digital

SAS Oracle OPS TruCluster PS DIGITAL UNIX AlphaServer 8x00

DIGITAL HiTest Notes

Part Number: EK-HDWUB-HN. C01

March 1998

Revision/Update Information: This is a revised document.

Digital Equipment Corporation
Maynard, Massachusetts

March 1998

Table of Contents

4 Interoperability Tests and Results

5 System Limits and Characterization Data

6 Problems and Resolutions

A Detailed Hardware Configuration

B Test Scripts

Figures

Tables

Preface

This document provides an overview of DIGITAL HiTest Suites and detailed technical information about interoperability test results for the SAS Oracle OPS TruCluster PS DIGITAL UNIX AlphaServer 8x00 HiTest Suite.

Audience

Primary users of this document are DIGITAL and Partners sales representatives and technical support personnel. Secondary audiences include product managers, customers, and the personnel responsible for installing, setting up, and operating a DIGITAL HiTest Suite.

Road Map

This document contains the following chapters:

1. Introduction – Provides a brief summary of the benefits of DIGITAL HiTest Suites and an overview of the Suite covered in this document.

2. Configuration Data – Includes tables of configuration data about the hardware and software components that define the Template, and special configuration rules if any.

3. System Installation and Setup – Presents useful information for installing and setting up this DIGITAL HiTest Suite.

4. Interoperability Tests and Results – Describes how the tests were set up (including database organization), what data and programs were placed on what disks, and how the tests were run.

5. System Limits and Characterization Data – Summarizes any system limitations or characterization data that were identified during testing.

6. Problems and Resolutions – Discusses any problems and resolutions that were discovered during testing.

Appendix A: Detailed Hardware Configuration – Contains more detailed information about the hardware and software components listed in the Configuration Data chapter.

Appendix B: Test Scripts – Contains detailed test scripts.

Feedback and Ordering Information

What our readers think of this or any other DIGITAL documentation is important to us. If you have any comments, we would appreciate hearing from you. Send your comments to: *reader-comments@digital.com*.

Please reference the document title and part number (EK-HDWUB-HN. C01) in your correspondence about this document.

Copies of this and other DIGITAL documents can be ordered by calling 1-800-DIGITAL.

1
Introduction

DIGITAL HiTest Suite and Its Advantages

DIGITAL HiTest Suites are guidelines for configuring a set of prequalified computer systems. A HiTest Suite often contains all the hardware and software needed for a complete customer solution. DIGITAL HiTest Suites can be used as a basis for configuring systems that satisfy a wide set of customer requirements. Typically, Suites target specific markets such as Data Warehousing or Enterprise Applications.

DIGITAL Product Management and Engineering select the components and design the configurations in each HiTest Suite to ensure high system reliability, application performance, and upgradability. A Suite's hardware and software components have been successfully tested for interoperability.

A HiTest Suite specifies allowed ranges of hardware and software components, as well as each component's part number, description, and revision information. These specifications are listed in the *DIGITAL HiTest Template*.

The components in a HiTest Suite are organized into two groups, the *DIGITAL HiTest Foundation* and the *DIGITAL HiTest AppSet*. The HiTest Foundation includes the hardware, operating system, middleware, and database software. The HiTest Foundation can be used as a base on which any customer-desired applications can be installed. The HiTest AppSet includes the software specific to one class of customer solutions.

Configuring a DIGITAL HiTest Suite is straightforward. Select components from the HiTest Template to configure a DIGITAL HiTest System. Any system configured as specified in the DIGITAL HiTest Template can be called a DIGITAL HiTest System.

The HiTest Suite is documented in the *DIGITAL HiTest Notes*. The HiTest Notes list the HiTest Foundation and HiTest AppSet components. HiTest Notes also describe the testing of the Suite and include configuration details, installation instructions, tuning parameters, problems encountered and their solutions, and system diagrams.

Some components listed in the HiTest Foundation or AppSet may be optional. If the minimum quantity is zero (0), then the component is optional. If the minimum quantity is one or more, then you must order at least the minimum quantity.

The maximum quantities represent the largest group of components that were tested for interoperability with all the other components in the Suite. Although it may be possible to place more than the specified maximum quantity of a component on a DIGITAL system, extensive interoperability testing was not done at that level and such a system would not be considered a DIGITAL HiTest System.

You can select any combination of components with quantities ranging from the minimum to the maximum specified. Occasionally, special configuration rules give further guidance or restrict configurations. These rules appear in the Configuration Data chapter of the HiTest Notes.

A customer can include the Suite-specified hardware and software they need and then layer on additional software. Other types of hardware, called *add-on hardware*, can also be added to a DIGITAL HiTest System. The add-on hardware is specified in the Configuration Data chapter of the HiTest Notes, and in the HiTest Systems Web Pages, available through the following URLs:

`http://cosmo.tay.dec.com` (Intranet)
`http://cosmo.tay.dec.com/public/configsys/config_systems.htm`
`http://www.partner.digital.com:9003/cgi-bin/comet` (Internet)

Even though the customer may install application software that is not specified in the Suite, the customer and DIGITAL still experience the advantages of knowing that all of the Suite-based hardware and software interoperates correctly. Of course, the full benefit of configuring a system from a HiTest Suite is obtained when the system includes only specified HiTest Foundation and AppSet components.

Overview of This DIGITAL HiTest Suite

The SAS Oracle OPS TruCluster PS DIGITAL UNIX AlphaServer 8x00 HiTest Suite consists of the following software components:

- SAS System, Release 6.12
- DIGITAL UNIX, Version 4.0B
- TruCluster Production Server, Version 1.4A
- Oracle 7 Server, Version 7.3.3

The SAS Data Warehouse brings core technologies together in a single, powerful, and integrated software environment. This open and integrated solution encompasses everything from back-end access to legacy systems to front-end decision support and business intelligence capabilities. SAS tables are optimized for decision support processing and are, therefore, ideal for the data warehouse environment.

2
Configuration Data

This chapter describes the tested DIGITAL HiTest Configuration Suite including the hardware, software, and firmware components, and their revision levels. Special configuration rules are explained if required.

Hardware and Software Components

Table 2-1 and Table 2-2 identify the range of hardware and software components that can be configured using the SAS Oracle OPS TruCluster PS DIGITAL UNIX 8x00 HiTest Suite. These two tables form the DIGITAL HiTest Template.

Table 2-3 lists the optional System Management Station hardware and software.

Table 2-4 lists the revision levels of the components.

The DIGITAL HiTest Template consists of three categories:

- **AppSet Software** – Includes software specific to one class of customer solutions, in this case the Data Warehousing market.
- **Foundation Hardware** – Includes the base system, storage, and other hardware options.
- **Foundation Software** – Includes the operating system, middleware, and database software.

When ordering an item from a HiTest Template, select a quantity that is within the minimum/maximum range for the item. If the minimum quantity is zero (0), then the component is optional. If the minimum quantity is one or more, then order at least the minimum quantity, but not more than the maximum quantity. The maximum quantity represents the greatest number of components that were tested for interoperability with all the other components in the Suite.

For more details on the HiTest Suite hardware configuration, see Appendix A.

Table 2-1: DIGITAL HiTest Template – AppSet Software and Foundation Hardware

SAS Oracle HiTest AppSet TruCluster DIGITAL UNIX AlphServer 8x00 HiTest Foundation Hardware				
For documentation and updates: http://cosmo.tay.dec.com and http://www.partner.digital.com:9003/cgi-bin/comet				
Line Item	**Description**	**Part Number**	**HiTest Range**	
			Min	**Max**
AppSet Software				
1	**The SAS System, Release 6.12** For SAS Institute products, call 919-677-8000. See Table 2-2 for detailed license requirements. **Note:** This AppSet is not required when the foundation hardware and software is ordered for use with a non-HiTest application.	**SAS Institute**	1	1
Foundation Hardware				
2	*Select two base systems*: **AlphaServer 8400 5/625, 2 GB System** **AlphaServer 8400 5/625, 4 GB System** *Hardware includes*: • Two 5/440 MHz CPUs; 4 MB cache • I/O module with four I/O channels (KFTHA-AA) • 2 or 4 GB of memory • 4.3 GB 3.5” SCSI disk • One BN21K-02 PCI FWD SCSI cable • One DWZZB-VW SCSI signal converter • 600 MB CD-ROM drive • One KZPSA-BB, PCI FWD SCSI controller • KZPAA-AA, PCI FNSE SCSI ctlr, for CD-ROM only • BN21H-02 2-meter SCSI cable • DE500-AA 10/100 Mbit Etherworks 32-bit NIC • BA660-AB StorageWorks Plug-in-unit • One DWLPB-AA, PCI 12-slot Plug-in-unit (8400) • Two H7263-*Ax* non-BBU capable 48 VDC power regulators; includes power cord • Shielded console cable **Note:** Systems ordered in the Americas or Asia Pacific include the keyboard. *Software includes*: • Factory Installed Software • DIGITAL UNIX Operating System base license • Unlimited User and Server Extension licenses • ServerWORKS • Internet AlphaServer Admin Software • DECevent and BMC Patrol Agent	**DA-292FF-CA** **DA-292FG-CA**	2	2
3	**Dual 5/440 CPU module for 8400**	**756P2-AX**	0	6
4	**2 GB Memory module for 8400** **Note:** This HiTest Template supports a memory range from 2 to 8 GB per system. When selecting memory options, stay within the Template’s 8 GB maximum.	**MS7CC-FA**	0	6
5	**PCI Expansion Box**	**DWLPB-BA**	2	2
6	**PCI one-port FWD SCSI controller**	**KZPSA-BB**	7	11
7	**SCSI Y cable, 68-pin**	**BN21W-0B**	4	8
8	**SCSI Differential terminator**	**H879-AA**	4	8

SAS Oracle HiTest AppSet TruCluster DIGITAL UNIX AlphServer 8x00 HiTest Foundation Hardware				
For documentation and updates: http://cosmo.tay.dec.com and http://www.partner.digital.com:9003/cgi-bin/comet				
Line Item	**Description**	**Part Number**	**HiTest Range**	
			Min	**Max**
9	**3 m SCSI-3 cable "P" straight/90º**	**BN21K-03**	2	4
10	**20 m SCSI-3 cable "P" straight/90º**	**BN21K-20**	5	7
11	**SCSI Trilink connectors**	**H885-AA**	4	8
12	**SCSI cable (HSZ50 controller pair connection)**	**BN21L-0B**	2	4
13	**StorageWorks Storage Array** *Hardware includes:* • SW800-GA Cabinet Assembly with Power Controller • SW8XP-BA (Backup) Power Controller • 3 BA350-MB Disk Controller Shelves • 3 BA35X-HF (Backup) Power Supplies for Disk Controller Shelves • 18 BA350-JA Disk Shelves • 18 BA35X-HA (Backup) Power Supplies for Disk Shelves	**SW822-GA**	1	2
14	**StorageWorks RAID Array Dual SCSI Controller** *Each HSZ52-AJ Controller Option includes:* • 2 HSZ50 256 MB SCSI Controllers • An External Cache Battery Backup Power Supply which is installed in place of a (Backup) power supply in the associated disk storage shelves • 2 2-meter cables to interconnect Controllers with external cache batteries • 2 H885-AA SCSI Tri-link Connectors, H879-AA SCSI Terminator, and a BN21L-0B cable to interconnect the 2 SCSI Controller	**HSZ52-AJ**	2	4
15	**4.3 GB 7200 RPM UltraSCSI Disk**	**DS-RZ1CB-VW**	72	144
16	**Memory Channel Hub**	**CCMHA-AA**	1	2
17	**PCI to Memory Channel controller**	**CCMAA-BA**	2	4
18	**Memory Channel cable**	**BC12N-10**	2	4
19	**PCI to FDDI Adapter-Dual Attach**	**DEFPA-DB**	2	2
20	**Multimode Fiber-Optic Duplex Cable**	**BN34D-10**	2	2
21	**Fast Ethernet NIC, PCI adapter**	**DE500-AA**	2	2
22	**10BaseT Twisted-Pair Ethernet cable**	**BN25G-07**	4	4
23	**176 Cart Tape base library unit with 6 TZ89 drives**	**DS-TL896-BA**	1	1
24	**System Management Station**	**See Table 2-2**	0	1

Table 2-2: DIGITAL HiTest Template – Foundation Software

SAS Oracle TruCluster DIGITAL UNIX AlphaServer 8x00 Foundation Software						
For documentation and updates: http://cosmo.tay.dec.com and http://www.partner.digital.com:9003/cgi-bin/comet						
Line Item	**Description**	**Part Number**	**HiTest Range**		**Required By**	
			Min	**Max**	**Fnd**†	**App**†
Foundation Software						
1	Factory installed software and licenses for: • DIGITAL UNIX Operating System, Version 4.0B, Base License and Unlimited Interactive User License • DIGITAL UNIX Server Extensions, Version 4.0B	**Included with item 2**	2	2	Yes	Yes
2	**DIGITAL UNIX Operating System, V4.0C Software Media Kit** **Note:** DIGITAL UNIX Version 4.0C installs as Version 4.0B on this system. Software media and documentation required for first system on site.	**QA-MT4AA-H8**	See Note	See Note	Yes	Yes
3	**DIGITAL CD-ROM Software Library for DIGITAL UNIX Systems layered products** **Note:** Software media and documentation required for first system on site.	**QA-054AA-H8**	2	2	Yes	Yes
4	**TruCluster Production Server V1.4**	**QB-3RLAQ-AA**	2	2	Yes	Yes
5	**ServerWORKS**	**Included with item 2**	2	2	Yes	Yes
6	**Developers' Toolkit for DIGITAL UNIX, Version 4.0B** *Select a license:* • **Traditional Licenses** • **Concurrent Use License** **Note:** Includes DEC C for DIGITAL UNIX, ladebug, and dbx	**QL-MT5AQ-AA** **QL-MT5AM-3B**	1	1	Opt'l	Opt'l
7	**DEC C++ Version 5.5 for DIGITAL UNIX Systems** *Select a license:* • **Traditional Licenses** • **Concurrent Use License**	**QL-MTRAQ-AA** **QL-MTRAM-3B**	1	1	Opt'l	Opt'l
8	**DEC COBOL Version 2.4 for DIGITAL UNIX Systems** *Select a license:* • **Traditional Licenses** • **Concurrent Use License**	**QL-2BZAQ-AA** **QL-2BZAM-3B**	1	1	Opt'l	Opt'l
9	**DIGITAL Fortran for DIGITAL UNIX Alpha Systems, run-time library, Version 3.6I**	**Included with item 2**	1	1	Opt'l	Opt'l
10	**Micro Focus COBOL Compiler and Development Tools Version 4.0A for DIGITAL UNIX Micro Focus COBOL Developer Suite Package**	**QP-5S9AM-3B**	1	1	Opt'l	Opt'l
11	**DEC FUSE V3.0 for DIGITAL UNIX V4.0 Systems**	**QL-092AM-3B**	1	1	Opt'l	Opt'l

SAS Oracle TruCluster DIGITAL UNIX AlphaServer 8x00
Foundation Software

For documentation and updates: http://cosmo.tay.dec.com and http://www.partner.digital.com:9003/cgi-bin/comet

Line Item	Description	Part Number	HiTest Range		Required By	
			Min	Max	Fnd†	App†
12	**Oracle7 Server for DIGITAL UNIX V7.3.3,** including: • Enterprise Backup Utility (OEBU) 2.1.0.1.2 • Oracle7 Parallel Server 7.3.3 • Oracle Server Manager 2.3.3 • Parallel Query Option • PL/SQL • SQL*Net (include TCP/IP protocol adapter) • SQL*Plus 3.3.3 Oracle contacts: Call: 1-800-ORACLE1, E-mail: infodec@us.oracle.com, or URL `http://www.oracle.com`	**Oracle**	1	1	Yes	Yes
13	**StorageWorks Platform kit (DIGITAL UNIX)** includes: • HSOF software V5.1 • StorageWorks Command Console	**QB-5JCAB-SA**	1	1	Yes	Yes
14	**HSZ50 Array Controller Operating Software (HSOF) V5.1 LIC/MCD/No DOC Kit**	**QB-5CJAB-SB**	3	7	Yes	Yes
15	**StorageWorks PLUS**, which includes: • Networker Save and Restore for DIGITAL UNIX V4.3 • LSM V4.1 Logical Storage Manager • AdvFS Utilities V4.0B	**QB-5RYAQ-AA**	2	2	Yes	Yes
16	**NetWorker Save and Restore Database Module for Oracle, Version 2.0 license**	**QL-3P4AQ-AA**	1	1	Yes	Opt'l
17	**NetWorker Save and Restore Database Module for Oracle, Version 2.0 media kit**	**QL-3P4AA-H8**	1	1	Yes	Opt'l
18	**NSR Jukebox Tier 3 License**	**QL-04WAL-3B**	1	1	Yes	Yes
19	**Hard copy of this Suite's HiTest Notes**	**EK-HDWUB-HN**	1	1	Yes	Yes

