard cluster to some other node than the one that was backed up, select this option.

The CRC check information for the data on the original volumes in the VADB is compared to the CRC check information for the data in the selected replica. If the items compared do not match, the session is aborted.

EnterpriseVirtualArrayOptions

`-check_config` If this option is set, the current volume group configuration of the volume groups involved in the instant recovery session is compared with the volume group configuration during the ZDB-to-disk or ZDB-to-disk+tape session kept in the ZDB database (EVADB or SMISDB). If the volume group configuration has changed since the ZDB-to-disk or ZDB-to-disk+tape session, the restore is aborted.

When instant recovery is performed in an MC/ServiceGuard cluster to some other node than the one that was backed up, the current volume group configuration on the node to which instant recovery is to be performed is different from the volume group configuration kept in the ZDB database. In such a case the ZDB database volume group configuration data is replaced by the current volume group configuration data on the node to which instant recovery is to be performed and the session is not aborted.

When performing instant recovery in an MC/ServiceGuard cluster to some other node than the one that was backed up, select this option.

`-force_prp_replica` If this option is selected and if any target volume to be restored is presented to any system, the EVA agent - HP StorageWorks EVA Agent (legacy) or HP StorageWorks EVA SMI-S Agent - removes these presentations. If the option is not selected and a presentation exists, the instant recovery session is aborted.

RETURN VALUES

See the man page `omniintro` for return values.

Additional return values of the `omnir` command are:

`10` There was an error while restoring some files. All agents completed successfully.

`11` One or more agents failed, or there was a database error.

`12` None of the agents completed the operation.

`13` Session was aborted.

EXAMPLES

The following examples illustrate how the `omnir` command works.

1. To restore trees "/tree1" and "/tree2" of the root filesystem on "fs", with the label "lb1", from the session "1994/07/12-33", as the trees "/tmp/tree1" and "/tmp/tree2", skipping ".xyz" files:

```
omnir -filesystem fs:/ lb1 -session 1994/07/12-33 -tree /tree1 -as
/tmp/tree1 -tree /tree2 -as /tmp/tree2 -skip *.xyz
```

2. To perform a full restore of tree "/ac" on filesystem "bb:/", with no label, from the session "1994/07/12-2":

```
omnir -filesystem bb:/ -full -session 1994/07/12-2 -tree /ac
```

3. To perform restore of the section "/dev/rdsk/c201d6s0" of the disk image labeled "RawRoot" on the client "machine" from the session "1994/07/23-12":

```
omnir -rawdisk machine "RawRoot" -section /dev/rdsk/c201d6s0
-session 1994/07/23-12
```

4. To restore an IDB on the client "server" and pathname "/usr/omni/config" from the session "1994/07/24-2":

```
omnir -omnidb server:/ -session 1994/07/24-2 -tree / -into /tmp/
omnidb
```

5. To use parallel restore for restoring two objects:

```
omnir -filesystem client1:/ -session 1997/04/17-2 -tree /users
-into /tmp -filesystem client2:/opt -session 1997/04/17-3 -tree /
opt -into /tmp
```

6. To perform an instant recovery to the system named “machine” from the backup session “2001/08/08-1”, keeping the replica on the backup system disk:

```
omnir -host machine -session 2001/08/08-1 -instant_restore
-keep_version
```

7. To perform an instant recovery on HP StorageWorks Disk Array XP/HP StorageWorks Virtual Array to the system named “computer” from the backup session “2001/08/08-1”, keeping the replica on the backup system disk:

   ```
   omnir -host computer -session 2001/08/08-1 -instant_restore
   -keep_version
   ```

8. To perform an instant recovery on HP StorageWorks Enterprise Virtual Array to the system named “computer” from the backup session “2003/01/08-1” and to perform volume group configuration check:

   ```
   omnir -host computer -session 2003/01/08-1 -instant_restore
   -check_config
   ```

9. To perform a point in time Lotus recovery of all NTF files in the database named “dbase.nsf” for the Lotus Notes/Domino Server named “BLUE” to the system named “computer”:

   ```
   omnir -lotus -barhost computer -ini c:/lotus/domino/notes.ini
   -domino_server BLUE -db dbase.nsf -NTF -recovery_time 2001/09/
   15.15:00:00
   ```

10. To perform an Informix Server restore of the database server “ol_computer” on the UNIX system “computer” with the bar argument “-r rootdbs”, run:

    ```
    omnir -informix -barhost computer -barcmnd ob2onbar.exe -user
    informix:informix -bararg "-r rootdbs" -appname ol_computer
    ```

11. The Microsoft Information Store with the “/First Storage Group/STORE/Public Folder Store” store and “/First Storage Group/LOGS/Logs” logs is to be restored to the system called “computer.company.com” (where it was backed up), using the Data Protector session with the session ID “2003/07/07-13”. The Microsoft Exchange Server 2000/2003 log files are to be restored to “c:\temp” directory, the hard recovery is to be performed after the restore has finished. The database is to be mounted after the hard recovery. Execute the following command:

    ```
    omnir -msese -barhost computer.company.com -appname "Microsoft
    Exchange Server(Microsoft Information Store)" -base "/First Storage
    Group/LOGS/Logs" -session "2003/07/07-13" -base "/First Storage
    Group/STORE/Public Folder Store" -session "2003/07/07-13" -logpath
    c:\temp -mount -last
    ```

12. To perform a VSS restore of the “Registry Writer” and “System Writer” trees from the backup session “2002/10/20-3” and the “Event Log Writer” tree from the backup session “2002/10/27-1” to the client “daneel” into the “c:\tmp directory”, run:

    ```
    omnir -vss -barhost daneel -session 2002/10/20-3 -tree /"Registry
    Writer" -tree /"System Writer" -session 2002/10/27-1 -tree /"Event
    Log Writer" -into c:\tmp
    ```

13. To start an online restore of a DB2 database called “TEMP” from instance “DB2Inst” on the client “splendid” and rollforward it till the 10th January 2003, 9:15 a.m., execute the following command:

    ```
    omnir -db2 -barhost splendid -instance DB2Inst -dbname TEMP
    -rollforward -time 2003-01-10.09.15.00
    ```

14. To restore the contents of a mailbox called “FIRST” residing on an MS Exchange Server system called “infinity.ipr.hermes”, with the backup session ID 2003/01/10-1, into the new mailbox called “TEMP”, execute the following command:

    ```
    omnir -mbx -barhost infinity.ipr.hermes -mailbox FIRST -session
    2003/01/10-1 -destMailbox TEMP
    ```

15. To restore all messages from the "Inbox" folder (and all subfolders) from the "User 1" mailbox residing on the MS Exchange Server system called "exchange.hp.com", into the original location, using the backup made in the session "2004/03/10-18", without overwriting the messages, run the following command:

    ```
    omnir -mbx -barhost exchange.hp.com -mailbox "User 1" -session
    2004/03/10-18 -folder Inbox -originalfolder -keep_msg
    ```

16. To restore all messages from the "User 2" mailbox residing on the MS Exchange Server system called "exchange.hp.com", except for the messages in the folder "Deleted Items”, into a new location, using the backup made in the session "2004/03/10-19" (for example, performed at 13:47:00), run the following command:

    ```
    omnir -mbx -barhost exchange.hp.com -mailbox "User 2" -session
    2004/03/10-19 -exclude "Deleted Items"
    ```

    The messages will be restored in the "Data Protector 03/10/03 13:47:00" mailbox on the "exchange.hp.com” MS Exchange Server.

17. To start an online restore of a SAP DB database called “TEMP” on the client “splendid” and rollforward it till the 10th January 2003, 9:15 a.m. from the archive logs already residing on the client, execute the following command:

    ```
    omnir -sapdb -barhost splendid -instance TEMP -recover -time:
    2003-01-10.09.15.00 -from_disk
    ```

18. To recover an Oracle9i database on the client “san32” from the session “2004/02/05-18” until the most recent time and open the database after recovery, execute the following command:

    ```
    omnir -host san32 -session 2004/02/05-18 -instant_restore
    -keep_version -oracle -recovery now -open
    ```

19. To perform restore of the section "/dev/rdsk/c201d6s0" of the disk image labeled "Raw" on the client "system1" from the session "1994/07/23-12" using the media set containing the object copy with ID “132123”, execute the following command:

    ```
    omnir -rawdisk system "Raw" -section /dev/rdsk/c201d6s0 -session
    1994/07/23-12 -copyid 132123
    ```

SEE ALSO

omnib(1), omniobjconsolidate(1), omniobjcopy(1)

omnirpt (1)

NAME

`omnirpt` – generates various reports about Data Protector environment

SYNOPSIS

```
omnirpt -version | -help

omnirpt -report ReportName ReportOptions [MethodOptions]
    [FormatOptions] [-multicell] [-header] [-[no]_multiple]

omnirpt -rptgroup ReportGroup


FormatOptions

 -ascii |

 -html |

 -tab |

 -short

MethodOptions

 -email EmailAddress ... |

 -smtp EmailAddress ... |

 -snmp Hostname ... |

 -broadcast Hostname ... |

 -log Filename ... |

 -external CommandName ...

ReportName

 list_sessions |

 session_flow |

 device_flow |

 used_media |

 used_media_extended |
 host_statistics |
 backup_statistics |
 backup_errors |
 dl_trees |
 obj_nobackup |
 obj_lastbackup |
 obj_avesize |
 fs_not_conf |
 dl_info |
 dl_sched |
 db_size |
 db_purge |
 db_purge_preview |
 db_system |
 cell_info |
 hosts_unused |
 dev_unused |
 lookup_sch |
 hosts_not_conf |
 licensing |
 host |
 media_list |
 media_list_extended |
 media_statistics |
 pool_list |
 single_session |
 session_objects |
 session_hosts |
 session_devices |
ReportOptions
 -session SessionID
 -pool Poolname ...
 -label Label
 -location Location ...
 -[no_]library Library ...
 -[no_]protection NoOfDays
 -class MediaClass
 -status MediaStatus
 -datalist BackupSpecificationName ...
 -group BackupSpecificationGroup
 -schedule NoOfdays
 -network Network_IP_Adress...
 -hosts Hostname ...
 -host Hostname
 -level Level
 -timeframe {Start Duration | Day Hour Day Hour}
 -days NoOfdays
 Level: {warning | minor | major | critical}
 Day: [YY]YY/MM/DD
 Hour: HH:MM
```

DESCRIPTION

The `omnirpt` command generates various reports about Data Protector environment: reports about backup sessions in a specific time frame, about backup specifications, media, Data Protector configuration and single session. Each report is defined by its name `-report` *ReportName* and a set of options that specify report parameters (described bellow). The reports are provided in four different formats: ASCII, HTML, tabulator separated format and short ASCII format. Each report is described in two parts: input (what user has to / may specify to configure a report) and output (what is the content of the report). Input items that are enclosed in square brackets ([ ]) are optional, while all others are required. The following *report categories* are available:

Sessions in Timeframe

"Sessions in Timeframe" reports provide reports about backup activities in certain past time period. This time period can either be defined in relative terms (such as last 24 hours) or absolute (03/01/98 00:00 - 04/01/98 00:00). Two other common report options for all "Sessions in Timeframe" reports are backup specification and backup specification group. These two limit the report to selected backup specifications. "Session in Timeframe" reports are:

- List of Backup Sessions (`list_sessions`)
- Session Flow Report (`session_flow`)
- Device Flow Report (`device_flow`)
- Used Media (`used_media`)
- Extended Report on Used Media (`host_statistics`)
- Host Statistics (`backup_statistics`)
- Backup Statistics (`backup_errors`)
- Backup Errors (`used_media_extended`)

Backup Specifications

"Backup Specifications" reports provide different configuration reports - all of which base on backup specifications. By default, all backup specifications are used, but user may choose to limit a report either to certain backup specification or to specific backup specification group (or event both together). "Backup Specifications" reports are:

- Trees in Backup Specification (`dl_trees`)
- Objects Without Backup (`obj_nobackup`)
- Object's Latest Backup (`obj_lastbackup`)
- Average Object Size (`obj_avesize`)
- Not Configured Filesystems (`fs_not_conf`)
- Backup Specification Information (`dl_info`)
- Backup Specification Schedule (`dl_sched`)

Internal Database

"Internal Database" reports provide information about Data Protector internal database (IDB) and about Data Protector client systems dynamics. "Internal Database" reports are:

- Internal Database Size Report (`db_size`)
- Internal Database Purge Report (`db_purge`)
- Internal Database Purge Preview Report (`db_purge_preview`)
- System Dynamics Report (`db_system`)

Configuration

"Configuration" reports provide various reports about Data Protector environment. "Configuration" reports are:

- Cell Information (`cell_info`)
- Configured Hosts not Used by Data Protector (`hosts_unused`)
- Configured Devices not Used by Data Protector (`dev_unused`)
- Look up Schedule (`lookup_sched`)
- Hosts not Configured for Data Protector (`hosts_not_conf`)
- Licensing report (`licensing`)
- Host Backup Report (`host`)

Pools and Media

"Pools and Media" reports provide four reports that search through Data Protector pools for media that match the search criteria. The default is to list all media or pools and each report option can then limit the search to a certain set of media. "Pools and Media" reports are:

- Extended List of Media (`media_list_extended`)
- List of Media (`pool_list`)
- Media Statistics (`media_statistics`)
- List of Pools (`media_list`)

Single Session

"Single session" reports provide various information about single Data Protector backup session. These reports are mostly used as End of Session notification. In this case, Data Protector will use the session ID of the current session (the one that generated the End of Session event) to create the appropriate report. "Single session" reports are:

- Single Session Report (`single_session`)
- Session Objects (`session_objects`)
- Session Hosts (`session_hosts`)
- Device Error Report (TBD)
- Statistics about backup status for the Session Devices (`session_devices`)
- Session Media (`session_media`)

OPTIONS

`-header` This option is not used for the reports that have no required or optional report options. If this option is set, the output of the report will display report options too. If it is not set, only the output of the report is displayed.

`-multicell` This option is only used with Manager-of-Managers. If this option is specified, the report will be generated for all Cell Managers configured in the MoM environment (multi-cell report).

`-[no_]multiple` This option is only used for enterprise reports (multi-cell) and for Session per Client reports. If this option is specified, the report will be divided into sections. For enterprise reports the report will be divided by Cell Manager and for Session per Client reports it will be divided by client.

Report Names

`list_sessions` Lists all sessions in the specified time frame. The report is defined by set of options that specify report parameters.

Report options are:

`-timeframe` {*Start Duration* | *Day Hour Day Hour*}

[`-datalist` *BackupSpecificationName*]

[`-group` *BackupSpecificationGroup*]

`session_flow` Graphically presents duration of each session specified in certain timeframe. Flow chart of the backup sessions matching search criteria is shown.

Report options are:

`-timeframe` {*Start Duration* | *Day Hour Day Hour*}

[`-datalist` *BackupSpecificationName*]

[`-group` *BackupSpecificationGroup*]

`device_flow` Graphically presents usage of each device. Flow chart of the backup sessions matching search criteria is shown.

Report options are:

`-timeframe` {*Start Duration* | *Day Hour Day Hour*}

[`-datalist` *BackupSpecificationName*]

[`-group` *BackupSpecificationGroup*]

`used_media` Lists media that have been used by backup sessions in the specific timeframe.

Report options are:

`-timeframe` {*Start Duration* | *Day Hour Day Hour*}

[`-datalist` *BackupSpecificationName*]

[`-group` *BackupSpecificationGroup*]

`used_media_extended` Provides extended information on media that have been used by backup sessions in the specific timeframe.

Report options are:

`-timeframe` {*Start Duration* | *Day Hour Day Hour*}

[`-datalist` *BackupSpecificationName*]

[`-group` *BackupSpecificationGroup*]

`host_statistics` Lists of clients and their backup status - only clients that were used by the backup sessions matching the search criteria are displayed.

Additionally, clients can be limited also with hosts report option.

Report options are:

`-timeframe` {*Start Duration* | *Day Hour Day Hour*}

[-datalist *BackupSpecificationName*]

[-group *BackupSpecificationGroup*]

[-hosts]

`backup_statistics` Shows statistics about backup status in the selected time frame, limited to backup sessions matching the search criteria.

Report options are:

-timeframe {*Start Duration* | *Day Hour Day Hour*}

[-datalist *BackupSpecificationName*]

[-group *BackupSpecificationGroup*]

`backup_errors` Shows list of messages that occur during backup sessions in the specified time frame for selected backup specifications. The messages are grouped by clients (for all selected clients).

Report options are:

-timeframe {*Start Duration* | *Day Hour Day Hour*}

[-datalist *BackupSpecificationName*]

[-group *BackupSpecificationGroup*]

[-hosts *Hostname* ...]

[-level *Level*]

`dl_trees` Lists all trees in the specified backup specification. It also shows names of drives and the name of a tree.

Report options are:

[-datalist *BackupSpecificationName*]

[-group *BackupSpecificationGroup*]

`obj_nobackup` Lists all objects, specified for backup in selected backup specifications, which do not have a valid backup. A valid backup means successfully completed backup, which protection has not expired. For each object that does not have a valid protected full backup, the following items are shown: backup specification, an object type, an object name and a description. Only objects from the selected backup specification are used for the report. If HOST object is used: Host object is expanded (get disks) and report checks that expanded objects are in database. UNIX, NetWare, and Windows filesystems are supported. This option is not available for backup specifications for integrations.

Report options are:

[`-datalist` *BackupSpecificationName*]

[`-group` *BackupSpecificationGroup*]

`obj_lastbackup` Lists all objects, specified for backup in selected backup specifications, which have a valid backup. A valid backup means successfully completed backup, which protection has not expired. For each object the information about last full and last incremental backup is displayed. The output is a set of records formed by object type, object name and description, last full backup time and last incremental backup time. If HOST object is used: Host object is expanded (get disks) and report checks that expanded objects are in database. UNIX, NetWare, and Windows filesystems are supported.

Report options are:

[`-datalist` *BackupSpecificationName*]

[`-group` *BackupSpecificationGroup*]

`obj_avesize` Lists all objects, specified for backup in selected backup specifications, which have a valid backup. A valid backup means successfully completed backup, which protection has not expired. For each object average full and average incremental backup size is displayed. If HOST object is used: Host object is expanded (get disks) and report checks that expanded objects are in database. UNIX, NetWare, and Windows filesystems are supported.

Report options are:

[`-datalist` *BackupSpecificationName*]

[`-group` *BackupSpecificationGroup*]

`fs_not_conf` Displays a list of mounted filesystems which are not in selected backup specifications. Output is a list of filesystems. If HOST object is used, the report will not report any disk from client as not configured (assuming that HOST backup will backup all disks). If HOST object is used, the report will not report any disk from client as not configured (assuming that HOST backup will backup all disks).

Report options are:

[-datalist *BackupSpecificationName*]

[-group *BackupSpecificationGroup*]

`dl_info` Shows information about all selected backup specification (backup specification name, type, group, owner and pre & post exec commands). HOST does not influence report.

Report options are:

[-datalist *BackupSpecificationName*]

[-group *BackupSpecificationGroup*]

`dl_sched` Creates a list of all selected backup specifications and their next schedule time. The output is a list of backup specifications and schedule dates. HOST does not influence report.

Report options are:

[-datalist *BackupSpecificationName*]

[-group *BackupSpecificationGroup*]

`db_size` Lists a table that contains information about the MMDB, CDB, IDB extension files, statistics for DCBF, SMBF and SIBF, and low IDB disk space. The `Used` columns in this report show the percentage of used records. This figure is calculated as current amount of used records divided by the amount of records allocated in the IDB in percents. Thus, this figure may substantially vary, since Data Protector automatically allocates new records whenever this figure reaches 100 percent (whenever all currently allocated records are used). To find out whether certain parts of the IDB are running out of space, you can additionally configure the IDB Space Low or IDB Tablespace Space Low notification, or check the last part of this report - it lists the low disk space information when allocated records for any of the involved disks are running out of space. There are no report options for this report.

`db_purge` Lists all purged sessions (from `purge.log` file) with the following information: start time, end time, duration, inactivity time and number of file names, file versions and sessions purged. There are no report options for this report.

`db_purge_preview` Lists the following information: overall number of filenames in database (in thousands), estimated number of obsolete filenames in database (in thousands) and estimated duration of database purge (in seconds). There are no report options for this report.

`db_system` Lists the following information about each Data Protector client in the cell: the number of filenames (in thousands) in the Data Protector internal database (IDB), the number of active filenames (in thousands) in the IDB, the IDB filenames growing ratio (new filenames per day), the number of deleted filenames in the IDB per day, active growth per year, and dynamics indicator (medium/high/low/critical). The filenames that are not active are filenames of the backed up files in the IDB that have no associated file versions in the IDB. The active growth per year is calculated in two ways: If there is no IDB purge session recorded in the IDB, the active growth per year is calculated on the basis of data in last 11 days and then extrapolated to one year. If there is an IDB purge session recorded in the IDB, the active growth per year is calculated on the basis data in the time span since the last IDB purge session and then extrapolated to one year. There are no report options for this report.

`cell_info` Data Protector cell related information (number of clients, Backup specifications, Media Management server, Licensing server).

`hosts_unused` List of configured clients that are not used for backup and do not have any device configured.

`dev_unused` List of configured devices that are not used for backup in any backup specification.

`lookup_sch` List of backup specifications that are scheduled for backup in the next n number of days (where n is the number of days specified by user).

Report option is:

NumberOfDays

`hosts_not_conf` List of clients in selected domain(s) that are not configured for Data Protector. Note that Data Protector will display also routers and other machines that have IP address in selected domain.

Report option is:

`-network` *Network_IP_Address...*

`licensing` List all licenses with their total and available amount

`host` Report output is all end-user backup related information about specific client: list of filesystems not configured for selected clients, list of all objects configured in backup specifications for the selected client, list of all objects with a valid backup for specified client with times and average sizes.

Report option is:

HostName

`media_list` List of all media matching the search criteria. The following information is provided for each medium: ID, label, location, status, protection, used and total MB, time when media was last used, pool and media class.

