

November 2001  
15VZ-1101A-WWEN

Prepared by Enterprise Storage  
Group

Compaq Computer Corporation

**Contents**

|  |           |
|--|-----------|
| <b>Introduction.....</b>                               | <b>3</b>  |
| Oracle 9i Database Server .....                        | 3         |
| SANworks Enterprise<br>Volume Manager.....             | 3         |
| What is a Data Warehouse? .....                        | 3         |
| Additional References .....                            | 4         |
| Key Terms.....   | 4         |
| <b>Using Clones.....</b>                               | <b>6</b>  |
| <b>Using Snapshots .....</b>                           | <b>7</b>  |
| <b>Transporting Oracle 9i Data.....</b>                | <b>8</b>  |
| Point-in-time Copies.....                              | 8         |
| External Tables and Flat<br>Files .....                | 9         |
| Transportable Tablespaces.....                         | 12        |
| <b>Using EVM with Data Marts .....</b>                 | <b>13</b> |
| What is a Data Mart? .....                             | 13        |
| Moving Data from the<br>Warehouse to a Data Mart ..... | 14        |
| <b>Conclusions .....</b>                               | <b>14</b> |
| <b>Contact Us.....</b>                                 | <b>15</b> |

# Compaq SANworks™ Enterprise Volume Manager version 2.0 and Data Warehousing using Oracle 9i

*Abstract:* This paper addresses using Compaq SANworks™ Enterprise Volume Manager (EVM) version 2.0 to improve the extraction and transportation procedures for Oracle 9i databases. Data warehouses typically use an Extract, Transport, Transform and Load (ETL) methodology that involves several lengthy and time-consuming steps, such as data exports and network copies. This paper will show how to use clones and snapshots to improve these processes in conjunction with Oracle transportable tablespaces and how to create Point-In-Time Copies (PITC) of Oracle 9i Databases. These processes are equally applicable to Oracle 8i databases except where noted.

## Notice

© 2001 Compaq Information Technologies Group, L.P.

Compaq and the Compaq logo are trademarks of Compaq Information Technologies Group, L.P. in the U.S. and/or other countries. Microsoft, Windows, and Windows NT are trademarks of Microsoft Corporation in the U.S. and/or other countries. All other product names mentioned herein may be trademarks of their respective companies.

Compaq shall not be liable for technical or editorial errors or omissions contained herein. The information is provided “as is” without warranty of any kind and is subject to change without notice. The warranties for Compaq products are set forth in the express limited warranty statements accompanying such products. Nothing herein should be construed as constituting an additional warranty.

Compaq SANworks Enterprise Volume Manager 2.0 and Data Warehousing using Oracle 9i  
Solution Guide prepared by Enterprise Storage Group

First Edition (October 2001)  
Document Number 15VZ-1101A-WWEN

# Introduction

## Oracle 9i Database Server

Oracle9i is the complete Internet infrastructure that helps you deploy applications faster while lowering your cost of computing. The Oracle9i Database delivers the industry's highest performance, reliability, and security for thousands of popular applications.

## SANworks Enterprise Volume Manager

Compaq SANworks™ Enterprise Volume Manager (EVM) version 2.0 delivers more of the precious resource that no one seems to have enough of: Time. EVM customers have found that backing up their production data from point-in-time copies allows them to keep their business-critical applications available during the backup process. They have also found value in using snapshots and clones for application testing and data migration. This paper will demonstrate how to use EVM version 2.0 to speed up the data migration and movement necessary for data warehouse environments.

## What is a Data Warehouse?

A data warehouse is a relational database that is designed for analysis rather than transaction processing. A data warehouse usually contains historical data that is derived from transaction data, as well as from other sources. The warehouse separates the analysis workload from transaction workloads and consolidates data from several sources.

In addition to the database, a data warehouse environment usually implements an Extract, Transport, Transform and Load (ETL) methodology. The warehouse contains an online analytical processing (OLAP) engine and client analysis tools. Other applications and processes typically manage data collection and delivery. SANworks Enterprise Volume Manager version 2.0 offers a significant improvement to the process of gathering and distributing that information.