† Fnd = Foundation, App = AppSet

Table 2-3: System Management Station Template

SAS Oracle HiTest AppSet System Management Station				
For documentation and updates: http://cosmo.tay.dec.com and http://www.partner.digital.com:9003/cgi-bin/comet				
Line Item	**Description**	**Part Number**	**HiTest Range**	
			Min	**Max**
Management Station Hardware				
Note: This HiTest Suite supports the use of a system management station. When the management station option is included, this HiTest Template identifies the items required. When system management is provided through other means, this option may be omitted without invalidating the HiTest Suite.				
1	**DIGITAL PC 5100** *Hardware includes*: • 200 MHz Pentium CPU with MMX • 512 KB secondary cache • 32 MB memory • Integrated Fast Ethernet (10/100) • 16X CD-ROM • PCI 64-bit S3 ViRGE/GX graphics controller (with 2 MB) • 3.2 GB disk drive • 1.44 MB floppy *Software includes*: • Windows NT Workstation 4.0 (factory installed) **Note:** A functionally equivalent 80X86 system may be substituted without invalidating this HiTest Template.	**FR-DAB04-AF**	1	1
2	**Country Kit, North American**	**FR-PC94K-AA**	1	1
3	**32 MB SDRAM dual-bank DIMM Memory**	**FR-PCCAM-EC**	1	1
4	**Diamond 56.6 K Modem** **Note:** Used for page notification.	**FR-PCXFA-AA**	0	1
5	*Select one high-resolution monitor:* **21" (19.6" view) 1600 x 1200 @ 75 Hz** **19" (18" view) 1600 x 1200 @ 75 Hz** **17" (16" view) 1280 x 1024 @ 75 Hz**	 **FR-PCXAV-WZ** **FR-PCXAV-TZ** **FR-PCXAV-YZ**	1	1
Software Installed on Management Station				
6	**Windows NT Workstation 4.0** Contact Microsoft at `http://www.microsoft.com`, or `ftp://ftp.microsoft.com/bussys/winnt` or call (800) 360-7561	**Included with item 1**	1	1
7	**Windows NT Service Pack 3** Contact Microsoft at `http://www.microsoft.com`	**Microsoft**	1	1
8	**Hummingbird Exceed, Version 6.0** Contact `http://www.hummingbird.com`	**Hummingbird**	1	1
9	**DIGITAL ServerWORKS Manager, Version 3.0**	**Included with the base system**	1	1
10	**StorageWorks Command Console, V1.1B**	**Included with StorageWorks kit**		

SAS Oracle HiTest AppSet System Management Station				
For documentation and updates: http://cosmo.tay.dec.com and http://www.partner.digital.com:9003/cgi-bin/comet				
Line Item	**Description**	**Part Number**	**HiTest Range**	
			Min	**Max**
11	*Choose one BMC product:* **BMC PATROLWATCH for ServerWORKS, V3.2** **BMC PATROL Operator Console Windows NT** **BMC PATROL Developer Console Windows NT** **Note:** BMC products that are included with the AlphaServer are on the ServerWORKS Quick Launch CD. BMC PATROL Developer Console includes BMC PATROL Operator Console. Contact BMC at: `http://www.bmc.com`	**Included with base system** **BMC** **BMC**	0	1
12	**BMC PATROL Agent for Windows NT, license and media**	**QB-5KKAB-WA**	0	1
13	**BMC Operating System Knowledge Module for Windows NT, license and media**	**QB-5KLAB-WA**	0	1
Software Installed on Managed System(s)				
14	**Base UNIX systems management tools**	**Included with DIGITAL UNIX**	1	1
15	**DIGITAL UNIX Management Agent**	**Included with item 9**	1	1
16	**StorageWorks Command Console Agent**	**Included with item 10**	1	1
17	**BMC PATROL Agent for UNIX**	**Included with the AlphaServer**	0	1
18	**BMC Operating System Knowledge Module** **Note: W*** refers to the class: WA - Desktop; WB - Workgroup; WC - Departmental; WD - Enterprise Server	**QB-5KLAA-W***	0	1

Table 2-4: Component Revision Levels

Hardware Component	Hardware	Firmware	Software
KZPAA-AA	0002	–	–
FWD SCSI controller (KZPSA-BB)	N01	A10	–
Array controller (HSZ50-CF)	A01	HSOF V5.1Z-0	–
4.3 GB disks (RZ29B-VA)	–	0016	–
5/440 MHz CPU (756P2-AX)	–	–	–
Fast Ethernet adapter (DE500-AA)	00012	–	–
SCSI CAM Medium Change Driver	–	–	V3.1A
DS-TL896-BA	–	V1.10	–
PALcode	–	V1.21	–
AlphaServer 8400 console	–	4.9-20	–
2 GB memory module (MS7CC-FA)	B01	–	–
4.3 GB 3.5 in. disk (DS-RZ1CB-VW)	B03	–	–

System I/O module (KFTHA-AA)	D03	–	–
StorageWorks Plug-in unit (BA660-AB)	A01	–	–
StorageWorks Shelf Power Supply (DS-BA356-HH)	B01	–	–
PCI Plug-in unit (DWLPB-AA)	A02	–	–
PCI Plug-in unit (DWLPB-BA)	A02	–	–
Software Component	**Version/Revision**	**Patch Level**	
DIGITAL UNIX	4.0B	Rev564 40BAS0005	
TruCluster for DIGITAL UNIX	1.4A	–	
DIGITAL UNIX Driver for Memory Channel	1.4A	–	
EISA System Config Kit	V1.9	–	
DEC C++ for DIGITAL UNIX Systems	V5.5	–	
DEC COBOL for DIGITAL UNIX Systems	V2.4	–	
Micro Focus COBOL for DIGITAL UNIX	V4.0U1	–	
Oracle7	7.3.3	–	
Oracle7 Parallel Server	7.3.3	–	
Oracle Server Manager	2.3.3	–	
Parallel Query Option	7.3.3	–	
PL/SQL	7.3.3	–	
SQL*Net	2.3.3	–	
SQL*Plus	3.3.3	–	
Oracle Enterprise Backup Utility (OEBU)	2.1.0.1.2	510127	

Special Configuration Rules

There are no special configuration rules for this SAS Oracle OPS TruCluster PS DIGITAL UNIX HiTest system.

3
System Installation and Setup

This chapter presents information that is useful when installing and setting up a DIGITAL HiTest System configured from this DIGITAL HiTest Suite. System preparation includes installation of the hardware, operating system, and applications.

Hardware Installation

The hardware was installed and interconnected as shown in Appendix A.

TruCluster Production Server

The specific hardware rules and restrictions that govern TruCluster Production Server configurations include the following:

1. Set the boot_reset variable to on:
   ```
   >>> set boot_reset on
   ```
2. Ensure the KZPSA adapters have different SCSI ID numbers (see Table 3-1). Set the four shared SCSI bus controllers in the first 8x00 to SCSI ID 7. For the maximum configuration Template, set the controllers in the second 8x00 to SCSI ID 6.
3. Enable fast SCSI bus speed for each KZPSA adapter (see Table 3-1).
4. Connect each KZPSA adapter to a shared SCSI bus with a “Y” cable.
5. Disable internal KZPSA SCSI termination. Remove the internal termination resistors (Z1, Z2, Z3, Z4, and Z5) on each KZPSA-BB on shared SCSI buses.
6. Set jumper memory Channel PCI adapters for standard mode (see *Memory Channel Users Guide*).
7. Tightly screw memory Channel link cables to ensure proper grounding.
8. Apply memory Channel Hub power before the member systems are booted.

Table 3-1: SCSI Controller Configuration

Step	Action	Result

1	`>>> show kzpsa*`	`KZPSAa0_fast 1`
		`KZPSAa0_host_id 7`
		`KZPSAa0_termpwr 1`
		`KZPSAa1_fast 1`
		`KZPSAa1_host_id 7`
		`KZPSAa1_termpwr 1`
		`KZPSAa2_fast 1`
		`KZPSAa2_host_id 7`
		`KZPSAa2_termpwr 1`
		`KZPSAa3_fast 1`
		`KZPSAa3_host_id 7`
		`KZPSAa3_termpwr 1`
		`KZPSAa4_fast 1`
		`KZPSAa4_host_id 7`
		`KZPSAa4_termpwr 1`
		If any values need to be changed, refer to the next steps.

2	`>>> set KZPSAa0_host_id 6`
	`>>> set KZPSAa1_host_id 6`
	`>>> set KZPSAa2_host_id 6`
	`>>> set KZPSAa3 host_id 6`
		Changes the SCSI ID of the KZPSA-BB controllers from 7 to 6 on the second AlphaServer.

3	`>>> set KZPSAa0_fast 1`
	`>>> set KZPSAa1_fast 1`
	`>>> set KZPSAa2_fast 1`
	`>>> set KZPSAa3_fast 1`
	`>>> set KZPSAa4_fast 1`
		Fast SCSI mode is enabled when the value is set to 1 (0 = slow mode). Set this value for the controllers on both systems.

HSZ50 Array Controller Setup

To set up the HSZ50 Array Controllers, do the following:

1. Set the number of SCSI target IDs (0 through 3):

   ```
   CLI > SET THIS_CONTROLLER ID = (0, 1, 2, 3)
   ```

2. Set failover to place the controllers in a dual-redundant configuration.

   ```
   CLI > SET FAILOVER COPY=THIS_CONTROLLER
   ```

3. Restart both controllers by pressing the OCP reset button or entering the following commands:

   ```
   CLI > RESTART OTHER_CONTROLLER
   CLI > RESTART THIS_CONTROLLER
   ```

4. Enter the following commands to determine whether the preceding parameters are set correctly:

   ```
   CLI > SHOW THIS_CONTROLLER FULL
   CLI > SHOW OTHER_CONTROLLER FULL
   ```

5. Set preferred paths to balance the load and improve the performance of the HSZ array controller pair:

```
CLI > SET THIS_CONTROLLER PREFERRED_ID = (0, 1)
CLI > SET OTHER_CONTROLLER PREFERRED_ID = (2, 3)
```

6. Run the CONFIG program to locate and add devices to the array controller's configuration:

```
CLI > RUN CONFIG
```

7. Set up the RAIDsets using the following commands:

```
CLI > ADD RAIDSET name POLICY=BEST_PERFORMANCE
RECONSTRUCTION=NORMAL disk disk disk disk
```

8. Initialize the RAIDsets:

```
CLI > INITIALIZE name CHUNKSIZE = 256
```

   Refer to the Disk Storage Configuration section for more information.

9. If a host-accessible logical unit needs to be created from the RAIDset, do the following:

```
CLI > ADD UNIT "unit name" "RAIDset name" WRITEBACK_CACHE
```

Disk Storage Configuration

Use the StorageWorks HSZ50 Array Controller Utility to configure the StorageWorks HSZ50 controllers and disks (see the *DIGITAL StorageWorks HSZ50 Array Controller, HSOF Version 5.1 CLI Reference Manual*.)

To configure the controllers and disks, do the following:

1. Configure the disks attached to the HSZ50 controllers as RAID-5 sets. RAID-5 provides disk striping with distributed parity, but provides only the capacity of *n*-1 devices as shown. Each stripeset consists of six RZ29Bs at 4.3 GB per device for a total of 25.8 GB of disk space. There is 21.5 GB exported usable space per RAIDset.

   Where *n* is the number of drives:

$$\frac{(n-1)}{n} * (n * \text{size of each device}) = \text{disk space per RAIDset}$$

$$\frac{(6-1)}{6} * (6 * 4.3\ \text{GB}) = 21.5\ \text{GB usable space}$$

2. Set the chunk size for the Oracle 7 database to 256 disk blocks (based on a 32K db_block_size and having 4 db blocks per chunk).

$$\frac{(db\_block\_size) * (\#\ of\ db\ blocks\ per\ chunk)}{512} = chunk\_size$$

$$\frac{(32{,}768) * (4)}{512} = 256\ disk\ blocks$$

3. Enable the write-back cache option in the HSZ50 Array controller. The write-back cache allocates cache memory to both read and write operations. This allows the I/O to report completion faster.

Operating System Installation

This section describes how to install the DIGITAL UNIX Version 4.0B operating system with all kernel options. Install all DIGITAL UNIX subsets and the TruCluster software for TruCluster Production Server operation. For information on resizing partitions, refer to the *DIGITAL UNIX Installation Guide*.

Swap Space

When installing DIGITAL UNIX, use the Deferred Swap Mode to reduce the amount of swap space required, and to reduce the system overhead required for mapping the address space during the creation of processes:

```
mv /sbin/swapdefault /sbin/swapdefault.old
```

Configuring Additional Swap Space

A secondary swap space can also be assigned during installation to enable complete crash dumps when using 8 GB of memory.

You can assign additional swap space by including entries in `/etc/fstab` as the following example shows:

```
/dev/rz0b swap1 ufs sw 0 2
```

UNIX Kernel Build

Due to a problem in C-shell, you cannot set address space high enough to map a 6 GB SGA. Therefore, the value for maxdsize must be changed in /sys/cont/nodename to the proper memory size of the system. This is not a dynamic setting and a kernel rebuild must be performed.

UNIX Kernel Parameters (/etc/sysconfigtab)

Table 3-2 lists the parameter settings used for the UNIX kernel. The settings are based on a 2 GB minimum and a 8 GB maximum memory configuration, suggested in the *Oracle7 Server for Digital UNIX Installation and Configuration Guide* or refer to the *System Tuning and Performance Management Guide*.

Set the parameter values of *ubc-minpercent* and *ubc-maxpercent* to minimize the Unified Buffer Cache, which allows a larger Oracle buffer cache.

Depending on specific customer requirements for balancing a system to use both Oracle and an application such as the SAS System, *ubc-minpercent*, *ubc-maxpercent*, and *gh-chunks* may be set differently to make more physical memory available to the application.

Table 3-2 shows the parameter settings used for the UNIX kernel.

Table 3-2: Kernel Parameters

Parameter	Minimum Configuration Value	Maximum Configuration Value	Comments
vm parameters:			
vm-maxvas	2146483648	8589934592	Maximum virtual address space for user maps.
vm-mapentries	400	400	Maximum number of virtual memory map entries that a user map can have.
ubc-minpercent	1	1	Percentage of memory at which page stealing from the UBC is prohibited.
ubc-maxpercent	2	2	Maximum percentage of memory that the UBC can consume.
gh-chunks	0	1540	Enables the use of granularity hints to tune performance of shared memory segments on large memory systems.
new-wire-method	1	0	Disable lightweight wiring when enabling gh-chunks.
rt parameters:			
aio-max-num	512	512	The maximum number of concurrent asynchronous I/O requests that can be outstanding on the system at any point in time.
aio-max-percent	2	2	The percentage of physical memory that the asynchronous I/O database can occupy. This limits the maximum number of concurrent asynchronous I/O requests that can be set by aio-max-num.
proc:			
max-proc-per-user	1024	1024	Maximum number of processes one user can run simultaneously.
max-threads-per-user	1024	1024	Maximum limit of threads a user can create.
max-per-proc-data-size	2147483648	8589934592	Maximum value for the upper limit per process.
max-per-proc-address-space	1610612736	8589934592	Maximum value for the upper limit per process.
ipc parameters:			
shm-max	1069547520	2139095040 8 MB aligned address	Maximum shared memory segment size.
shm-seg	32	32	Maximum attached shared memory segments per process.
msg-max	8192	8192	Maximum message size.
msg-mni	1024	1024	Number of System V message queue identifiers.

Parameter	Minimum Configuration Value	Maximum Configuration Value	Comments
msg-tql	4096	4096	Number of System V message headers.
sem-aem	16384	16384	Adjust on exit maximum value.
sem-mni	200	200	Number of message queue identifiers.
sem-msl	200	200	Number of semaphores per ID.
sem-opm	200	200	Maximum number of operations per semop call.
sem-ume	200	200	Maximum number of undo entries per process.
sem-vmx	32767	32767	Semaphore maximum value.
ssm-thres-hold	8388608	0	Disable when enabling gh-chunks. These parameters are mutually exclusive.

TruCluster Production Server

The TruCluster software environment supports up to four systems connected via the Memory Channel interconnect. Install and set up the TruCluster Production Server as follows:

1. Register the TruCluster Production Server Software Version 1.4 software license (TCR-UA).
2. Load and mount the Associated Products Volume 2 CD-ROM to `/mnt`.
3. Use the `setld -l /mnt/TCR141` command to load the TruCluster Production Server Software subsets. The installation procedure starts after the subsets are loaded.
4. Install all mandatory and optional subsets.
5. Enter the IP name and address for the cluster interconnect (Memory Channel).
6. Ensure the system is in ASE 0 and enable the ASE logger.

   The kernel is automatically rebuilt.
7. Identify the shared SCSI buses.
8. Move the new kernel to the root file system.
9. Reboot the systems and use `asemgr` to add the ASE members.
10. Install TCR141 Patch Kit patches: OSF-TCR1.4A-18SEP1997.

For more detailed information, refer to the *TruCluster Production Server Software – Hardware Configuration and Software Installation Manual*.

TruCluster Available Service Environment (ASE) Services

The TruCluster Production Server includes the Available Server Environment and extends these features with the addition of the Distributed Raw Disk (DRD) Service. The implementation of a high availability system, as described in this document, requires the use of ASE services. The specific services used in this configuration are DRD and Network File System (NFS).

Forethought is necessary in planning your ASE services. Partitions of a single RAIDset cannot be shared among multiple ASE services due to ASE limitations regarding ownership and locking.