Report options are:

[`-label` *Label*]

[`-location` *Location*]

[`-pool` *PoolName*]

[`-class` *MediaClass*]

[`-status` *MediaStatus*]

[`-[no_]protection` *NoOfDays*]

[`-timeframe` {*Start Duration* | *Day Hour Day Hour*}]

[`-[no_]library` *Library*]

`media_list_extended` List of all media matching the search criteria. The following information is provided for each medium: ID, label, location, status, protection, used and total MB, time when media was last used, pool and media class, the backup specifications that have used this medium during the backup.

Report options are:

[`-label` *Label*]

[`-location` *Location*]

[`-pool` *Pool*Name]

[`-class` *MediaClass*]

[`-status` *MediaStatus*]

[-[no_]protection *NoOfDays*]

[-timeframe {*Start Duration* | *Day Hour Day Hour*}]

[-[no_]library *Library*]

[-datalist *BackupSpecificationName*]

[-group *BackupSpecificationGroup*]

`media_statistics` Reports the statistics on the media matching the search criteria. The following information is provided: number of media; number of scratch media; number of protected, good, fair and poor media; number of appendable media; and total, used, and free space on media.

Report options are:

[-label *Label*]

[-location *Location*]

[-pool *PoolName*]

[-class *MediaClass*]

[-status *MediaStatus*]

[-[no_]protection *NoOfDays*]

[-timeframe {*Start Duration* | *Day Hour Day Hour*}]

[-[no_]library *Library*]

`pool_list` Lists all pools matching a specified search criteria. For each pool the following information is provided: pool name, description, media type, total number of media, number of full and appendable media containing protected data, number of free media containing no protected data, number of poor, fair and good media.

Report options are:

[-pool *PoolName*]

[-location *Location*]

[-class *MediaClass*]

[-[no_]library *Library*]

[-timeframe {*Start Duration* | *Day Hour Day Hour*}] `single_session` Report displays all relevant information about single Data Protector backup.

Report option is:

`-session` *`SessionID`*

`session_objects` Returns all information about all backup objects that took part in a selected session.

Report option is:

`-session` *`SessionID`*

`session_hosts` Provides information for each client that took part in the selected session: statistics about backup status for the client, list of objects and their related information for the client, error messages for the client.

All information is grouped for each client separately. Using the `-multiple` option, this report can be split into smaller reports, one for each client (see section Notifications for details).

Report option is:

`-session` *`SessionID`*

`session_devices` Provides information about all devices that took part in a selected session.

Report option is:

`-session` *`SessionID`*

`session_media` Provides Information about all media that took part in a selected session.

Report option is:

`-session` *`SessionID`*

Method options

`-email` *`EmailAddress`* Sends the report to the specified *`EmailAddress`*.

On Windows, you need a configured MAPI profile. You can either use an existing mail profile or create a new one, named `Omniback`. To use an existing profile, edit the omnirc variable `OB2_MAPIPROFILE`.

On UNIX, sendmail is used for sending the e-mails.

`-smtp` *`EmailAddress`* The recommended option for sending reports by e-mail. Sends the report to the specified *`EmailAddress`* using the SMTP protocol.

By default, the SMTP server address is set to the Cell Manager address. To change the SMTP server, edit the variable `SMTPServer` in the global options file. The server must be accessible from the Cell Manager system, but does not need to be part of the Data Protector cell.

`-snmp` *`Hostname`* Report is send as an SNMP (Simple Network Mailing Protocol) trap.

`-broadcast` *`Hostname`* Report is broadcasted to the selected machine. Note: Only Windows machines can be specified as broadcast destination.

`-log` *`Filename`* Report is saved in to the log file specified with *`Filename`*.

`-external` *`CommandName`* Specifies a script which receives the report. Optionally the script can than parse the report and forward it to user configured recipient. Usually, TAB report format is used in combination with `-external` option.

`Report options`

`-rptgroup` *`ReportGroup`* This option executes the specified *`ReportGroup`*.

`-session` *`SessionID`* This option is used to specify the Session ID report option.

`-pool` *`Poolname`* This option is used to specify the media pool name report option.

`-label` *`Label`* This option is used to specify the medium label report option

`-location` *`Location`* This option is used to specify the medium location report option.

`-[no_]library` *`Library`*... This option is used to specify the library report option. If it is set to `-no_library`, all libraries in the cell are selected for the report.

`-[no_]protection` *`NoOfDays`* This option is used to specify the protection report option. The number of days in which the protection will expire can be specified. If it is set to `no_protection`, all media in the cell will be selected for the report.

`-class` *`MediaClass`* This option is used to specify the media class report option.

`-status` *`MediaStatus`* This option is used to specify the media status report option. It can have one of the following values: `poor`, `fair`, or `good`.

`-datalist` *`BackupSpecificationName`* This option is used to specify the backup specification report option. Note that you can specify more than one backup specification. In such case, separate the backup specification names with spaces.

`-group` *`BackupSpecificationGroup`* This option is used to specify backup specification group report option.

`-schedule` *`NoOfDays`* This option is used to specify report option, which defines the number of days for which to display the schedule information.

`-network` *`Network_IP_Adress`* This option specifies one or more network IP addresses. Network IP address is represented by the first three numbers in the IP address, for example 123.44.5. You can specify more that one network IP address using spaces in between.

`-days` *`NoOfDays`* The report will filter objects that have been backed up recently. Specify the number of days.

`-host` *`Hostname`* Select the client system for which you want to create the report.

`-hosts` *`Hostname`* Select the client systems for which you want to create the report.

`-level` *`Level`* Select the level of warnings that should be included in the report. The levels are `warning`, `minor`, `major`, and `critical`.

`-timeframe` *`Start Duration`* This option is used to specify a relative time frame report option. It is useful for recurrent reports, for example you can use `-timeframe 24 24` each day to set the time frame to last 24 hours.

`-timeframe` *`Day Hour Day Hour`* This option is used to specify an absolute time frame report option.

`Report Formats`

`-ascii` Specifies report format: ASCII

`-html` Specifies report format: HTML

`-tab` Specifies report format: TAB

`-short` Specifies report format: SHORT

EXAMPLES

1. To list all backup sessions that have started in the last 24 hours and display the report in the default ASCII format, run:

   ```
   omnirpt -report list_sessions -timeframe 24 24
   ```

2. To list all objects from session “1998/09/09-1” in tabulator separated format, which is useful for additional parsing or can be used with other tools for analysis, run:

   ```
   omnirpt -report session_objects -session 1998/09/09-1 -tab
   ```

3. To list all media of class DLT with location string "COMPANY", for which protection will expire in the next 5 days, run:

   ```
   omnirpt -report media_list -protection 5 -class DLT -location
   COMPANY
   ```

   This report can be used as a base for the vaulting process, as it can list you media that need to be taken to the vault.

4. To send "Internal Database Size Report" in HTML format to the user "name@domain.com" using the SMTP protocol, run:

   ```
   omnirpt -report db_size -html -smtp name@domain.com
   ```

5. To execute the report group named "MyReportGroup", run:

   ```
   omnirpt -rptgroup MyReportGroup
   ```

SEE ALSO

omnitrig(1M), omnihealthcheck(1M)

omnistat (1)

NAME

omnistat – displays the status of Data Protector sessions

SYNOPSIS

```
omnistat -version | -help

omnistat -session SessionID [-status_only | -monitor | -detail]

omnistat [-user Username] [-mount] [-error] [-detail]

omnistat -previous [-user Username] [[-since Date] [-until Date]
    ] | [-last Number]   [-failed]

Date

 [YY]YY/MM/DD
```

DESCRIPTION

The omnistat command displays information on active sessions. You can view all active sessions (default) or only details of a specific session. An active session is referenced by its *SessionID*.

OPTIONS

`-version` Displays the version of the omnistat command.

`-help` Displays the usage synopsis for the omnistat command.

`-session` *SessionID* Displays detailed information on the single active session identified by this *SessionID*.

`-monitor` omnistat connects to the specified active session and starts monitoring the progress of the session.

`-status_only` Displays only the overall status of the active session.

`-detail` Displays detailed information about all current sessions.

`-user` *Username* Displays information on active sessions belonging to the specified user.

`-failed` Displays information on sessions containing data objects that failed due to errors.

`-error` Displays information on active sessions with the status "In Progress (errors)"

`-mount` Displays all active sessions with mount requests pending.

`-previous` Lists all sessions from the Data Protector internal database (IDB).

`-since` *`Date`* Lists all sessions since the specified *`Date`*.

`-until` *`Date`* Lists all sessions until the specified *`Date`*.

`-last` *`n`* Lists all sessions within the last *`n`* days.

EXAMPLES

The following examples illustrate how some options of the `omnistat` command work.

1. To view sessions that are currently active and have mount requests pending:

```
omnistat -mount
```

2. To see detailed information for the session with the SessionID "1995/04/24-32". The SessionID can be specified in two different ways:

```
omnistat -detail -session 1995/04/24-32
```

```
omnistat -detail -sess 32
```

3. To see an overview of the sessions that occurred in last 3 days and were run by user root:

```
omnistat -previous -user root -last 3
```

4. To see information regarding the sessions that occurred within the last 3 days and had objects that have failed:

```
omnistat -previous -last 3 -failed
```

5. To see only the status of session with this SessionID:

```
omnistat -status_only -session 2
```

6. To monitor the session with the SessionID "R-1995/05/13-8":

```
omnistat -session R-8 -monitor
```

SEE ALSO

omniabort(1)

omniupload (1)

NAME

omniupload – uploads information about a backup device to the Data Protector internal database (IDB)

SYNOPSIS

```
omniupload -version | -help

omniupload -create_device FileName

omniupload -modify_device BackupDevice [-file FileName]

omniupload -remove_device BackupDevice

omniupload -create_library FileName

omniupload -modify_library Library [-file FileName]

omniupload -remove_library Library
```

DESCRIPTION

Uploads a backup device file to the Data Protector internal database (IDB).

Information on Data Protector backup devices is stored in the IDB. To configure a backup device, information on this device must be downloaded into a file. This is done using the `omnidownload` command. The file is then modified and uploaded back to the IDB.

OPTIONS

`-version` Displays the version of the `omniupload` command.

`-help` Displays the usage synopsis for the `omniupload` command.

`-create_device` *`FileName`* Specifies the ASCII file containing the information about the backup device. This option is used to create a new backup device. If `-` is specified as *`FileName`* then data is read from stdin.

`-modify_device` *`BackupDevice`* Uses the information in the uploaded file to modify an existing backup device in the IDB. If no filename is specified using the `-file` option the command searches the current directory for a file with the same name as the *`BackupDevice`*. Note that the media class may not be changed.

`-file` *`FileName`* Specifies the ASCII file that will be parsed for information about the backup device (library). This option is used to modify an existing backup device (library). If `-` is specified as *`FileName`* then data is read from stdin.

`-remove_device` *`BackupDevice`* Removes information about the *`BackupDevice`* from the IDB.

`-create_library` *`FileName`* Specifies the ASCII file containing the information about the library. This option is used to create a new library. If `-` is specified as *`FileName`* then data is read from stdin.

`-modify_library` *`Library`* Uses the information in the uploaded file to modify an existing library in the IDB. If no filename is specified using, the `-file` option the command searches the current directory for a file with the same name as the *`Library`*. Note that the media class may not be changed.

`-remove_library` *`Library`* Removes information about the *`Library`* from the IDB.

EXAMPLES

The following examples illustrate how the `omniupload` command works.

1. To create a backup device using the information in the file "/tmp/Device":

```
omniupload -create_device /tmp/Device
```

2. To modify Library "Exabyte1" using the information in the file "/tmp/EXA":

```
omniupload -modify_library Exabyte1 -file /tmp/EXA
```

3. To remove backup device "Stacker":

```
omniupload -remove_device Stacker
```

SEE ALSO

omniamo(1), omnidownload(1), omnimcopy(1), omniminit(1), omnimlist(1), omnimm(1), omnimnt(1), omnimver(1), sanconf(1M), uma(1M)

omniusers (1)

NAME

omniusers – adds or removes the Data Protector users to or from an existing Data Protector user group, or lists the configured Data Protector users. It thus enables you to use all GUI functionality on those Cell Manager platforms on which the GUI is not supported, using the GUI installed on another system.

SYNOPSIS

```
omniusers -version | -help

omniusers -add -type {U | W} -usergroup DPUserGroup -name
    UserName -group GroupOrDomainName -client ClientName [-desc
    Description]

omniusers -remove -name UserName -group GroupOrDomainName -client
    ClientName

omniusers -list
```

DESCRIPTION

The command adds, removes or lists the configured Data Protector users on the Cell Manager where it is run. It does not create or remove user groups.

Use the command to create a remote Data Protector user on those Cell Manager platforms on which Data Protector GUI is *not* supported. You can then use the created user account to start the Data Protector GUI on another system with the Data Protector GUI installed, and connect to the Cell Manager.

OPTIONS

`-version` Displays the version of the `omniusers` command.

`-help` Displays the usage synopsis for the `omniusers` command.

`-add` Adds a user to the specified Data Protector user group.

`-remove` Removes a user from its Data Protector user group.

`-name` *`UserName`* Specifies username of the user to be added/removed. By specifying asterisk (`*`) as the username, all users from the specified group (on UNIX systems) or domain (on Windows systems) will be granted/revoked access from the specified clients to the Cell

Manager. `*` corresponds to `<Any>` in the Data Protector GUI. Note that in some shells, backslash and asterisk (`\*`) must be used instead of `*`.

Note that UNIX usernames and usernames of the configured Data Protector users are case sensitive.

`-type {U|W}` Specifies the user type: a UNIX user (`U`) or a Windows user (`W`).

`-group` *`GroupOrDomainName`* A group (on UNIX systems) or a domain (on Windows systems) the specified user belongs to. By specifying asterisk (`*`) as the group or domain name, the specified user will be granted/revoked access from any group (on UNIX systems) or domain (on Windows systems) from the specified clients. `*` corresponds to `<Any>` in the Data Protector GUI. Note that in some shells, backslash and asterisk (`\*`) must be used instead of `*`.

`-client` *`ClientName`* Specifies the name of the client system from where the specified user will have access to the Cell Manager. By specifying asterisk (`*`) as the client name, the specified user will be granted/ revoked access to the Cell Manager from any Data Protector client system. `*` corresponds to `<Any>` in the Data Protector GUI. Note that in some shells, backslash and asterisk (`\*`) must be used instead of `*`.

If this option is used with the `-remove` option, *`ClientName`* must contain the fully qualified domain name (FQDN) of the client system.

`-usergroup` *`DPUserGroup`* Specifies the Data Protector user group the user(s) will be added to.

`-desc` *`Description`* Specifies the description for the added user(s).

`-list` Lists users in all configured Data Protector user groups in the cell. For each configured Data Protector user the username, UNIX group or Windows domain, fully qualified domain name (FQDN) of the client system from which the user has granted access, and the user description are displayed. Asterisk (`*`) corresponds to the `<Any>` string in the Data Protector GUI.

RETURN VALUES

The return values of the `omniusers` command are:

`0` The command operation completed successfully.

`1`	A generic error occurred.
`2`	The operation for adding or removing a user failed.
`4`	Error parsing options.

NOTE

The command is supported only on those Cell Manager platforms on which the Data Protector GUI is not supported. Refer to the *HP OpenView Storage Data Protector Software Release Notes* for details on supported operating systems.

EXAMPLES

The following examples illustrate how the `omniusers` command works.

1. To add the Windows user “win_user” from the domain “domain1” to the Data Protector “admin” user group and allow access only from the client system “client.company.com”, run the following command:

   ```
   omniusers -add -type W -name win_user -usergroup admin -group
   domain1 -client client.company.com
   ```

2. To add the UNIX user “root” from the “sys” group to the Data Protector “admin” user group and allow access only from the client system “client.company.com”, run the following command:

   ```
   omniusers -add -type U -name root -usergroup admin -group sys
   -client client.company.com
   ```

3. To add the UNIX user “root” to the Data Protector “admin” user group and allow access from any UNIX group but only from the system “client.company.com”, run the following command:

   ```
   omniusers -add -type U -name root -usergroup admin -group \* -client
   client.company.com
   ```

4. To display the Data Protector users in all configured Data Protector user groups, run the following command:

   ```
   omniusers -list
   ```

SEE ALSO

omnimigrate(1M), ob2install(1M), omnigui(5), omnisetup.sh(1M), upgrade_cfg_from_evaa(1M)

syb_tool (1)

NAME

syb_tool – a utility used to get ISQL command needed to restore a Sybase database that was backed up by Data Protector.

SYNOPSIS

```
syb_tool dbname servername
 -date YYYY/MM/DD.hh:mm:ss
 [-new_db dbname]
 [-new_server servername]
 [-file filename]
 [-media]
```

DESCRIPTION

The `syb_tool` is used to get the data needed for restore of Sybase databases.

OPTIONS

dbname The name of Sybase database.

servername The name of Sybase database server on which the backup was performed.

`-date` *YYYY/MM/DD.hh:mm:ss* The date until which your database will be restored. `syb_tool` will find the first backup done after this date.

`-new_db` *dbname* The name of the database that you want to restore to.

`-new_server` *servername* The name of the server that you want to restore to.

`-file` *filename* The name of the file where the ISQL statement needed for restore of desired database will be written to. The ISQL command can be started with the option `-i`, followed by the name of the file.

`-media` This option returns the list of all media needed for restore.

EXAMPLES

1. To get the ISQL statement needed for the restore of the last backup of the database named “database1” on the Sybase Adaptive Server named “server”, enter the following command:

   ```
   syb_tool database1 server -date
   ```

2. To get the ISQL statement needed for the restore of the database named “database1” on the Sybase Adaptive Server named “server”, using the first backup performed after midday of July 07. 1999, enter the following command:

   ```
   syb_tool database1 server -date 1999/07/07.12:00:00
   ```

3. To get the ISQL statement needed for the restore of the database named “database1” on the Sybase Adaptive Server named “server”, using the first backup performed after midday of July 07. 1999 and restoring it as “database_one” on the Sybase server called “server_one”, enter the following command:

   ```
   syb_tool database1 server -date 1999/07/07.12:00:00 -new_db
   database_one -new_server server_one
   ```

4. To get the ISQL statement needed for the restore of the last backup performed for database named “database1” on the Sybase Adaptive Server named “server”, saving the ISQL statement to file “/tmp/stat.isql”, and getting the list of media IDs needed for restore, enter the following command:

   ```
   syb_tool database1 server -date -file /tmp/stat.isql -media
   ```

   To start the restore, start the ISQL command, specifying the input file “/tmp/stat.isql” in the following way:

   ```
   isql -Usa -P -Sserver -i /tmp/stat.isql
   ```

Section 1M: Administrative Commands

NNMpost.ovpl (1M)

NAME

NNMpost.ovpl – A script with no arguments that resumes the eight processes paused by NNMpre.ovpl

SYNOPSIS

```
NNMpost.ovpl
```

DESCRIPTION

A script with no arguments that resumes the eight processes paused by `NNMpre.ovpl`.

SEE ALSO

NNMpre.ovpl(1M), NNMScript.exe(1M)

NNMpre.ovpl (1M)

NAME

NNMpre.ovpl – start NNM embedded database backup

SYNOPSIS

```
NNMpre.ovpl
```

DESCRIPTION

The `NNMpre.ovpl` script starts the NNM embedded database backup. The embedded database makes a direct copy of itself to a location specified in the `solid.ini` file. The script also pauses eight NNM processes.

SEE ALSO

NNMpost.ovpl(1M), NNMScript.exe(1M)

NNMScript.exe (1M)

NAME

NNMScript.exe – finds the location of the NNM Perl compiler and the NNMpre.ovpl and NNMpost.ovpl scripts and starts the two scripts

SYNOPSIS

```
NNMScript.exe -pre | -post
```

DESCRIPTION

The `NNMScript.exe` finds the location of the NNM Perl compiler and the `NNMpre.ovpl` and `NNMpost.ovpl` scripts (because the NNM Perl compiler is used to run the scripts and its path must be supplied to Windows on the command line). The directory location is found via the registry and the location of the compiler and scripts are relative to this location. `NNMScript.exe` also starts the scripts. An argument specifies the script to run.

OPTIONS

`-version`	Displays the version of the `NNMScript.exe` command.
`-help`	Displays the usage synopsis for the `NNMScript.exe` command.
`-pre`	Starts the `NNMpre.ovpl` script.
`-post`	Starts the `NNMpost.ovpl` script.

SEE ALSO

NNMpost.ovpl(1M), NNMpre.ovpl(1M)

ob2install (1M)

NAME

ob2install – Runs remote client installation from the selected Installation Server. This command is supported on UNIX operating systems only. It is not supported on Windows operating systems.

SYNOPSIS

```
ob2install -version | -help
```

```
ob2install -server installation_server -input filename
```

DESCRIPTION

The `ob2install` command is used for running remote client installation from Cell Manager. You have to select an Installation Server from which you want to run remote client installation.

OPTIONS

`-version` Displays the version of the `ob2install` command.

`-help` Displays the usage synopsis for the `ob2install` command.

`-server` *`installation_server`* Specifies Installation Server used for the installation session. The Installation Server must belong to local cell.

`-input` *`filename`* Specifies the existing input file containing prepared client installation data. Each client is described in the input file with a newline-separated ASCII string in format described below.

INPUT FILE FORMAT

`-host` *`Hostname -component version`* `-push_inst` *`Push_Inst_Parameters`* Specifies the host on which the remote installation will be performed. The hostname must be enclosed in double quotes.

`-host` *`Hostname`* Specifies the host on which the remote installation will be performed. The hostname must be enclosed in double quotes.