Data warehouse extracts typically enable the database to continue to operate while the extract is used for analysis or modeling. For example, a subset of a customer database could be extracted to create customer profiles for various purposes such as targeted marketing. Extracts can also model buying patterns of products during various times of the year. The result will be better optimization of purchasing to supply raw materials for the products purchased during a particular time of year. Using EVM version 2.0, the extracts can be done repeatedly--even during peak hours--to optimize the extracted data without disrupting the production database.

---

**Note:** Many resources address designing and implementing data warehouses. This paper will not attempt to duplicate any of those efforts.

---

## Additional References

For more information, see the following:

*Enterprise Volume Manager and Oracle8 Best Practices* available at:

<http://www.compaq.com/products/sanworks/evm/index.html>

Under **Product Information**, select **Solution Guides**.

*Compaq SANworks Enterprise Volume Manager Guide to the Oracle Storage Compatibility Program* available at:

<http://www.compaq.com/products/sanworks/evm/index.html>

Under **Product Information**, select **Solution Guides**.

*Compaq SANworks™ Enterprise Volume Manager for Compaq Tru64 Unix Installation Guide* available at:

<http://www.compaq.com/products/sanworks/evm/index.html>

Under **Product Support**, select **Technical Documentation**.

*Guidelines for Using Snapshot Storage Systems for Oracle Database* by Nabil Osorio and Bill Lee, Oracle Corporation

*Oracle 8 Backup and Recovery Handbook* by Rama Velpuri and Anand Adkoli, Oracle Press

*Oracle9i Data Warehousing Guide* available at <http://technet.oracle.com/>

*Oracle 9i Database Utilities* available at <http://technet.oracle.com/>

## Key Terms

This document uses the following terms and definitions:

**Business Continuance Volume (BCV)** - A generic term for a storage unit that is created from a standard production unit. A BCV unit can subsequently be used in production tasks, such as backups, application migration, data mining, and testing. A key feature of a BCV unit is that it can be accessed while the production volume remains online for normal I/O. The term BCV unit does not imply the use of any specific technology.

**Clone** - The capability to produce a disk image by breaking off a mirrored member of a mirror set.

**Data Manipulation Language (DML)** – Structured Query Language (SQL) statements that manipulate data. For example, SELECT, INSERT, UPDATE, and DELETE.

**Logical Unit Number (LUN)** - A number that distinguishes distinct physical or logical disk drives on a SCSI-based controller.

**Primary Database** - The production database that receives changes.

**Recovery** - Restoring an Oracle Database to the point-in-time of failure. Oracle can be set up so that no single failure will cause a loss of data.

**Redundant Array of Independent Disks (RAID)** - Storing the same data in different places (thus, redundantly) on multiple hard disks. By placing data on multiple disks, I/O operations can overlap in a balanced way, improving performance. Therefore, storing data redundantly increases fault tolerance.

**Rolling Clone** – Like a clone, this is a mirror member separated from a mirror set. However, a rolling clone switches the separated member between two or more disks, allowing for multiple near-instantaneous splits of mirror members. When it is time to refresh the BCV, the members “roll” back and forth so that one member is normalizing while the other is split off. This process completely eliminates the time to normalize the clone before splitting it off.

**Secondary Database** - A copy of a production Oracle database used for reporting purposes.

**Small Computer Systems Interface (SCSI)** - An interface used to communicate to storage, either direct attached or networked.