Distributed Raw Disk (DRD) Service

The shared storage on which the common database resides is provided by the Distributed Raw Disk (DRD) service of DIGITAL UNIX TruCluster software. DRD service allows an application such as Oracle Parallel Server (OPS) to provide high-performance, parallel access to Oracle database storage media with fast failover from multiple cluster member systems. OPS is designed to take advantage of DIGITAL clustering technology for better scalability and availability than the single system product.

The requirements of DRD for the amount of overall storage required to support application, system, and database processes depend on the peak load of the critical applications and the expected growth with expansion for the future. The `asemgr` facility is provided to help assign physical storage (device/partition level) to DRD entity and node location of a DRD service.

When setting up the DRD service, consider the following:

- Plan to place data that is accessed by different instances on different DRD services. This can reduce memory channel traffic and Oracle PCM lock requirements.
- Use a symbol link to assign meaningful names to all the DRD devices to improve manageability.

Set up DRD services for two cluster members (depot1 and depot2) using `asemgr`.

Table 3-3 lists the full DRD service configuration for the maximum configuration.

Table 3-3: DRD Service Configuration (Maximum)

Service Name	Placement Policy	DRD Device Special File	Underlying Storage
drd_depot1	mcdepot1	/dev/rdrd/drd1	/dev/rrzb8a
		/dev/rdrd/drd2	/dev/rrzb8b
		/dev/rdrd/drd3	/dev/rrzb8d
		/dev/rdrd/drd4	/dev/rrzb8e
		/dev/rdrd/drd5	/dev/rrzb8f
		/dev/rdrd/drd6	/dev/rrzb8g
		/dev/rdrd/drd7	/dev/rrz9a
		/dev/rdrd/drd8	/dev/rrz9b
		/dev/rdrd/drd9	/dev/rrz9g
		/dev/rdrd/drd10	/dev/rrz9h
		/dev/rdrd/drd11	/dev/rrzb9a
		/dev/rdrd/drd12	/dev/rrzb9b
		/dev/rdrd/drd13	/dev/rrzb9g
		/dev/rdrd/drd14	/dev/rrzb9h
		/dev/rdrd/drd15	/dev/rrzc9a
		/dev/rdrd/drd16	/dev/rrz32a
		/dev/rdrd/drd17	/dev/rrz32b
		/dev/rdrd/drd18	/dev/rrz32g
		/dev/rdrd/drd19	/dev/rrz32h
		/dev/rdrd/drd20	/dev/rrzb32a
		/dev/rdrd/drd21	/dev/rrzb32b
		/dev/rdrd/drd22	/dev/rrzb32g
		/dev/rdrd/drd23	/dev/rrzb32h
		/dev/rdrd/drd24	/dev/rrz33a
		/dev/rdrd/drd25	/dev/rrz33b
		/dev/rdrd/drd26	/dev/rrz33g
		/dev/rdrd/drd27	/dev/rrz33h
		/dev/rdrd/drd28	/dev/rrzb33a
		/dev/rdrd/drd29	/dev/rrzb33b
		/dev/rdrd/drd30	/dev/rrzb33g
		/dev/rdrd/drd31	/dev/rrzb33h
		/dev/rdrd/drd63	/dev/rrzc9g
		/dev/rdrd/drd64	/dev/rrzc32a
		/dev/rdrd/drd65	/dev/rrzc9h
		/dev/rdrd/drd68	/dev/rrzc9b
		/dev/rdrd/drd69	/dev/rrzc32b

drd_depot2	mcdepot2	/dev/rdrd/drd32	/dev/rrzb16a
		/dev/rdrd/drd33	/dev/rrzb16b
		/dev/rdrd/drd34	/dev/rrzb16c
		/dev/rdrd/drd35	/dev/rrzb16d
		/dev/rdrd/drd36	/dev/rrzb16e
		/dev/rdrd/drd37	/dev/rrzb16f
		/dev/rdrd/drd38	/dev/rrzb16g
		/dev/rdrd/drd39	/dev/rrz17a
		/dev/rdrd/drd40	/dev/rrz17b
		/dev/rdrd/drd41	/dev/rrz17g
		/dev/rdrd/drd42	/dev/rrz17h
		/dev/rdrd/drd43	/dev/rrzb17a
		/dev/rdrd/drd44	/dev/rrzb17b
		/dev/rdrd/drd45	/dev/rrzb17g
		/dev/rdrd/drd46	/dev/rrzb17h
		/dev/rdrd/drd47	/dev/rrz40a
		/dev/rdrd/drd48	/dev/rrz40b
		/dev/rdrd/drd49	/dev/rrz40g
		/dev/rdrd/drd50	/dev/rrz40h
		/dev/rdrd/drd51	/dev/rrzb40a
		/dev/rdrd/drd52	/dev/rrzb40b
		/dev/rdrd/drd53	/dev/rrzb40g
		/dev/rdrd/drd54	/dev/rrzb40h
		/dev/rdrd/drd55	/dev/rrz41a
		/dev/rdrd/drd56	/dev/rrz41b
		/dev/rdrd/drd57	/dev/rrz41g
		/dev/rdrd/drd58	/dev/rrz41h
		/dev/rdrd/drd59	/dev/rrzb41a
		/dev/rdrd/drd60	/dev/rrzb41b
		/dev/rdrd/drd61	/dev/rrzb41g
		/dev/rdrd/drd62	/dev/rrzb41h
		/dev/rdrd/drd66	/dev/rrzc40a
		/dev/rdrd/drd71	/dev/rrzc40b
		/dev/rdrd/drd67	/dev/rrzc40g
		/dev/rdrd/drd70	/dev/rrzc40h

Network File Systems (NFS) Service

File system space is required by the applications, which are simultaneously executing across nodes, and is also required to remain available to the surviving node during node failure cases. Maintain synchronization of these files across both nodes.

The ASE NFS service is provided to address this need and provide shared file system services regardless of the underlying host location. The file system can be mounted and shared from both nodes through the ASE NFS service, which transparently migrates the services to an available host node. For more information on the NFS Service, refer to the *TruCluster Production Server Software Cluster Administration Guide*.

Table 3-4 shows the ASE Service Configuration information.

Table 3-4: ASE NFS Service Configuration (Maximum)

Service Name	Placement Policy	DRD Device Special File	Underlying Storage
sas_ase	balanced	/sas1	/dev/rz16c /dev/rzc33c
		/sas2	/dev/rzc17c /dev/rzc41c
oracle_ase	balanced	/oracle_733	/dev/rz8a

Application Installation

This section describes installing the Foundation Software and Appset Software including middleware, database software, and data warehouse application software.

DIGITAL Layered Products

The full list of DIGITAL layered products in the SAS Oracle OPS TruCluster PS HiTest AppSet appears in Table 2-1. Install these products using standard defaults.

Install system management products Networker Save and Restore (NSR) for DIGITAL UNIX and NSR Database Module for Oracle (DMO).

NSR must be installed before DMO; otherwise there is no preferred sequence for installation of these products. Follow the installation instructions for each product as well as the additional information provided in this section.

Note

To have the proper privileges, the user who performs the restores using the Oracle7 Enterprise Backup Utility must be included in the Operator Group set in `/etc/passwd`.

Setting Up the Tape Library

Set up the automated tape library (jukebox) for backup:

1. Execute the command `MAKEDEV.MC mcxx`.

   Where xx = (SCSI Bus ID * 8) + y
   y = value of the LUN

   This command produces `/dev/mcxx` files.

   To use the jukebox with NSR, configure it by executing the NetWorker jb_config command. This utility creates and stores the necessary resource data that NetWorker requires to manipulate the jukebox.

2. The jb_config command prompts you for the information in Table 3-5.

Table 3-5: Jukebox Values

Jukebox Device Type:	TL896
jukebox name:	TL896
jukebox device path:	/dev/mc66
jukebox tape drive path:	/dev/nrmt0h
jukebox tape drive type:	TZ89
jukebox tape drive path:	/dev/nrmt1h
jukebox tape drive type:	TZ89
jukebox tape drive path:	/dev/nrmt2h
jukebox tape drive type:	TZ89
jukebox tape drive path:	/dev/nrmt3h
jukebox tape drive type:	TZ89
jukebox tape drive path:	/dev/nrmt4h
jukebox tape drive type:	TZ89
jukebox tape drive path:	/dev/nrmt5h
jukebox tape drive type:	TZ89
jukebox bar code reader enabled:	Y
jukebox volume labels to match bar code labels:	Y

3. Label tapes using the following command. All 76 tapes are labeled, although a jukebox can hold a maximum of 176 tapes.

```
nsrjb -L -j TL896 -v -Y -f /dev/nrmt0h -b oracle -S 1-76
```

Setting Up Networker Save and Restore (NSR)

Use the NetWorker Administrator's utility, nwadmin, to set up NSR according to the *NetWorker Save and Restore Database Module for Oracle Administrator's Guide.*

- Setting Up NetWorker Directives

  Set the NetWorker Directives for Oracle using the Directives window of the Customize pull-down menu in the NetWorker Administrator window.

  To set Oracle directives, create the following directive named Oracle:

```
<</>>
forget
ignore
+obkasm:*
```

  The Oracle directive will be associated with the client performing the backup/restore. The `obkasm` is a DMO-specific program that specifies how a set of files is to be backed up and recovered. This directive instructs the client to run `obkasm` on all files specified in the backup/restore.

The Oracle directive's instructions tell the process to:

<</>>	Start at root
forget	Forget previous directives
ignore	Ignore directives in the file system (.nsr files)
+obkasm:*	Run obkasm on all files under root (the + indicates all subdirectories)

- Setting Up a NetWorker Group

  Set up a single NetWorker Group for Oracle from the Groups window of the NetWorker Administrator Customize pull-down menu, according to the *DMO Administrator's Guide.* For example, set up one Group called Oracle to correspond with the environment variable NSR_GROUP used by the client processes using Oracle7 Enterprise Backup Utility.

- Setting Up a NetWorker Client

  Set up NetWorker Client for Oracle7 database backup from the Clients window of the Client Setup pull-down menu.

  In the configuration, NetWorker Save and Restore and the database resided on a single system, therefore it was defined using NetWorker Administrator. Depot 1 was specified as the client system name (this is where the database resides); and Oracle was specified as the directive; and as the group.

- Setting Up Backup Schedules

  Set up backup schedules for NetWorker Save and Restore.

- Setting Up NetWorker Volume Pools

  Set up a volume pool to include all tape volumes.

- Setting Up NetWorker Save and Restore Server

  Set up the NetWorker Save and Restore Server to correspond with the settings in *Oracle Enterprise Backup Utility (OEBU)* for parallelism, according to the *NetWorker Administrator's Guide, UNIX Version.*

  Table 3-6 shows the NSR Server Settings set up for a parallelism of five.

Table 3-6: NSR Server Settings

Item	Setting
Parallelism	12
Active Devices	6
Sessions per Device	2

Oracle Installation

When you install Oracle, be sure to do the following:

- Install Oracle software and DIGITAL UNIX on separate disks to eliminate I/O contention.
- Install Oracle7 Parallel Server, Version 7.3.3, including the Parallel Query Option, PL/SQL, SQL*Net, SQL*Plus, and Oracle7 Enterprise Backup Utility (OEBU) using Oracle Installer from the Oracle7 Server CD.
- Install Oracle on shared storage using standard defaults.

Oracle Parallel Server

The implementation of a shared database using the Oracle Parallel Server (OPS) on a DIGITAL UNIX TruCluster requires some special considerations to ensure the proper and efficient use of the application.

Consider the following when planning the database.

- Distributed Raw Disk Service (DRD) ownership is critical to OPS. The Oracle DBA user must own the DRDs.
- Tablespace Configurations

  Separating table and indexes helps performance by allowing concurrent reads of indexes and their associated table data while minimizing I/O contention. Placing the indexes in their own tablespaces allows the assignment of Parallel Cache Management (PCM) locks specific to data files in that tablespace. This helps to reduce possible block pinging.
- Redo Logs

  Each instance must have a unique set of redo logs, but they must be accessible by the other nodes in the Parallel Server to maintain integrity during an instance or node failure. One of the remaining instances will detect the failure and apply the failed instance's redo log to the database to provide consistency across the shared database. Hence, the placement of the redo logs for both instances on DRDs of the shared storage.

  Each instance is assigned a thread number and each redo log group is enabled for a specific thread. There is a slight performance benefit when using local DRDs to configure for a given instance, so each node is the server for DRDs used for the instance's redo logs.

  The commands for adding the redo threads for an additional instance are as follows:

```
alter database add logfile thread 2
          '/oracle/links/LOGA_2' size 50M reuse,
          '/oracle/links/LOGB_2' size 50M reuse,
          '/oracle/links/LOGC_2' size 50M reuse;
alter database enable public thread 2;
```

  Table 3-7 lists the redo logs.

Table 3-7: Redo Logs

File name (link)	Size (MB)	DRD Number	DRD Service
/oracle/links/LOGA_1	50	drd2	drd_depot1
/oracle/links/LOGB_1	50	drd3	" "
/oracle/links/LOGC_1	50	drd4	" "
/oracle/links/LOGA_2	50	drd35	drd_depot2
/oracle/links/LOGB_2	50	drd36	" "
/oracle/links/LOGC_2	50	drd37	" "

- Rollback Segments

  The creation of one tablespace per instance (node) is not a requirement, but was used in testing to allow each instance private rollback segments that would produce only local DRD access. This slightly more optimal I/O path is likely to produce benefits in an update-intensive environment. One tablespace for each instance is created to hold rollback segments using a local DRD. Then, rollback segments are created in the tablespace and assigned to the instance using the Oracle initialization parameter ROLLBACK_SEGMENTS, which should be in the instance-specific Oracle parameter file.

- Control Files

  The control files need to be accessed by all instances in the Oracle Parallel Server Environment. To meet this requirement, two control files were distributed across two separate DRDs, that are served by different cluster nodes that use two unique storage controllers.

Partition Views

Partition views are supported in Oracle 7.3 for read-only environments such as data warehouses. Very large tables, as used in a star schema, can be divided into multiple physical tables using partitioning criteria. Data warehouse maintenance operations such as periodic large data loads and associated index creation may result in significant down time of critical tables. Through the use of partitioned views, data can be loaded into individual partitions and their indexes created independently without impacting availability. Availability is briefly effected while the union-all view is created to unite the underlying tables.

An additional benefit of managing the underlying tables independently is a major reduction in the amount of TEMP storage required for index creation. As seen in the star schema model, TEMP space requirements are approximately equal in size to that of the large table at the star center. However, by partitioning this large table, mimimum TEMP requirements are consistent with the approximate size of an individual underlying table, not the view as a whole.

As well as increasing manageability, queries may also benefit from partition views where the optimizer chooses to eliminate partitions not needed for the query.

See the *Oracle7 Server Tuning Manual,* Appendix C (Part No. A32537-1).

Oracle Initialization Parameters

Implementing an Oracle Parallel Service requires some parameters to be identical across all instances. Use of both a common and instance-specific parameter file allows the flexibility of constant and variable parameters across instances. The common initialization parameter file is included in the instance-specific parameter file, through use of the Oracle IFILE parameter.

The majority of the initialization parameters were consistent across the instances, but a few like ROLLBACK_SEGMENTS, THREAD, and INSTANCE_NUMBER were uniquely defined in the instance-specific parameter file.

Table 3-8 shows the optimizations made to the Oracle7 server and database for SAS Oracle TruCluster PS.

Table 3-8: Oracle 64-Bit Option Parameters

Parameter	Minimum Configuration Value	Maximum Configuration Value	Comment
db_block_size	32768	32768	DSS and data warehousing applications benefit from a Big Oracle Block (BOB).
db_block_buffers	32000	192617	Buffer cache based on the db_block_size of 32 KB. This number should be maximized to provide the highest cache hit ratio without adversely affecting the memory requirements of other Oracle and system processes.
vlm_sga_base_address	not used	0X400000000	Use for SGA in excess of 3 GB.
async_write	1	1	Enables the DIGITAL UNIX asynchronous I/O feature which allows the DB writer to perform multiple writes to multiple disks simultaneously without waiting for previous writes to finish.
sort_area_size	209715200	26843546	Memory area allocated to each process/thread performing sorts. This parameter should be maximized during intensive sort operations, such as index creation on large tables. This parameter is highly sensitive to the parallel degree of the table being sorted, as each thread will allocate this amount of space.
sort_direct_writes	true	true	Setting this parameter will allocate memory in addition to the sort area to perform sort writes directly to disk, bypassing the buffer cache. Setting this parameter to auto disables the sort_write_buffer_size parameter.
sort_write_buffer_size	131072	131072	When sort_direct_writes is set to true, this parameter must be set to a value between 32768 and 131072, that is a multiple of 32768.
cache_size_threshold	180000	180000	This parameter controls the buffer cache space reserved for table scans. Tables with this or fewer Oracle blocks will be cached in the SGA during full table scans.