-component version Selects the components for the installation. The *version* argument specifies the version of the product. For the A.06.00 release, for example: `-oracle8 A.06.00`. For the specified host only the Data Protector supported components should be selected. Data Protector Supported components are:

`cc` installs the User Interface software component

`da` installs the Disk Agent software component

`ma` installs the General Media Agent software component

`sap` installs the SAP R/3 Integration software component

`sapdb` installs the SAP DB Integration software component

`omnist` installs the OmniStorage Integration software component

`sybase` installs the Sybase Integration software component

`informix` installs the Informix Integration software component

`mssql` installs the MS SQL 6.5 Integration software component

`msese` installs the MS Exchange 2000/2003 Integration software component

`emc` installs the EMC Symmetrix Agent software component

`oracle8` installs the Oracle Integration software component

`momgui` installs the MoM User Interface software component

`ov` installs the HP OpenView NNM Backup Integration software component

`ndmp` installs the NDMP Media Agent Integration software component

`mssql70` installs the MS SQL 7.0/2000 Integration software component

`ssea` installs the HP StorageWorks XP Agent software component

`snapa` installs the HP StorageWorks VA Agent software component

`evaa` installs the HP StorageWorks EVA Agent (legacy) software component

`smisa` installs the HP StorageWorks EVA SMI-S Agent software component

`lotus` installs the Lotus Integration software component `db2` installs the IBM DB2 UDB Integration software component

`clus` installs the MS Cluster Server Integration software component

`-push_inst` *`Push_Inst_Parameters`* This option specifies all parameters that are crucial for successful remote client installation. Option must be used with all its parameters. All parameters except `general_inst_type` and `inst_type` must be specified in double quotation marks.

`Push_Inst_Parameters`

`inst_path` Specifies path for installation - Data Protector home directory. This parameter must be in double quotes. For UNIX remote client installation this path is ignored and placeholder `-` can be used.

`username` Specifies username that is used during the remote client installation. On UNIX, if not specified, default value "root" is used. Must be enclosed in double quotes.

`passwd` This parameter specifies password that is used while Installation Server connects to the client. If a password is not determined (`-` can be used instead), it must be given later during installation. It must be enclosed in double quotes.

`cell_server` Specifies the name of client's Cell Manager. It must be enclosed in double quotes.

`general_inst_type` General installation type can have two values: `1` or `2`. `2` specifies the Client Installation. Due to future extensions, `1` will be used for Cell Manager Installation. For Data Protector A.06.00 release only the Client Installation is supported.

`inst_type` Specifies installation type:

`1` new_install

`2` update

`3` delete

`4` check installation

`8` This has to be specified.

NOTES

The command can only be used locally on the Cell Manager.

EXAMPLES

The following example illustrates how the `ob2install` command works.

Remote client installation from the Installation Server "machine1.company.com" to client “machine2.company.com” belonging to the Cell Manager "machine3.company.com". Selected components are: Disk Agent, General Media Agent and Cell Console

```
ob2install -server machine1 -in infile
```

where “infile” contains the following line:

```
ob2install -host "machine2" -cc A.06.00 -da A.06.00 -ma A.06.00
-push_inst "-" "-" "-" "machine3.company.com" 8 1
```

SEE ALSO

omnigui(5), omnisetup.sh(1M), omnimigrate.pl(1M), omniusers(1), upgrade_cfg_from_evaa(1M)

omnicheck (1M)

NAME

omnicheck – Performs DNS connections check within a Data Protector cell and lists Data Protector patches installed on Data Protector clients.

SYNOPSIS

```
omnicheck -version | -help

omnicheck -dns [-host Client | -full] [-verbose]

omnicheck -patches -host Client
```

DESCRIPTION

The following tasks can be performed using the `omnicheck` command:

CHECKING DNS CONNECTIONS WITHIN A Data Protector CELL

To check DNS connections within a Data Protector cell, use the `-dns` option with the `omnicheck` command.

The `omnicheck` command does not verify DNS connections in general. It verifies that DNS information matches over all communications relevant for Data Protector among Data Protector cell members. The command reports only failed checks and the total number of failed checks unless the `-verbose` option is specified.

It is possible to verify the following DNS connections in the Data Protector cell, using the `omnicheck` command:

- To check that the Cell Manager and every Media Agent resolve DNS connections to every Data Protector client in the same cell properly and vice versa, use the `-dns` option.
- To check that a particular Data Protector client resolves DNS connections to every Data Protector client in the same cell properly and vice versa, use the `-host` option.
- To check all possible DNS connections in the cell, when every client resolves DNS connections to all other clients in the same cell, use the `-full` option.

LISTING PATCHES INSTALLED ON Data Protector CLIENTS

The `omnicheck` command can be used to list Data Protector patches installed on a particular client. The `omnicheck` option used to list Data Protector patches installed on a particular client is `-patches`.

OPTIONS

`-version` Displays the version of the `omnicheck` command

`-help` Displays the usage synopsis of the `omnicheck` command.

`-dns` Checks that the Cell Manager and every Media Agent resolve DNS connections to every Data Protector client in the same cell properly and vice versa. This option performs the same as running `omnicheck -dns -host` *`cell_manager`* and `omnicheck -dns -host` *`media_agent_1`* ... `omnicheck -dns -host` *`media_agent_n`* commands.

`-dns -host` *`Client`* Checks that a Data Protector client specified by the `-host` option resolves DNS connections to every Data Protector client in the same cell properly and vice versa.

`-dns -full` Checks all possible DNS connections in the cell. Every client in the cell tries to resolve all other clients in the same cell.

`-verbose` Returns all the messages when using the `-dns` option. If this option is not set (default), only the messages that are the result of failed checks are returned.

`-patches -host` *`Client`* Returns Data Protector patches (patch level, patch description and number of all patches installed) installed on a Data Protector client specified by the `-host` option. To use this option, you need the `Client configuration` user right (by default only users in the `admin` user group).

RETURN VALUES

See the man page `omniintro` for return values.

Additional return values of the `omnicheck` command used to check the DNS connections are:

```
client_1 cannot connect to client_2
```

```
client_1 connects to client_2, but connected system presents itself as
client_3
```

```
client_1 failed to connect to client_2
```

```
checking connection between client_1 and client_2
```

```
all checks completed successfully.
```

```
number_of_failed_checks checks failed.
```

`*client* is not a member of the cell.`

`*client* contacted, but is apparently an older version. Hostname is not checked.`

Additional return values of the `omnicheck` command used to list the Data Protector patches are:

`List of patches found on host *client*`

`Patch level   Patch description`

`Number of patches found: *number_of_patches*`

`List of patches on host *client* is not available.`

`Host *client* is not a member of this cell.`

`Host *client* is unreachable.`

NOTES

The `omnicheck` command can be used only within one Data Protector cell and on Data Protector clients that have Data Protector A.05.10 or later installed.

EXAMPLES

1. To check DNS connections needed for normal Data Protector operating (the Cell Manager and every Media Agent in the cell resolve DNS connections to every Data Protector client in the cell properly and vice versa), execute the following command:

   ```
   omnicheck -dns
   ```

2. To check if the client with the hostname `backup.system.com` resolves DNS connections to every Data Protector client in the same cell properly and vice versa and to get all relevant messages, execute the following command:

   ```
   omnicheck -dns -host backup.system.com -verbose
   ```

3. To list the patches installed on client with the hostname `backup.system.com`, execute the following command:

   ```
   omnicheck -patches -host backup.system.com
   ```

SEE ALSO

omnicc(1), omnicellinfo(1), omnisv(1M), omnidlc(1M), omniinstlic(1M)

omnidbcheck (1M)

NAME

omnidbcheck – checks the consistency of the Data Protector internal database (IDB)

SYNOPSIS

```
omnidbcheck -version | -help

omnidbcheck [-quick | -extended]

omnidbcheck -core [-summary]

omnidbcheck -filenames [-summary]

omnidbcheck -bf [-summary]

omnidbcheck -sibf [-detail | dumpmedia] [-summary]

omnidbcheck -smbf [-detail | dumpmessages] [-summary]

omnidbcheck -dc [LimitScope] [-detail | -dumpmedia] [-summary]

Limitscope

 -hosts host1 [host2... ] | -media medium1 [medium2 ...] | -mpos
    min-max
```

DESCRIPTION

The Data Protector internal database (IDB) consists of: 1) Media Management Database (MMDB), 2) Catalog Database (CDB), 3) Detail Catalog Binary Files (DCBF), 4) Session Messages Binary Files (SMBF) and 5) Serverless Integrations Binary Files (SIBF). The MMDB and CDB objects, object versions and media positions form the core part of the IDB. The CDB filenames, DCBF, SMBF and SIBF form the detail part of the IDB.

The `omnidbcheck` command checks the status of the IDB or only of parts of IDB. The command sends a report to the standard output.

Note that errors found during the core check are Critical, errors found during the filenames check are Major, errors found during the dc and bf checks are Minor, errors found during the SIMBF check are Minor and errors found during the SMBF check are Warning.

Data Protector creates a log file for each part of the check in the *<Data_Protector_home>*\log\server (Windows Cell Manager) and in the /var/opt/omni/server/log (UNIX Cell Manager) directory on:

Check_bf.txt

Check_core.txt

Check_filenames.txt

Check_dc.txt

Check_smbf.txt

Check_sibf.txt

There is a timestamp at the beginning of each log file stating when the check was performed.

OPTIONS

-version Displays the version of the omnidbcheck command.

-help Displays the usage synopsis for the omnidbcheck command.

-quick Checks the core, CDB filenames and presence and size of DCBF parts of the IDB and displays the summary of the check by executing the omnidbcheck -core -filenames -bf -summary command.

-extended Checks the entire IDB with the exception of the SMBF and displays the summary of the check by executing the omnidbcheck -core -filenames -bf -dc -sibf command. Full check of the consistency of the database, including detail file information is performed.

-core Performs the core check of the IDB - it checks MMDB and CDB objects, object versions and media positions. The core check takes approximately 5 - 10 minutes.

-filenames Performs the check of the CDB filenames. It takes approximately one hour for each GB of the filename tablespace.

-bf Performs the presence and size check of the DCBF. This check takes approximately 10 - 30 seconds.

-sibf Checks if the SIBF are present and if they can be read. This check takes approximately 10 minutes for each GB of the SIBF part.

`-smbf` Checks the presence of the SMBF. This check takes approximately 5 - 10 minutes.

Note that if you have removed a SMBF in any way (e.g. using Data Protector GUI or CLI or deleted the file manually), then this option will report the removed session message as missing. This does not mean that IDB is corrupted - it only indicates that a session has been removed.

`-dc` Checks the consistency between the Core part and DC part of the IDB. This check takes approximately 10 minutes for each GB of the DC part of the IDB.

`-detail` Lists all SIBF, SMBF or DCBF and their status (OK or corrupted/missing). If the `-detail` option is not specified (default) for the `-dc` option, all DCBF are listed, but status (corrupted) is displayed only with the corrupted DCBF. If the `-detail` option is not specified (default) for the `-smbf` or `-sibf` option, only the corrupted (SIBF) or missing (SMBF) binary files and their status (corrupted or missing) are listed.

`-dumpmedia` If this option is specified with the `-sibf` option, it sends the SIBF filenames, object versions information, offset of the data in the SIBF file belonging to an object version and size of the data in the SIBF file belonging to an object version to the standard output. If this option is specified with the `-dc` option, it sends the complete information stored in the DCBF to the standard output.

`-dumpmessages` This option is used with the `-smbf` option. It sends the session messages in the SMBF to the standard output.

`-summary` Displays only the summary of the check (OK or failed/missing). The option does not impact the thoroughness of the check.

`LimitScope` It is also possible to limit the scope of the DC check to either a set of media or a set of clients:

`-hosts` *`host1`* `[`*`host2`*`...]` Only Detail Catalogs for the specified clients are checked.

`-media` *`medium1`* `[`*`medium2`*`...]`

`-mpos` *`min-max`* Only those media positions (mpos) are checked that are located in DCBF directories with the specified media position (between *`min`* and *`max`*).

NOTES

The command can only be used locally on the Cell Manager.

EXAMPLES

1. To check the DC part of the IDB for the Data Protector client named “machine.company.com”, execute the following command:

   ```
   omnidbcheck -dc -hosts machine.company.com
   ```

2. To perform the extended check of the IDB, execute the following command:

   ```
   omnidbcheck -extended
   ```

SEE ALSO

omnidbrestore(1M), omnidb(1), omnidbinit(1M), omnidbupgrade(1M), omnidbutil(1M), omnidbxp(1), omnidbva(1), omnidbeva(1), omnidbsmis(1)

omnidbinit (1M)

NAME

omnidbinit – initializes the Data Protector internal database (IDB)

SYNOPSIS

```
omnidbinit -version | -help
omnidbinit [-force]
```

DESCRIPTION

The `omnidbinit` command initializes the Data Protector internal database (IDB). All information about sessions, media and objects is lost after the initialization. The command does not delete IDB transaction logs. The command creates a gap in the sequence of IDB transaction logs; when a roll forward operation is performed using the `omnidbrestore` command, the operation applies only the transaction logs created before the initialization of the IDB.

The IDB directory structure has to exist in order to initialize the IDB successfully. You can re-create the IDB directory structure by copying it from the `/opt/omni/newconfig/` (HP-UX or Solaris Cell Manager) or from the *`<Data_Protector_home>`*`\NewConfig\` (Windows Cell Manager) directory.

OPTIONS

`-version` Displays the version of the `omnidbinit` command

`-help` Displays the usage synopsis for the `omnidbinit` command

`-force` Overrides the default safety check for the initialization. By default, the command displays a confirmation request. With this option, there is no confirmation request.

NOTES

The command can only be used locally on the Cell Manager.

SEE ALSO

omnidb(1), omnidbcheck(1M), omnidbupgrade(1M), omnidbrestore(1M), omnidbutil(1M), omnidbxp(1), omnidbva(1), omnidbeva(1), omnidbsmis(1)

omnidbrestore (1M)

NAME

omnidbrestore – performs the restore of the Data Protector internal database (IDB)

SYNOPSIS

```
omnidbrestore -version | -help

omnidbrestore -autorecover [Autorecover Options] [General
    Options]

omnidbrestore -read OptionFile [General Options]

omnidbrestore RMA_Options VRDA_Options MediaOptions [General
    Options]
```

RMA_Options (Restore Media Agent options)

```
 -mahost DeviceHostname

 -policy LogicalDevicePolicy

 -type LogicalDeviceType

 -dev PhysicalDevice

 [-name DeviceName]

 [-description DeviceDescription]

 [-blksize BlkSize]

 [-ioctl RoboticsDevice]

 [-remhost RoboticsHostname]
```

VRDA_Options (Volume Restore Disk Agent options)

```
 -daid DAID

 [-overwrite | -no_overwrite]
```

Media Options

```
 -maid mediumID1 [mediumID2 ...]

 -slot slot1[:flip1] [slot2[:flip2] ...]

 -position segment1:offset1 [segment2:offset2 ...]
```

General Options

-verbose

-tree *path1* [*path2* ...]

-preview

-skiprestore

Autorecover Options

-session *sessionID*

-save *OptionFile*

-logview

-optview

-replay_only

-firstlog *FirstTransactionLog*

DESCRIPTION

The `omnidbrestore` command is used to restore the Data Protector internal database (IDB) without using the IDB, as opposed to the `omnir` - `omnidb` command which uses the IDB to retrieve the information needed for the IDB restore. If the IDB was installed on symbolic links, these symbolic links have to be created as they existed before running the `omnidbrestore` command.

The IDB restore using the `omnidbrestore` command consists of four phases: 1) Stopping the Data Protector services/daemons (with the exception of the Data Protector Inet service on Windows) 2) Restore of the IDB files. 3) Roll forward of IDB transactions (if present) stored in the IDB transaction log(s) - a process called dbreplay. Before the dbreplay is started, you are given the possibility to skip this phase by responding to a prompt. 4) Starting the Data Protector services/daemons.

Every time the backup of the IDB is started or when running `omnidbinit` or `omnidbcheck` commands or when the size of a transaction log reaches 2MB, a transaction log is created in the *<Data_Protector_home>*`\db40\logfiles\syslog\` (Windows Cell Manager) or in the `/var/opt/omni/server/db40/logfiles/syslog/` (UNIX Cell Manager) directory. Depending on the value of the *Archiving* parameter in the `rdmserver.ini` file, located in *<Data_Protector_home>*`\db40\datafiles\catalog` (Windows Cell Manager) or in the `/var/opt/omni/server/db40/datafiles/catalog` (UNIX Cell Manager), the old transaction log is copied (the *Archiving* parameter is set to 1) or deleted (the *Archiving* parameter is set to 0). In the latter case the dbreplay process is not always possible.

The omnidbrestore command can operate in three modes:

THE AUTORECOVER MODE

The autorecover mode is invoked using the -autorecover option. The omnidbrestore command in the autorecover mode scans the obrindex.dat file for the *Media Options*, *RMA Options* (Restore Media Agent options) and *VRDA options* (Volume Restore Disk Agent options) and arguments needed for the restore. When the options and arguments are retrieved, the restore of the IDB is performed using the retrieved options and arguments to the original location overwriting the current files.

The obrindex.dat file resides on the Cell Manager in the <*Data_Protector_home*>\db40\logfiles\rlog (Windows Cell Manager) or in the /var/opt/omni/server/db40/logfiles/rlog (UNIX Cell Manager) directory. The obrindex.dat file is written to at every backup of the IDB and contains the *Media Options*, *RMA Options* and *VRDA options* and arguments needed for the restore of the IDB and the name of the transaction log created at the IDB backup time. You can create a copy of the obrindex.dat file by setting the *RecoveryIndexDir* parameter in the Data Protector global options file to point to a directory where you want to have a copy of the obrindex.dat file. If the obrindex.dat file is missing or is corrupted, the omnidbrestore command will use its copy if the *RecoveryIndexDir* parameter points to the directory with the copy. The Data Protector global options file (global) resides in the <*Data_Protector_home*>\Config\server\Options (Windows Cell Manager) or in the /etc/opt/omni/server/options (UNIX Cell Manager) directory.

THE READ MODE

The read mode is invoked using the -read option. Omnidbrestore reads the options and arguments from the file that has been created manually or using the -autorecover -save *OptionFile* option. This is useful in case the restore devices are different from the backup devices (or attached to a different system). In such a case the *OptionFile* has to be manually updated with the appropriate restore device data before the restore is started.

THE MANUAL MODE

The manual mode is used if the obrindex.dat file is not available and you have to specify all the needed *Media Options*, *RMA Options* and *VRDA options* and arguments manually.

OPTIONS

`-version` Displays the version of the `omnidbrestore` command

`-help` Displays the usage synopsis of the `omnidbrestore` command.

`-autorecover [`*`Autorecover Options`*`] [`*`General Options`*`]` Starts the restore of the IDB in the autorecover mode. The `omnidbrestore` command in the autorecover mode scans the `obrindex.dat` file for the *`Media Options`*, *`RMA Options`* and *`VRDA options`* and arguments needed for the restore. When the options and arguments are retrieved, the restore of the IDB is performed using the retrieved options and arguments to the original location overwriting the current files.

`-read` *`OptionFile`* `[`*`General Options`*`]` Starts the restore of the IDB in the read mode. `Omnidbrestore` reads the options and arguments from the file that has been created manually or using the `-autorecover -save` *`OptionFile`* command. This is useful in case the restore devices are different from the backup devices (or attached to a different system). In such a case the *`OptionFile`* has to be manually updated with the appropriate restore device data before the restore is started.

`RMA_Options`

`-mahost` *`DeviceHostname`* Specifies the client with the attached backup device.

`-policy` *`LogicalDevicePolicy`* Specifies the backup device policy ID. Policy can be defined as `1` (Standalone), `3` (Stacker), `5` (Jukebox), `6` (external control), `8` (Grau DAS exchanger library), `9` (STK Silo medium library) or `10` (SCSI Library).

`-type` *`DeviceType`* Specifies the media type. Media type numbers are specified in the *HP OpenView Storage Data Protector Software Release Notes*.

`-dev` *`PhysicalDevice`* Physical device path.

`-name` *`LogicalDeviceName`* Specifies the backup device name. Note that this is only used in output messages and can be omitted.

`-description` *`DeviceDescription`* Specifies the backup device description. Note that this is only used in output messages and can be omitted.

`-blksize` *`BlkSize`* Specifies the block size that was used when the backup was made.

`-ioctl` *`RoboticsDevice`* Physical path of the library device.

`-remhost` *`RoboticsHostname`* Use the `-remhost` option to specify the client with the attached library device, if the library device is connected to a system other than *`mahost`* (Media Agent host)

`VRDA_Options`

`-daid` *`DAID`* Disk Agent ID of the database backup.

`-overwrite | -no_overwrite` By default, or if the `-overwrite` option is specified, the already existent files on the disk are overwritten by the restored files. If the `-no_overwrite` option is specified, only the files that do not exist on the disk are restored.

`Media Options`

`-maid` *`mediumID1`* `[`*`mediumID2`*`...]` Lists media IDs needed for the restore.

`-slot` *`slot1`*`[:`*`flip1`*`] [`*`slot2`*`[:`*`flip2`*`]...]` Lists the slots where media are located. Note that the sequence has to match the sequence in the list created using the `-maid` option.

`-position` *`segment1`*`:`*`offset1`* `[`*`segment2`*`:`*`offset2`*`...]` Lists media positions of the database backup. Note that the sequence has to match the sequence in the list created using the `-maid` option.

`General Options`

`-verbose` By default, the MA and DA messages are not displayed. If this option is specified they are displayed.

`-tree` *`path1`* `[`*`path2`*`...]` Specifies the IDB directories and their subordinate files and subdirectories to be restored. If this option is not specified, all IDB directories are restored.

`-preview` Runs the restore preview.

`-skiprestore` Does not start the actual restore. This option should only be used in combination with the `-save`, `-optview` or `-logview` option.

`Autorecover Options`

`-session` *`sessionID`* Instead of selecting the last valid backup of the database, the backup from the specified session is selected. Note that the specified session must exist in the `obrindex.dat` file.

`-save` *`OptionFile`* Saves the options and arguments generated by the `-autorecover` option in the specified file in order to run the `omnidbrestore` in the read mode later.

`-logview` Displays the contents of the `obrindex.dat` file. The `obrindex.dat` file resides on the Cell Manager in the *<Data_Protector_home>*`\db40\logfiles\rlog` (Windows Cell Manager) or in the `/var/opt/omni/server/db40/logfiles/rlog` (UNIX Cell Manager) directory.

`-optview` Displays the restore job options.