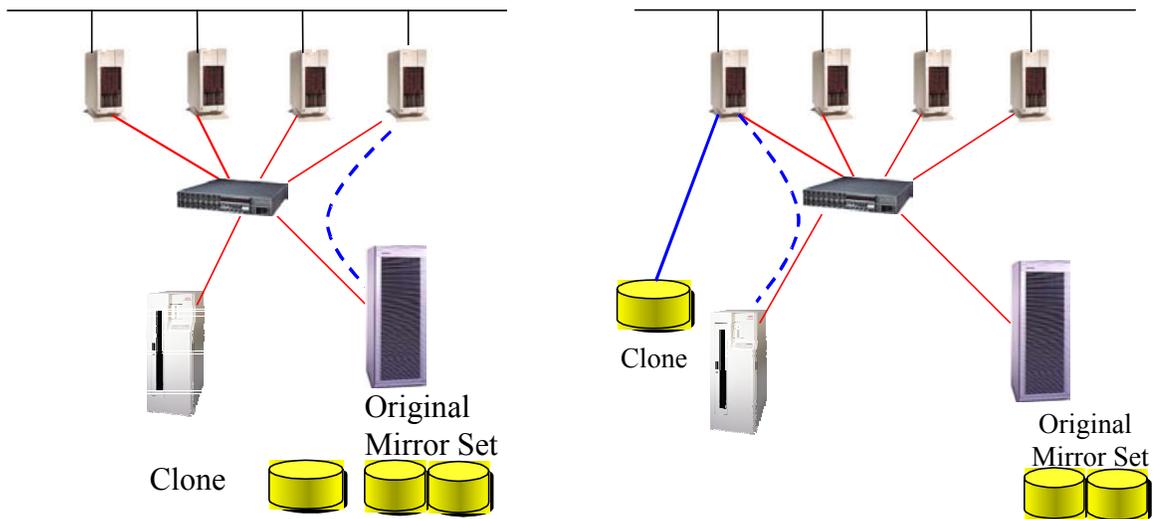
**Snapshot** – Sometimes referred to as *write on update* in the test documentation. The snapshot records when it is broken off, but does not make a full copy of the data at that point. As blocks are changed, prior to the change, a before-image copy of the blocks is stored in the snapshot area.

**Storage Area Network (SAN)** - A high-speed special-purpose network that interconnects different kinds of data storage devices with an associated data server on behalf of a larger network of users.

**System Change Number (SCN)** - The System Change Number is a crucial data structure that defines a committed version of the database at a precise moment in time. When a transaction commits, it is assigned an SCN that uniquely identifies the transaction.

## Using Clones

A clone, often referred to as a third mirror, is a point-in-time copy of a drive or a set of drives. Clones can be made from single drives, stripe sets, or mirror sets as well as striped and mirrored disk configurations. When a clone is initiated, the controller creates a copy of the existing logical unit onto a similarly configured new set of drives and gives the clone a new logical unit number. If the original LUN has three or more mirror members (the level of redundancy is actually definable as part of an EVM 2.0 job), then one of those members is split off immediately without adding a new member. In either case, once the clone is complete, the new LUN can be mounted on either the local or a remote host. At this point the drive is an independent entity and can be modified, copied, or backed-up. Figure 1 shows how the cloning process is accomplished.



**Figure 1: The Cloning Process Before and After Split**

In this example, a clone is created from a mirror set. Once the cloning process completes, the clone is mounted on a different server within the SAN and backed-up to tape.

The main advantages to clones are:

- Clones remove spindle contention for the drives by copying data to a completely different set of drives.
- Clones provide a level of high availability because they are an independent drive with a point-in-time copy of your data. This copy can be used to restore the data in case of a problem with the data on the original drives.
- Only the number of drives attached to the controller pair limits the number of clones.

## Using Snapshots

Snapshots differ from clones in that the data is not physically copied to other disks from the outset. When a snapshot is initiated, information about the disk (metadata) is recorded in cache, and a new LUN is created. From this point on, when data on the original LUN is modified, the original data blocks (also referred to as a *before image*) are recorded on the snapshot disks, and the new data is written to the original LUN. The controller then uses the metadata to track modifications between the two LUNs.

There are several advantages to this approach:

- Unlike a clone, there is no delay to create a full copy.
- Only the changed blocks are actually copied to the snapshot LUN.
- Snapshots do not have to renormalize when the process is complete.