Parameter	Minimum Configuration Value	Maximum Configuration Value	Comment
ccf_io_size	786432	786432	Determines the number of bytes per write when creating a contiguous file.
db_file_multiblock_ read_count	4	4	Due to an Oracle limit of 128 KB per I/O, this parameter was set to 4, which is the maximum based on the formula max_io_size/db_block_size (128 KB/32 KB).
compatible	7.3.3	7.3.3	This parameter is required to be set to operable.
parallel_max_servers	72	72	Setting this parameter greater than 1 allows the Oracle RDBMS to take advantage of the parallel query option, for tables where the parallel degree has been set. This parameter should minimally be set to the largest degree specified on any table. Index creation on a table with a parallel degree set will also use this option; however two threads will be created for each degree. Therefore, this parameter should be set to twice the degree for full benefit on index creation.
shared_pool_size	3500000	18000000	This parameter affects the performance of the library cache (shared SQL and PL/SQL areas) and the dictionary cache. This parameter may be reduced if the cache hit ratio is not adversely affected. If cache misses are prevalent, the open_cursors parameter may need to be increased.
log_checkpoint_interval	100000000	100000000	Set to sufficiently large number to disable time-based checkpoints forcing checkpoints only to occur on log switches.
gc_db_locks	191021	191021	This is the total number of Parallel Cache Management (PCM) locks covering the data blocks in the SGAs of a Parallel Server.
gc_rollback_segments	41	41	The maximum number of rollback segments systemwide. This parameter is set to the total number of rollback segments acquired by all instances in a parallel server.
V773_plans_enabled	true	true	Determines whether bitmap access paths will be considered for regular indexes on the tables that have at least one bitmap index.
B_tree_bitmap_plans	not used	true	Enable to use binary and bitmap indexes in combinations.

Enabling Process Limits for the Oracle DBA User Account

Edit the Oracle DBA User Account Process Settings to enable use of larger parameter process values that are required for the Oracle 64-bit option.

```
limit datasize unlimited
limit stacksize unlimited
limit memoryuse unlimited
limit addresspace unlimited
```

For more information, see *Oracle7 Server for Digital UNIX 7.3.3*. The following settings are an example using C shell. Refer to the *Oracle7 Installation Guide for Digital UNIX* for other shells.

Setting Up the Oracle7 Enterprise Backup Utility

This section describes how to set up the Oracle7 Enterprise Backup Utility. Before installing the Oracle7 Enterprise Backup Utility (OEBU), it is necessary to create a database to store the backup catalog.

1. Setup SQL*NET for the catalog database BKCT and the target database Consumer Packaged Goods (CPG) database.

   OEBU uses SQL*NET; therefore, appropriate entries must be made in tnsnames.ora and listener.ora.

   The following are the entries in tnsnames.ora and listener.ora for the catalog database, BKCT and the target database, CPG.

   ```
   $ORACLE_HOME/network/admin/tnsnames.ora

   BKCT.world=
     (DESCRIPTION=
           (ADDRESS=
              (PROTOCOL=IPC)
              (HOST=depot1)
              (KEY=BKCT)
           )
           (CONNECT_DATA=
              (SID=BKCT)
              (GLOBAL_NAME=BKCT.world)

           )
     )
   CPG1.world=
     (DESCRIPTION=
           (ADDRESS=
              (PROTOCOL=IPC)
              (HOST=depot1)
              (KEY=CPG1)
           )
           (CONNECT_DATA=
              (SID=CPG1)
              (GLOBAL_NAME=CPG1.world)
           )
     )


   $ORACLE_HOME/network/admin/listener.ora


   LISTENER =
     (ADDRESS_LIST =
           (ADDRESS =
              (PROTOCOL = IPC)
              (HOST=DEPOT1)
              (KEY=BKCT)
           )
     )
   (ADDRESS_LIST =
           (ADDRESS =
              (PROTOCOL = IPC)
              (HOST=DEPOT1)
   ```

```
            (KEY=CPG1)
          )
  )
STARTUP_WAIT_TIME_LISTENER = 0
CONNECT_TIMEOUT_LISTENER = 10
TRACE_LEVEL_LISTENER = 16
trace_directory_listener=/tmp
SID_LIST_LISTENER =
  ( SID_LIST =
    (SID_DESC =
       (SID_NAME = BKCT)
       (ORACLE_HOME = /oracle_733/app/oracle/product/7.3.2)
       (PROGRAM=oracle)
     )
    (SID_DESC =
       (SID_NAME = CPG1)
       (ORACLE_HOME = /oracle_733/app/oracle/product/7.3.2)
       (PROGRAM=oracle)
     )
  )
```

2. Start the SQL*Net listener:

   ```
   lsnrctl start
   ```

3. Create database BKCT to store the backup catalog.

4. Install the OEBU executables using the Oracle Installer. Do this after NSR and DMO are installed because it has to link against them.

   During installation, select Legato NetWorker as the third-party media management software vendor. Enter /usr/shlib as the directory path for the Networker API library.

5. Register the target database in the catalog using the OBACKUP REGISTER operation.

   For a full discussion of installing OEBU, see *Oracle7 Enterprise Backup Utility™ Installation Guide for DIGITAL UNIX, Release 2.1.0.1.2* (January 1996, Oracle Corp., Part No. A45307-1).

Oracle Patches

Apply the following patch to the system.

Patch # 510127 - TNS Authentication

Applying this patch will allow the OEBU target database to be registered successfully as discussed in Chapter 6.

SAS System Installation

Install the SAS System, Release 6.12 using the SAS System Installation Manager from the SAS System distribution media. See *Installation Instructions for the SAS System Under UNIX Environments, Release 6.12, TS040*, (Cary, NC:SAS Institute Inc., 1996).

Making SAS Available to Users

Use the link method to make the SAS System available to users, as described in the *Installation Instructions for the SAS System*.

Define a symbolic link for SAS as follows:

```
ln -s /oracle_733/sas612/sas /usr/bin/sas
```

Note

When issuing the `ln` command, you must be root user.

Editing the SAS Configuration File

Edit the SAS configuration file, config.sas612, to provide sufficient memory for the large SAS data set size as shown in Table 3-9.

Table 3-9: SAS Configuration File Settings

Setting	Comment
-memsize 4096m	-`memsize` limits the amount of memory allocated by the SAS System. Default is 32 MB.
-sortsize 4048m	-`sortsize` limits the amount of memory allocated during sorting operations. Default is 16 MB.
-work /sas2	-`work` specifies where the SAS work library is created. This is a temporary work library: SAS data sets created in this temporary file are deleted when the system terminates. Default is `/usr/tmp`.

4
Interoperability Tests and Results

This chapter describes how the tests were set up (including database organization), what data and programs were placed on what disks, and how the tests were run.

This chapter describes:

- Test Environment
- Test Tools
- Test Configuration
- Test Process and Results

Overview of Results

Interoperability testing was performed successfully on the SAS Oracle OPS TruCluster PS HiTest Suite.

Oracle database tests were executed on the AlphaServer 8x00 TruCluster to verify the ability to process queries, in which all data resides in a local Oracle7 database on shared storage. Tests were conducted in both minimum and maximum configurations using a warm cache in all cases, and all users were local terminal users.

Subsequently, SAS system database tests were conducted with the SAS system running scripts to test forecasting and statistical modeling functions.

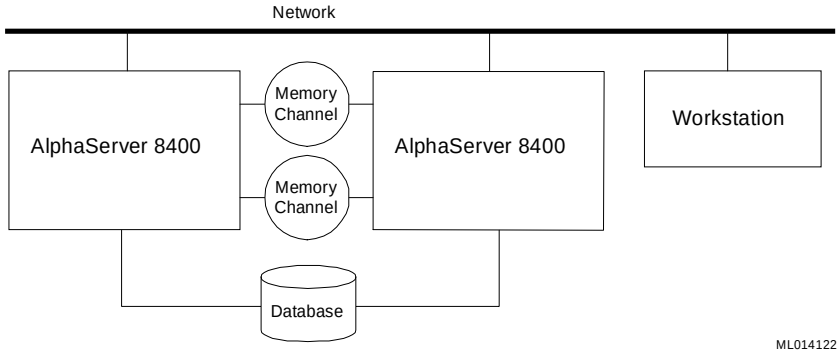
All tests demonstrated typical data warehousing type queries and verified interoperability of the SAS Oracle OPS TruCluster PS DIGITAL UNIX AlphaServer 8x00 HiTest Suite components.

Test Environment

Figure 4-1 shows the SAS Oracle OPS TruCluster PS DIGITAL UNIX AlphaServer 8x00 test environment.

Appendix A shows detailed test configurations of the DIGITAL UNIX AlphaServer 8x00 Foundation hardware.

Figure 4-1: Test Environment



Test Tools

All tests were executed as local terminal users, manually or via UNIX shell scripts.

Consumer Packaged Goods Database

The Consumer Packaged Goods database contains data that represents sales information for a variety of products across geographical areas of the US, including sales volume, channels, and other characteristics that were used as the test database. The minimum configuration database used represents one year of sales data. The database used to test the maximum configuration represents two years of sales data.

The database was built using the Consumer Packaged Goods Database Demo scripts provided by Oracle Corporation. The Consumer Packaged Goods Database represents typical marketing and sales data for a consumer products manufacturing firm. The data provides 24 months of sales data in a data warehouse, star schema, optimized for decision support.

Test Scripts

Test scripts were used to demonstrate and test the performance capabilities of the SAS Oracle TruCluster PS DIGITAL UNIX AlphaServer 8x00 HiTest Suite, in a Data Warehousing and Decision Support application. The scripts were designed to emulate typical decision support questions about the historical activity of a product sales environment.

Test Configuration

The database consists of distributed raw disks. Each volume is a six disk RAID-5 stripeset. Volumes are arranged on dual controller HSC50 pairs.

Storage Considerations

Performance considerations guided the placement of data files. The Oracle data files are primarily pathed across four HSZ50 Array Controller pairs. In general, data and the associated indexes should be placed on different controllers. However, by using the table partitioning strategy, maximum fan-out across all available I/O channels of both data and indexes can be achieved. This is done by combining data and indexes from different table partitions on individual RAIDsets sharing a controller. As shown in Table 4-2 (Maximum Configuration), the Oracle data files for the 24 FACTS partitions (representing 24 months of sales data) are matched across I/O channels and the 24 FACTSINDEX tablespaces are similarly placed, but in the reverse order. This configuration maximizes I/O access paths, while minimizing I/O contention for data load and data retrieval.

Also consider the SAS advfs file systems. In support of Oracle Parallel Server, the database resides on distributed raw devices which are members of ASE DRD services (refer to Table 3-3). The SAS advfs file systems are members of an ASE NFS service (Table 3-4), for failover purposes. Partitions of an individual RAID volume cannot be members of multiple ASE services due to ASE limitations regarding ownership and locking.

The minimum requirement for TEMP storage is approximately equivalent to the size of a single table partition. Optionally, additional TEMP files can be used to build multiple indexes in parallel. Six TEMP data files where chosen as a reasonable trade-off between storage requirements and system resource utilization benefits of parallel operations. However, optimal placement could not be achieved due to storage requirements for SAS and the ASE service limitation previously explained. Bitmap index data files where placed likewise, to optimize on storage space rather than performance.

Minimum Configuration

The minimum configuration database includes 12 data files comprising the 12 SALES_FACT partitions. The indexes for the SALES_FACT tables consist of 12 data files, 2 data files of TEMP space were used to build the SALES_FACT_INDEXES. The database used to test the minimum configuration represents one year of sales data. Table 4-1 shows the minimum configuration of the file systems and database shared storage map.

Table 4-1: Systems and Database Storage Map (Minimum Configuration)

SCSI Bus	Underlying Storage	ASE Service	DRD Device (File System Type)	File System and Database Links	Size
SCSI Bus #1	**HSZ50 Controller**				
	rrz8a	oracle_ase	(ufs)	/oracle_733	5112 M
	rrz8b				5112 M
	rrz8g				5112 M
	rrz8h				5112-M
	rrzb8a	drd_depot1	drd1	DIMENSIONS	1024 M
	rrzb8b	drd_depot1	drd2	LOGA_1	1024 M
	rrzb8d	drd_depot1	drd3	LOGB_1	1024 M
	rrzb8e	drd_depot1	drd4	LOGC_1	1024 M
	rrzb8f	drd_depot1	drd5	CTL_1	1024 M
	rrzb8g	drd_depot1	drd6	RBS_1	1024 M
	rrzb8h		drd38	TEMP1	14305 M
	rrz8c	sas_ase	(advfs)	/sas1	20449 M
	HSZ50 Controller				
	rrz9a	drd_depot1	drd7	FACT1	5112 M
	rrz9b	drd_depot1	drd8	FACT3	5112 M
	rrz9g	drd_depot1	drd9	FACT5	5112 M
	rrz9h	drd_depot1	drd10	FACT7	5112 M
	rrzb9a	drd_depot1	drd11	FACT9	5112 M
	rrab9b	drd_depot1	drd12	FACT11	5112 M
	rrzb9g	drd_depot1	drd13	FACTINDEX2	5112 M
	rrzb9h	drd_depot1	drd14	FACTINDEX4	5112 M
	rrzc9a	drd_depot1	drd15	FACTINDEX6	5112 M
	rrzc9b	drd_depot1	drd16	FACTINDEX8	5112 M
	rrzc9g	drd_depot1	drd17	FACTINDEX10	5112 M
	rrzc9h	drd_depot1	drd18	FACTINDEX12	5112 M
SCSI Bus #2	**HSZ50 Controller**				
	rrz16a		(ufs)	/data	20449 M
	rrzb16a	drd_depot2	drd19	SYSTEM	1024 M
	rrzb16b	drd_depot2	drd20	DIMENINDEX	1024 M
	rrzb16c	drd_depot2	drd21	CTL_2	1024 M
	rrzb16d	drd_depot2	drd22	LOGA_2	1024 M
	rrzb16e	drd_depot2	drd23	LOGB_2	1024 M
	rrzb16f	drd_depot2	drd24	LOGC_2	1024 M
	rrzb16g	drd_depot2	drd25	RBS_2	1024 M
	rrzb16h		drd39	TEMP2	13281 M
	rrzb16c	sas_ase	(advfs)	/sas2	20449 M

SCSI Bus	Underlying Storage	ASE Service	DRD Device (File System Type)	File System and Database Links	Size
	HSZ50 Controller				
	rrz17a	drd_depot2	drd26	FACT2	5112 M
	rrz17b	drd_depot2	drd27	FACT4	5112 M
	rrz17g	drd_depot2	drd28	FACT6	5112 M
	rrz17h	drd_depot2	drd29	FACT8	5112 M
	rrzb17a	drd_depot2	drd30	FACT10	5112 M
	rrzb17b	drd_depot2	drd31	FACT12	5112 M
	rrzb17g	drd_depot2	drd32	FACTINDEX1	5112 M
	rrzb17h	drd_depot2	drd33	FACTINDEX3	5112 M
	rrzc17a	drd_depot2	drd34	FACTINDEX5	5112 M
	rrzc17b	drd_depot2	drd35	FACTINDEX7	5112 M
	rrzc17g	drd_depot2	drd36	FACTINDEX9	5112 M
	rrzc17h	drd_depot2	drd37	FACTINDEX11	5112 M

Maximum Configuration

The maximum configuration includes 24 data files for the 24 SALES_FACT*xx* table partitions. The indexes for the SALES_FACT table consists of 24 data files, six data files of TEMP space were used to build the SALES_FACT_INDEXES. The database used to test the maximum configuration is representative of two years of sales data. Table 4-2 shows the maximum configuration of the file systems and database-shared storage map.