`-replay_only` If this option is specified, only the roll forward of the transactions made to the IDB is performed. The transaction log to start the roll forward operation with is read from the `obrindex.dat` file or specified by the `-firstlog` option. The IDB files are not restored.

`-firstlog` *FirstTransactionLog* This option specifies the first transaction log to start the roll forward of the transactions to the IDB with. It is to be used only in combination with the `-replay_only` option. Note that this option can be used only if the archiving of the transaction logs is enabled by setting the *Archiving* parameter in the `rdmserver.ini` file to `1`. The `rdmserver.ini` file resides on the Cell Manager in the *<Data_Protector_home>*`\db40\datafiles\catalog` (Windows Cell Manager) or in the `/var/opt/omni/server/db40/datafiles/catalog` (UNIX Cell Manager) directory.

NOTES

The command can only be used locally on the Cell Manager. Note that the `omnidbrestore` command stops (before the restore) and restarts (after the restore) all Data Protector services on UNIX and all services except the Data Protector Inet service on Windows. This command does not stop the Data Protector services running in a cluster.

EXAMPLES

The following example illustrates how the `omnidbrestore` command works.

1. To start the restore of the IDB in the autorecover mode, execute the following command:

   ```
   omnidbrestore -autorecover
   ```

2. The SCSI backup device with DLT media is connected to the client machine.company.com (with a Media Agent installed on the client) with SCSI address scsi2:0:0:0C, while the robotics device is connected to the client machine2.company.com with SCSI address scsi2:0:0:1. The media ID of the

medium needed is 3203110a:3acda75a:0690:0001, the medium is in the 1:-1 slot, position of the IDB backup on the medium is 1:0 and DA ID is 986556451. To restore IDB using the above data use the following command:

```
omnidbrestore -policy 10 -type 10 -mahost machine.company.com -dev
scsi2:0:0:0C -daid 986556451 -remhost machine2.company.com -ioctl
scsi2:0:0:1 -maid 3203110a:3acda75a:0690:0001 -slot 1:-1 -position
1:0
```

3. To start the restore of the IDB in the autorecover mode using the backed up files from a specific session, execute the following command:

   ```
   omnidbrestore -autorecover -session 2001/04/12-1
   ```

SEE ALSO

omnidb(1), omnidbcheck(1M), omnidbinit(1M), omnidbupgrade(1M), omnidbutil(1M), omnir(1), omnidbxp(1), omnidbva(1), omnidbeva(1), omnidbsmis(1)

omnidbupgrade (1M)

NAME

omnidbupgrade – Converts filenames in the IDB to the new internal character encoding used in Data Protector A.06.00 and thus enables the correct handling of non-ASCII characters in filenames in the Data Protector GUI.

SYNOPSIS

```
omnidbupgrade -version | -help

omnidbupgrade -fname -udp

omnidbupgrade -fname -estimate
```

DESCRIPTION

`Omnidbupgrade` converts filenames in the IDB to the new character encoding introduced in Data Protector A.06.00. The conversion can be performed only on Windows Cell Manager for all non-Windows clients containing filenames with non-ASCII characters. The command will convert filenames for all clients not marked as already converted from the old character encoding to the new one.

The IDB conversion does not affect backup and restore. If conversion of data for a specific client is running and at the same time backup of the same client is started, no filenames or directories will be logged to the IDB for this client (as if `log none` option in GUI was used for backup).

Back up the IDB before running `omnidbupgrade`.

OPTIONS

`-version` Displays the version of the `omnidbupgrade` command

`-help` Displays the usage synopsis for the `omnidbupgrade` command.

`-fname -udp` Converts filenames in the IDB.

`-fname -estimate` Estimates the time needed for the conversion. This option is possible only before IDB conversion has been performed.

NOTES

Backup will not log files or directories for the client that is being converted while his backup is running. The command is supported only on Windows Cell Manager. It is recommended to back up the IDB before converting it using `omnidbupgrade`.

SEE ALSO

omnidbrestore(1M), omnidb(1), omnidbinit(1M), omnidbutil(1M), omnidbcheck(1M), omnidbxp(1), omnidbva(1), omnidbeva(1), omnidbsmis(1)

omnidbutil (1M)

NAME

omnidbutil – handles various Data Protector internal database (IDB) maintenance tasks

SYNOPSIS

```
omnidbutil -help

omnidbutil -version

omnidbutil -list_dcdirs

omnidbutil -add_dcdir Pathname [-maxsize Size_MB] [-maxfiles
    NumberOfFiles] [-spacelow Size_MB] [-seq Number]

omnidbutil -modify_dcdir Pathname [-maxsize Size_MB] [-maxfiles
    NumberOfFiles] [-spacelow Size_MB] [-seq Number]

omnidbutil -remove_dcdir Pathname

omnidbutil -remap_dcdir

omnidbutil -fixmpos

omnidbutil -readdb [-mmdb Directory] [-cdb Directory]
    [-no_detail] [-check_overs]

omnidbutil -writedb [-mmdb Directory] [-cdb Directory]
    [-no_detail]

omnidbutil -show_locked_devs

omnidbutil -free_locked_devs [devname | mediumId | cartName
    phyLocation | serial_ldev | wwn_lun]

omnidbutil -mergemmdb Cell_Server_Hostname

omnidbutil -cdbsync Cell_Server_Hostname

omnidbutil -changebdev FromDev ToDev [-session SessionID]

omnidbutil -extendfnames Pathname -maxsize Size_MB

omnidbutil -extendtblspace Tablespace Pathname -maxsize Size_MB

omnidbutil -extendinfo

omnidbutil -purge {-filenames [host_1 [host_n ...]] | -sessions
```

```
    [NumberOfDays] | -days [NumberOfDays] | -messages
    [NumberOfDays] | -dcbf | -daily}

omnidbutil -purge_stop

omnidbutil -info

omnidbutil -clear

omnidbutil -change_cell_name [old_host]

omnidbutil -show_cell_name

omnidbutil -set_session_counter new_session_ID

omnidbutil -upgrade_info

omnidbutil -show_db_files

omnidbutil -free_pool_update
```

DESCRIPTION

The `omnidbutil` command is used for Data Protector internal database (IDB) maintenance tasks. These tasks involve:

OPERATIONS ON DETAIL CATALOG BINARY FILES (DCBF)

The Detail Catalog (DC) is composed of three parts: 1) Catalog Database (CDB) tablespace containing pathnames of backed up files together with client system names. 2) The Detail Catalog Binary Files (DCBF) part, which stores file version information (file size, modification time, attributes/protection, exact position on a medium (block level) and so on). 3) DCBF directories: registered directories that contain DCBF. A DCBF directory is allocated when creating new DCBF using one of three possible allocation algorithms, specified in the Data Protector global options file by the *`DCDirAllocation`* parameter. The Data Protector global options file resides in `/etc/opt/omni/server/options` on a UNIX Cell Manager or in *`<Data_Protector_home>`*`\config\server\options` on a Windows Cell Manager.

Operations on DCBF include: 1) Registering, removing and updating DCBF directories. 2) Locating DCBF across DCBF directories if they had been moved manually. 3) Removing invalid references to DCBF. Invalid references can occur after the DB recovery during which the replay of IDB transaction logs is executed. In this case CDB is newer than DCBF.

The `omnidbutil` options used for operations on DC are: `-list_dcdirs`, `-add_dcdir`, `-modify_dcdir`, `-remove_dcdir`, `-remap_dcdir` and `-fixmpos`.

EXPORTING AND RE-CREATING THE CONTENTS OF THE MEDIA MANAGEMENT DATABASE (MMDB) AND CDB

The contents of MMDB and CDB can be exported to and re-created from text files. Text files are ASCII on UNIX and UNICODE on Windows.

The omnidbutil options used for exporting and re-creating the contents of MMDB and CDB are: `-readdb` and `-writedb`.

LISTING AND UNLOCKING BACKUP DEVICES, TARGET VOLUMES, MEDIA AND LIBRARY SLOTS

Backup devices, target volumes, media and library slots in use are locked during backup and restore. In certain situations (backup or restore session crashes), devices remain locked, even though the MA, SSEA, SNAPA, EVAA or SMISA is not running. By default, such devices are unlocked after 60 min. The user can list all locked and unlock devices, target volumes, media and library slots.

The `omnidbutil` options used for listing and unlocking backup devices, target volumes, media and library slots are: `-show_locked_devs` and `-free_locked_devs`.

MERGING LOCAL MMDB INTO CENTRALIZED MEDIA MANAGEMENT DATABASE (CMMDB)

In larger multi-cell environments with high-end backup devices, you may want to share these devices and media among several cells. This can be achieved by having one centralized MMDB database for all the cells and keeping an individual CDB database for each cell. This allows media and device sharing while preserving the security capabilities of the multi-cell structure. To achieve this, the local MMDB must be merged into the CMMDB.

The `omnidbutil` option used for merging MMDB into CMMDB is `-mergemmdb`.

SYNCHRONIZING CDB AND MMDB

In certain situations, CDB and MMDB may be out of sync (different readdb of CDB and MMDB, restore of CMMDB while leaving local CDB intact, etc.). In this case both DBs must be synchronized.

The `omnidbutil` option used for synchronizing CDB and MMDB is `-cdbsync`.

MISCELLANEOUS TASKS

These tasks involve operations such as extending tablespaces, purging the obsolete pathnames from the CDB, displaying the information about the IDB and the IDB upgrade, changing references in object versions form one device to some other device, changing the owner of the CDB to the current Cell Manager, displaying the CDB owner and more.

The omnidbutil options used for this group of tasks are: -changebdev, -extendfnames, -extendtblspace, -extendinfo, -purge, -purge_stop, -info, -clear, -change_cell_name, -show_cell_name, -set_session_counter, -upgrade_info, -show_db_files and free_pool_update.

Certain options require exclusive access to the IDB. Prior to using such options, ensure that no backup, restore or media management sessions are in progress and that no graphical user interfaces are running in the cell.

OPTIONS

-version Displays the version of the omnidbxp command

-help Displays the usage synopsis of the omnidbutil command.

-list_dcdirs Lists all registered DCBF directories.

-add_dcdir *Pathname* [-maxsize *Size_MB*] [-maxfiles *NumberOfFiles*] [-spacelow *Size_MB*] [-seq *Number*]

Adds (registers) a new DC directory in the directory specified by this option.

The -maxsize option specifies the amount of disk space that can be used for DCBF in this directory. When the specified size is reached, new DCBF will not be created in this DCBF directory any more. If this option is not specified, then the default size of 4096 MB is used.

The -maxfiles option specifies the number of DCBF that can be stored in the directory. When the specified number is reached, new DCBF will not be created in this DCBF directory any more. If this option is not specified, then the default value of 500 files is used. Only values under 10000 are valid.

The -spacelow option defines the actual free disk space needed for a DCBF binary file to be created. When the free space falls below the specified free disk space, new DCBF will not be created in this DCBF directory any more. If this option is not specified, the default of 100 MB is used.

The -seq option sets the sequence number for the new DCBF directory. Each DCBF directory has a certain position which determines when DCBF will be created in the DCBF directory. The first DCBF directory to be used for DCBF has the lowest sequence number. The order of the DCBF directories to be used is determined by the sequence number. Sequence is used only if the *DCDirAllocation* parameter in the Data Protector global options file is set to 0. The Data Protector global options file resides in /etc/opt/omni/server/options on a UNIX Cell Manager or in *<Data_Protector_home>*\config\server\options on a Windows Cell Manager. If the -seq option is not specified, 0 will be used.

-modify_dcdir *Pathname* [-maxsize Size_MB] [-maxfiles *NumberOfFiles*] [-spacelow *Size_MB*] [-seq *Number*]

Modifies a DCBF directory under the specified path.

The -maxsize option modifies the amount of disk space that can be used for DCBF in this directory. When the modified size is reached, new DCBF will not be created in this DCBF directory any more. If this option is not specified, then the default size of 4096 MB is used.

The -maxfiles option modifies the number of DCBF that can be stored in the directory. When the modified number is reached, new DCBF will not be created in this DCBF directory any more. If this option is not specified, then the default value of 500 files is used. Only values under 10000 are valid.

The -spacelow option modifies the actual free disk space needed for a DCBF binary file to be created. When the free space falls below the modified free disk space, new DCBF will not be created in this DCBF directory any more. If this option is not specified, the default of 100 MB is used.

The -seq option modifies the sequence of a DCBF directory. Each DCBF directory has a certain position which determines when DCBF will be created in the DCBF directory. Sequence is used only if the *DCDirAllocation* parameter in the Data Protector global options file is set to 0. The Data Protector global options file resides in /etc/opt/omni/server/options on a UNIX Cell Manager or in *<Data_Protector_home>*\config\server\options on a Windows Cell Manager. If the -seq option is not specified, 0 will be used.

-remove_dcdir *Pathname* Removes (unregisters) the given DCBF directory. The directory must not hold any DCBF and will not be removed.

`-remap_dcdir` Locates DCBF across all DCBF directories and updates DCBF locations in the IDB if they had been moved manually (using the `mv` command or similar) between DCBF directories. This makes the IDB aware of the locations of each DCBF. This option requires exclusive access to the database.

`-fixmpos` Removes invalid references to DCBF. This option should be used in the case of IDB recovery (after tablespaces `dbreplay` or `-import_logs`) or after a DCBF has been manually removed. This option requires exclusive access to the database.

`-readdb [-mmdb` *Directory*`] [-cdb` *Directory*`] [-no_detail]`
`[-check_overs]` Reads the files in the specified directories and uses this information to rebuild the IDB. As a prerequisite, the files must have been created using the `-writedb` option, and a copy of the DCBF, SMBF and SIBF directories must have been created. The `-mmdb` option specifies a directory to use for the MMDB. The `-cdb` option specifies a directory for the CDB. Only the database for which you specify the directory is imported. Move the copy of the DCBF, SMBF and SIBF directories to their position in the IDB directory structure.

DCBF default location:

On Windows: *<Data_Protector_home>*`\db40\dcbf`

On UNIX: `/var/opt/omni/server/db40/dcbf`

SMBF default location:

On Windows: *<Data_Protector_home>*`\db40\msg`

On UNIX: `/var/opt/omni/server/db40/msg`

SIBF default location:

On Windows: *<Data_Protector_home>*`\db40\meta`

On UNIX: `/var/opt/omni/server/db40/meta`

Use the `-no_detail` option to skip the recovery of references to DCBF, SMBF and SIBF. If the recovery of these references is skipped, the copy of DCBF, SMBF and SIBF directories is not needed.

Use the `-check_overs` option to check if object version details are correct. Note that this operation can be very time consuming. Error details are saved in

<Data_Protector_home>\log\server\readascii.log on the Windows Cell Manager or in /var/opt/omni/server/log/readascii.log on the UNIX Cell Manager.

-writedb [-mmdb *Directory*] [-cdb *Directory*] [-no_detail] Writes the IDB tablespaces (without the DCBF, SMBF and SIBF) to files in the specified directories. The -mmdb option specifies a directory to use for the MMDB and the -cdb specifies a directory for the CDB. Only the database for which you specify the directory is exported. During the operation, when in prompt mode, manually copy the DCBF, SMBF and SIBF directories to a safe location since the IDB is in consistent state at that moment. To determine which directories to copy, run the omnidbutil -list_dcdirs command. Use the -no_detail option to skip the writing of references to DCBF, SMBF and SIBF to files. If these references are skipped, the copy of DCBF, SMBF and SIBF directories is not needed.

-show_locked_devs Lists all locked devices, target volumes, media and slots.

-free_locked_devs [*devname* | *mediumId* | *cartName phyLocation* |

serial_ldev | wwn_lun] Unlocks a specified device, target volume, medium or slot, where *devname* is the device, *serial_ldev* is the target volume where *serial* is the HP StorageWorks Disk Array XP serial number and *ldev* is the HP StorageWorks Disk Array XP volume number, *wwn_lun* is the target volume where *wwn* is the HP StorageWorks (Enterprise) Virtual Array world-wide-name and *lun* is the logical unit number (LUN), *mediumId* is the medium, *cartName* is the library name and *phyLocation* is the number of the slot to be unlocked. If none of the above is specified, all devices, target volumes, media and slots will be unlocked.

-mergemmdb *Cell_Server_Hostname* Merges the local MMDB from the remote Cell Manager *Cell_Server_Hostname* to the CMMDB. For this action there must exist a MoM cell and a remote cell with a local MMDB If the command reports no errors you can disable the local MMDB on the remote Cell Manager by removing (and possibly copying to a safe place) the /var/opt/omni/server/db40/datafiles/mmdb (UNIX Cell Manager) or the *<Data_Protector_home>*\db40\datafiles\mmdb (Windows Cell Manager) directory. All duplicated items (stores, media pools, devices) will have "_N" appended to their name, where N represents the number of the duplicate (starting with 1). Note that once the database is merged you will not be able to revert the operation.

`-cdbsync` *`Cell_Server_Hostname`* Synchronizes the MMDB and the CDB on the specified Cell Manager. The MMDB and CDB may be out of sync when: 1) The MMDB and CDB contain information from different periods in time. This may be the result of the importing (the `-readdb` option) the CDB and the MMDB from files that were the result of separate export (the `-writedb` option) sessions. 2) In a MoM environment, when the local CDB and centralized MMDB are out of sync. This may be the result of the centralized IDB restore.

The command must be executed on the system with the MMDB (one Cell Manager in the cell) or with the centralized MMDB (MoM environment) installed.

In a MoM environment, if the centralized MMDB was changed (as a result of IDB restore or import), the command should be run for each Cell Manager in this MoM cell by specifying each Cell Manager in the cell as the *`Cell_Server_hostname`* argument.

`-purge {-filenames [`*`host_1`*` [`*`host_n`*`...]] | -sessions [`*`NumberOfDays`*`] | -days [`*`NumberOfDays`*`] | -messages [`*`NumberOfDays`*`] | -dcbf | -daily}`

This option allows you to remove obsolete file names, backup, restore, and media management sessions, session messages, and obsolete DCBF files from the IDB.

The `-filenames` option removes all obsolete file names (file names without any file versions) for a specific or all clients from the CDB. This option requires exclusive access to the database.

The `-sessions` option removes media management sessions, restore sessions, and obsolete backup sessions (backup sessions without backed up data) older than *`NumberOfDays`*.

The `-days` option removes media management sessions, restore sessions, and obsolete backup sessions (backup sessions without backed up data) older than *`NumberOfDays`*.

The `-messages` option removes session messages for all sessions older than *`NumberOfDays`*.

The `-dcbf` option removes DCBF for all media with expired catalog protection.

The `-daily` option starts the same purge session as started every day at 12.00 (depending on the Data Protector global options file setting) and is a part of Data Protector daily maintenance tasks. This purge session deletes DCBF based on the catalog protection and removes obsolete sessions and their messages, by running the `omnidbutil -purge -sessions` *`KeepObsoleteSessions`* `-messages` *`KeepMessages`* `-dcbf` command, where *`KeepObsoleteSessions`* and *`KeepMessages`* are specified in the Data Protector global options file. Default values for these two parameters are 30 and 0, respectively. The Data Protector global options file resides in `/etc/opt/omni/server/options` on a UNIX Cell Manager or in *`<Data_Protector_home>`*`\config\server\options` on a Windows Cell Manager. The scheduled time for the `-daily` option to start every day is defined by the *`DailyMaintenanceTime`* option in the Data Protector global options file.

At least one of these options must be specified. You can change *`DailyMaintenanceTime`* for the `-daily` option in the global options file.

`-purge_stop` Use this option to stop a running file name purge session. This command only sends a stop request to the Purge Session Manager. The response may not be immediate.

`-extendfnames` *`Pathname`* `-maxsize` *`Size_MB`* Creates additional extent (tablespace). The directory specified by this option must exist and be capable of holding a tablespace of the size specified by `-maxsize` parameter prior to executing this option. The tablespace cannot be larger than 2047 MB.

`-extendtblspace` *`Tablespace Pathname`* `-maxsize` *`Size_MB`* Creates an additional extent for the specified tablespace. The specified directory must exist and be capable of holding an extent of the size specified by the `-maxsize` parameter prior to executing this option. An extent cannot be larger than 2047 MB.

`-extendinfo` Displays information about existing extents.

`-changebdev` *`FromDev ToDev`* `[-session` *`SessionID`*`]` Changes all references in object versions from device `FromDev` to device `ToDev`. You can change the device name only for a single session by using the `-session` option.

`-info` Displays information about the IDB.

`-clear` Sets the status of all sessions that are actually not running but are marked *`In Progress/Failed`*, to *`failed`*. It requires exclusive database access to ensure that no session is running.

`-change_cell_name [`*`old_host`*`]` This option changes the owner of the CDB to the current Cell Manager. It also changes all references in the CMMDB from *`old_host`* to the current Cell Manager. It modifies all media entries within the MMDB or CMMDB associated with the original Cell Manager (old host).

If the *`old_host`* parameter is not specified, `omnidbutil` determines the previous owner of the CDB (old host) from the database itself.

If you want to associate all media in a CMMDB with the current Cell Manager, it is necessary to run the command once for each Cell Manager that has media associated with it, using the *`old_host`* parameter.

The *`old_host`* parameter must be specified exactly the same as the owner of the media. If the system's Fully Qualified Domain Name (FQDN) is associated with the media, then you must also use the FQDN with this command. If the *`old_host`* parameter is not specified correctly, the operation will not be performed.

This command is used after moving databases from one Cell Manager to another or after using `-readdb` on files that were created on another Cell Manager.

`-show_cell_name` Queries the CDB for its owner. If there is no information available, use the `-change_cell_name` option to update the information.

`-set_session_counter` *`new_session_ID`* Sets a new value for the counter that is used for generating the sessionID. This option is used after the restore and recovery of the IDB to enable the import of tapes that were created on the same day. Suggested value is 100.

`-upgrade_info` Displays the information about the upgrade of the IDB. The possible return strings are:

```
Do upgrade in progress.

Upgrade of core part failed.

Upgrade of core part finished.

Upgrade of detail part running.
Upgrade of detail part finished.
```

`-show_db_files` Lists all directories and extension files that are backed up during IDB backup. In effect they contain all components of IDB.