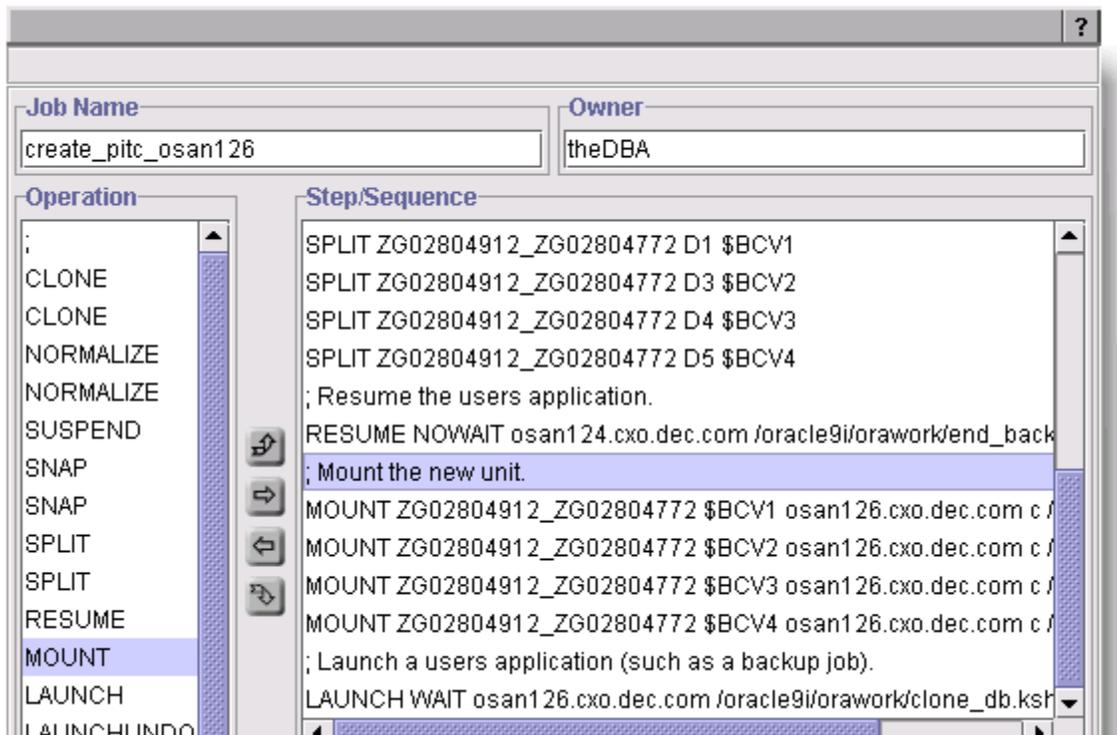
Other than these differences, snapshots can be treated the same as clones. Snapshots can be used for backups, and both the original LUN and snapshot LUN can be updated. However, because snapshots depend on the original drives, they are not suitable for high-availability solutions.

# Transporting Oracle 9i Data

## Point-in-Time Copies

A Point-In-Time Copy (PITC) of an Oracle 9i database is an exact copy of the database frozen in time. PITCs can be used for backups, reporting, ad hoc query, reconciliation, and many other business-critical functions. Although technically not a data warehouse, a PITC still satisfies the need to offload reporting and other time-sensitive functions to a separate server. Because PITCs typically receive a lot of activity and may span many LUNs, Compaq recommends using EVM's cloning capabilities to create the PITCs.

A portion of a sample clone job to copy all database drives (four in this case) is shown below.



**Figure 2: Clone All Database Drives**

Make sure all data files, redo logs, and archive logs are on drives that are capable of being cloned. (For example, Raid 0, 1 or 0+1). The control files should not be cloned; rather, use the `alter database backup controlfile to trace` command. If control files are copied, inconsistencies with the system change number can occur, preventing the PITC database from starting. Edit the trace file to provide the necessary steps to create a new control file and copy the file to the remote server.

```
SQL> connect sys/change_on_install@dw as sysdba
Connected.
SQL> alter database backup controlfile to trace;
Database altered.
```