Table 4-2: File Systems and Database Shared Storage Map (Maximum Configuration)

SCSI Bus	Underlying Storage	ASE Service	DRD Device (File System Type)	File System and Database Links	Size
SCSI Bus #1	**HSZ50 Controller**				
	rrz8a	oracle_ase	(ufs)	/oracle_733	5112 M
	rrz8b				5112 M
	rrz8g				5112 M
	rrz8h				5112 M
	rrzb8a	drd_depot1	drd1	DIMENSIONS	1024 M
	rrzb8b	drd_depot1	drd2	LOGA_1	1024 M
	rrzb8d	drd_depot1	drd3	LOGB_1	1024 M
	rrzb8e	drd_depot1	drd4	LOGC_1	1024 M
	rrzb8f	drd_depot1	drd5	CTL_1	1024 M
	rrzb8g	drd_depot1	drd6	RBS_1	1024 M
	rrzb8h				14305 M
	rrzc8c		(ufs)	/data1	20449 M
	HSZ50 Controller				
	rrz9a	drd_depot1	drd7	FACT1	5112 M
	rrz9b	drd_depot1	drd8	FACT7	5112 M
	rrz9g	drd_depot1	drd9	FACT13	5112 M

SCSI Bus	Underlying Storage	ASE Service	DRD Device (File System Type)	File System and Database Links	Size
	rrz9h	drd_depot1	drd10	FACT19	5112 M
	rrzb9a	drd_depot1	drd11	FACTINDEX6	5112 M
	rrzb9b	drd_depot1	drd12	FACTINDEX12	5112 M
	rrzb9g	drd_depot1	drd13	FACTINDEX18	5112 M
	rrzb9h	drd_depot1	drd14	FACTINDEX24	5112 M
	rrzc9a	drd_depot1	drd15	TEMP1	5112 M
	rrzc9b	drd_depot1	drd68	BITMAP1	5112 M
	rrzc9g	drd_depot1	drd63	TEMP2	5112 M
	rrzc9h	drd_depot1	drd65	BITMAP2	5112 M
SCSI Bus #2	**HSZ50 Controller**				
	rrz16c	sas_ase	(advfs)	/sas1	20449 M
	rrzb16a	drd_depot2	drd32	SYSTEM	1024 M
	rrzb16b	drd_depot2	drd33	DIMENINDEX	1024 M
	rrzb16c	drd_depot2	drd34	CTL_2	1024 M
	rrzb16d	drd_depot2	drd35	LOGA_2	1024 M
	rrzb16e	drd_depot2	drd36	LOGB_2	1024 M
	rrzb16f	drd_depot2	drd37	LOGC_2	1024 M
	rrzb16g	drd_depot2	drd38	RBS_2	1024 M
	rrzb16h				13281 M
	rrzb16c		(ufs)	/data2	20449 M
	HSZ50 Controller				
	rrz17a	drd_depot2	drd39	FACT2	5112 M
	rrz17b	drd_depot2	drd40	FACT8	5112 M
	rrz17g	drd_depot2	drd41	FACT14	5112 M
	rrz17h	drd_depot2	drd42	FACT20	5112 M
	rrzb17a	drd_depot2	drd43	FACTINDEX5	5112 M
	rrzb17b	drd_depot2	drd44	FACTINDEX11	5112 M
	rrzb17g	drd_depot2	drd45	FACTINDEX17	5112 M
	rrzb17h	drd_depot2	drd46	FACTINDEX23	5112 M
	rrzc17c	sas_ase	(advfs)	/sas2	20449 M
SCSI Bus #4	**HSZ50 Controller**				
	rrz32a	drd_depot1	drd16	FACT3	5112 M
	rrz32b	drd_depot1	drd17	FACT9	5112 M
	rrz32g	drd_depot1	drd18	FACT15	5112 M
	rrz32h	drd_depot1	drd19	FACT21	5112 M
	rrzb32a	drd_depot1	drd20	FACTINDEX4	5112 M
	rrzb32b	drd_depot1	drd21	FACTINDEX10	5112 M
	rrzb32g	drd_depot1	drd22	FACTINDEX16	5112 M
	rrzb32h	drd_depot1	drd23	FACTINDEX22	5112 M
	rrzc32a	drd_depot1	drd64	TEMP3	5112 M
	rrzc32b	drd_depot1	drd69	TEMP4	5112 M

SCSI Bus	Underlying Storage	ASE Service	DRD Device (File System Type)	File System and Database Links	Size
	rrzc32g				5112 M
	rrzc32h				5112 M
	HSZ50 Controller				
	rrz33a	drd_depot1	drd24	FACT4	5112 M
	rrz33b	drd_depot1	drd25	FACT10	5112 M
	rrz33g	drd_depot1	drd26	FACT16	5112 M
	rrz33h	drd_depot1	drd27	FACT22	5112 M
	rrzb33a	drd_depot1	drd28	FACTINDEX3	5112 M
	rrzb33b	drd_depot1	drd29	FACTINDEX9	5112 M
	rrzb33g	drd_depot1	drd30	FACTINDEX15	5112 M
	rrzb33h	drd_depot1	drd31	FACTINDEX21	5112 M
	rrzc33c	ase_ase	(advfs)	/sas1	20449 M
SCSI Bus #5	**HSZ50 Controller**				
	rrz40a	drd_depot2	drd47	FACT5	5112 M
	rrz40b	drd_depot2	drd48	FACT11	5112 M
	rrz40g	drd_depot2	drd49	FACT17	5112 M
	rrz40h	drd_depot2	drd50	FACT23	5112 M
	rrzb40a	drd_depot2	drd51	FACTINDEX2	5112 M
	rrzb40b	drd_depot2	drd52	FACTINDEX8	5112 M
	rrzb40g	drd_depot2	drd53	FACTINDEX14	5112 M
	rrzb40h	drd_depot2	drd54	FACTINDEX20	5112 M
	rrzc40a	drd_depot2	drd66	TEMP5	5112 M
	rrzc40b	drd_depot2	drd71	BITMAP3	5112 M
	rrzc40g	drd_depot2	drd67	TEMP6	5112 M
	rrzc40h	drd_depot2	drd70	BITMAP4	5112 M
	HSZ50 Controller				
	rrz41a	drd_depot2	drd55	FACT6	5112 M
	rrz41b	drd_depot2	drd56	FACT12	5112 M
	rrz41g	drd_depot2	drd57	FACT18	5112 M
	rrz41h	drd_depot2	drd58	FACT24	5112 M
	rrzb41a	drd_depot2	drd59	FACTINDEX1	5112 M
	rrzb41b	drd_depot2	drd60	FACTINDEX7	5112 M
	rrzb41g	drd_depot2	drd61	FACTINDEX13	5112 M
	rrzb41h	drd_depot2	drd62	FACTINDEX19	5112 M
	rrzc41c	sas_ase	(advfs)	/sas2	20449 M

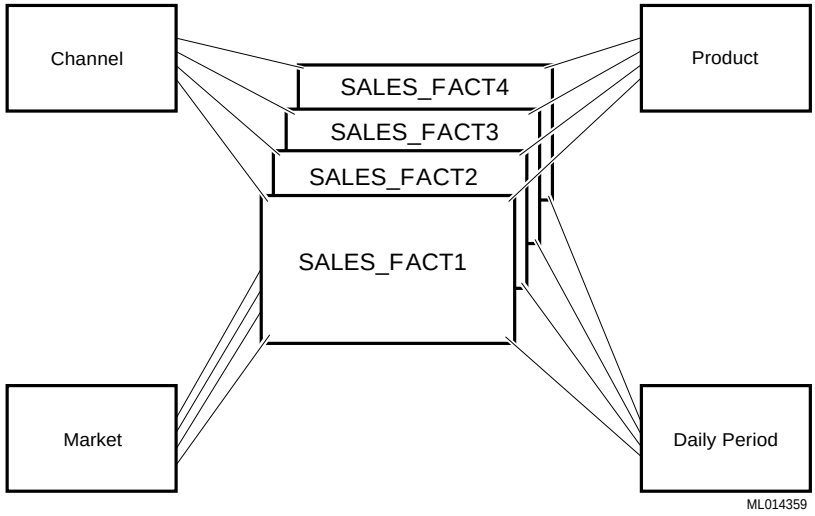
Test Process and Results

Interoperability of the SAS Oracle TruCluster PS DIGITAL UNIX AlphaServer 8x00 HiTest Suite was tested by loading data to the database using the SQL*Loader and creating the indexes required to access the stored data. Then, test queries were made to the data using Oracle SQL*PLUS and SAS applications to demonstrate typical data warehousing type queries.

Building the Database

The logical database design is based on a simple star schema with the large SALES_FACT view at the center and dimension tables around it. The SALES_FACT view is a union of tables, partitioned by month. The dimension tables are PRODUCT, MARKET, CHANNEL and DAILY_PERIOD. A composite index is created on each of the underlying partitions of the SALES_FACT view, where the index columns reference the unique key of each dimension table as shown in the following figure.

Figure 4-2: Logical Database Design



The shared storage is configured as 6 disk RAID-5 sets. The chunk size was set to to 256 disk blocks (131072 bytes). For optimal performance when writing a contiguous file, as occurs during tablespace creation, the oracle parameter ccf_io_size should be set to the number of disk spindles multiplied by the the chunk size. This value allows for concurrent asynchronous writes across all spindles of the RAIDset, 1 chunk/device.

ccf_io_size = 786432 bytes

Maximum Configuration

The database consists of 24 FACTS*xx*, 24 FACTSINDEX*xx* and 4 BITMAP*xx* tablespaces each 5112 MB in size. The TEMPFILE tablespace consists of 6 datafiles each 5112 MB in size. The 20 GB RAID-5 sets are disk labeled with 4 equal partitions for this purpose. Additionally, two 20 GB RAID-5 sets are labeled with 1 GB partition to store the SYSTEM (190 MB), DIMENSIONs (10 MB), DIMINDEX(10 MB) tablespaces, control files and redo logs (1000 MB/log).

The FACTS*xx* and FACTINDEX*xx* tablespaces must be created serially due to limitations in Oracle7 with data dictionary lock requirements. The extent size for these tablespaces is 1000 MB, with the exception of the initial extent of the FACTS*xx* tablespaces set to 64K. A small initial extent for the tablespaces to be loaded by SQL*Loader was chosen to minimize unusable space. These 5112 MB tablespaces each took about 12 minutes to create.

The BITMAP*xx* tablespaces where also created serially, but used a small extent size of 10 MB to facilitate a high degree of parallelism when creating the bitmap indexes.

TEMPFILE tablespace consists of multiple datafiles; therefore, the create tablespace operation allocates the first datafile, then the remaining datafiles are allocated in parallel using the alter tablespace operation. The extent size is set 200 MB. The alter operation, allocating the remaining 5 datafiles took an additional 12 minutes.

In total, the database consists of 58 datafiles of 5112 MB each for a total of 290 GB. The elapsed time for building the database was approximately 11 hours. This is a drawback of using partitioned views because system resources cannot be fully used to parallelize operations to reduce database build time.

Minimum Configuration

The database consists of 12 FACTS*xx* and 12 FACTSINDEX*xx* each 5112 MB in size. The TEMPFILE tablespace consists of 2 datafiles of 14305 MB and 13281 MB in size. The extent sizes are the same as that specified previously for the maximum configuration database. The total size of the database is approximately 147 GB in size and took approximately 6 hours to build.

Database Load, SQL*Loader

Planning sufficient storage for a dedicated data load volume is necessary in data warehouse environments, where data from production systems is regularly loaded in bulk. An inadequate storage strategy may cause bottlenecks, slowing the data load rate and limiting the use of multiple threads.

Data generation was partitioned by month, where one month's data is generated and formatted in a flat file. The flat file was used as input to SQL*Loader. In all cases, each stream targeted one data file, loading one month's data of approximately 98 million rows from a unique 4.3 GB flat file. The datafiles are loaded to 92% full, on average.

Minimum Configuration

One 20 GB file system was used as the load volume, for loading up to four months of data concurrently. Loads of two months and four months were tested, where 1/2 the target database files are on remote DRDs. Loads to remote DRDs took about 4% longer on average.

Table 4-2: Data Load for Minimum Configuration

Data Files	# of Rows	Size	Elapsed Time	I/O Rate
2	191977004	9520 MB	1 hr 48 min	5.2 GB/hr
4	383954008	19040 MB	3 hr 20 min	5.6 GB/hr

Maximum Configuration

Two 20 GB file systems were used as the load volumes for loading 6 months of data concurrently across 6 HSZ50 controllers. Each file system is locally mounted to the DRDs of the target database files.

Table 4-3: Data Load for Maximum Configuration

Data Files	# of Rows	Size	Elapsed Time	I/O Rate
1	97562084	4838 MB	1 hr 35 min	3.0 GB/hr
6	579078176	28717 MB	1 hr 37 min	17.3 GB/hr

Binary Index

Parallel index creation, provided by the Oracle Parallel Query option, is an important database management function. The Oracle Parallel Query option was tested as part of this suite. Parallel index creation was used during creation of each index on the SALES_FACT*xx* tables.

Calculating Index Storage

During index creation, it is important to allocate sufficient space for temporary storage for the index to build successfully. Due to the time required to build an index on a data warehouse, miscalculating these parameters can be costly.

Temporary space requirements should be estimated at a minimum of 10% above the required index space.

Calculate initial index and temporary space required to allow the creation of a concatenated index SF_KEY*xx* for the related SALES_FACT*xx* table.

Calculate the space requirements using the following procedure:

1. Obtain, or if necessary estimate, the total number of rows in the largest SALES_FACT*xx* table:

   rows = 97562084 (31 day month)

2. Calculate the index block header size, using the formula:

index block header size	= **fixed header + variable header** = 113 bytes + initrans*24 bytes = 113 bytes + 2*24bytes = 161 bytes

3. Calculate available data bytes per block as follows:

data bytes per block	= **DB block size - index block header size** = 32768 bytes - 161 bytes = 32607 bytes

4. Calculate the entry column size by summing the length of the columns in the index. The four columns used in this index are product_id, chan_id, market_id, and day, which are declared as 3 char (6) and a date field (7).

entry column size	= **sum (length of 4 columns)** = 6+6+6+7 = 25 bytes

5. Calculate the bytes per entry as follows:

bytes per entry	= **entry column size + entry header + rowid + F + V** = 25 + 2 + 6 + 5 + 0 = 38 bytes Givens: entry column size = 25 entry header = 2 bytes rowid = 6 bytes F = fixed length bytes/entry (# of columns 127 bytes) = 5 (4 specified columns plus 1 for rowid) V = variable length byte (0 for all calculations)

6. Calculate total index blocks required as follows:

total index blocks	1.05*(#rows*bytes per entry) = ---------- ((data bytes per block/bytes per entry)*byte per entry)) = 1.05 * ((97562084 * 38)/((32607/38) * 38)) = 119383 index blocks **Note:** The multiplier 1.05 is used to account for the index branch nodes and the use of parallel index create will most likely result in files that are not 100% used.
index bytes	= index blocks * db block size = 119383 * 32768 = 3911942144 bytes
index MB	= index bytes / (1024 * 1024) = 3911942144 / 1048576 = 3731 MB

7. The parallelism and extent sizes used to enhance performance will adversely affect efficiency of the storage utilization for the index. To account for this behavior, a 10% safety margin is built into the calculation of final index space as follows:

Final index space (MB)	= index MB * 1.1 = 3731 MB *1.1 = 4104.1 MB

8. Determine (whole number) the maximum number of extents in a single datafile. For this database, maximum number of extents is calculated as follows:

Maximum number of extents per file	= datafile size/extent size = 5112MB/1000MB = 5

9. Determine total number of files required to hold one SF_KEY*xx* index as follows:

index files	final index space (MB) = ---------- (extent size * number of extents per file) = 4104.1 MB/(1000MB*5) = 0.82 (always rounded up) = 1

For more information on calculating space for indexes, see *Oracle7 Server Administrator's Guide, Release 7.3*.

Creating the Binary Index

The concatenated SF_KEY*xx* must be created for each SALES_FACT*xx* table partition and then combined to form the union all view. Multiple indexes can be created concurrently, but concurrency is limited by the amount of TEMP space available. For each index to be built concurrently, an equal amount of TEMP space plus some additional overhead must be planned.

The Oracle parameter PARALLEL_MAX_SERVERS must be set to a value equal, or greater than, the sum of the parallel degree of each table for which an index is concurrently created, multiplied by two. The degree of parallelism for each index creation is implemented by two cooperating query servers, one for I/O and one for sorting. The parallel degree for index creations was enabled by altering the default degree on each SALES_FACT*xx* table.

The following svrmgr command shows the alter operation for one table, SALES_FACT1, as an example:

```
svrmgr> alter table SALES_FACT1 parallel (degree 4);
```

To maximize memory utilization during index creation, consider the following:

- Total amount of memory (per node) available for sort operations

  The total amount of memory used can be maximized through enabling the SORT_DIRECT_WRITES parameter. This parameter setting allows the SGA to be bypassed when performing index creation. For the purposes of index creation, the SGA can be greatly reduced by minimizing the buffer cache, maximizing the memory available for sort operations. For example, the maximum configuration (8 GB of memory) used a total of 6 GB of memory for sort operations, by reducing the buffer cache to 1000 MB.

  Use caution when enabling the UNIX kernel parameter gh-chunks; reduction in the SGA will not automatically return the memory for process use. An additional measure must be taken to reduce the dedicated shared memory region by decreasing the gh-chunks parameter, thus allowing the increase in pageable memory available to the sort processes, see Table 3-2.

- Amount of memory available per process thread

  The second consideration regarding memory allocation is at the process thread level and is controlled by the Oracle parameter SORT_AREA_SIZE. This value is allocated for each sort process thread. The maximum value for this parameter should never exceed the total pageable memory available divided by the number of sort process threads; where the sort process threads equals the sum of the parallel degrees of each table involved in concurrent index create operations per node.

  Table 4-4 shows how multiple indexes where built concurrently. The parallel degree of the tables as well as the sort_area_size parameter are shown. The elapsed time refers to the longest running create operation.

Table 4-4: Index Parameters

# of indexes	Degree	sort_area_size	Elapsed Time
Minimum Configuration			
2 (1/node)	4	419430400	1 hr 15 min
4 (2/node)	2	419430400	2 hr 24 min
6 (3/node)	2	146800640	2 hr 56 min
Maximum Configuration			
3	4	536870912	1 hr 3 min
6 (3/node)	4	536870912	1 hr 21 min

Union All Partition View

The partitions views are defined through the use of check constraints where the DAY column is constrained to a calendar month. The following example is an alter operation defining the constraint for one table partition. This operation took approximately 1 hour. Constraints are defined for each table partition; however, the alter table operations can be performed in parallel:

```
alter TABLE SALES_FACT1 add constraint sf1 check (day between
     '01-JAN-95' and '31-JAN-95');
```

The union all view is then created as shown in the following example. This operation took less than a minute for both the minimum and maximum configurations with 12 and 24 underlying table partitions respectively:

```
create view sales_fact as
        select * from sales_fact1 union all
        select * from sales_fact2 union all
        select * from sales_fact3 union all
        select * from sales_fact4 union all
                          :
                          :
        select * from sales_factn;
```

Bitmap Index

Oracle7 Server, Version 7.3.3 provides powerful functionality, in the form of bitmap indexes. The use of bitmap indexes has the potential to significantly reduce index storage space requirements, and improve performance on queries that involve low cardinality columns.

Table 4-5 shows the tablespace configurations for the Bitmap Indexes.