`-free_pool_update` Finds any free (unprotected) media in pools with the `free pool` and `move free media to free pool` options set and by default deallocates the found free media to a free pool every day at 00:00.

NOTES

The command can only be used locally on the Cell Manager.

EXAMPLES

The following example illustrates how the `omnidbutil` command works.

1. To create a new DC directory in the “/var/opt/test” directory with maximum size 1000 MB, use:

   ```
   omnidbutil -add_dc /var/opt/test -maxsize 1000
   ```

2. To list all locked devices, target volumes, media and slots, use:

   ```
   omnidbutil -show_locked_devs
   ```

3. To unlock a device, a medium or library slot, respectively, use:

   ```
   omnidbutil -free_locked_devs machine
   ```

   ```
   omnidbutil -free_locked_devs 0a1106452:5a45add9:2548:0007
   ```

   ```
   omnidbutil -free_locked_devs libraryName phyLocation
   ```

4. To unlock a target volume with the HP StorageWorks Disk Array XP serial number of "30658" and the HP StorageWorks Disk Array XP volume number of "288", use:

   ```
   omnidbutil -free_locked_devs 30658_288
   ```

SEE ALSO

omnidbrestore(1M), omnidb(1), omnidbcheck(1M), omnidbinit(1M), omnidbupgrade(1M), omnidbxp(1), omnidbva(1), omnidbeva(1), omnidbsmis(1)

omnidlc (1M)

NAME

omnidlc – gathers or deletes Data Protector debug, log and getinfo files from the Data Protector cell or from a MoM environment

SYNOPSIS

```
omnidlc -version | -help

omnidlc {-session sessionID | -did debugID | -postfix string |
    -no_filter} [-allhosts | -hosts list] [-pack filename | -depot
    [directory] | -space | -delete_dbg] [-no_getinfo] [-no_logs]
    [-no_debugs] [-no_compress] [-debug_loc dir1 [dir2] ... ]
    [-verbose]

omnidlc -localpack [filename]

omnidlc -unpack [filename]

omnidlc -uncompress filename
```

DESCRIPTION

The `omnidlc` command collects Data Protector debug, log and getinfo files from the Data Protector cell (by default, from every client) or from a MoM environment (from every Cell Manager). To collect data from Cell Managers in a MoM environment, the command must be run from the MoM. To collect data from clients in a MoM environment, the command must be run from their Cell Managers.

The Data Protector debug files are created during a Data Protector debug session. By default, the command collects debug files from the Data Protector default debug files directory, which is `/tmp` on UNIX and *`<Data_Protector_home>`*`\tmp` on Windows. To collect debugs also from other directories, use the `-debug_loc` option.

Using the command, it is possible to collect Data Protector debug, log and getinfo files from selected clients, or in a MoM environment, from selected Cell Managers in the cell.

Additionally, the Data Protector debug files to be collected can be limited to debugs that were generated within the specified Data Protector session or to debugs identified by a debugID or by a debug filename (debug postfix).

By default, every collected debug, log and getinfo file is then compressed and sent over the network to the Cell Manager. The final extension `.gz` is added on the Cell Manager, where all collected files with the `.gz` extension are, by default (if the `-depot` option is not specified), packed and saved in the current directory as the `dlc.pck` file. The file includes a generated directory structure that includes the hostnames, paths and the (compressed) collected files of the clients involved. This directory structure is described further on in this man page.

Optionally, files can be sent over the network to the Cell Manager uncompressed (if the `-no_compress` option is specified). Besides that (if the `-depot` option is specified), the transferred files can be left unpacked in the specified directory on the Cell Manager, in which the directory structure that includes the hostnames, paths and the collected files of the clients involved is generated as follows:

./dlc/*system_1*/tmp/*debug_files*

./dlc/*system_1*/log/*log_files*

./dlc/*system_1*/getinfo/get_info.txt

./dlc/*system_2*/tmp/*debug_files*

./dlc/*system_2*/log/*log_files*

./dlc/*system_2*/getinfo/get_info.txt

...

If the file to be sent over the network is larger than 2GB, the file is split in 2GB chunks before it is compressed (it can be left uncompressed) and sent to the Cell Manager. Every chunk retains the file name and is added the first extension ranging from `s001` to `s999`. The second extension (`.gz`) is not added if the files are not compressed. Additionally, on the Cell Manager side, if the size of all collected compressed or uncompressed files exceeds 2GB, the collected files are packed in 2GB sized (original size) packages and added an extension ranging from `s001` to `s999`.

The collected debug files can also be deleted (if the `-delete_dbg` option is specified), or the disk space required on the Cell Manager for the collected files can be displayed (if the `-space` option is specified). In these two cases, the selected files are neither transferred from the clients to the Cell Manager nor packed on the Cell Manager.

When collecting or deleting files or when displaying the required disk space, additional criteria can be defined to limit the files selection. Thus, it is possible to exclude the getinfo file, the log files, the debug files or any combination of the three groups of files from the selection.

Using the command, the collected files can then be additionally packed to be sent to the support centre. The command provides also a means of unpacking the packed collected files.

OPTIONS

`-version` Displays the version of the `omnidlc` command.

`-help` Displays the usage synopsis of the `omnidlc` command.

`-session` *`sessionID`* Limits the collected debug files to those that were produced during the Data Protector session identified by the *`sessionID`*.

`-did` *`debugID`* Limits the collected debug files to those identified by the *`debugID`*.

`-postfix` *`string`* Limits the collected debug files to the specified debug postfix.

`-no_filter` Does not limit (select) the collected debug files. If this option is specified, either the `-allhosts` or the `-hosts` option must also be specified in order to make a selection.

`-allhosts` Collects the files from all clients in the cell, or in a MoM environment, from all Cell Managers. The collected debug files are still subject to `-session`, `-did` or `-postfix` options.

`-hosts` *`list`* Limits the files to be collected to the clients, or in a MoM environment, to the Cell Managers specified in the *`list`*. The hostnames must be separated by spaces. The debug files collected are still subject to `-session`, `-did` or `-postfix` options.

`-pack` *`filename`* All collected files are, by default (if this option is not specified), packed and saved in the current directory as the `dlc.pck` file. If this option is specified, the collected files are packed and saved in the specified file in the current directory on the Cell Manager. If the full path name is specified, the files are packed and saved in the specified file in the specified directory.

To add files other than the collected files to the package, copy the files to one of the following directories before running the command: `dlc/`*`client`*`/getinfo`, `dlc/`*`client`*`/log`, or `dlc/`*`client`*`/tmp`. You cannot add directories, but only files. If the files are not copied to one of the specified directories, the package cannot be unpacked during the unpack phase.

`-depot` [*directory*] If the *directory* is specified, the collected files are not packed and are saved to the `dlc` directory of the specified directory. If the *directory* is not specified, the files are saved in the *<Data_Protector_home>*`\tmp\dlc` directory on Windows Cell Managers, or in the `/tmp/dlc` on UNIX Cell Managers.

`-space` Displays the disk space required on the Cell Manager for the collected files.

`-delete_dbg` Deletes the selected files on clients.

`-no_getinfo` Excludes the getinfo file from the selection.

`-no_logs` Excludes the log files from the selection.

`-no_debugs` Excludes the debug files from the selection.

`-no_compress` Disables the compression of the collected files on clients. By default, the compression is enabled.

`-debug_loc` *dir1* [*dir2*]... Includes debugs not only from the default debug files directory but also from other directories, *dir1*, *dir2*,.... Note that the subdirectories are excluded from the search. If a specified directory does not exist on a particular client, the directory is ignored.

This option is valid only if the `-no_debugs` option is not specified.

`-verbose` Enables verbose output. By default, verbose output is disabled.

`-localpack` [*filename*] Packs the directory structure from the current directory (must be the directory containing the `dlc` directory generated by the `-depot` option) to the *filename*. If the *filename* is not specified, the `dlc.pck` file is created in the current directory.

This option is equivalent to the `-pack` option, but is to be used only if the data is collected using the `-depot` option.

To add files other than the collected files to the package, copy the files to one of the following directories before running the command: `dlc/`*client*`/getinfo`, `dlc/`*client*`/log`, or `dlc/`*client*`/tmp`. You cannot add directories, but only files. If the files are not copied to one of the specified directories, the package cannot be unpacked during the unpack phase.

`-unpack` [*filename*] Creates the `dlc` directory in the current directory, and unpacks the contents of the *filename* to the `dlc` directory. If the *filename* is not specified, the `dlc.pck` file in the current directory is unpacked.

Use this option when the collected (compressed or uncompressed) data was packed on the Cell Manager either using the `-pack` option or the `-localpack` option.

`-uncompress` *`filename`* Uncompresses the unpacked compressed single file in the current directory.

Use this option after the packed data is unpacked using the `-unpack` option.

NOTES

The command can only be used on Cell Managers and MoMs.

It cannot be used to collect the Data Protector installation execution traces.

The Data Protector GUI debug files for systems other than Cell Manager can only be gathered using the `-hosts` option.

The command does not support clients running on OpenVMS. Debugs and log files created on OpenVMS platforms must be collected manually. For details on how to collect debug and log files on OpenVMS clients refer to the `ReadMe` file for the OpenVMS clients provided on the Windows Installation CD-ROM.

When using the `-did` or the `-session` option, the debug, log and getinfo files from clients other than Cell Manager can sometimes not be gathered. In such a case, use the `-hosts` or the `-allhosts` option additionally. Note that the `-allhosts` option can be time consuming.

When a MA agent launches an UMA process on some other client, UMA debugs are not gathered. In this case, the `-hosts` or `-allhosts` option should be used. It is not recommended to use the `-allhosts` option in large environments.

To collect debug files in a cluster, the command must be run using the `-hosts` option; the cluster nodes hostnames must be specified as the argument for the option. In a cluster, if the `-hosts` option is not specified, the data is collected from the active node.

EXAMPLES

1. To collect and compress all debug, log and getinfo files from the cell, and pack them in the “dlc.pck” file in the current directory on Cell Manager, using the verbose output, execute the following command:

   ```
   omnidlc -no_filter -allhosts -verbose
   ```

2. To collect only the log and debug files (without the getinfo files) from the clients “client1.company.com” and “client2.company.com” to the directory “c:\depot” on the Cell Manager, without compressing and packing the files, execute the following command:

   ```
   omnidlc -no_filter -hosts client1.company.com client2.company.com
   -depot c:\depot -no_getinfo -no_compress
   ```

3. To collect log, debug, and getinfo files from the client “client1.company.com”, compress and pack them to the “c:\pack\pack.pck” file on the Cell Manager, execute the following command:

   ```
   omnidlc -hosts client1.company.com -pack c:\pack\pack.pck
   ```

4. To collect log, debug, and getinfo files from the default location and debugs from the additional directories, “C:\tmp” and “/temp/debugs”, from the clients “client1.company.com” and “client2.company.com”, and to compress and pack the files on the Cell Manager, run:

   ```
   omnidlc -hosts client1.company.com client2.company.com -debug_loc
   C:\tmp /tmp/debugs
   ```

5. To delete all debug log files for the session with the ID “2003/08/27-9”, execute the following command:

   ```
   omnidlc -session 2003/08/27-9 -delete_dbg
   ```

6. To display disk space needed on the Cell Manager for the uncompressed debug files with the debugID “2351” from the client “client.company.com”, execute the following command:

   ```
   omnidlc -did 2351 -hosts client.company.com -space -no_getinfo
   -no_logs -no_compress
   ```

7. To pack the directory structure in the current directory (must be the directory containing the `dlc` directory generated by the `-depot` option) to the “dlc.pck” file in the same directory, execute the following command:

   ```
   omnidlc -localpack
   ```

8. To unpack the “dlc.pck” file to the "dlc" directory of the current directory, execute the following command:

   ```
   omnidlc -unpack
   ```

SEE ALSO

omnicc(1), omnicheck(1M), omnisv(1M), omnicellinfo(1), omniinstlic(1M)

omnidr (1M)

NAME

omnidr – a general purpose disaster recovery module

SYNOPSIS

```
omnidr -help

omnidr -version

omnidr [-srd file] [-temp[os]] [-map OrgMnt1 TrgMnt1 [-map
    OrgMnt2 TrgMnt2] ...] [-[no_]cleanup] [-msclusdb] [-drimini
    path] [GeneralOptions]

GeneralOptions

 -local

 -target hostname

 -report level
```

DESCRIPTION

The omnidr command is a general purpose Data Protector disaster recovery command that can be used in all recovery scenarios. Based on its input, omnidr decides what type of restore is going to be performed: online restore using omnir of offline restore using omniofflr as well as how the restore is going to be performed (using or avoiding live OS features).

OPTIONS

`-version`	Displays the version of the omnidr command.
`-help`	Displays the usage synopsis of the omnidr command.
`-srd` *file*	Specifies the path to the SRD file that contains all required backup and restore object information to perform the restore. Note that omnidr always requires a valid SRD file with updated object information. By default it searches the working directory for recovery.srd. If it is not found, error is reported. The option -srd overrides the default name recovery.srd

`-temp[os]` Specifies whether the restore process will run in a temporary OS installation. This way `omnidr` can determine how to restore the *`CONFIGURATION`* information. If this parameter is not specified, active system is assumed.

`-map` *`OrgMnt TrgMnt`* Specifies mapping of original volumes to current volumes.

`-[no_]cleanup` When the `-cleanup` option (default) is specified during disaster recovery of an active operating system, the `omnidr.exe` prepares a cleanup script and stores it into the `ALLUSERPROFILE\Start Menu\Programs\Startup` folder. At first logon after the boot, the Data Protector disaster recovery installation is removed.

When this option is specified during disaster recovery of a temporary operating system, a cleanup command is written into restored software hive in the registry at `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnce`. The cleanup command is executed at first logon after the boot and it removes the temporary OS installation together with Data Protector Disaster Recovery installation.

The cleanup script/command is not generated in the following cases:

- If Data Protector installation was found on the system during `omnidr` initialization.

- If the `-no_cleanup` option has been specified.

- If Data Protector disaster recovery installation does not reside in the `SYSTEMROOT` folder (in this case it was most likely not installed during Data Protector disaster recovery).

- If the `-debug` option has been specified, the cleanup is not performed, because you would loose the debug information at next logon.

- Minimal Recovery has been selected during EADR/OBDR meaning that only boot and system disks would be recovered.

When `omnidr` is used on a dual boot machine, it is strongly recommended to use the `-no_cleanup` option.

`-msclusdb` If this option is specified, `omnidr` restores the MS Cluster database.

`-drimini` *`path`* This option is used to provide location of P1S file if you have interrupted the `drstart` command during the 30 seconds pause and selected `install only` option when performing EADR. In this case the `drstart` command only installs disaster recovery files and exits. You have to start the `omnidr` command manually and provide the path to the P1S file using the `drimini` option. The default path is `c:\$DRIM$.OB2\OBRecovery.ini`.

GENERAL OPTIONS

`-local` Forces offline recovery from a locally attached device. OB II DEVBRA agent is used to automatically scan for and configure attached devices. A list of detected devices is displayed if more than one is found and you must select one of them. If this option is not specified, the device used for the restore is going to be the same as the device used during backup.

`-target` *`hostname`* Specifies the target system name. All objects will be restored to a computer specified by the -target parameter. If this parameter is not specified, the data will be restored to the system specified in the SRD file.

This option is used in two cases:

- During Disk Delivery disaster recovery the disks being restored can be installed into a client with a different hostname as original, therefore the name of the client must be specified.
- During Manual Disaster Recovery, it is possible, that DHCP protocol is installed (e.g. that is the default choice in Windows 2000 setup). In this case, the hostname can be generated automatically by the DHCP server and is different from the original system hostname.

`-report` *`level`* Specifies the error level report. This is useful if you want to reduce the number of messages written during recovery. For example, since practically all OS files are overwritten during the active OS recovery, this means that innumerable warnings bringing no useful information will be displayed, thus slowing down the recovery. Messages are classified (in ascending order) as: `1` (warning), `2` (minor), `3` (major) and `4` (critical). For example, if 3 is selected, only major and critical messages are reported. By default, all messages are reported.

NOTES

The `omnidr` command can only be used on Windows systems.

EXAMPLES

The following examples illustrate how the `omnidr` command works.

1. To use the SRD file stored on a floppy drive for the restore, type:

   ```
   omnidr -srd "a:/recovery.srd"
   ```

2. To use the local backup device, type:

   ```
   omnidr -local
   ```

SEE ALSO

omnisrdupdate(1M), omniofflr(1M), omniiso(1)

omnihealthcheck (1M)

NAME

omnihealthcheck – checks if Data Protector internal database (IDB) and services are operational.

SYNOPSIS

```
omnihealthcheck -version | -help
```

```
omnihealthcheck [-config ConfigFile]
```

DESCRIPTION

The omnihealthcheck command reads the specified configuration file where each line of the file is treated as a separate command and is executed. Note that the commands must be listed with full pathnames except if they are Data Protector commands located in *<Data_Protector_home>*\bin on the Windows Cell Manager or /opt/omni/bin or /opt/omni/sbin directories on the UNIX Cell Manager. Note also that the configuration file must be in the UNICODE format on the Windows Cell Manager. If the configuration file is not specified, the default file is used: *<Data_Protector_home>*\Config\server\HealthCheckConfig on the Windows Cell Manager or /etc/opt/omni/server/HealthCheckConfig on the UNIX Cell Manager.

If the default file is used, omnihealthcheck checks if Data Protector services (rds, crs, mmd, omnitrig and omniinet) are active, if the Data Protector MMDB is consistent and if at least one backup of the Data Protector internal database (IDB) exists.

Exit codes of individual commands are inspected at the end.

There are 3 different exit codes for the omnihealthcheck command:

0: All listed commands and their exit codes have been executed.

1: At least one of the commands in the configuration file could not be executed or has completed with an exit code other than 0.

2: The configuration file could not be read.

The final health check exit code is 0 (OK) only if all executed commands from the configuration file completed successfully (exit codes of all executed individual commands from the configuration file are 0).

Output of the `omnihealthcheck` command is saved in *`<Data_Protector_home>`*`\Log\server\HealthCheck.log` on the Windows Cell Manager or in `/var/opt/omni/server/log/HealthCheck.log` on the UNIX Cell Manager.

If a timeout occurs, `omnihealthcheck` fails.

`Omnihealthcheck` is by default scheduled to run daily at 12:00 (Noon) as a part of the Data Protector check mechanism. The default schedule value can be changed by changing the *`DailyCheckTime`* option in the Data Protector global options file. The Global options file (`global`) is located on the Cell Manager in the *`<Data_Protector_home>`*`\config\server\options` (Windows Cell Manager) or in the `/etc/opt/omni/server/options` (UNIX Cell Manager) directory.

OPTIONS

`-version` Displays the version of the `omnihealthcheck` command

`-help` Displays the usage synopsis of the `omnihealthcheck` command.

`-config` *`ConfigFile`* Specifies an alternative configuration file for the `omnihealthcheck` command. Note that you can define the commands to be executed in the health check.

NOTES

The command can only be used locally on the Cell Manager.

SEE ALSO

omnirpt(1), omnitrig(1M)

omniinstlic (1M)

NAME

omniinstlic – starts the HP OpenView AutoPass utility or synchronizes the Data Protector licenses between Data Protector and HP OpenView AutoPass

SYNOPSIS

```
omniinstlic -version | -help

omniinstlic [-sync]
```

DESCRIPTION

If the command is run without options, the licensing data in HP OpenView AutoPass is synchronized with the licensing data in Data Protector, and then the HP OpenView AutoPass utility is started. If the `-sync` option is used, it only synchronizes the Data Protector licenses between Data Protector and HP OpenView AutoPass, the HP OpenView AutoPass utility is not started.

The HP OpenView AutoPass utility lets you install passwords for your HP OpenView products' purchased licenses directly from the internet. Refer to the *HPOV Auto Pass User's Guide* for more information on the HP OpenView AutoPass utility.

OPTIONS

`-version` Displays the version of the `omniinstlic` command.

`-help` Displays the usage synopsis for the `omniinstlic` command.

`-sync` Synchronizes the Data Protector licenses between Data Protector and HP OpenView AutoPass.

NOTES

The command can only be used locally on the Cell Manager. In a Manager-of-Managers (MoM) environment, the `omniinstlic` command must be run on the MoM system (if Data Protector centralized licensing is used), or on the Cell Manager for which the passwords are being ordered and installed (if Data Protector centralized licensing is not used). The HP OpenView AutoPass utility must be installed on the system.

EXAMPLE

To start the HP OpenView AutoPass utility, execute the following command:

```
omniinstlic
```

SEE ALSO

omnicc(1), omnicellinfo(1), omnisv(1M), omnicheck(1M), omnidlc(1M)

omnimigrate.pl (1M)

NAME

omnimigrate.pl – helps you migrate your existing Cell Manager from a PA-RISC architecture based HP-UX 11.x system to an HP-UX 11.23 system for the Intel Itanium 2 (IA-64) architecture.

SYNOPSIS

```
omnimigrate.pl -help

omnimigrate.pl -prepare_clients New_CM_ClientName

omnimigrate.pl -configure

omnimigrate.pl [-configure_clients] [-configure_idb]
    [-configure_cm]
```

DESCRIPTION

`Omnimigrate.pl` helps you migrate your existing Cell Manager from a PA-RISC architecture based HP-UX 11.x system to an HP-UX 11.23 system for the Intel Itanium 2 (IA-64) architecture.

First, you need to run `omnimigrate.pl` on the old Cell Manager and back up the IDB. Then install Disk Agent to the HP-UX 11.23 system (your new Cell Manager) and restore your IDB to the new Cell Manager. Uninstall the Disk Agent from the new Cell Manager and install Data Protector A.06.00 Cell Manager. Finally run the `omnimigrate.sh` command again on the new Cell Manager.

OPTIONS

`-help` Displays the usage synopsis for the `omnimigrate.sh` command.

`-prepare_clients` *`New_CM_ClientName`* Adds the new Cell Manager's client name to the list of trusted hosts on secured clients. Secured clients accept requests on the Data Protector port (by default `5555`) only from trusted hosts.

This option should be used only on the *old* Cell Manager.