Note that if the destination uses different path names, the *create controlfile* script should be modified to reflect this difference. For example, if the Oracle mount points on the production server are /oradata1, /oradata2, /oradata3, and /oradata4, but the point-in-time copy mounts on /oradata1BCV, /oradata2BCV, /oradata3BCV, and /oradata4BCV, the *create controlfile* command will need to be changed to reflect this difference; otherwise, the point-in-time copy will not start.

```
STARTUP NOMOUNT pfile=?/admin/dw/init.ora
CREATE CONTROLFILE REUSE DATABASE "DW" NORESETLOGS ARCHIVELOG
  MAXLOGFILES 5
  MAXLOGMEMBERS 5
  MAXDATAFILES 100
  MAXINSTANCES 1
  MAXLOGHISTORY 454
LOGFILE
  GROUP 1 '/oralog1BCV/redo01.log'  SIZE 100M,
  GROUP 2 '/oralog2BCV/redo02.log'  SIZE 100M,
  GROUP 3 '/oralog3BCV/redo03.log'  SIZE 100M
DATAFILE
  '/oradata1BCV/dw/system01.dbf', # was /oradata1
  ...
  '/oradata4BCV/dw/users01.dbf'    # was /oradata4
CHARACTER SET US7ASCII
;

RECOVER DATABASE
ALTER SYSTEM ARCHIVE LOG ALL;
ALTER DATABASE OPEN;

ALTER TABLESPACE TEMP ADD TEMPFILE '/oradata1BCV/dw/temp01.dbf' REUSE;
```

## External Tables and Flat Files

Oracle 9i offers a new feature called *external tables*. An external table allows Oracle 9i databases to read a flat file and manipulate it using standard SQL as if the flat file were an ordinary table. Prior to Oracle 9i, this process required that a flat file be loaded into a temporary table via SQL\*Loader where the temporary table was used for the transformation operations. EVM Snapshot capabilities offer an ideal solution to quickly move data between servers after the data has been extracted into flat files.

Below is a sample snapshot job to move a flat file to the Oracle 9i server. Note that since we are simply moving flat files, we do not need to run any server jobs to put the database in backup mode prior to performing the snapshot.