Table 4-5: Bitmap Index Tablespace Configuration

Tablespace	Contents	Datafile Name	MB Size
BITMAP_IDX1	sf_key_pid	/oracle/links/BITMAP1	5112 MB
BITMAP_IDX2	sf_key_mid	/oracle/links/BITMAP2	5112 MB
BITMAP_IDX3	sf_key_cid	/oracle/links/BITMAP3	5112 MB
BITMAP_IDX4	sf_key_did	/oracle/links/BITMAP4	5112 MB

This index type was tested by implementing four indexes on the SALES_FACT*xx* columns, used in the concatenated SF_KEY*xx* indexes. To completely replace the b-tree indexes on the SALES_FACT*xx* tables, four bitmap indexes must be created, as only one column is allowed in a bitmap index. Table 4-6 shows the creation times for each index.

All 24 indexes for each SALES_FACT*xx* partition were created concurrently.

Table 4-6: Creation Times for Bitmap Index

Index Name	Column Type	Cardinality	Elapsed Time
sf_key_pid*xx*	CHAR (6)	522	121 min
sf_key_mid*xx*	CHAR (6)	1002	80 min
sf_key_did*xx*	DATE	2189	108 min
sf_key_cid*xx*	CHAR (6)	41	82 min

Elapsed time represents the longest running index creation, average elapsed times are less.

Table 4-7 shows the default values for the parameters most likely to have a significant impact on the performance during the creation of bitmap indexes. These values were used in the creation of all of the bitmap indexes in the maximum configuration.

Table 4-7: Bitmap Parameter Settings

Parameter	Values
bitmap_merge_area_size	1048756
create_bitmap_area_size	8388608
v733_plans_enabled	true

These parameter values are used by each of the query servers in a parallel index create and, as with binary tree indexes, parallel creation of bitmap indexes is implemented using cooperating query servers. For more information on the use of bitmap indexes, see the *Oracle7 Server Tuning, Release 7.3* (June 1996, Oracle Corp., Part No. A32537-1).

Oracle Database Queries

Functional verification of the ability to perform query operations was demonstrated using five SQL join scripts and a full table scan. These queries exercised functionality of the RDBMS server, SQL*Plus, and Parallel Query.

The scripts were designed to emulate typical decision support questions about the historical activity of a product sales environment. In most cases, the result of these queries would be used to generate sales trends.

The following five queries were designed to search the database in varying ways to exercise the database. All queries returned results grouped by month.

Query 1

Query 1 asks “What was the product share of a specific brand of cereal as compared to other cereals in the same product category, in a particular state in a particular type of store?” The information was grouped by month to show market trends.

The business question asked is:

“How did 20 oz. Wheat Flakes do in 1995 as compared to all types of wheat flakes in supermarkets in the state of Connecticut?”

Query 2

Query 2 compares the sales of a specific product, in a particular outlet in a region, against the sales of the same product through all channel outlets. The information is grouped by month to show market trends.

The business question asked is:

"What percentage of sales of 15 oz. Wheat Flakes were made in the Safeway stores in NY and PA as compared to all outlets in the NY and PA areas?"

Query 3

Query 3 compares the market share of a product in a particular type of store, in a particular market location, to sales of all types of outlets in the region. The information was grouped by month to show market trends.

The business question asked is:

"How are 10 oz. Wheat Flakes doing in convenience stores in Bridgeport Connecticut as compared to the entire northeast region?"

Query 4

Query 4 compares the market share of a particular product, in a particular type of store, in a particular market location to all sales of competitive products in the same market location. The information is grouped by month to show market trends.

The business question asked is:

"What was the market share of 20 oz. Wheat Flakes in Connecticut supermarkets in 1995?"

Query 5

Query 5 compares the product share of a given product, combining several areas, to total sales across the same areas.

The business question asked is:

"What was the market share of 20 oz. Wheat Flakes across 10 test market areas?"

Oracle Database Tests

This section describes Oracle database tests. All tests were performed with a warm buffer cache. All times are reported in minutes and represent the elapsed time of the query. The "Min" column represents the shortest elapsed time of the query for any of the simultaneous users. Likewise, the "Max" column represents the longest elapsed time for the query across all simultaneous users.

The scripts for each of the queries are detailed in Appendix B.

Test Series One – Sequential Queries on Single Node

This test series demonstrates one to eight users running queries 1 to 5 sequentially on a single node of the 2 node OPS TruCluster where approximately half the database resides on local Distributed Raw Disks (DRDs) and the remaining data is on remote DRDs. The test series is repeated on both the minimum and maximum configurations. The database contains 2 years of sales fact data in the maximum configuration and 1 year of data in the minimum. Queries 1 to 5 will access the same 12 months of data to generate sales trends for that year. The query results are the same for both the minimum and maximum configurations, and are grouped by month.

The minimum configuration consists of 2 GB of memory with 2 CPU's in each cluster node. Shared storage consists of 2 SCSI buses, 2 controller pairs, and 12 (6 member) RAID-5 stripesets.

As indicated by the timed results shown in Table 4-8, overall performance degrades by 24% as the workload increases from a single user to 8 simultaneous users. As the number of users surpasses the number of CPUs, performance significantly degrades: 2% at 2 users, 8% at 4 users, and 24% at 8 users. Although, query 4 is the longest running query in this series, it scales fairly well with only a 14% performance hit at 8 users; whereas query 3, a relatively short-running query, is dramatically impacted by over 60%.

Table 4-8: Sequential Queries on Single Node – Minimum Configuration

Query	1 User		2 Users		4 Users		8 Users	
	Min	Max	Min	Max	Min	Max	Min	Max
Query 1	2.10	2.10	2.41	2.42	2.60	2.60	3.08	3.12
Query 2	2.81	2.81	2.80	2.80	2.96	2.97	3.38	3.39
Query 3	1.23	1.23	1.26	1.26	1.46	1.46	1.96	1.98
Query 4	13.53	13.53	13.63	13.64	14.09	14.11	15.34	15.36
Query 5	1.07	1.07	1.09	1.09	1.32	1.32	1.75	1.77

The maximum configuration consists of 8 GB of memory and 8 CPUs per cluster node. The database and storage is double that of the minimum configuration in both size and bandwidth.

The same test series was performed on the maximum configuration with unexpected results, shown in Table 4-9. A major concern is the drastically different behavior of queries 2 and 4 and the timed results in comparison to those previously seen in the minimum configuration (Table 4-8). Investigation of the query execution plans indicates the Oracle 7 optimizer has chosen different plans for these queries. Furthermore, trace information does not indicate that partition elimination has occurred in the maximum configuration with the 12 partitions of the second years data (1 per month), as expected. In depth investigation would be required to predict the actions of the optimizer with partitioned views as they increase in number.

Also, considering other varying factors in CPU, memory and storage between these configurations, it is impossible to draw conclusions regarding performance and scalability from this testing. It is important to note that the cumulative factors have resulted in vastly different execution plans, and an existing potential for variances in performance. Of particular concern is the choices made by the cost-based optimizer regarding partitioned views, as this behavior was not apparent in previous testing of Oracle 7 without partitioning. The following example is, in part, taken from the query 2 trace information. It highlights the different operations and resulting row counts incurred by the optimizer.

Query 2 Trace - Minimum Configuration

```
    Rows   Execution Plan
--------   ----------------------------------------------------------

       0   SELECT STATEMENT   GOAL:  CHOOSE
      24    SORT (UNIQUE)
      24     UNION-ALL
  182500      SORT (GROUP BY)
  283500       HASH JOIN
    2189        TABLE ACCESS  GOAL: ANALYZED (FULL) OF 'DAILY_PERIOD'
  182500         NESTED LOOPS
     800          MERGE JOIN (CARTESIAN)
      20           NESTED LOOPS
       1            TABLE ACCESS   GOAL: ANALYZED (BY ROWID) OF
                       'PRODUCT'
       1             INDEX   GOAL: ANALYZED (UNIQUE SCAN) OF
                         'P_PRODUCT' (UNIQUE)
    1002            TABLE ACCESS   GOAL: ANALYZED (FULL) OF 'MARKET'
      40           SORT (JOIN)
      41            TABLE ACCESS   GOAL: ANALYZED (FULL) OF 'CHANNEL'
  182500          VIEW OF 'SALES_FACT'
  182500           UNION-ALL (PARTITION)
   15500            FILTER
   15500             TABLE ACCESS   GOAL: ANALYZED (BY ROWID) OF
                         'SALES_FACT'
       :
       :
       :
```

Query 2 Trace - Maximum Configuration

```
    Rows   Execution Plan
--------   ----------------------------------------------------------

       0   SELECT STATEMENT   GOAL:  CHOOSE
      24    SORT (UNIQUE)
      24     UNION-ALL
  182500      SORT (GROUP BY)
  285000       HASH JOIN
    2189        TABLE ACCESS  GOAL: ANALYZED (FULL) OF 'DAILY_PERIOD'
  584000        HASH JOIN
    1002         TABLE ACCESS  GOAL: ANALYZED (FULL) OF 'MARKET'
18286500          NESTED LOOPS
      40          NESTED LOOPS
       1           TABLE ACCESS   GOAL: ANALYZED (BY ROWID) OF
                      'PRODUCT'
       1            INDEX   GOAL: ANALYZED (UNIQUE SCAN) OF
                       'P_PRODUCT' (UNIQUE)
      41           TABLE ACCESS   GOAL: ANALYZED (FULL) OF 'CHANNEL
18286500           VIEW OF 'SALES_FACT'
18286500            UNION-ALL (PARTITION)
  776550            FILTER
  776550             TABLE ACCESS   GOAL: ANALYZED (BY ROWID) OF
                        'SALES_FACT'
       :
       :
       :
```

In general, the use of partitioned views with Oracle 7 should be carefully considered regarding scalability in performance versus the benefits associated with data warehouse maintenance and storage requirements. Further analysis of the Oracle 7 cost-based optimizer is out of the scope of this document. Significant changes in Oracle 8 are likely to correct these issues.

Table 4-9: Sequential Queries on Single Node – Maximum Configuration

Query	1 User		2 Users		4 Users		8 Users	
	Min	Max	Min	Max	Min	Max	Min	Max
Query 1	0.56	0.56	0.36	0.38	0.38	0.40	0.38	0.42
Query 2	9.78	9.78	9.33	9.91	9.58	10.83	9.87	11.10
Query 3	2.21	2.21	2.23	2.32	2.32	2.38	2.52	3.17
Query 4	1.12	1.12	0.98	0.99	1.07	1.10	1.08	1.18
Query 5	2.62	2.62	2.59	2.60	2.69	2.91	2.77	3.02

Test Series Two – Concurrent Queries on Single Node

This test series demonstrates how the same workload shown in Test Series One can use multiple processes per user to reduce the total elapsed execution time. For each user, 5 processes concurrently run one query each and the total elapsed time is measured by the single longest running query of any user. In the previous test series, all users where fetching the same data simultaneously; whereas, this scenerio varies the buffer cache by having different queries running concurrently.

The minimum configuration 5-process test shown in Table 4-10 shows a 38% improvement in total elapsed time over the same one-user workload in the previous test. As the workload increases, overall performance improvements decrease to 24% at the 8-user/40-process test. All total elapsed times in this test series for each set of users are superior than the previous scenerio; however, individual query performance may suffer. This demonstrates the ability to fully use system resources and improve throughput for the entire workload at the potential cost in individual query response time.

Table 4-10: Concurrent Queries on Single Node – Minimum Configuration

Query	1 User 5 Processes		2 Users 10 Processes		4 Users 20 Processes		8 Users 40 Processes	
	Min	Max	Min	Max	Min	Max	Min	Max
Query 1	5.30	5.30	4.01	4.02	4.64	4.68	6.48	6.84
Query 2	3.38	3.38	4.49	4.50	3.83	3.89	6.90	7.34
Query 3	1.40	1.40	1.58	1.59	2.30	2.34	6.03	6.36
Query 4	12.90	12.90	13.99	13.99	14.38	14.43	19.06	19.39
Query 5	1.34	1.34	1.38	1.39	1.96	2.02	4.89	5.47

Repeating the same test series on the maximum configuration as shown in Table 4-11, query 2 further demonstrates the cost of an inefficient execution plan. The extraneous rows of data processed by this plan impede the performance of the remaining queries as the cache hit ratio decreases.

Table 4-11: Concurrent Queries on Single Node – Maximum Configuration

Query	1 User 5 Processes		2 Users 10 Processes		4 Users 20 Processes		8 Users 40 Processes	
	Min	Max	Min	Max	Min	Max	Min	Max
Query 1	0.38	0.38	0.53	0.56	0.93	1.18	1.88	2.09
Query 2	9.19	9.19	10.05	10.31	12.21	12.80	18.46	20.48
Query 3	2.44	2.44	2.74	2.80	4.95	5.23	9.76	10.58
Query 4	1.14	1.14	1.32	1.33	2.64	2.77	5.28	6.00
Query 5	2.87	2.87	3.31	3.64	5.20	5.72	10.55	11.98

Test Series Three – Sequential Queries on Two Node OPS

This test expands Test Series One to a two node Oracle7 Parallel Server (OPS) environment demonstrating scalability and load balancing. Table 4-12 and Table 4-14 indicate the minimum and maximum query execution times across both nodes.

Table 4-12: Sequential Queries on Two Node OPS – Minimum Configuration

Query	2 Users 1 Process/ Node		4 Users 2 Processes/ Node		8 Users 20 Processes/ Node		16 Users 40 Processes/ Node	
	Min	Max	Min	Max	Min	Max	Min	Max
Query 1	2.44	2.52	.2.55	2.61	2.84	2.90	3.29	3.39
Query 2	2.85	2.89	2.87	2.95	3.07	3.17	3.51	3.62
Query 3	1.29	1.30	1.30	1.33	1.47	1.52	1.62	2.12
Query 4	13.99	14.06	13.76	13.99	14.11	14.49	15.26	15.77
Query 5	1.06	1.12	1.09	1.16	1.32	1.36	1.75	1.89

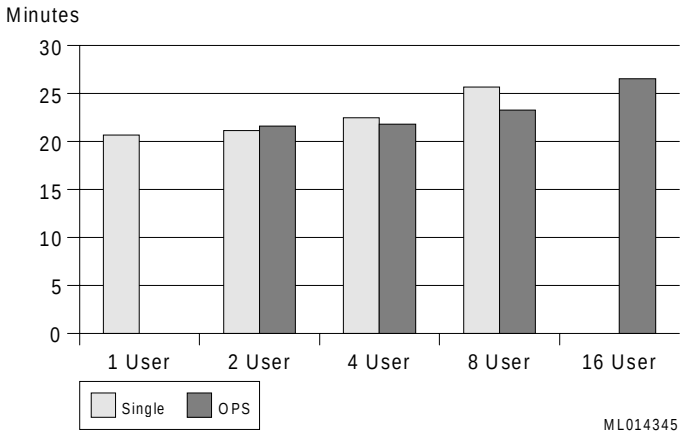
Table 4-13 compares the maximum elapsed times for each query to the single node results from Table 4-8. At a workload of 2 users, the OPS environment cannot benefit from the cache hits seen on the single node where multiple users can share cached buffers. However, as the workload increases the OPS advantage is apparent.

Table 4-13: OPS vs. Single Node Comparison – Minimum Configuration

Query	2 Users		4 Users		8 Users		16 Users	
	Single	OPS	Single	OPS	Single	OPS	Single	OPS
Query 1	2.42	2.52	2.60	2.61	3.12	2.90	N/A	3.39
Query 2	2.80	2.89	2.97	2.95	3.39	3.17	N/A	3.62
Query 3	1.26	1.30	1.46	1.33	1.98	1.52	N/A	2.12
Query 4	13.64	14.06	14.11	13.99	15.36	14.49	N/A	15.77
Query 5	1.09	1.12	1.32	1.16	1.77	1.36	N/A	1.89
Total Elapsed Time:	21.21	21.89	22.46	22.04	25.62	23.44	N/A	26.79

At the 4-user workload, balancing the users across the 2-node OPS environment shows a 2% performance improvement. When doubling the workload to 8 users, performance improves by approximately 10%. As the workload increases, the performance gains of the 2-node OPS cluster over a single node increases, as shown in Figure 4-2.

Figure 4-3: Single Node vs. 2 Node OPS Performance Comparison



To further demonstrate scalability of the OPS environment, performance degradation as the workload increases can be measured in both environments individually. In the single node, tests response times degraded by 21% at the 8 user test from the 2 user. In comparison, the OPS performance degradation for the same workload is only 7%.

This analysis reinforces the balancing benefit of clusters.

Repeating the same test series on the maximum configuration, Table 4-14 shows the minimum and maximum query execution times across both nodes.