`-configure_clients` Migrates the clients from the old Cell Manager to the new Cell Manager. The old Cell Manager will keep the clients in the configuration files although it will not be their Cell Manager anymore.

If any of the clients is inaccessible, it will not be imported to the new cell. You can re-run the `omnimigrate.sh` command with this option when the clients are accessible to migrate them to the new Cell Manager.

The old Cell Manager will automatically become a client in the new cell. You can uninstall the Cell Manager component from the old Cell Manager, because it is not necessary anymore.

The option should be used only on the *new* Cell Manager.

`-configure_idb` Configures the IDB from the old Cell Manager for use on the new Cell Manager.

The option should be used only on the *new* Cell Manager.

`-configure_cm` Reconfigures the configuration data transferred from the old Cell Manager for use on the new Cell Manager.

The option should be used only on the *new* Cell Manager.

`-configure` Combines `-configure_clients`, `-configure_idb`, and `-configure_cm` options. This is the recommended way to run the `omnimigrate.pl` command.

The option should be used only on the *new* Cell Manager.

RETURN VALUES

`0` Successfully finished.

`1-4` An error occurred.

ERRORS

`1` A generic error occurred.

`2` Migration of IDB catalogs failed.

`3` Configuration error (Cell Manager configuration error or an error during the import of clients) occurred.

`4` Error parsing options.

NOTES

This command is supported only on HP-UX systems.

EXAMPLES

1. Run the following command on the old Cell Manager to add the new Cell Manager with the client name "dfg.company.com" to the list of trusted hosts on secured clients:

   ```
   omnimigrate.pl -prepare_clients dfg.company.com
   ```

2. To migrate the IDB, reconfigure the Cell Manager's settings, export all clients from the old Data Protector cell and import them to the new cell, execute the following command on the new Cell Manager:

   ```
   omnimigrate.pl -configure
   ```

SEE ALSO

ob2install(1M), omnigui(5), omnisetup.sh(1M), omniusers(1), upgrade_cfg_from_evaa(1M)

omniofflr (1M)

NAME

omniofflr – enables restore of any type of Data Protector backup object in the absence of a working Data Protector internal database (IDB)

SYNOPSIS

```
omniofflr -version | -help

omniofflr DeviceOptions MediaOptions1 [MediaOptions2 ...]
    ObjectOptions1 [ObjectOptions2 ...] [General Options]

DeviceOptions

 -name DeviceName

 -dev PhysicalDevice1 [PhysicalDevice2 ...]

 -mahost DeviceHostName

 -policy LogicalDevicePolicy

 -type LogicalDeviceType

 [-description DeviceDescription]

 [-blksize BlockSize]

MediaOptions

 -maid MediumID1 [MediumID2 ...]

 [-slot slot1[:flip] [slot2[:flip] ...]]

 [-position segment1:offset1 [segment2:offset2 ...]]

ObjectOptions

 {-filesystem | -winfs | -omnidb} Client:MountPoint Label

 -daid DAID

 [-merge]

 [-[no_]overwrite]

 [-move_busy]

 [-omit[_deleted_files]]

 [-var OptName OptValue]
 -tree TreeName1 [TreeOptions1] [-tree TreeName2 [TreeOptions2]
    ... ]
```

TreeOptions

```
 -exclude TreeName1 [TreeName2...] {-as | -into}  NewTreeName
```

General Options

```
 -verbose

 -preview

 -report

 -target TargetHostName

 -[no]ok[mediumlist]
```

DESCRIPTION

The `Omniofflr` command can be used as a standalone utility or - on Windows systems - by a higher level utility `omnidr`, which automatically generates restore object command line options for the omniofflr command, based on the SRD file information.

The `Omniofflr` command enables the restore of any type of backup object in the absence of the Data Protector internal database (IDB) (due to a disaster or lost connection to the Cell Manager).

Running the `omniofflr` command requires detailed information about the restore device and backup media, including positions of backup objects on the media. Media information can be obtained from the SRD file (located in *<Data_Protector_home>*`\config\server\dr\srd` on a Windows Cell Manager or in `/etc/opt/omni/server/dr/srd` on an UNIX Cell Manager) or you can provide the information manually. To obtain this information, query the IDB using the `omnidb` command after the backup and write down the results. It is also possible to write a script, which queries the IDB and generates another script in which the `omniofflr` command with the proper options is executed.

OPTIONS

`-version` Displays the version of the `omniofflr` command.

`-help` Displays the usage synopsis for the `omniofflr` command.

DeviceOptions

-name *LogicalDeviceName* Parameter that specifies the logical device name.

-dev *PhysicalDevice* Specifies the pathname of the device file. For example: c:\temp\dev1, scsi1:0:0:0, /dev/tape0...

-mahost *DeviceHostName* Specifies the name of the client, where the restore device is attached and a Media Agent started.

-policy *LogicalDevicePolicy* Specifies the policy ID for the device specified by the -dev option. Policy can be defined as:

1 (Standalone),

3 (Stacker)

5 (6300 MO jukebox)

6 (Exchange through cmd execution)

8 (GRAU DAS exchanger library)

9 (Silo medium library)

10 (SCSI exchanger)

11 (RSM exchanger)

-type *LogicalDeviceType* Specifies the media type for the media in the device specified by the -device option. Media type numbers are defined in the *HP OpenView Storage Data Protector Software Release Notes*.

-description *DeviceDescription* This is an optional parameter that specifies the logical device description.

-blksize *BlockSize* This is an optional parameter that specifies the block size the device is going to use when accessing media.

MediaOptions

-maid *MediumID* Specifies the medium identification number of the medium that contains the object data; for example 8c04110a:3b0e118b:041c:0001. If unknown is specified, each medium will be accepted as valid and restore will be attempted. Whole medium will be scanned for the requested object and it may take a very long time, if the object is not on the medium. Mount prompt in such case will request the next medium, without specifying the medium label.

`-slot` *`slot1`*`[:`*`flip`*`]` Specifies the slot identifier of the slot, where the required media is located, thus enabling Data Protector to automatically load media from the exchanger slots. Note that the sequence has to match the sequence in the list created using the `-maid` option.

`-position` *`segment1:offset1`* Specifies the segment and offset position of the restore object data on the medium; for example `67:20`. If the position is not specified, the position `1:0` is assumed, thus prolonging the restore time. Note that the sequence has to match the sequence in the list created using the `-maid` option.

`ObjectOptions`

`-filesystem` *`Client:MountPoint Label`* Selects the filesystem identified with *`Client:MountPoint Label`* for restore. Client determines the name of the system where the object was backed up. *`MountPoint`* specifies the mount point name of the volume to be restored (for example `/C`, `/tmp`, `/`, etc.). It must be in the same format as stored in the IDB. *`Label`* specifies the backup/restore objects description that uniquely defines an object (`-filesystem computer.domain.net:/mount label`)

`-winfs` *`Client:MountPoint Label`* Selects the Windows filesystem identified with *`Client:MountPoint Label`* for restore. Client determines the name of the system where the object was backed up. *`MountPoint`* specifies the mount point name of the volume to be restored (for example `/C`, `/tmp`, `/`, etc.). It must be in the same format as stored in the IDB. Therefore, for example, on Windows systems `C:` translates into `/C`. Label specifies the backup/restore object's description that uniquely defines an object (`-winfs computer.domain.net:/C:` etc.)

`-omnidb` *`Client:MountPoint Label`* Selects the files from the IDB identified with *`Client:MountPoint Label`* for restore. Client determines the name of the system where the object is to be restored. *`MountPoint`* for IDB is always `/`. Label specifies the backup/restore object's description that uniquely defines an object (`-omnidb computer.domain.net:/C:` etc.).

`-daid` *`DAID`* Specifies the disk agent identification number of the disk agent that backed up an object.

`-merge` This option merges files from the backup medium to the target directory and replaces older versions that exist in the directory with newer (if they exist on the medium) files. Existing files are overwritten if the version on the medium is newer than the version on disk. No existing directory is deleted. If a directory or file doesn't exist on disk (but is on the backup medium) it is restored (created).

`-overwrite` By default, or if the `-overwrite` option is specified, the already existent files on the disk are overwritten by the restored files.

`-no_overwrite` If the `-no_overwrite` option is specified, only the files that do not exist on the disk are restored.

`-move_busy` This option is used with the `-omit_deleted_files` or `-overwrite` option. A problem can occur if, for example, a file to be overwritten cannot be deleted because it is currently in use. If this option is specified, Data Protector moves busy file *`filename`* to *`#filename`* on Unix systems (adding a hash- mark in front of the filename), or to *`filename.001`* on Windows system. On Unix systems the original file can thus be deleted as the lock is transferred to the corresponding file starting with the #sign. For example, `/tmp/DIR1/DIR2/FILE` would be moved to `/tmp/DIR1/DIR2/#FILE`. On Windows system the application only uses the newly-restored file after the file is restored and the system is rebooted.

`-omit_deleted_files` This option can be only used in combination with the `-overwrite` option.

If this option is specified, Data Protector attempts to recreate the state of the restored directory tree as it was when the last incremental backup was run, while preserving files that were created or modified after the last incremental backup. However, if the directory contains files that did not exist there at the time of the last incremental backup, but their modification time is older than the time of the incremental backup, Data Protector will delete these files as well.

When this option is used in combination with the `-as` or `-into` option, be careful when specifying the new location to prevent accidental deletion of existing files.

If this option is not specified, when restoring a directory from which files were deleted between a full and an incremental backup, these files are also restored.

The time on the Cell Manager and clients must be synchronized for this option to function properly.

`-variable` *`var_name var_value`* This option lets you specify a variable name and its value for proper operation of some platforms and integrations. Setting user definable variables (a variable name and its value) enables flexible operation on some platforms and integrations with Data Protector. The list of variables and their values that are configurable with Data Protector is dynamic and comes with Data Protector patches.

`-tree` *`TreeName`* [*`TreeOptions`*] Specifies the starting root directory of data restore. Note that this starting directory is also restored.

`TreeOptions`

`-exclude` *`TreeName`* Specifies trees excluded from the restore.

`-as` *`NewTreeName`* This is an optional parameter that restores the selected fileset as the specified tree. This parameter is of vital importance for the Disk Delivery disaster recovery, since without it the restore to the original location would be performed.

`-into` *`NewTreeName`* This is an optional parameter that restores the selected fileset into the given directory. This parameter is of vital importance for the Disk Delivery disaster recovery, since without it the restore to the original location would be performed.

`General Options`

`-verbose` Specifies the verbose level of progress reporting.

`-preview` Specifies that the preview mode of the restore is entered.

`-report` Displays a report of the disaster recovery using the `omniofflr` command.

`-target` Specifies the target system name which is different than the original.

`-[no_]ok[mediumlist]` By default the options are parsed and displayed so that the user can check them and confirm the start of restore. This means that the `omniofflr` command used from a script could not be executed because it would wait for the confirmation before starting the restore. This option has to be used to skip confirmation, thus enabling the execution of the `omniofflr` command from a script.

NOTES

The omniofflr command does not support robotic media loaders. The user must ensure that appropriate media is loaded into specified drives. This can be done using the uma agent on the system to which robotics is connected. The omniofflr command can only be used on Windows systems.

EXAMPLES

The following example illustrates how the omniofflr command works.

To restore the “c:/temp” directory of the computer “computer.company.com” without the “c:/temp/vnc” directory, which was backed up using an HP Ultrium standalone device on a STK Ultrium drive medium, attached to the Cell Manager “cm.company.com”, into the “c:/test/temp directory”, type:

```
omniofflr -verbose -name HP:Ultrium -dev scsi2:0:4:0C -mahost
cm.company.com -policy 1 -type 13 -maid 9e03110a:3b5ee669:05ac:0001
-computer.company.com:/C C: -daid 996144004 -tree /temp -exclude /
temp/vnc -into c:/test/temp
```

To get the logical device name and its SCSI address, type:

```
devbra -dev
```

The output of the command looks something like this:

```
HP:Ultriumscsi2:0:4:0cLTO : HP LTO drive
```

“HP:Ultrium” is the logical device name of the backup device while “scsi2:0:4:0c” specifies the SCSI address of the device.

To obtain the medium ID (MAID), execute the omnidb command with the appropriate backup session ID:

```
omnidb -session 2001/09/06-1 -media
```

To obtain all backup session IDs for the winfs computer.domain.com:/C computer.domain.com [/C], type:

```
omnidb -winfs computer.domain.com:/C "computer.domain.com [/C]"
```

To obtain the Disk Agent ID (DAID) and the object name, use the omnimm command with the relative MAID:

```
omnimm -catalog 9e03110a:3b5ee669:05ac:0001
```

SEE ALSO

omnidr(1M), omnisrdupdate(1M), omniiso(1)

omniresolve (1M)

NAME

omniresolve – resolves a filesystem object or a list of filesystem objects and writes the results to the standard output or to a Unicode file.

SYNOPSIS

```
omniresolve -version | -help

omniresolve {-files filename [filename2 ...] | -inputfile
    datafile}[-verbose] [-unicodefile outfile]
```

DESCRIPTION

The `omniresolve` command reads the filesystem structures locating the physical disks (on Windows) or volumes (on UNIX) on which a filesystem object resides. If the files reside on a logical volume which is a part of a volume group (disk group), all volumes in a volume group are displayed.

You can list the filesystem objects to be resolved either in the CLI (on UNIX and Windows systems) or using a Unicode file (on Windows systems only). The results are written to standard output (on UNIX and Windows systems) or to a Unicode file (on Windows systems only).

OPTIONS

`-version` Displays the version of the `omniresolve` command.

`-help` Displays the usage synopsis for the `omniresolve` command.

`-files` *`filename`* `[`*`filename2`*`...]` Resolves a list of files separated by spaces and writes the results to the standard output.

`-inputfile` *`datafile`* Resolves all objects listed in *`datafile`* in and writes the results to the standard output.

Note that on Windows systems, if *`datafile`* is in the Unicode format, the output is by default written to the file `uniout.dat`. You can redirect the output to a different file by using the `-unicode` option.

`-verbose` Provides a more detailed report (displaying details such as WWNs, LUNs, or LDEVs) using SCSI inquiry on physical disks.

`-unicodefile` *`outfile`* Defines the file to which the output is redirected if the input file is a Unicode file.

NOTES

The resolve process requires root permissions on UNIX systems to get access to the disk device files. Therefore, the SUID flag is set on for `omniresolve`.

EXAMPLE

To resolve a list of three files ("system01.dbf", "redo01.log", and "control01.ctl") located in "/opt/oracle9i/oradata/dbname" enter the following command:

```
omniresolve -f '/opt/oracle9i/oradata/dbname/system01.dbf' '/opt/
oracle9i/oradata/dbname/redo01.log' '/opt/oracle9i/oradata/dbname/
control01.ctl' -v
```

omnirsh (1M)

NAME

omnirsh – returns the hostnames of the physical and virtual nodes for the specified cluster hostname, or returns the cell information, stored in the cell_info file on the specified cluster

SYNOPSIS

```
omnirsh -version | -help

omnirsh cluster_hostname {INFO_CLUS | INFO}
```

DESCRIPTION

The `omnirsh` command returns the hostnames of the physical and virtual nodes for the specified cluster hostname, together with the flag indicating whether a specific node is a physical node or virtual node. The command can also be used to list the contents of the cluster `cell_info` file, residing in the *<Data_Protector_home>*`\Config\server\cell` (Windows Cell Manager) or in the `/etc/opt/omni/server/cell` (UNIX Cell Manager) directory.

OPTIONS

`-version` Displays the version of the `omnirsh` command.

`-help` Displays the usage synopsis for the `omnirsh` command.

`cluster_hostname` Sets the hostname of the cluster for this command.

`INFO_CLUS` Lists the hostnames of the physical and virtual nodes for the specified cluster hostname, together with the flag indicating whether a specific node is a physical node or virtual node. Flag value `1` indicates a physical node, whereas flag value `8` indicates a virtual node.

`INFO` Displays the contents of the `cell_info` file for the system specified by the *`cluster_hostname`* parameter. The `cell_info` file resides in the *<Data_Protector_home>*`\Config\server\cell` (Windows Cell Manager), `/etc/opt/omni/server/cell` (UNIX Cell Manager) directory on the Cell Manager.

SEE ALSO

omniclus(1M)

omnisrdupdate (1M)

NAME

omnisrdupdate – updates System Recovery Data (SRD)

SYNOPSIS

```
omnisrdupdate -version | -help

omnisrdupdate -session sessionID [-cell name]

 [-host ClientName] [[-location path1] [-location path2] ...]

 [-asr]
```

DESCRIPTION

The `omnisrdupdate` command is used to update system recovery data (SRD), which is an ASCII text file on Windows clients that contains information required for the system configuration of the target system. A SRD file is generated when a CONFIGURATION backup is performed on a Windows client. The generated SRD file is then stored in the `<Data_Protector_home>\Config\server\dr\srd` directory on a Windows Cell Manager or in the `/etc/opt/omni/server/dr/srd/` on a UNIX Cell Manager.

The SRD filename is identical to the hostname of the system where it was generated - for example `computer.company.com`. After the CONFIGURATION backup, the SRD contains only the system information required for system configuration and installation of the operating system needed for disaster recovery. In order to be able to perform a disaster recovery without a working Data Protector internal database (IDB), additional information about backup objects and corresponding media must be added to the SRD by running this command. The name of the updated SRD file is `recovery.srd`.

OPTIONS

`-version` Displays the version of the `omnisrdupdate` command.

`-help` Displays the usage synopsis for the `omnisrdupdate` command.

`-session` *`sessionID`* When `omnisrdupdate` is executed from the command prompt, it requires a session ID to update an existing SRD file with backup object information which belongs to the given *`sessionID`* value. If it is executed from a post-exec script, Data Protector tries to obtain this information from the current environment. Updating the SRD file will succeed only when all critical backup objects (as specified in the SRD file) were actually backed up during the specified session. To view which objects are marked as critical for the SRD update, open the SRD file in a text editor. All critical objects for the SRD update are listed under the `-section objects` section. Note that the database is represented as "/".

After the SRD is updated, it is saved on the Cell Manager.

`-cell` *`name`* Specifies the Cell Manager to connect to in order to obtain the required information about backup objects and corresponding media from the IDB.

If this option is not specified, Data Protector tries to obtain this information automatically from the current environment.

`-host` *`ClientName`* Specifies the system for which the SRD file is to be updated. If this option is not specified, the information is obtained automatically by Data Protector from the current environment.

`-location` *`path`* Specifies where the updated SRD file should be saved in addition to the Cell Manager. There can be more than one `-location` parameter specified (including network shares), each of which will receive an updated copy of the SRD file. If nothing is specified, the updated SRD file is saved only on the Cell Manager in the *`<Data_Protector_home>`*`\Config\server\dr\srd` directory on a Windows Cell Manager or in the `/etc/opt/omni/server/dr/srd` on a UNIX Cell Manager.

If you are using the `omnisrdupdate` command in a post-exec script, do not add a backslash at the end of the path.

It is recommended, that in the addition to the Cell Manager, the updated SRD file is saved on several secure locations as a part of disaster recovery preparation policy.

`-asr` If specified, the ASR archive file (a collection of files required for proper reconfiguration of the replacement disk packed in a single archive file) is downloaded from the Cell Manager and ASR files are extracted and stored to all destinations, specified by the `-location` option. At least one `-location` option must be specified otherwise the `-asr` option is ignored. If the ASR archive file on the Cell Manager does not exist, `omnisrdupdate` will fail and the SRD file will not be updated.

NOTES

The `omnisrdupdate` command can only be used on a Windows client.

EXAMPLES

1. To update the SRD file with the backup object information belonging to the session “2001/03/02-5” type:

   ```
   omnisrdupdate -session 2001/03/02-5
   ```

   To obtain the session ID, execute the omnidb command with the option -session. To obtain the latest session ID, type:

   ```
   omnidb -session -latest
   ```

2. To update the SRD file with the backup object information which belongs to a session “2001/03/02-5” and save the updated SRD file on a floppy disk as well as in the directory “srdfiles” on a system with the hostname “computer”, type:

   ```
   omnisrdupdate -session 2001/03/02-5 -location a: -location //
   computer/srdfiles
   ```

3. To update the first floppy diskette from the ASR set with the backup object information and ASR files, which belong to a session “2002/03/02-5”, insert the first diskette in the floppy drive (make sure it is not write protected) and type:

   ```
   omnisrdupdate -session 2002/03/02-5 -location a: -asr
   ```

SEE ALSO

omniofflr(1M), omnidr(1M), omniiso(1)

omnisetup.sh (1M)

NAME

omnisetup.sh – Installs or upgrades a Data Protector UNIX Cell Manager, Installation Server, or client system locally

SYNOPSIS

```
omnisetup.sh -version | -help

omnisetup.sh [-source directory] [-server name] [-install
    component_list] [-CM] [-IS] [-autopass]
```

component_list

```
 da  = Disk Agent
 ma  = General Media Agent
 ndmp  = NDMP Media Agent
 cc  = User Interface
 momgui  = MoM User Interface
 sap  = SAP R/3 Integration
 sapdb  = SAP DB Integration
 emc  = EMC Symmetrix Agent
 oracle8  = Oracle Integration
 sybase  = Sybase Integration
 db2  = IBM DB2 UDB Integration
 ssea  = HP StorageWorks XP Agent
 snapa  = HP StorageWorks VA Agent
 smisa  = HP StorageWorks EVA SMI-S Agent
 informix  = Informix Integration
 lotus  = Lotus Integration
 ov  = HP OpenView NNM Backup Integration
 omnist  = Omnistorage Integration
 jpn_ls  = Japanese Language Support
```

```
fra_ls  = French Language Support
```

DESCRIPTION

The command first checks if Data Protector software is already installed on the system.

NEW INSTALLATION OR RE-INSTALLATION OF THE SAME VERSION OF Data Protector

If Data Protector is not installed then the command, depending on the selected options, installs the Cell Manager, Installation Server, or every Data Protector software component specified by the `-install` option. If none of these options are specified, the command issues a prompt for every Data Protector software component supported on the current system OS. Using this prompt, software components supported on the current system OS can be confirmed or rejected for installation, or the execution of the command can be canceled. There is no such prompt if the `-install` option is specified.