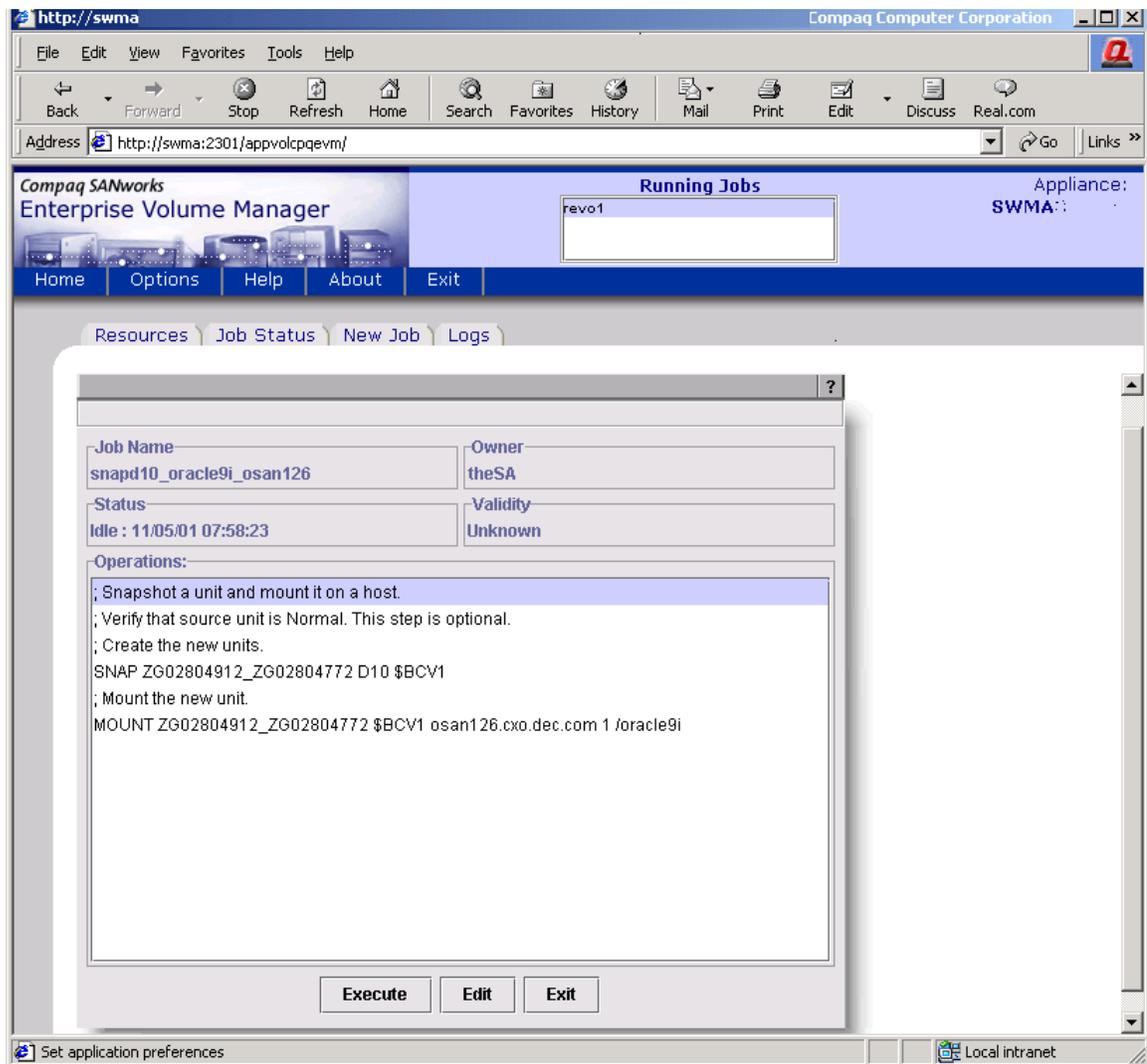


Figure 3: Moving Flat Files Using Snapshots

Within Oracle, in order to use the flat file as an external table, the customer will need to create a directory pointing to the mount point where the snapshot has been mounted. Note that Oracle must have write access to the directory in order to write log and bad files. The text box below contains the necessary SQL statements and sample data to demonstrate the use of Oracle 9i external tables.

```
CREATE DIRECTORY ext_tab_dir AS '/oradata';

CREATE TABLE contact (
  first_name CHAR(15),
  last_name CHAR(20),
  year_of_birth CHAR(4))
ORGANIZATION EXTERNAL (
  TYPE ORACLE_LOADER
  DEFAULT DIRECTORY ext_tab_dir
  ACCESS PARAMETERS (
    fields terminated by ","
    (first_name ,
     last_name ,
     year_of_birth char(4) )
  )
  location ('contact.dat') );
```

```
Alvin,Tolliver,1976
Kenneth,Baer,1963
Mary,Dube,1973
```

```
SQL> select * from contact;
```

| FIRST_NAME | LAST_NAME | YEAR |
|------------|-----------|------|
| Alvin      | Tolliver  | 1976 |
| Kenneth    | Baer      | 1963 |
| Mary       | Dube      | 1973 |

SQL\*Loader may also be used depending on the customer's particular needs. The EVM 2.0 processing is identical in both cases; however, when using versions of Oracle prior to 9i, SQL\*Loader must be used.

## Transportable Tablespaces

Transportable tablespaces, introduced in Oracle 8, offer some of the widest and most efficient options for moving data to the data warehouse. Transportable tablespaces import and export extremely quickly. In the past, the downside to transportable tables has been the time it takes to move the data files over the network (either via ftp or nfs), especially for large tablespaces. EVM version 2.0 snapshots and clones greatly accelerate data movement. The time necessary to coalesce the tablespaces is only a matter of minutes and not hours as when using network-based transfers.

Below are the steps necessary to export the orders\_ts tablespace.

```
sql> alter tablespace orders_ts read only;
$ exp "system/password as sysdba" file=orders_20010806.dmp -
  transport_tablespace=y tablespace=orders_ts
sql> alter tablespace orders_ts read write;
```

***After exporting the tablespace, use EVM 2.0 to snap or clone the necessary drives and move the tablespace's data files to the data warehouse.***

Next, import the tablespace on the data warehouse with the following commands:

```
$ imp "system/password as sysdba" file=orders_20010806.dmp -
  transport_tablespace=y datafiles=(/u10/oradata/order_ts.dbf)
sql> alter tablespace orders_ts read write;
```

Once the tablespace has been imported, a couple of options are available. The first option is to load the data into the data warehouse. For example:

```
SQL> commit;
SQL> set transaction use rollback segment big_rs;
SQL> insert into orders select * from daily_orders;
SQL> commit;
```

In this case, an EVM snapshot is ideal to quickly move the information between the transaction processing (OLTP) database and the data warehouse.