Table 4-14: Sequential Queries on 2 Node OPS – Maximum Configuration

Query	2 Users 5 Processes		4 Users 10 Processes		8 Users 20 Processes		16 Users 40 Processes	
	Min	Max	Min	Max	Min	Max	Min	Max
Query 1	0.34	0.35	0.36	0.37	0.37	0.39	0.36	0.42
Query 2	8.88	9.31	9.32	9.94	9.38	10.34	9.60	11.41
Query 3	2.20	2.35	2.42	2.47	2.34	2.45	2.18	2.91
Query 4	1.01	1.07	1.00	1.06	1.00	1.11	1.03	1.36
Query 5	2.49	2.63	2.67	2.95	2.65	2.94	2.54	3.37

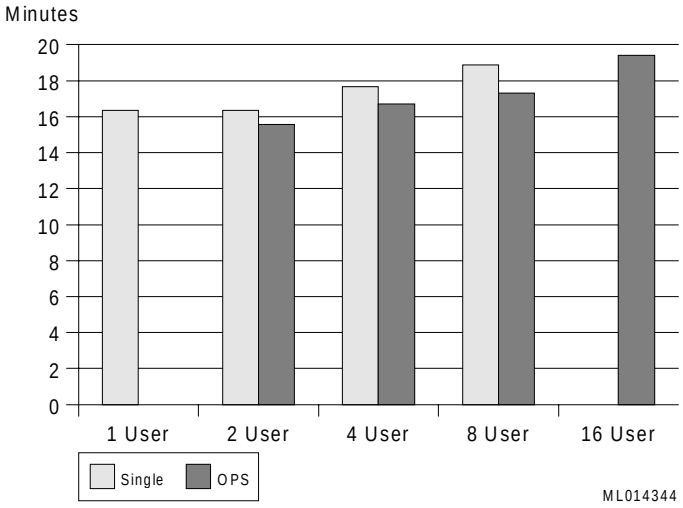
As seen in the minimum configuration, performance improves by 3% to 10% over the single node results of Test Series One as the workload increased from 2 users to 8 users.

Table 4-15: OPS vs. Single Node Comparison – Maximum Configuration

Query	2 Users		4 Users		8 Users		16 Users	
	Single	OPS	Single	OPS	Single	OPS	Single	OPS
Query 1	0.38	0.35	0.40	0.37	0.42	0.39	N/A	0.42
Query 2	9.91	9.31	10.83	9.94	11.10	10.34	N/A	11.41
Query 3	2.32	2.35	2.38	2.47	3.17	2.45	N/A	2.91
Query 4	.99	1.07	1.10	1.06	1.18	1.11	N/A	1.36
Query 5	2.60	2.63	2.91	2.95	3.02	2.94	N/A	3.37
Total Elapsed Time:	16.20	15.71	17.62	16.79	18.89	17.23	N/A	19.47

Using the same analysis to demonstrate cluster scalability, performance degradation is measured as the workload increases in both environments independently. The single node test response time degradation is 17% at the 8 user work level over the 2 user work level. Whereas, the OPS tests response time degradation for the same workloads is only 10%.

Figure 4-4: Single Node vs. 2 Node OPS Performance Comparison



Test Series Four – Parallel Query

To exercise the Parallel Query Option a full table scan of the union all view sales_fact*xx* partitions was performed by obtaining the row count of the view. Note that Internode Parallel Query (IPQ) was not used due to limitations in this functionality in Oracle 7.3 on DIGITAL UNIX. The following SQL statement includes the necessary hint required to process multiple partitions in parallel:

```
select /*+ full(s) parallel (s,24) */ count(*)
      from sales_facts;
```

The parameter for the parallel degree in the previous statement was set to 24 in the minimum configuration and 48 in the maximum configuration. In both cases, this resulted in 2 parallel query threads per partition.

Configuration	Degree	Row Count	Elapsed Time	I/O Rate
Minimum	24	1148714860	57 Min.	993 MB/min.
Maximum	48	2297429720	74 Min.	1530 MB/min.

Bitmap Index Tests

Bitmap indexes were unsuccessfully tested in combination with the binary indexes on the union all view partitions. Several attempts through the use of hints resulted in either the binary indexes being used or full table scans performed on all partitions. This was verified by examining the execution plans. Further investigation is necessary into the behavior of the cost-based optimizer to determine functionality. Previous testing of bitmap indexes for Oracle 7.3, in the absence of binary indexes, proved functional but not optimal and did not include partitioned tables.

The parameter, b_tree_bitmap_plans=true was enabled for using binary and bitmap indexes in combination; however, this does not appear to function for the queries tested. Any solution is highly dependent on query access patterns and the cardinality of the columns used.

A different set of test queries involving fewer columns and lower cardinality is necessary to exercise this functionality.

Backup and Restore Tests

Offline database backup using NSR and OEBU was executed to demonstrate interoperability. Additional tests were run with varying degrees of parallelism (for example, tape devices and I/O streams) and with local and remote DRD access.

The backup I/O rates are greatly effected by the storage layout of tablespaces and data files. Also of importance is the sequence in which the tablespaces and data files are created (or added) to the database. OEBU determines the order of backup streams by ordering the tablespaces by create date. All data files associated with a tablespace are then ordered by their create dates before the next tablespace in chronological order is chosen. Therefore, not only the placement of data files, but the order in which data files are added during database build, must be considered to achieve maximum fan-out of the OEBU I/O streams for optimal performance.

Table 4-16 shows the results of the backup tests performed. I/O rates are an average of the rate at which data is read from the database. Actual data written to tape may vary depending on compression.

Table 4-16: Backup Test Results

Parallelism	Devices	DRDs	Elapsed Time	I/O Rate
12	6	50% remote	3 hrs. 31 min.	77 GB/hr.
12	6	local	2 hrs. 47 min.	97 GB/hr.
6	6	50% remote	4 hrs. 23 min.	62 GB/hr.
6	6	local	2 hrs. 57 min.	92 GB/hr.
2	2	local	5 hrs. 48 min.	47 GB/hr.

Table 4-16 shows significant improvement in the I/O rate if achieved when all DRD's are local to the backup node. However, little improvement was made when increasing the parallelism from 6 to 12. First, with parallelism of 12, the read I/O is constrained by 2 streams accessing a single RAID-5 set simultaneously. This could have been avoided if the tablespaces were added to the database in the order which would provide the widest fan-out at the RAIDset level instead of just at the controller level. Second, multiple I/O streams per tape device may cause contention. Increasing the dedicated KZPSAs to one per tape drive is recommended for optimal performance.

Observations have indicated a maximum sustained I/O bandwidth of a single HSZ50 at approximately 9.5 MB/sec, and a limit of approximately 13 MB/sec for a dual pair of HSZ50s. This infers the theoretical maximum read performance of this configuration to be 52 MB/sec. Actual throughput of the backup operation including all internal processing and compression is somewhat less than this.

Due to an isolated hard error on a tape drive, the restore operation did not complete and could not be verified at this time.

SAS System Database Tests

Functional testing of SAS included interactive script execution, where the interactive mode used the X-window display manager. Scripts retrieved data from the existing Oracle Consumer Packaged Goods Data Warehouse via SAS/Access. Forecasting and statistical modeling test scripts were executed.

Forecasting Tests

Scripts were executed to create data sets for forecasting at a daily, a weekly, and a monthly level result.

Test One

Daily forecast for a single market, single channel, single product, and two years of daily data to produce a 30-day ahead forecast on a daily level. The view of the graphed results contained the last 30 days and the 30 days ahead forecast.

Test Two

Weekly forecast for a single market, single channel, single product and two years of data aggregated to week-ending levels to produce a 12-week ahead forecast. The results contained data for the last 12 weeks and the forecast 12 weeks.

Test Three

Monthly forecast for a single market, single channel, single product, and two years of data aggregated to week-ending levels to produce a 12-week ahead forecast. The view of the graphed results contained all data points and forecast months.

Statistical Modeling Tests

Scripts were executed to create both views and data sets for statistical modeling to detect differences between stores, channels groups, products, and so on.

Test One

Build a statistical model of how channel and monthly factors effect daily sales using daily level data for a single market, single product, and two years of data. Examine the interaction between channel and month.

Test Two

Build two statistical models looking at channel effects and channel and month effects with no interactions using eight months of data aggregated to the monthly channel for a single market and a single product.

Test Three

Build a repeated measure analysis to examine channel group differences over time using eight months data aggregated to the monthly channel level for a single product and a single market.

5
System Limits and Characterization Data

This chapter describes any system limits that may have been determined as a result of the testing, along with information about the system characterization during testing.

There were no system limits discovered during integration testing of this HiTest Suite.

6
Problems and Resolutions

This chapter describes any problems that may have been encountered during the testing. Where appropriate, a solution for each problem is given which provides the system manager or user with a fix or workaround.

The following problems were identified during testing:

Foundation Hardware

I/O Adapter Failure

Problem PCIA error panics cause system crashes under heavy I/O load.

Resolution Two KSPSA-BB adapters were found to be defective and were replaced.

Shared Storage Availability

Problem Shared storage can become unavailable when certain hard errors are reported on a shared SCSI bus due to a defective KZPSA-BB adapter. The adapter failure may cause the SCSI bus to hang on other cluster nodes, resulting in a single point of failure.

Resolution Replace faulty adapter.

Operating System

Setting Address Space Unlimited

Problem Cannot set “Limit Address Space Unlimited” in C-Shell on the 8 GB memory configuration.

Resolution Set limit-h address 4117776K, then issue the “Limit Address Space Unlimited” command until a kernel rebuild can be done. Then edit the /sys/conf/nodename file and change the value for maxdsize to the memory size and then do a kernel rebuild.

Application

Oracle7 Database Issues

Multi-Block Read Count Limitation

Problem There is an undocumented limitation on multi-block read count. Due to a maximum of 128 KB per I/O, the maximum setting is 4 according to the formula: maximum db_file_multiblock_read_count = max_io_size/db_block_size (128 KB/32KB).

Resolution This parameter was set to 4 because 32K (db_blocks) was used, the maximum based on the formula.

OEBU Register Operation Fails

Problem OEBU register operation fails with an error:
ORA-12641: TNS: authentication service failed to initialize

Resolution Install Oracle patch 510127 as indicated in Chapter 3 or disable authentication by adding this line to sqlnet.ora: sqlnet.authentication_services=(NONE). However, this resolution will cause a "connect internal" problem.

Password Required for Connect Internal Command

Problem The Connect Internal command in the svrmgr1 requires a password.

Resolution Install Oracle patch 510127 as indicated in Chapter 3 or remove disabling of authentication by removing this line in sqlnet.ora:
sqlnet.authentication_services=(NONE)
However, this will cause a TSN authentication error with OEBU.

Advanced Network Option (ANO) Partially Installed

Problem The ANO is partially installed although not selected during Oracle product installation.

Workaround Install the ANO option, then de-install the ANO option.

Compatibility of Oracle 7.3.3 Control File

Problem Startup of the database fails after initial database creation with the following error:

ORA-0021: control file version 7.3.3.0.0
incompatible with ORACLE version 7.0.1.2.0.0

Resolution Include the following parameter in the Oracle initialization file: compatible=7.3.3

Parallelization of Partition View

Problem Full table scan of partition view does not parallelize across multiple partitions; each table partition is accessed serially.

Workaround Use a SQL hint for the parallelism on full table scan and also use a table alias.

Networker Save and Restore

Problem Use of NSR_DEBUG causes backup operation to hang and fully consume all CPU resources (brtp processes) after writing approximately 300 MB of data to tape.

Workaround Disable NSR_DEBUG.

A
Detailed Hardware Configuration

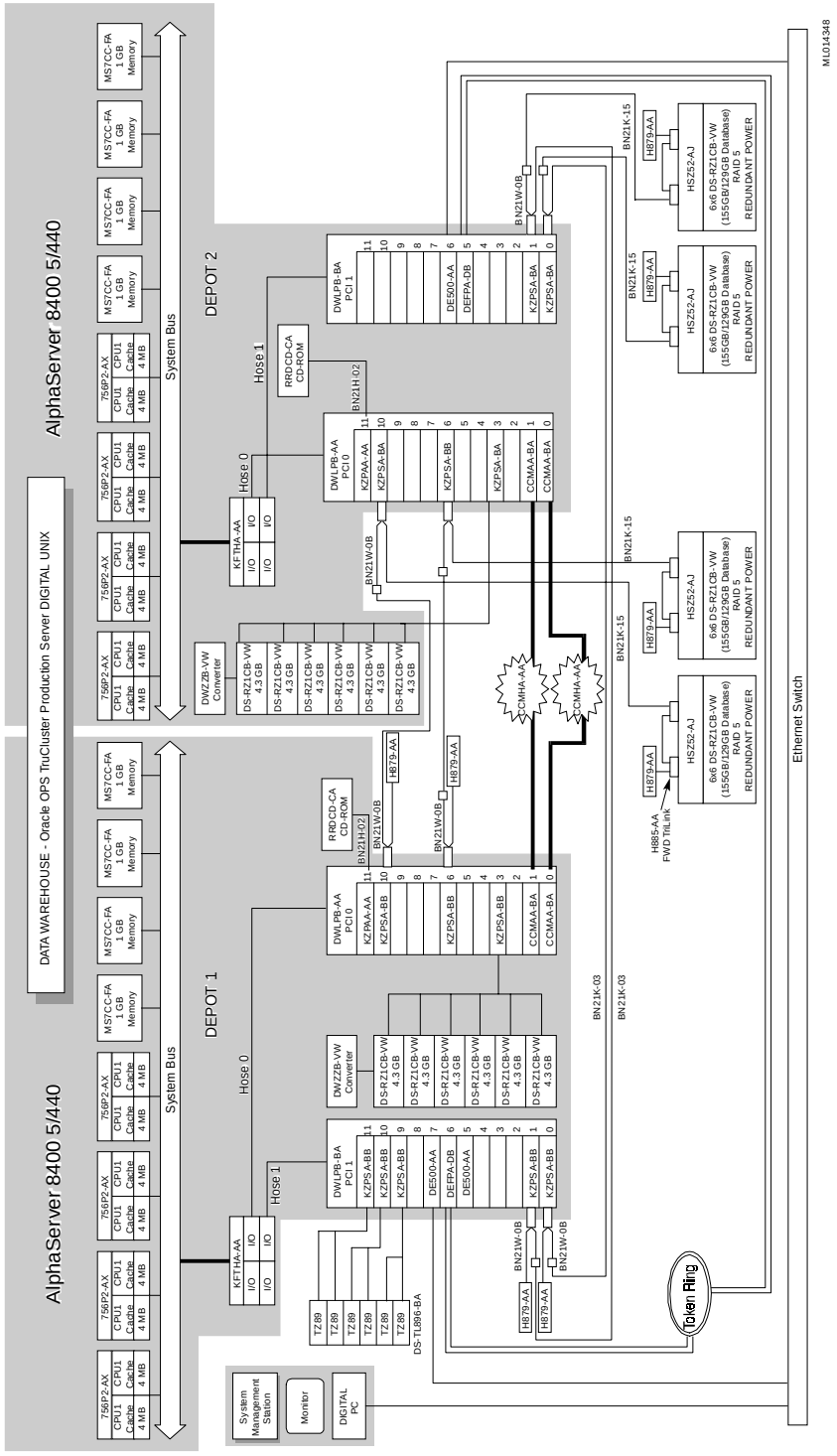
This appendix describes the minimum and maximum hardware configuration for the following:

- System Diagram
- AlphaServer 8x00 configurations, including:
  - System Centerplane
  - PCI backplanes (slot usage)
- Configuration Cabling

System Diagram

Figure 6-1 shows the maximum configuration of this HiTest suite.

Figure 6-1: System Diagram



AlphaServer 8x00 Configurations

Figure 6-2 and Table 6-1 show the AlphaServer 8x00 centerplane and describe the minimum and maximum hardware configurations used in this HiTest Template.

Figure 6-2: AlphaServer 8x00 Centerplane

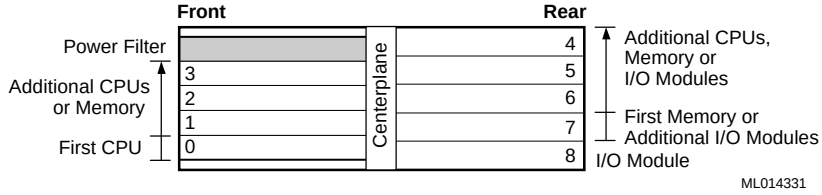


Table 6-1: AlphaServer 8x00 Usage (Minimum and Maximum Configuration)

Slot	Minimum Configuration Options	Maximum Configuration Options	Description
0	KN7CE-AB	KN7CE-AB	5/440 MHz Dual CPU
1	KN7CE-AB	KN7CE-AB	5/440 MHz Dual CPU
2		KN7CE-AB	5/440 MHz Dual CPU
3		KN7CE-AB	5/440 MHz Dual CPU
4	MS7CC-FA	MS7CC-FA	2 GB Memory Module
5		MS7CC-FA	2 GB Memory Module
6		MS7CC-FA	2 GB Memory Module
7		MS7CC-FA	2 GB Memory Module
8	KFTHA-AA	KFTHA-AA	System I/O Module (4 channel)

AlphaServer 8x00 PCI Slot Usage

Figure 6-3 and the following tables show the PCI slot usage for the minimum and maximum configurations of this HiTest Template.