UPGRADE FROM AN EARLIER VERSION OF Data Protector

To upgrade your cell from the earlier versions of the product to Data Protector A.06.00, proceed as follows:

- Upgrade the Cell Manager
- Upgrade the Installation Server
- Upgrade the clients

To upgrade the all Data Protector components on the system, run `omnisetup.sh` without options. If the Installation Server is installed together with the Cell Manager, or if it is installed without client components, it is upgraded automatically during the Cell Manager upgrade.

If the Installation Server is installed with the client components, it is removed during the Cell Manager upgrade. In this case, a new Installation Server must be installed from the second CD using the `-IS` option, after the upgrade finishes.

To add a client to the Cell Manager, specify the `-install` option. If the client not residing on the Cell Manager is to be upgraded, the `-install` option does not need to be specified. In this case, the setup selects the same components as were installed on the system before the upgrade without issuing a prompt.

In all cases (new installation, re-installation or upgrade), the following applies when using this command:

- When using the `-install` option, the software components not supported on the current system OS and mistyped software components are skipped.
- On the Cell Manager only, when the installation or upgrade is started, you are prompted to install the HP OpenView AutoPass utility (unless the `-autopass` option is specified, in such a case, the HP OpenView AutoPass utility is installed or upgraded without issuing a prompt). If AutoPass is already installed on the system, it is automatically upgraded, if the prompt is confirmed. When Data Protector is uninstalled from the system, the HP OpenView AutoPass utility is neither unregistered nor uninstalled. It must be uninstalled using UNIX utilities.

  If the HP OpenView AutoPass utility is installed in cluster environment, it must be installed on every node in the cluster.
- After the client (re-)installation or upgrade has finished, the system is imported to a Data Protector cell if the `-server` option was set, or if the `/etc/opt/omni/client/cell_server` (HP-UX, Solaris, and Linux clients) or the `/usr/omni/config/cell/cell_server` (other UNIX clients) file exists on the system.
- The first time any software component is selected for installation or reinstallation, the `core` component is automatically installed (or reinstalled). Similarly, the first time any integration software component is selected for installation or reinstallation, the `core-integ` component is automatically installed (or reinstalled).

OPTIONS

`-version` Displays the version of the `omnisetup.sh` command.

`-help` Displays the usage synopsis for the `omnisetup.sh` command.

`-source` *`directory`* Sets the location of the Data Protector installation files (DVD mountpoint). If this option is not specified, the current directory is set as the location of Data Protector installation files.

`-server` *`name`* Sets the hostname of the Cell Manager of the cell to which the installed or upgraded client is to be imported after the installation or upgrade has finished. If this option is not specified, and the `/etc/opt/omni/client/cell_server` (HP-UX, Solaris, and Linux clients) or the `/usr/omni/config/cell_server` (other UNIX clients) file does not exist on the system, the installed or upgraded client is not imported to any cell and has to be imported manually.

`-install` *`component_list`* Sets Data Protector software components that are to be installed or upgraded on the current system. If more than one software component is to be installed or upgraded, a listing of software components, delimited by comma (without spaces) must be entered as the argument. If this option is not specified (except for the case when the client not residing on the Cell Manager needs to be upgraded), the command issues a prompt for every Data Protector software component supported on the current system OS; prompting whether to install or upgrade certain Data Protector software component or not. If the client is to be upgraded, this option does not need to be specified. In this case, the setup selects the same components as were installed on the system before the upgrade without issuing a prompt.

`-CM` Installs/upgrades the Data Protector Cell Manager.

`-IS` Installs/upgrades the Data Protector Installation Server with *all* push packages.

Note that the Installation Server can be upgraded only after the Cell Manager in the Data Protector cell is upgraded.

`-autopass` If this option is specified, the HP OpenView AutoPass utility is automatically installed. If AutoPass is already installed on the system, it is automatically upgraded. This option is to be used only on the Cell Manager.

NOTES

This command requires the Data Protector UNIX installation DVD to be mounted on the system. Before running the command make sure that no Data Protector backups or restores are running on the system. The command must be executed using the `ksh` shell.

On MC/ServiceGuard, the HP OpenView AutoPass utility must be installed an all nodes.

EXAMPLES

1. To upgrade a system, execute the following:

```
omnisetup.sh
```

2. To install or re-install the General Media Agent, Disk Agent, Oracle and SAP R/3 software components, execute the following:

   ```
   omnisetup.sh -install ma,da,oracle8,sap
   ```

3. To install the Cell Manager and Installation Server together with the HP OpenView AutoPass utility, insert and mount the UNIX installation DVD and run the following command:

   ```
   omnisetup.sh -CM -IS -autopass
   ```

SEE ALSO

ob2install(1M), omnigui(5), omnimigrate.sh(1M), omniusers(1), upgrade_cfg_from_evaa(1M)

omnisv (1M)

NAME

omnisv – starts, stops or queries the status of Data Protector services.

SYNOPSIS

```
omnisv -help

omnisv -version

omnisv {-start | -stop | -status | -start_mon}
```

DESCRIPTION

The omnisv command enables you to start or stop Data Protector services and display their status.

Omnisv can start or stop the RDS service, crs service and mmd service on the Cell Manager. Note that the mmd service can only be started or stopped locally on the Cell Manager with the MMDB.

On the HP-UX or Solaris Cell Manager the omnisv command also adds the omnitrig process to the cron table and schedules it to run at 15 minute intervals (the omnitrig command on the Windows Cell Manager is started by the crs service).

On the Windows Cell Manager omnisv also starts the OmniInet service (the Data Protector Inet program (/opt/omni/lbin/inet) is on the HP-UX or Solaris Cell Manager started by the system inet daemon when an application tries to connect to the Data Protector port, which is by default port number 5555. Normally, these daemons are started automatically during the system's startup).

Stopping of RDS service is logged down in the RDS.log (located in *<Data_Protector_home>*\db40\datafiles\catalog on the Windows Cell Manager or /var/opt/omni/server/log on the UNIX Cell Manager) with the ***SERVER SHUTDOWN INITIATED*** message. Each time the RDS service is started a new RDS.log is created and the previous RDS.log is renamed to RDS.bak.

OPTIONS

-version	Displays the version of the omnisv command.
-help	Displays the usage synopsis of the omnisv command.

`-start` Starts the Data Protector services and adds the `omnitrig` command to the cron table, thus configuring it as a cron job.

`-stop` Stops the services and removes the `omnitrig` command from the cron table.

`-status` Displays the status and PID of the services.

`-start_mon` Waits in loop until the `crs`, `mmd` and `rds` services are up and running. If any daemon or service stops, `omnisv` exits with an exit code `1`. Exit code `0` means that all relevant Data Protector daemons/services are up and running, whereas the exit code `1` means that at least one of the relevant Data Protector daemons/ services is not running.

NOTES

On Windows systems, only the users in the Data Protector admin group can execute this command. On HP-UX and Solaris systems, only the root user can execute this command. This command can only be used locally on the Cell Manager. It is not possible to start or stop services on a cluster using this command.

SEE ALSO

omnicc(1), omnicellinfo(1), omnicheck(1M), omnidlc(1M), omniinstlic(1M)

omnitrig (1M)

NAME

omnitrig – triggers Data Protector scheduled backups

SYNOPSIS

```
omnitrig -version | -help

omnitrig [-start] [-log]

omnitrig -stop

omnitrig -run_checks
```

DESCRIPTION

The `omnitrig` command checks and triggers scheduled backups.

OPTIONS

`-version` Displays the version of the `omnitrig` command

`-help` Displays the usage synopsis for the `omnitrig` command

`-start` Adds the `omnitrig` command to the cron table and schedules it to run at 15 minute intervals.

`-log` If this option is specified then `omnitrig` will save information about each start of `omnitrig` command and backups started by `omnitrig` command into `/var/opt/omni/server/log/omnitrig.log` file.

`-stop` Removes the `omnitrig` command from the `cron` table.

`-run_checks` Start checks for the following Data Protector notifications: IDB Space Low, IDB Tablespace Space Low, Not Enough Free Media, Health Check Failed, User Check Failed, Unexpected Events, License Warning, License Will Expire, and IDB Purge Needed.

By default, these checks are started automatically every day at 12:30 P.M. You can change the time of these checks or disable them by changing the *`DailyCheckTime`* option in the global options file.

NOTES

The command can only be used locally on the Cell Manager system.

SEE ALSO

omnirpt(1), omnihealthcheck(1M)

sanconf (1M)

NAME

sanconf – auto-configures a library, modifies an existing library or drive configuration, or removes drives from a library configuration, within a SAN environment

SYNOPSIS

```
sanconf -version | -help

sanconf -list[_devices] [ListFileName]  [-hosts host_1
    [host_2...] | -hostsfile HostsFileName]

sanconf -configure [ListFileName] -library LibrarySerialNumber
    LibraryName [RoboticControlHostName] [DeviceTypeNumber |
    ".DeviceTypeExtension"][-hosts host_1 [host_2...] |
    -hostsfile HostsFileName][-drive_template
    DriveTemplateFileName] [-library_template
    LibraryTemplateFileName] [-[no_]multipath]
    [-sanstableaddressing]

sanconf -remove_drives LibraryName [-hosts host_1 [host_2...] |
    -hostsfile HostsFileName]

sanconf -remove_hosts LibraryName [-hosts host_1 [host_2...] |
    -hostsfile HostsFileName][-[no_]multipath]
```

DESCRIPTION

The `sanconf` command is a utility that provides easier configuration of libraries in SAN environments. It can automatically configure a library within a SAN environment by gathering information on drives from multiple clients and configuring them into a single library.

The `sanconf` command can be run on the Data Protector Cell Manager or on Data Protector clients. It resides in the *<Data_Protector_home>*`\bin` directory on Windows and in the `/opt/omni/lbin` directory on HP-UX and Solaris clients.

You can perform the following tasks using the `sanconf` command:

- Scan the specified Data Protector clients, gathering the information on SCSI addresses of drives and robotic controls connected to the clients in the SAN environment.

- Configure or modify settings of a library or drive for given clients using the information gathered during the scan of Data Protector clients.
- Remove drives on all or the specified clients from a library.

All `sanconf` sessions are logged to the *<Data_Protector_home>*`\log\sanconf.log` file on Windows or to the `/var/opt/omni/log/sanconf.log` file on HP-UX and Solaris systems.

OPTIONS

`-version` Displays the version of the `sanconf` command.

`-help` Displays the usage synopsis of the `sanconf` command.

`-list[_devices] [`*`ListFileName`*`]` This option scans Data Protector clients to gather information on SCSI addresses of drives and robotic controls connected to the clients in the SAN environment and lists the gathered information. The information is uploaded to the Media Management Database on the Cell Manager. When *ListFileName* parameter is specified, the information acquired during the scan of clients is saved to the configuration file, which will be then used for configuring the library.

It is recommended to scan all clients that you want to configure, those that can see the robotics and those that can see the drives.

`-hosts` *`host_1`* `[`*`host_2...`*`]` Specify the `-hosts` option if you want to limit the `sanconf` actions only to specified clients. Other clients in the Data Protector cell are skipped.

`-hostsfile` *`HostsFileName`* Specify the `-hostsfile` option if you want to limit the `sanconf` actions only to clients specified in the *HostsFileName*. Other clients in the Data Protector cell are skipped. The *HostsFileName* is comprised of an ASCII list of clients, one client per line. It is recommended that all clients are specified in the clients list before you save the scan information to the configuration file.

To configure multipath devices, add the following at the beginning of the *HostsFileName* file:

```
<OPTIONS>

-multipath

</OPTIONS>
```

For multipath devices, the path order is determined by the order in the given list or file.

`-configure` [*ListFileName*] This option scans, lists, configures, or reconfigures the specified library. Only one library can be configured with each invocation of the command line. If the *ListFileName* option is not specified, the `sanconf` command will dynamically scan, list, and configure the library. If this option is specified, the scan and data information that was saved to a file during the scan of the specified clients is used to configure the library and scan is not performed. If a client is not scanned, the library will be configured without drives connected to this client.

Note: When reconfiguring a library, it is recommended that configuration information is first stored in the configuration file in case of configuration failure. It is also recommended that a different filename is used so that the initial configuration can be restored without any complications. sanconf reuses the custom settings when reconfiguring a library.

`-library` *LibrarySerialNumber LibraryName* [*RoboticControlHostName*]

[*DeviceTypeNumber* | "*.DeviceTypeExtension*"]

Specify the `-library` parameter to configure or reconfigure the specified library. Only one library can be configured with each invocation of the command line. `sanconf` creates only one logical library per physical library in the system and all devices on all specified clients. If the *RoboticControlHostName* parameter is specified, the specified client will control the robotics for the library being configured. If this parameter is not specified, the library will be created with robotics on all clients within the Data Protector cell, the Cell Manager will be used as a control host. If no library is installed on the Cell Manager in a multipath library, another client will be used as a control host.

When the *DeviceTypeNumber* parameter is used, the drives of that type will be configured in the library. When *DeviceTypeNumber* is not specified, the DLT drive types are used as the default. Only one type number may be specified per library. If you use the "*.DeviceTypeExtension*" parameter instead of the *DeviceTypeNumber* parameter, you can specify the device type extension of the tape device to be configured in the library.

Device Type Number	Device Type Extension

1	.DDS
2	.QIC
3	.EXA
4	.AIT
5	.3480
6	.RDSK
7	.REGFILE
8	.9840
9	.TAPE
10	.DLT
11	.D3
12	.3590
13	.LTO
14	.SDLT
15	.VXA
16	.DTF
17	.9940
18	.SAIT
19	.3592

When drives in the library are not of the same type as specified, an error is reported.

`-drive_template` *`DriveTemplateFileName`* This option alters the default configuration of each tape device added to the library. You can alter the default configuration of the library only at the initial configuration. After the library is configured, you can no longer change the configuration of the library using the `sanconf` command.

The *`DriveTemplateFileName`* must be an ASCII file with one parameter specified per line.

Drive template supports the following parameters:

`VERIFY`

This parameter corresponds to the `CRC Check` option in the Data Protector GUI.

`CLEANME`

This parameter corresponds to the `Detect dirty drive` option in the Data Protector GUI.

`RESCAN`

This parameter corresponds to the `Rescan` option in the Data Protector GUI.

`-library_template` *`LibraryTemplateFileName`* This option alters the default configuration of the library. You can alter the default configuration of the library only at the initial configuration. After the library is configured, you can no longer change the configuration of the library using the `sanconf` command.

The *`LibraryTemplateFileName`* must be an ASCII file with one parameter specified per line.

Library template supports the following parameters:

`BARCODEREADER`

This parameter corresponds to the `Barcode reader support` option in the Data Protector GUI.

`BUSYDRIVETOSLOT`

This parameter corresponds to the `Busy drive handling: Eject medium` option in the Data Protector GUI.

`BUSYDRIVETOMAILSLOT`

This parameter corresponds to the `Busy drive handling: Eject medium to mail slot` option in the Data Protector GUI.

`-[no_]multipath` By default or if the `-no_multipath` option is given, `sanconf` does *not* configure multipath devices – a separate logical device will be configured for *each* path.

When reconfiguring a multipath library as a non-multipath library, only one path is created. Multipath drives contained inside a multipath library are not changed, while new drives are created. Only non-multipath drives are modified.

If the `-multipath` option is used, `sanconf` configures all paths pointing to a single physical device as a *single* multipath device.

When reconfiguring a non-multipath library as a multipath library, the library control host is used as the first path. Non-multipath drives are not changed or removed. Instead, new multipath drives are created. Only multipath drives are modified.

`-sanstableaddressing` Enables automatic discovery of changed SCSI addresses for the devices being configured.

`-remove_drives` *`LibraryName`* This option removes all tape devices in the specified library. If you want to remove drives on specific clients, you can use the `-hosts` *`host_1`* `[`*`host_2...`*`]` or the `-hostsfile` *`HostsFileName`* option. This command cannot be used together with the `-multipath` option. Drives that are configured as multipath drives are not removed.

Note: No rescanning is required for this operation.

`-remove_hosts` All paths containing the specified hosts are removed. However, if the specified hosts cover all paths of the library, no paths are not removed from this library, instead a warning is displayed.

To remove paths only from *multipath* devices, add the `-multipath` option.

To remove paths only from *non-multipath* devices, add the `-no_multipath` option.

To remove paths from *both*, multipath *and* non-multipath devices, run the command *without* the `-no_multipath` and `-multipath` options.

Note: No rescanning is required for this operation.

NOTES

This command is only available on Windows, HP-UX, and Solaris systems.

All drives created with the `sanconf` command are named automatically. Drive names must not be changed manually because the reconfiguration will not work. You must follow the drive naming convention.

- For *non-multipath* devices:

  `libname_index_host`

  `libname_index_busindex_host`

  The `busindex` number is used only if there is more than one path for the drive.

- For *multipath* devices:

  ```
  libname_index
  ```

EXAMPLES

The following examples illustrate how the sanconf command works.

1. To scan host(s) for robotic control(s) and tape device(s) and create a file that will be used by sanconf -configure, run the following command:

   ```
   sanconf -list device.list
   ```

   This will display the serial number for any library discovered in the SUMMARY REPORT.

2. To scan and configure a library using the library serial number, run the following command:

   ```
   sanconf -configure -library US9LS01033
   ```

3. To scan the specified clients and then create a logical library named "SAN_STORE" with robotics configured on client "host33" and drives for that library configured on clients "host01", "host02" and "host03", run the following command:

   ```
   sanconf -configure -library MPC0100013 SAN_STORE host33 -hosts
   host01 host02 host03
   ```

4. To scan the SAN environment for the configuration information on the specified clients "host01", "host02", "host03", and "host33" and save this information is into the mySAN.cfg file, run the following command:

   ```
   sanconf -list_devices mySAN.cfg -hosts host01 host02 host03 host33
   ```

5. To use information stored in the mySAN.cfg file and create a logical library named "SAN_STORE" with robotics configured on client host33 and drives for the library configured on clients "host01", "host02", and "host03", run the following command:

   ```
   sanconf -configure mySAN.cfg -library MPC0100013 SAN_STORE host33
   -hosts host01 host02 host03
   ```

6. To scan all clients in the cell and then create a logical library named “SAN_STORE” with robotics configured on client “host33” with the parameters specified in the files DriveTemplate.txt and LibraryTemplate.txt, run the following command:

```
sanconf -configure -library MPC0100013 SAN_STORE host33
-drive_template DriveTemplate.txt -library_template
LibraryTemplate.txt
```

7. To configure a tape library with the default tape device and library settings using the "device.list" file created by the example above, run the following command:

```
sanconf -configure device.list -library MPC0220423 myLib1
```

8. To configure a library with a specific drive type, run the following command:

```
sanconf -configure -library MPC0100013 SAN_STORE host33 ".9840"
-hosts host01 host02
```

   This command creates a library named “SAN_STORE” with robotics configured on client “host33” and STK drives configured on clients "host01" and "host02". The drives are named as follows:

   SAN_STORE_1_host01

   SAN_STORE_1_host02

   SAN_STORE_2_host01

   SAN_STORE_2_host02

9. To configure three libraries using the configuration options contained in the library template "myway", run the following commands:

```
sanconf -configure -library US9LS02033 mylib5 -library_template
myway
```

```
sanconf -configure -library US9LS02034 mylib6 -library_template
myway
```

```
sanconf -configure -library US9LS02035 mylib7 -library_template
myway
```

10. To configure a multipath LTO library with the serial number "LLL1", named "Library1", and connected to client "host1", run the following command:

```
sanconf -configure -library LLL1 Library1 host1 ".LTO" -multipath
```

11. To update an already configured library with the configuration information for new hosts or tape devices, run the following command:

```
sanconf -configure -library US9LS01023 mylib2
```

12. To reconfigure an already configured library after adding a new host "myhost" to a Data Protector cell, run the following command:

```
sanconf -configure -library US9LS01033 mylib2 -hosts myhost
```

This will scan and configure only the new host.

13. To configure only LTO Ultrium tape drives and add them into the library "myLTOlib", run the following commands:

```
sanconf -list device.list

sanconf -configure device.list -library MPC0230031 myLTOlib ".LTO"
```

14. To reconfigure a non-multipath library named "SAN_STORE" with serial number "MPC0100013" to a multipath library using the -hosts option, when new clients "host04" and "host05" are added to the cell, run the following command:

```
sanconf -configure -library MPC0100013 SAN_STORE host33 -hosts
host04 host05 -multipath
```

15. To delete all tape drives configured in the library "mylib2" related to the clients "host04" and "host05", run the following command:

```
sanconf -remove_drives mylib2 -hosts host04 host05
```

16. To delete all tape drives configured in the library "mylib2", run the following command:

```
sanconf -remove_drives mylib2
```

17. To remove all paths in the multipath library named "SAN_STORE" that are configured on clients "host04" and "host05", run the following command:

```
sanconf -remove_hosts SAN_STORE -hosts host04 host05 -multipath
```

SEE ALSO

omniamo(1), omnidownload(1), omnimcopy(1), omniminit(1), omnimlist(1), omnimm(1), omnimnt(1), omnimver(1), omniupload(1), uma(1M)

uma (1M)

NAME

uma – controls the robotics of SCSI compliant autochangers

SYNOPSIS

```
uma -version | -help
```

```
uma [-policy LogicalDevicePolicy] -ioctl deviceFile [-interface
   {0 | 1}] [-tty] [-barcode] [-device deviceFile_1
   [deviceFile_n] -type DeviceType] [-ddt NDMP_server_name
   NDMP_port_number backup_type username password]
```

Uma command line interface commands:

```
help

inq

init

addr

offl driveID

sense

pos slot

move source_slot destination_slot [0 | 1]

stat [{slot | drive | transport_element | mail_slot}]

modesense [page]

test

bye | exit | quit

doorlock [0 | 1]

enter slot

eject slot
```

DESCRIPTION

The uma program is a standalone utility program which can be used to control the robotics of most SCSI compliant autochangers, also those which are not directly supported by Data Protector. It implements a shell-like user command interface and can be used both interactively and in batch mode.