The second option is to exchange a partition for the table. This exchange is done if the data warehouse table is partitioned by a value in the imported table. For example, the warehouse table is partitioned by month. Now the database administrator (DBA) can add the most recent month as a partition and then exchange the current partition for the table.

```
SQL> alter table dw_orders add partition sept01_orders values
  less than (to_date_'1-oct-2001', 'dd-mon-yyyy');
SQL> alter table dw_orders exchange partition sept01_orders with
  table monthly_orders including indexes with validation;
```

This scenario brings up several issues. First, since adding a partition is a permanent addition to the data warehouse physical layout, a snapshot should not be used. In addition, because the BCV

in this example is a JBOD without RAID protection, a mirror member should be added to the unit. An example one-up EVM version 2.0 job would look like the following:

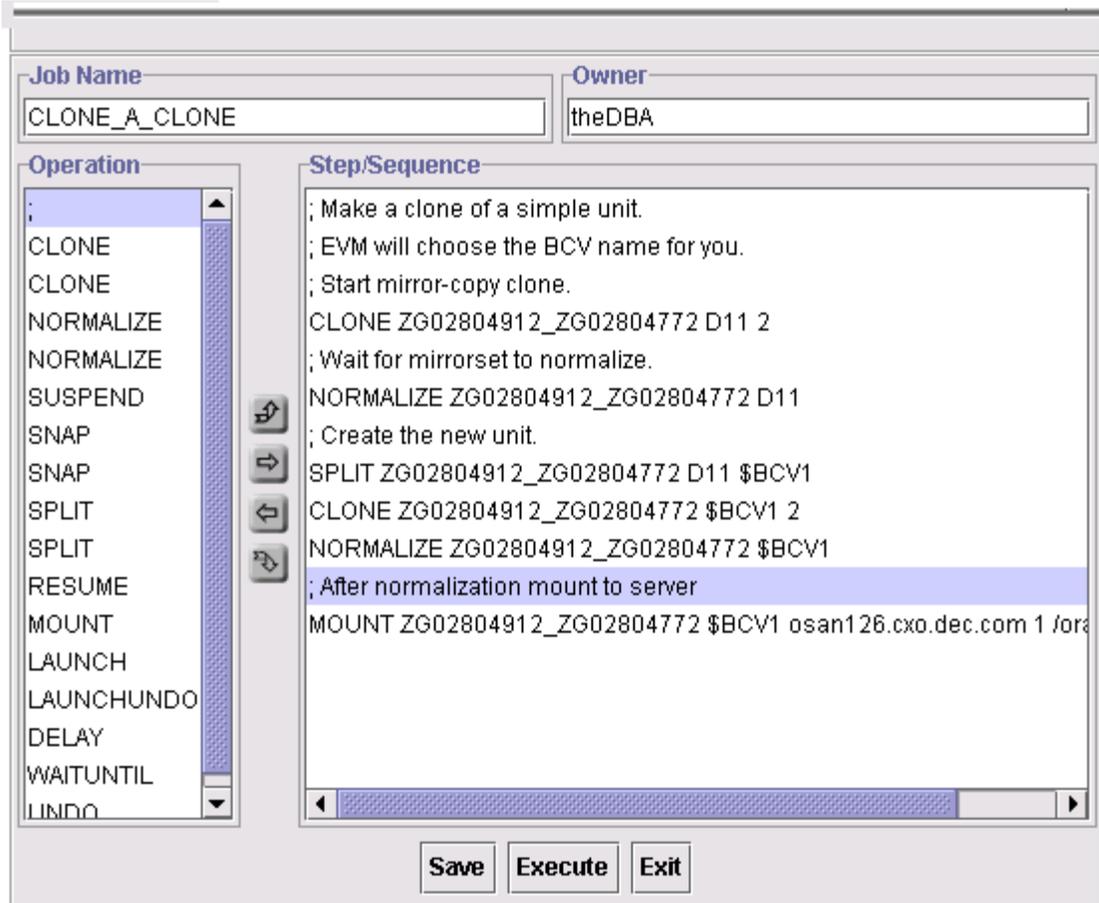


Figure 4: Sample EVM 2.0 Job Mirroring a Clone

Finally, take precautions to ensure that the EVM undo job cannot be run again. After the job has run, delete the job using the Job Status tab. Do not attempt to run the undo job.