Figure 6-3: AlphaServer 8x00 PCI Slot Usage

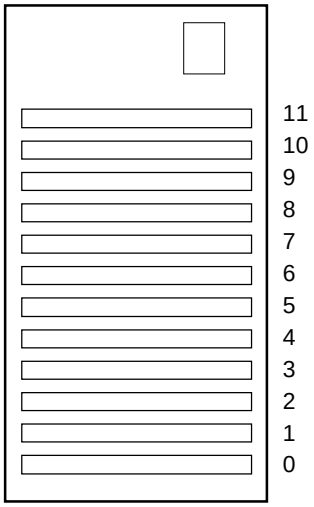


Table 6-2: PCI #0 (Depot1) Devices

Slots	Minimum Configuration Options	Maximum Configuration Options	Description
11	KZPAA-AA	KZPAA-AA	PCI one-port FNS SCSI controller
10	KZPSA-BB	KZPSA-BB	PCI one-port FWD SCSI controller
9	Available	Available	
8	Available	Available	
7	Available	Available	
6	KZPSA-BB	KZPSA-BB	PCI one-port FWD SCSI controller
5	Available	Available	
4	Available	Available	
3	KZPSA-BB	KZPSA-BB	PCI one-port FWD SCSI controller
2	Available	Available	
1	Available	CCMAA-BA	PCI to Memory Channel Controller
0	CCMAA-BA	CCMAA-BA	PCI to Memory Channel Controller

Table 6-3: PCI #1 (Depot1) Devices

Slots	Minimum Configuration Options	Maximum Configuration Options	Description
11	KZPSA-BB	KZPSA-BB	PCI one-port FWD SCSI controller
10	KZPSA-BB	KZPSA-BB	PCI one-port FWD SCSI controller
9	KZPSA-BB	KZPSA-BB	PCI one-port FWD SCSI controller
8	Available	Available	
7	DE500-AA	DE500-AA	Fast Ethernet Adapter
6	DEFPA-DB	DEFPA-DB	PCI to FDDI Adapter
5	DE500-AA	DE500-AA	Fast Ethernet Adapter
4	Available	Available	
3	Available	Available	
2	Available	Available	
1	Available	KZPSA-BB	PCI one-port FWD SCSI controller
0	Available	KZPSA-BB	PCI one-port FWD SCSI controller

Table 6-4: PCI #0 (Depot2) Devices

Slots	Minimum Configuration Options	Maximum Configuration Options	Description
11	KZPAA-AA	KZPAA-AA	PCI one-port FNS SCSI controller
10	KZPSA-BB	KZPSA-BB	PCI one-port FWD SCSI controller
9	Available	Available	
8	Available	Available	
7	Available	Available	
6	KZPSA-BB	KZPSA-BB	PCI one-port FWD SCSI controller
5	Available	Available	PCI one-port FWD SCSI controller
4	Available	Available	
3	KZPSA-BB	KZPSA-BB	PCI one-port FWD SCSI controller
2	Available	Available	
1	Available	CCMAA-BA	PCI to Memory Channel Controller
0	CCMAA-BA	CCMAA-BA	PCI to Memory Channel Controller

Table 6-5: PCI #1 (Depot 2) Devices

Slots	Minimum Configuration Options	Maximum Configuration Options	Description
11	Available	Available	
10	Available	Available	
9	Available	Available	
8	Available	Available	
7	Available	Available	Fast Ethernet Adapter
6	DE500-AA	DE500-AA	Fast Ethernet Adapter
5	DEFPA-DB	DEFPA-DB	PCI to FDDI Adapter
4	Available	Available	
3	Available	Available	
2	Available	Available	
1	Available	KZPSA-BB	PCI one-port FWD SCSI controller
0	Available	KZPSA-BB	PCI one-port FWD SCSI controller

Configuration Cabling

Table 6-6 lists the major cables in the HiTest Template.

Table 6-6: Configuration Cabling

Part Number	Qty	Description	From	To
BN21W-0B	8	SCSI Y cable, 68 pin	KZPSA-BB	Shared SCSI buses
BN21K-03	4	3 m SCSI-3 cable “P” straight/90°	SCSI Y-cable	SCSI Y-cable
BN21K-10	7	10 m SCSI-3 cable “P” straight/90°	SCSI Y-cable (9) KZPSA (2)	HSZ52 HSZ52
BN21L-0B	4	Shared SCSI bus	HSZ50	HSZ50
BN21N-02	24	8/16-bit SCSI bus	HSZ50	StorageWorks shelves
BC12N-10	4	Memory channel link	CCMAA-BA	CCMHA-AA
BN25G-07	4	10BaseT Twisted-Pair Ethernet Cable		

B
Test Scripts

This appendix shows examples of test scripts used for this HiTest suite:

- Functional test scripts using a forecasting model and a statistical model
- SQL queries that show the historical activity of a product sales environment

Functional Test Scripts

Functional testing of SAS included both interactive and batch modes of script execution, where interactive mode used the X-window display manager. Scripts retrieved data from the existing Oracle CPG Data Warehouse via SAS/Access.

Test One

Example of forecast script – forecast_month_dataset.sas

```
libname foo '/sas1';
goptions device=xcolor;
proc sql;
connect to oracle(user=cpg orapw=cpg);
create table forcastm as select * from connection to oracle(
select
 market,product,sum(dollar_sales) as dollar,month,channel,
sum(unit_sales) as unit
from market,product,
daily_period,channel,sales_fact where
market.market_id=sales_fact.market_id and
product.product_id=sales_fact.product_id and
channel.channel_id=sales_fact.channel_id and
 daily_period.day=sales_fact.day and market.market='Albany' and
channel.channel='Wal Mart' and
product.product='NUGN CTY WHT FLK 20 OZ' and
daily_period.month>'31-DEC-93' and month <'01-JAN-96'
group by market, product, month, channel);
quit;
```

```
proc forecast data=forcastm out=month outfull lead=12
              interval=dtmonth;
var dollar  unit;
id month ;
run;

data month; set month;
    date=datepart(month) ;  format date date7.;
run;
proc gplot data=month;
```

```
 plot dollar*date =_type_  /
 haxis='01jan94'd to '01dec96'd by month  href='01jan96'd;
 symbol1 i=none v=star h=1;
 symbol2 i=spline v=circle;
 symbol3 i=spline l=3;
 symbol4 i=spline l=3;
 legend  value=(  font=swissx) label=(font=swissx);
run;
quit;
```

Test Two

Example of statistical modeling script – channel_diff_month.sas

```
libname foo '/sas1';
proc sql;
connect to oracle(user=cpg orapw=cpg);
create table channeld as select * from connection to oracle(
select
 market,product,avg(dollar_sales) as dollar,month,channel,
avg(unit_sales) as unit,count(daily_period.day) as weight
from market,product,
daily_period,channel,sales_fact where
market.market_id=sales_fact.market_id and
product.product_id=sales_fact.product_id and
channel.channel_id=sales_fact.channel_id and
 daily_period.day=sales_fact.day and market.market='Wales 000' and
product.product='NUGN CTY WHT FLK 20 OZ' and
daily_period.month>'31-DEC-93' and month <'01-JAN-96'
group by market, product, month, channel
order by channel, month);
quit;

data month; set channeld;    date=datepart(month) ;  format date
date7.;
 if substr(channel,1,5)='Total' then delete;
run;

proc glm data=month; class channel;
model dollar= channel;
means channel/bon duncan;
run;quit;

proc glm data=month; class month channel;
model dollar= channel month;
means channel month/bon duncan;
run;quit;
```

SQL Queries

Query One

```
/*  1.  Star -- Product Share of Brand   */

select
SYSDATE,'All Wheat Flakes' Product, AL2.MONTH,
sum(AL5.UNIT_SALES) UNITS, sum(AL5.DOLLAR_SALES) DOLLARS, count(*),
DISTRICT, CHANNEL_GROUP CHNL
FROM PRODUCT AL4, SALES_FACT AL5,
CHANNEL AL1, DAILY_PERIOD AL2, MARKET AL3
WHERE (AL5.PRODUCT_ID=AL4.PRODUCT_ID
AND AL5.MARKET_ID=AL3.MARKET_ID
AND AL5.CHANNEL_ID=AL1.CHANNEL_ID
AND AL5.DAY=AL2.DAY)
AND (district='Connecticut'
AND CHANNEL_GROUP in('Supermarket')
AND BRAND in ('Quellogs Wheat Flakes')
AND YEAR=1995)
group by DISTRICT, CHANNEL_GROUP,
'All Wheat Flakes', AL2.MONTH, SYSDATE
UNION
select
SYSDATE, '20 Oz Wheat Flakes' Product, AL2.MONTH,
sum(AL5.UNIT_SALES) UNITS, sum(AL5.DOLLAR_SALES) DOLLARS, count(*),
DISTRICT, CHANNEL_GROUP CHNL
FROM PRODUCT AL4, SALES_FACT AL5,
CHANNEL AL1, DAILY_PERIOD AL2, MARKET AL3
WHERE (AL5.PRODUCT_ID=AL4.PRODUCT_ID
AND AL5.MARKET_ID=AL3.MARKET_ID
AND AL5.CHANNEL_ID=AL1.CHANNEL_ID
AND AL5.DAY=AL2.DAY)
AND (district ='Connecticut'
AND CHANNEL_GROUP in('Supermarket')
AND PRODUCT='QLGS WHT FLK 20 OZ'
AND YEAR=1995)
group by DISTRICT, CHANNEL_GROUP,
'20 Oz Wheat Flakes', AL2.MONTH, SYSDATE ;
```

Query Two

```
/*  2. Star -- Channel share of all channels  */

select
SYSDATE,  'All Channels' CHNL,  AL2.MONTH,
sum(AL5.UNIT_SALES) Units, sum(AL5.DOLLAR_SALES) Dollars, count(*),
'NY + PA' DISTRICT, PRODUCT
FROM PRODUCT AL4, SALES_FACT AL5,
CHANNEL AL1, DAILY_PERIOD AL2, MARKET AL3
WHERE (AL5.PRODUCT_ID=AL4.PRODUCT_ID
AND AL5.MARKET_ID=AL3.MARKET_ID
AND AL5.CHANNEL_ID=AL1.CHANNEL_ID
AND AL5.DAY=AL2.DAY)
AND DISTRICT in ('New York', 'Pennsylvania')
AND CHANNEL_GROUP in ('Supermarket','Convenience',
'Warehouse','Drug','Discount')
AND PRODUCT= 'QLGS WHT FLK 15 OZ'
```

```
AND YEAR=1995
group by SYSDATE, 'NY + PA', 'All Channels',
PRODUCT , AL2.MONTH
UNION
select
SYSDATE, CHANNEL CHNL,  AL2.MONTH,
sum(AL5.UNIT_SALES) Units, sum(AL5.DOLLAR_SALES) Dollars, count(*),
'NY + PA' DISTRICT , PRODUCT
FROM PRODUCT AL4, SALES_FACT AL5,
CHANNEL AL1, DAILY_PERIOD AL2, MARKET AL3
WHERE (AL5.PRODUCT_ID=AL4.PRODUCT_ID
AND AL5.MARKET_ID=AL3.MARKET_ID
AND AL5.CHANNEL_ID=AL1.CHANNEL_ID
AND AL5.DAY=AL2.DAY)
AND DISTRICT in ('New York' , 'Pennsylvania')
AND CHANNEL='Safeway'
AND PRODUCT='QLGS WHT FLK 15 OZ'
AND YEAR=1995
group by SYSDATE, 'NY + PA', CHANNEL,
PRODUCT, AL2.MONTH;
```

Query Three

```
/*  3.  Star -- Market share of Region  */


select
SYSDATE, 'Northeast Total' MARKET,  AL2.MONTH,
sum(AL5.UNIT_SALES) Units, sum(AL5.DOLLAR_SALES) Dollars, count(*),
CHANNEL_GROUP, PRODUCT
FROM PRODUCT AL4, SALES_FACT AL5,
CHANNEL AL1, DAILY_PERIOD AL2, MARKET AL3
WHERE (AL5.PRODUCT_ID=AL4.PRODUCT_ID
AND AL5.MARKET_ID=AL3.MARKET_ID
AND AL5.CHANNEL_ID=AL1.CHANNEL_ID
AND AL5.DAY=AL2.DAY)
AND (REGION='Northeast'
AND CHANNEL_GROUP in ('Convenience')
AND PRODUCT= 'QLGS WHT FLK 10 OZ'
AND YEAR=1995)
group by SYSDATE, 'Northeast Total', CHANNEL_GROUP,
PRODUCT , AL2.MONTH
UNION
select
SYSDATE, MARKET, AL2.MONTH,
sum(AL5.UNIT_SALES) Units, sum(AL5.DOLLAR_SALES) Dollars, count(*),
CHANNEL_GROUP, PRODUCT
FROM PRODUCT AL4, SALES_FACT AL5,
CHANNEL AL1, DAILY_PERIOD AL2, MARKET AL3
WHERE (AL5.PRODUCT_ID=AL4.PRODUCT_ID
AND AL5.MARKET_ID=AL3.MARKET_ID
AND AL5.CHANNEL_ID=AL1.CHANNEL_ID
AND AL5.DAY=AL2.DAY)
AND (MARKET='Bridgeport'
AND CHANNEL_GROUP in ('Convenience')
AND PRODUCT='QLGS WHT FLK 10 OZ'
AND YEAR=1995)
group by SYSDATE, MARKET, CHANNEL_GROUP,
PRODUCT, AL2.MONTH;
```

Query Four

```
/*  4. Star -- Product share of SubCategory -all competitive prods */

select
SYSDATE, 'All Wheat Products' Product, AL2.MONTH,
sum(AL5.UNIT_SALES) Units, sum(AL5.DOLLAR_SALES) Dollars, count(*),
DISTRICT, CHANNEL_GROUP CHNL
FROM PRODUCT AL4, SALES_FACT AL5,
CHANNEL AL1, DAILY_PERIOD AL2, MARKET AL3
WHERE (AL5.PRODUCT_ID=AL4.PRODUCT_ID
AND AL5.MARKET_ID=AL3.MARKET_ID
AND AL5.CHANNEL_ID=AL1.CHANNEL_ID
AND AL5.DAY=AL2.DAY)
AND (district='Connecticut'
AND CHANNEL_GROUP in('Supermarket')
AND BRAND IN ('Quellogs Wheat Flakes', 'Boast Weeties', 'Boast Oatey
Rounds',
'Quellogs Wheaten Rye')
AND YEAR=1995)
group by SYSDATE, DISTRICT, CHANNEL_GROUP,
'All Wheat Products', AL2.MONTH
UNION
select
SYSDATE, '20 Oz Wheat Flakes' Product, AL2.MONTH,
sum(AL5.UNIT_SALES) Units, sum(AL5.DOLLAR_SALES) Dollars, count(*),
DISTRICT, CHANNEL_GROUP CHNL
FROM PRODUCT AL4, SALES_FACT AL5,
CHANNEL AL1, DAILY_PERIOD AL2, MARKET AL3
WHERE (AL5.PRODUCT_ID=AL4.PRODUCT_ID
AND AL5.MARKET_ID=AL3.MARKET_ID
AND AL5.CHANNEL_ID=AL1.CHANNEL_ID
AND AL5.DAY=AL2.DAY)
AND (district ='Connecticut'
AND CHANNEL_GROUP in('Supermarket')
AND PRODUCT='QLGS WHT FLK 20 OZ'
AND YEAR=1995)
group by SYSDATE, DISTRICT, CHANNEL_GROUP,
'20 Oz Wheat Flakes', AL2.MONTH;
```

Query Five

```
/*  5. Star -- Product share of brand in 10 test markets aggregated */

select
SYSDATE, 'All Wheat Flakes' Product, AL2.MONTH,
sum(AL5.UNIT_SALES) Units, sum(AL5.DOLLAR_SALES) Dollars, count(*),
CHANNEL_GROUP CHNL, '10-States'
FROM PRODUCT AL4, SALES_FACT AL5,
CHANNEL AL1, DAILY_PERIOD AL2, MARKET AL3
WHERE (AL5.PRODUCT_ID=AL4.PRODUCT_ID
AND AL5.MARKET_ID=AL3.MARKET_ID
AND AL5.CHANNEL_ID=AL1.CHANNEL_ID
AND AL5.DAY=AL2.DAY)
AND (district in ('Connecticut',
'Delaware','Maine','Pennsylvania','New York',
'Oregon', 'Alaska', 'CA North', 'CA South','Washington')
AND CHANNEL_GROUP in('Supermarket')
AND BRAND IN ('Quellogs Wheat Flakes')
AND YEAR=1995)
```

```
group by
SYSDATE, 'All Wheat Flakes', AL2.MONTH, CHANNEL_GROUP, '10-States'
UNION
select
SYSDATE, '20 Oz Wheat Flakes' Product, AL2.MONTH,
sum(AL5.UNIT_SALES) Units, sum(AL5.DOLLAR_SALES) Dollars, count(*),
CHANNEL_GROUP CHNL, '10-States'
FROM PRODUCT AL4, SALES_FACT AL5,
CHANNEL AL1, DAILY_PERIOD AL2, MARKET AL3
WHERE (AL5.PRODUCT_ID=AL4.PRODUCT_ID
AND AL5.MARKET_ID=AL3.MARKET_ID
AND AL5.CHANNEL_ID=AL1.CHANNEL_ID
AND AL5.DAY=AL2.DAY)
AND (district in ('Connecticut',
'Delaware','Maine','Pennsylvania','New York',
'Oregon', 'Alaska', 'CA North', 'CA South','Washington')
AND CHANNEL_GROUP in('Supermarket')
AND PRODUCT='QLGS WHT FLK 20 OZ'
AND YEAR=1995)
group by
SYSDATE, '20 Oz Wheat Flakes', AL2.MONTH ,CHANNEL_GROUP, '10-States';
```