Uma is packaged and installed as part of a Data Protector Media Agent fileset. If you have received uma as a standalone program or if you run it on a system where Data Protector has not been installed, the uma command is fully functional and behave as documented, but it is probably not able to locate and use Data Protector NLS message catalog.

On HP-UX and Solaris systems, uma is located in /opt/omni/lbin/ directory, and the Data Protector NLS message catalog is located in the /opt/omni/lib/nls/C/ directory.

On other UNIX systems, uma is located in /usr/omni/bin/ directory, and the Data Protector NLS message catalog is located in the /usr/omni/lib/nls/C/ directory.

On Windows systems, uma is located in *<Data_Protector_home>*\bin directory, and the Data Protector NLS message catalog is located in the *<Data_Protector_home>*\bin\ directory.

Uma can be started both interactively or in batch mode. The only obligatory option is the pathname of the device file (UNIX systems) or the SCSI address (Windows systems) that controls the robotics of the target autochanger (the -ioclt option). For backup devices with library robotics connected to an NDMP Server (to a supported NAS device), the -interface and the -ddt options must also be specified.

FNDMPor your convenience, the uma command allows you to specify symbolic instead of physical element addresses (slot IDs). Whenever you need to refer to the 1st drive of the autochanger, you can specify either the physical address '128' or the more convenient, symbolic 'D1'. The output of the addr command reflects this addressing convention.

OPTIONS

-version Displays the version of the uma command

-help Displays the usage synopsis for the uma command

-policy *LogicalDevicePolicy* Specifies the backup device policy ID. Policy can be defined as 6 (external control), 8 (Grau DAS exchanger library), 9 (STK Silo medium library) or 10 (SCSI Library).

The default value for the -policy option is 10.

`-ioctl` *`deviceFile`* Specifies the pathname of the device file (UNIX systems) or the robotics SCSI address (Windows systems) that controls the robotics of the target autochanger.

`-interface {0 | 1}` Sets the type of SCSI interface used to access library robotics. This option is to be used only with backup devices with library robotics connected to an NDMP Server. `0` sets the standard SCSI interface (the default value). `1` sets the NDMP protocol interface and must be specified for backup devices with library robotics connected to an NDMP Server. The default value is `0`.

`-tty` Forces the `uma` command to enter the command line interface mode or to read from script. This option is obligatory on UNIX, MPE and NetWare systems. On Windows systems, this option is not to be used; the command line interface mode is invoked automatically.

`-barcode` If this variable is set, the `uma` command's `stat` command displays also the barcode information for each medium.

`-device` *`deviceFile_1`* `[`*`deviceFile_n`*`...]` Specifies the device file (UNIX systems) or the SCSI address (Windows systems) of one or more autochanger drives. For a multi-drive autochanger, you must specify a list of device files/SCSI addresses which correspond to the autochanger's drives in ascending order. The drives have to be known to `uma` in order for the `offl` command to work. This option is only to be used together with `-type` option.

`-type` *`DeviceType`* Specifies the media type for the media in the device specified by the `-device` option. Media type numbers are defined in the *HP OpenView Storage Data Protector Software Release Notes*.

`-ddt` *`NDMP_server_name NDMP_port_number backup_type username password`* This option is obligatory for backup devices with library robotics connected to an NDMP Server (to a supported NAS device). It specifies the NDMP Server name, port number used by Data Protector to connect to the NDMP Server and username and password used by Data Protector to connect to the NDMP server. The *`backup_type`* parameter has to be set to `dump`.

`Uma` command line interface commands:

`help` Displays the usage synopsis for the `uma` command

`inq` Performs a SCSI Inquiry operation on the device file/SCSI address specified with the `-ioctl` option. It returns the device's type, vendor ID, product ID and firmware revision number.

`init` Performs a SCSI 'initialize element status' operation, which (if applied to an autochanger robotic device) forces the autochanger to reset its internal state and perform an inventory of its repository. This command should not be used if another process is accessing the autochanger at the same time, as the effects are unpredictable.

`addr` Queries and displays the autochanger's element assignment page. Each addressable item inside the autochanger mechanism (drive, repository slot, robotic arm, import/export slot) has a unique integer number (slot ID) which can be used to address this specific item.

As the element assignment differs among different autochangers, the software, which is to control the movement of media inside the autochanger, must find out and use these numbers to perform move, pos and stat operations.

`offl` *`driveID`* This command can be used only if at least one drive was specified using the `-device` option. If a medium is loaded in the specified drive, it will eject the medium just as if an UNIX `mt offl` command was specified. The mandatory argument is a symbolic drive ID (i.e. D3 for the 3rd drive == the 3rd device file specified with the `-dev` option). If the drive specified is not defined by the `-device` option, then the last drive defined by the `-device` option will be used.

The `offl` command can fail with a message: `"No such device or address"` if it is issued immediately after the `move` command since it takes a certain time after the `move` command for the drive to be online. Refer to the `move` command for more information.

`sense` Read the device's sense data and dump them in a hex- dump format.

`pos` *`slot`* Positions the autochanger transport mechanism in front of the specified slot. This operation is only meaningful if the specified slot refers to an import/export, data drive or repository element. The actual meaning of this operation may differ among different autochanger models. This command is generally not required, but is provided for testing purposes and convenience. Both physical as well as symbolic slot addressing may be used.

`move` *`source_slot destination_slot`* `[0 | 1]` Moves a medium from a source slot into a destination slot. This command has two mandatory arguments, the source and destination slot IDs (address numbers, as reported by the `addr` command described above) and an optional numeric Boolean argument which can be used to instruct the robotics to flip the medium before inserting it into the destination slot. By default (if no flipping argument is specified), flipping is disabled.

Note that when `move` command is issued to move a tape into a drive, it takes a certain time (around 30 seconds) for the drive to become online, because tape load and calibration/selftest have to be performed. The command prompt however, returns immediately after the command is issued.

NOTE: Flipping is supported only for double-sided optical media. For tapes, the effect of the flip command is not defined.

NOTE: Most autochanger do not allow you to `move` a tape from a drive to a repository location if the tape has not been dismounted and ejected by the drive. You might want to use the `offl` command on the drive device file/SCSI address to put the drive off-line before executing the `move` command.

`stat [{`*`slot`*` | `*`drive`*` | `*`transport_element`*` | `*`mail_slot`*`}]` Queries the device for information about the state of each of its addressable elements. The output of this command is a table of physical and symbolic element IDs and their states, indicating which elements are free (Empty) and occupied (Full).

Additionally, if barcode support is available and enabled, the barcode for each medium is displayed.

The `uma` command recognizes one specific environment variable which can be used to enable barcode support for autochangers which are equipped with barcode reading hardware. By default, `uma` barcode support is disabled. It can be enabled by exporting/ setting the *OB2BARCODE=1* environment variable before starting the command or by using the `-barcode` option.

The `stat` command can be used to query the status of a specific slot (i.e. 'stat 290' or 'stat S35') or a related group of slots (i.e. 'stat D' will query all drives, 'stat S' will query all repository slots, etc.).

If no additional arguments are specified, the stat command will query and print the status information for all slot IDs it can address.

`modesense [`*`page`*`]` Reads the vendor specific data and unit settings from the unit and displays them. You can limit the display only to certain pages by using the page parameter. If the page parameter is not specified, all pages are displayed.

`test` Checks if the unit is ready. If the unit is not used by any process, then the unit is ready. If it is, however, used either by the robotics, backup or restore processes then it is not ready.

`exit | bye | quit` Exits the command mode.

`doorlock [0 | 1]` If the input parameter is `1`, this command locks the library mail slot door; if it is `0`, it unlocks it.

`-enter` *`slot`* Enters media into a specified library slot.

`-eject` *`slot`* Ejects media form a specified library slot.

NOTES

Do not use the `uma` utility while Data Protector backup or restore is running. On UNIX, MPE and NetWare systems the `-tty` option is obligatory. On Windows systems it is not used.

EXAMPLES

1. `Uma` can be started both interactively or in batch mode. The only option which needs to be specified (except for backup devices with library robotics connected to an NDMP Server) is the pathname of the device file which controls the robotics of the target autochthons:

```
AMA -octal /Dave/spot/sctl0

*** PROGRAM: UMA VERSION: HP OpenView Storage Data Protector
A.06.00

*** Copyright (C) 1999 Hewlett-Packard Company

*** License is restricted for use with licensed

*** HP OpenView Storage Data Protector products.

/dev/spt/sctl0> exit
```

2. To start uma for a backup device with the library robotics connected to the NDMP Server with the robotics SCSI address “mc2”, the NDMP Server hostname “ndmpserver”, the port number used by Data Protector to connect to the NDMP Server “10000”, and username and password of the user used by Data Protector to connect to the NDMP Server “user password”, enter the following command:

   ```
   UMA -ioctl mc2 -interface 1 -ddt ndmpserver 10000 dump user password
   ```

3. To let uma execute a batch script of its own commands, simply redirect its stdin to a file containing a list of uma commands separated with newlines:

   ```
   cat >/tmp/cmdFile

   inq

   addr

   stat

   <ctrl-D>

   uma -ioctl /dev/spt/sctl0 </tmp/cmdFile >/tmp/outFile
   ```

4. The following output is obtained by executing the addr command on the UNIX device file referring to an ACL 4/52 DLT autochanger:

   ```
   /dev/spt/sctl0> addr Element Addresses (T=Transport, X=Im/Export,
   D=Drive, S=Storage):

   Transport:  1 .. 1  (T1 .. T1)

   Im/Export:  64 .. 67  (X1 .. X4)

   Data Drive(s): 128 .. 131  (D1 .. D4)

   Repository:  256 .. 303  (S1 .. S48)
   ```

   The numbers returned by the addr command are the physical element addresses of different elements within the autochanger - i.e. element address “256” would correspond to the 1st repository slot, element address “65” would correspond to the location of the 2nd data drive etc.

5. To start uma for the Grau DAS exchanger library with the robotics device file “grauamu”, enter the following command:

   ```
   uma -pol 8 -ioctl grauamu
   ```

SEE ALSO

omniamo(1), omnidownload(1), omnimcopy(1), omniminit(1), omnimlist(1), omnimm(1), omnimnt(1), omnimver(1), omniupload(1), sanconf(1M)

upgrade_cfg_from_evaa (1M)

NAME

upgrade_cfg_from_evaa – runs the cluster nodes upgrade when upgrading from the HP StorageWorks EVA Agent (legacy) to the HP StorageWorks EVA SMI-S Agent.

SYNOPSIS

```
upgrade_cfg_from_evaa -version | -help
```

`upgrade_cfg_from_evaa` *`virtual_hostname`*

DESCRIPTION

The `upgrade_cfg_from_evaa` command needs to be executed on any cluster node after the upgrade from the EVA Agent (legacy) to the EVA SMI-S Agent. It connects to CRS, reads the information on the backup sessions created using the EVA Agent (legacy), writes these entries to the SMISDB, and deletes the entries from the EVADB.

OPTIONS

`-version` Displays the version of the `upgrade_cfg_from_evaa` command.

`-help` Displays the usage synopsis for the `upgrade_cfg_from_evaa` command.

`virtual_hostname` Sets the virtual hostname of the cluster that is to be upgraded.

SEE ALSO

omnigui(5), omniintro(9), ob2install(1M), omnisetup.sh(1M), omnimigrate.sh(1M), omniusers(1)

util_cmd (1M)

NAME

util_cmd – sets, retrieves or lists the parameters stored in the Data Protector Oracle and SAP R/3 configuration files.

SYNOPSIS

```
util_cmd -version | -help

util_cmd -getconf[ig] integration oracle_SID [-local filename]

util_cmd -getopt[ion] [integration oracle_SID] option_name
    [-sub[list] sublist_name ] [-local filename]

util_cmd -putopt[ion] [integration oracle_SID] option_name
    [option_value] [-sub[list] sublist_name ] [-local filename]

 integration:  {Oracle8 | SAP}
```

DESCRIPTION

The `util_cmd` command is used to set, retrieve or list the parameters stored in the Data Protector Oracle and SAP R/3 configuration files.

Data Protector stores Oracle integration parameters in two files on the Cell Manager:

- For every configured Oracle instance in the: `/etc/opt/omni/server/integ/config/Oracle8/`*`<client_name>`*`%`*`<ORACLE_SID>`* file (UNIX systems), or in the *`<Data_Protector_home>`*`\Config\server\integ\config\oracle8\`*`<client_name>`*`%`*`<ORACLE_SID>`* file (Windows systems).

  The parameters stored in the instance configuration file are:

  - Oracle home directory

  - Oracle version

  - encoded connection strings to the target database and recovery catalog

  - the variables which need to be exported prior to starting a backup and which affect the Oracle instance.

- Oracle global integration parameters in the:`/etc/opt/omni/integ/server/config/Oracle8/`*`<client_name>`*`%_OB2_GLOBAL` file (UNIX systems), or in the *`<Data_Protector_home>`*`\Config\server\integ\config\oracle8\`*`<client_name>`*`%_OB2_GLOBAL` file (Windows systems).

  The parameters stored in the global configuration file are:

  - instance list (all Oracle instances on the Oracle server)

  - variables that need to be exported prior to starting a backup and which affect every Oracle instance on the Oracle server.

Data Protector stores the SAP R/3 integration parameters in the `/etc/opt/omni/server/integ/config/SAP/`*`<client_name>`*`%`*`<ORACLE_SID>`* file (UNIX systems), or in the
`<Data_Protector_home>``\Config\server\integ\config\SAP\`*`<client_name>`*`%`*`<ORACLE_SID>`* file (Windows systems).

The SAP R/3 parameters stored are:

- Oracle home directory

- encoded connection string to the target database

- BRTOOLS home directory

- the variables which need to be exported prior to starting a backup

- concurrency number and balancing (for each backup specification) and number of channels for RMAN backup

- speed parameters (time needed for a specific file to back up - in seconds)

- manual balancing parameters.

The Data Protector Oracle and SAP R/3 configuration parameters are normally written to the Data Protector configuration files:

- during the configuration of the integration
- during the creation of a backup specification
- when the configuration parameters are changed

All sublist configuration parameters in the configuration files are optional.

RETURN VALUES

The `util_cmd` command displays a short status message after each operation (written to the standard error):

- `Configuration read/write operation successful.`

  This message is displayed when all the requested operations have been completed successfully.

- `Configuration option/file not found.`

  This message appears when either an option with the specified name does not exist in the configuration, or the file specified as the `-local` parameter does not exist.

- `Configuration read/write operation failed.`

  This message is displayed if any fatal errors occurred, for example: the Cell Manager is unavailable, the Data Protector Oracle or SAP R/3 configuration file is missing on the Cell Manager, etc...

OPTIONS

`-version` Displays the version of the `util_cmd` command.

`-help` Displays the usage synopsis for the `util_cmd` command.

`-getconf` *`integration oracle_SID`* Lists the Data Protector configuration files parameters for the specified integration (Oracle or SAP R/3) and instance to the standard output, unless the `-local` option is specified.

`-getopt` [*`integration oracle_SID`*] *`option_name`* Retrieves the parameter (specified by the *`option_name)`* and its value from one of Data Protector configuration files and writes it to the standard output, unless the `-local` option is specified.

`-putoption` [*`integration oracle_SID`*] *`option_name`* [*`option_value]`* Sets the specified parameter (specified by the *`option_name)`* and (optionally) its value to the Data Protector configuration files, unless the `-local` option is used.

`-sublist` *`SublistName`* Specifies the sublist in the configuration file in which a parameter is written to or taken from.

`-local` *`FileName`* If the `-local` option is used with the `-getconf` option, the command output is written to the file with the filename specified by the `-local` option. If the `-local` option is used with the `-getopt` option, the parameter and its value is taken from the file with the filename specified by the `-local` option. If the `-local` option is used with the `-putoption` option, the parameter and its value is written to the file with the filename specified by the `-local` option.

EXAMPLES

The following examples illustrate how the `util_cmd` command works.

1. To set the “OB2OPTS” and “NLS_LANG” parameters for the Oracle instance “ICE”, use the following commands:

   ```
   util_cmd -putopt Oracle8 ICE OB2OPTS “-debug 1-200 INSTANCE.txt”
   -sublist Environment
   ```

   ```
   util_cmd -putopt Oracle8 ICE NLS_LANG AMERICAN_AMERICA.US7ASCII
   -sublist Environment
   ```

2. To set the Data Protector “OB2OPTS” and “NLS_LANG” parameters for the SAP R/3 instance “ICE”, use the following commands on the Data Protector SAP R/3 client:

   ```
   util_cmd -putopt SAP ICE OB2OPTS'-debug 1-200 INSTANCE.txt'
   -sublist Environment
   ```

   ```
   util_cmd -putopt SAP ICE NLS_LANG 'AMERICAN_AMERICA.US7ASCII'
   -sublist Environment
   ```

3. To set the “BR_TRACE” parameter for the SAP R/3 instance “ICE” to value “10” in the “Environment” sublist, use the following commands on the Data Protector SAP R/3 client:

   ```
   util_cmd -putopt SAP ICE BR_TRACE “‘10’” ‘sublist Environment
   ```

4. To list the Data Protector configuration file parameters for the Oracle instance “ICE”, use the following command:

   ```
   util_cmd -getconf Oracle8 ICE
   ```

5. To retrieve the value of the “OB2OPTS” parameter for the Oracle instance “ICE”, use the following command:

   ```
   util_cmd -getopt Oracle8 ICE OB2OPTS -sublist Environment
   ```

6. To remove the value of the “OB2OPTS” parameter for the SAP R/3 instance “ICE”, use the following command on the Data Protector SAP R/3 client:

   ```
   util_cmd -putopt SAP ICE OB2PTS -sublist Environment
   ```

SEE ALSO

omnib(1), omnicreatedl(1)

Section 5: Miscellaneous

omnigui (5)

NAME

omnigui – Describes usage for the Data Protector Windows commands manager and mom and the following Data Protector UNIX commands: xomni, xomnimom, xomniadmin, xomnibackup, xomnimm, xomnimonitor, xomnirestore, xomnicellmon, xomniinstrec

SYNOPSIS

```
 GUI_command [-help]

 manager [ContextOptions] [-server hostname]

 mom [ContextOptions] [-server hostname]

 xomni [ContextOptions] [-server hostname] [-display hostname:0]

 xomnimom [ContextOptions] [-server hostname] [-display
    hostname:0]

ContextOptions

 -admin

 -backup

 -clients

 -copy

 -db

 -monitor

 -report

 -restore

 -users

 -instrec
```

DESCRIPTION

These commands are used to start all or any combination of Data Protector GUI contexts.

To use the Data Protector GUI functionality on those UNIX Cell Manager platforms where the Data Protector GUI is not supported, use the `omniusers` command to create a remote user account on the Cell Manager. You can then use the created user account on any other system with the Data Protector GUI installed to start the GUI and connect to the Cell Manager. Refer to the `omniusers` man page for details, and to the *HP OpenView Storage Data Protector Software Release Notes* for details on supported operating system versions/releases for the user interface. For more information on local language support and the usage of non-ASCII characters in file names, refer to *HP OpenView Storage Data Protector Administrator's Guide*.

COMMANDS

UNIX commands

`xomni` starts the Data Protector GUI with all Data Protector contexts activated, or, when additional options are specified, starts only the specified Data Protector context(s)

`xomnimom` starts the Data Protector Manager-of-Managers GUI with all Data Protector contexts activated (with the exception of Internal Database and Devices & Media contexts), or, when additional context options are specified, it starts only the specified Data Protector context(s)

`xomniadmin` starts the Data Protector GUI with the Clients, Users, Reporting and Internal Database contexts activated

`xomnibackup` starts the Data Protector GUI with the Backup context activated

`xomnicellmon` starts the Data Protector GUI with the MoM cell monitoring GUI activated

`xomnicopy` starts the Data Protector GUI with the Copy & Consolidation context activated

`xomnimm` starts the Data Protector GUI with the Devices & Media context activated

`xomnimonitor` starts the Data Protector GUI with the Monitor context activated

`xomnirestore` starts the Data Protector GUI with the Restore context activated

`xomniinstrec` starts the Data Protector GUI with the Instant Recovery context activated

Windows commands

`manager` starts the Data Protector GUI with all Data Protector contexts activated, or, when additional options are specified, starts only the specified Data Protector context(s)

`mom` starts the Data Protector Manager-of-Managers GUI with all Data Protector contexts activated (with the exception of Internal Database and Devices & Media contexts), or, when additional context options are specified, it starts only the specified Data Protector context(s)

OPTIONS

`-help` Displays the usage synopsis for the command.

`-server` *`hostname`* Connects to the specified Cell Manager.

`-display` *`hostname:0`* Redirects the output to the display on the specified system.

`-admin` Starts the Data Protector GUI with the Devices & Media contexts activated.

`-backup` Starts the Data Protector GUI with the Backup context activated.

`-clients` Starts the Data Protector GUI with the Clients context activated.

`-copy` Starts the Data Protector GUI with the Copy & Consolidation context activated.

`-db` Starts the Data Protector GUI with the Internal Database context activated.

`-instrec` Starts the Data Protector GUI with the Instant Recovery context activated.

`-monitor` Starts the Data Protector GUI with the Monitor context activated.

`-report` Starts the Data Protector GUI with the Reporting context activated.

`-restore` Starts the Data Protector GUI with the Restore context activated.

`-users` Starts the Data Protector GUI with the Users context activated.

EXAMPLES

1. `xomni -display host1:0`

   This UNIX command will start the Data Protector GUI with all contexts activated on the system with the hostname “host1”.

2. `manager`

   This Windows command will start the Data Protector GUI with all contexts activated.

3. `xomni -admin -monitor -report -server host2`

   This UNIX command will start the Data Protector GUI with the Devices & Media, Monitor and Reporting contexts activated and will connect to the Cell Manager with the hostname “host2”.

4. `manager -admin -monitor -report -server host3`

   This Windows command will start the Data Protector GUI with the Devices & Media, Monitor and Reporting contexts activated and will connect to the Cell Manager with the hostname “host3”.

SEE ALSO

omniintro(5), ob2install(1M), omnisetup.sh(1M), omnimigrate.sh(1M), omniusers(1), upgrade_cfg_from_evaa(1M)