## Using EVM with Data Marts

### What is a Data Mart?

A data mart is essentially a mini data warehouse that is typically restricted to either functional areas such as sales or along divisional boundaries such as a regional processing center. Because of the restricted nature of the data mart, it usually draws on fewer data sources. These sources can be both internal operational and external data as well as information from the centralized data warehouse. As a result, data marts can be classified into three areas: independent, dependent, and hybrid. An *independent* data mart draws all of its data from sources separate from the corporate data warehouse. A *dependent* data mart draws its information exclusively from the central, corporate data warehouse. The *hybrid* data mart draws information from both the corporate warehouse and non-warehouse sources.

## Moving Data from the Warehouse to a Data Mart

The methods of improving the transportation step for data warehouses apply to data marts as well. In the case of moving data from a warehouse to a data mart, the Extract, Transport, Transform and Load is often short cut because the data warehouse has already performed the transformation. Extracts can be done via Oracle export utility. Since the data is already in Oracle, transportable tablespaces become an efficient solution to moving the data.

When deciding whether to use a clone or snapshot, several issues should be weighed:

- How much activity will be happening on the data warehouse and the data mart?
- How frequently will the data mart need to be refreshed?
- How much update activity to the source data is happening on the warehouse?

If updates to the data mart are frequently required, snapshots can accomplish these updates quickly and efficiently. If, however, the data mart does not need to be refreshed frequently, then clones may offer the better option.

If the data mart is heavily used, clones may be desirable to prevent contention to the snapshot source volume. The same is true if the warehouse is highly utilized. Does the data mart simply need to extract a portion of the information sent from the data warehouse? Then snapshots can typically work most efficiently. A general rule of thumb is that the longer and heavier the data access on the data mart, the better a clone will work. The more often data needs to be refreshed, the better a snapshot will work.

## Conclusions

Compaq SANworks Enterprise Volume Manager version 2.0 can greatly improve data warehouse availability and accessibility. With the Extract, Transport, Transform and Load methodology, the slowest phase is often the transportation of the data. EVM version 2.0 snapshots and clones can quickly move data to and from the data warehouse, allowing for multiple extracts and enabling the repeated extraction of data when the data is required, not just during off-peak hours. The results will be better, more timely information with minimal impact to production databases and business processes.

## Contact Us

For comprehensive online support, refer to:

<http://www.compaq.com/>

For international information, refer to:

[http://www.compaq.com/corporate/overview/world\\_offices.html](http://www.compaq.com/corporate/overview/world_offices.html)

For a list of Compaq products, refer to:

<http://www.compaq.com/showroom/>

**Table 3. Departments and Telephone Numbers for the United States and Canada**

| <b>Department</b>   | <b>Telephone Number(s)</b>                          |
|---|---|
| Consumer Direct   | 1-800-888-0220                                      |
| Compaq DirectPlus   | 1-800-888-5858 (U.S.)                               |
| Compaq Partner Direct   | 1-800-888-5874                                      |
| Compaq Reseller Locator   | 1-800-345-1518 (Option 3)                           |
| Compaq Canadian Reseller Locator and Product Literature           | 1-800-567-1616                                      |
| Diskette Fulfillment (backup diskettes for preinstalled software) | 1-800-952-7689 (U.S.)<br>1-800-349-8498 (Canada)    |
| Compaq Product Information  | 1-800-345-1518 (U.S.)<br>1-800-567-1616 (Canada)    |
| Compaq Technical Support  | 1-800-OK-COMPAQ (U.S. and Canada)<br>1-800-652-6672 |