digital™

PeopleSoft Oracle TruCluster DIGITAL UNIX AlphaServer 4100

DIGITAL HiTest Notes

Part Number: EK–HPSUA–HN. A01

March 1997

Revision/Update Information: This is a new manual.

Version: Product Version 1.0

Digital Equipment Corporation
Maynard, Massachusetts

March 1997

Table of Contents

5 System Limits and Performance Data

6 Problems and Resolutions

A Detailed Hardware Configuration

Figures

Tables

Preface

This document provides an overview of DIGITAL HiTest Suites and detailed technical information about interoperability test results for the PeopleSoft Oracle TruCluster Production Server DIGITAL UNIX AlphaServer 4100 HiTest Suite.

Audience

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

Road Map

This document contains the following sections:

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 — Gives tables of configuration data about the hardware and software components that define the Template, and special configuration rules if any.

3. DIGITAL HiTest Suite Installation and Setup — Presents information useful when 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 Performance Data — Summarizes any system limitations or performance data that were identified during testing.

6. Problems and Solutions — Discusses any problems and solutions that were discovered during testing.

Appendix A: Detailed Hardware Configuration — Contains a more detailed treatment of the hardware and software components listed in the Configuration Data section.

Feedback and Ordering Information

What our readers think of this or any other DIGITAL documentation is important to us. If you have any comments, no matter how great or small, we'd appreciate hearing from you. Send your comments to: *reader-comments@digital.com*.

Please reference the document title and part number (EK-HPSUA-HN. A01) in your correspondence about this manual.

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 financials.

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 AppSet includes the software specific to one class of customer solutions.

Configuring a DIGITAL HiTest Suite is easy. Simply 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 section 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 section of the HiTest Notes, and in the HiTest Systems Web Pages, available through the following URLs:

```
http://cosmo.tay.dec.com              (Intranet)
http://www.partner.digital.com:9003 (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 base 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 PeopleSoft Oracle TruCluster DIGITAL UNIX AlphaServer 4100 HiTest Suite consists of the following software components:

- PeopleSoft Financial System (Benchmark kit)
  - Asset Management
  - General Ledger
  - Accounts Payable
  - Accounts Receivable
  - PSTools
- Oracle

This Suite will meet the enterprise financial server needs of medium to large organizations with more than 1000 employees and revenue of at least $100 million.

2
Configuration Data

This section describes the tested DIGITAL HiTest Configuration Suite, and includes the hardware, software, and firmware components that were tested together. Special configuration rules are explained if required.

Hardware and Software Components

Table 2-1 identifies the range of hardware and software components that can be configured using the PeopleSoft Oracle TruCluster Production Server DIGITAL UNIX AlphaServer 4100 HiTest Suit. The configuration ranges from a minimum of one AlphaServer 4100 base system to a maximum of three AlphaServer 4100 base systems.

The memory capacity of the AlphaServer 4100 Suite system is 4 GB.

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

NOTE

The benchmark version of PeopleSoft Online Financials 5.01 was used during the testing of this HiTest Suite.

Table 2-1: The HiTest Template

PeopleSoft *HiTest AppSet* Oracle, TruCluster, DIGITAL UNIX, AlphaServer 4100 HiTest Foundation				
For documentation and updates: http://cosmo.tay.dec.com *and* http://www.partner.digital.com:9003 *For hardcopy of this Suite's HiTest Notes, order* EK-HPSUA-HN.				
Line Item	**Description**	**Part Number**	**Tested Range**	
			Min	**Max**
AppSet Software				
1	**PeopleSoft Online Financials 5.01** *PeopleSoft contact: 1-888-773-8277*		1	1
Foundation Hardware				
2	*Select from 1 to 3 base systems:* **AlphaServer 4100 5/400 Drawer, DIGITAL UNIX** **AlphaServer 4000 5/400 Drawer, DIGITAL UNIX** *Hardware includes*: • 5/400-MHz CPU with 4-MB cache • 2-GB memory • PB2GA-JB TRIO64 1 MB Graphics • DE500-AA 10/100 Mbit Fast Ethernet • KZPDA-AA FW SCSI and cable • SCSI CD-ROM drive • RX23L-AB 1.44 MB Floppy drive • LK47W-A2 PS/2 style keyboard • Three-button PS/2 compatible mouse *Software includes*: • DIGITAL UNIX • DIGITAL EIP, consisting of the following services: – Management – Network Transport and Connectivity – File and Print – Internet – Database – Mail – Application Development – Presentation	**DA-51HAB-GB** **DA-53HEB-GB**	1	3
	Note: The maximum configuration is for three systems. When ordering two systems, the quantity is typically two-thirds of the maximum value, unless otherwise noted.			
3	*Select one of the following enclosures:* **Pedestal with StorageWorks shelf** **Cabinet with StorageWorks shelf** *Hardware includes*: • 4.3 GB Wide Disk (RZ29B-VW) • Pedestal or cabinet with one StorageWorks shelf and mounting kit. The Pedestal supports up to two more shelves; cabinet up to seven more.	**BA30P-AB/BB*** **H9A10-EL/EM***	1	3
4	*Add up to three additional CPUs to each 4100 system. Add up to one additional CPU to each 4000 systems.* **400 MHz CPU DIGITAL UNIX SMP UPG**	**KN304-BA**	3	9

* Where two part numbers or variants are separated by a "/", the first number applies to the Americas and Asia Pacific and the second number applies to Europe.

PeopleSoft *HiTest AppSet* Oracle, TruCluster, DIGITAL UNIX, AlphaServer 4100 HiTest Foundation				
For documentation and updates: http://cosmo.tay.dec.com *and* http://www.partner.digital.com:9003 *For hardcopy of this Suite's HiTest Notes, order* EK-HPSUA-HN.				
Line Item	**Description**	**Part Number**	**Tested Range**	
			Min	**Max**
	Note: The power supply included with the system drawer supports two CPUs, memory, and I/O adapters. Add a second supply to support four CPUs or for redundancy (N+1) with two CPU systems (AlphaServer 4000). Add a third supply for redundancy (N+1) with four CPUs (AlphaServer 4100).			
5	**450 Watt Power Supply**	**H7291-AA**	2	6
6	*Add a 2 GB memory option to each system.* **2 GB Memory Option**	**MS330-GA**	1	3
7	*There are 48 shared disks. Add four additional disks to each system for local storage:* **4.3 GB 7200 RPM 16-bit Wide SCSI Disks**	**RZ29B-VW**	52	60
8	**StorageWorks Subsystem** *Hardware includes*: • Cabinet assembly • Cable distribution unit • 18 Shelf assemblies • 3 controller shelf assemblies • Power supplies • SCSI cables (controller to shelf) • 8-bit personality modules	**SW822-GA/GB***	2	2
9	**PCI one-port FWD SCSI controller**	**KZPSA-BB**	4	12
10	*Order one for each KZPSA-BB ordered with the first system.* **10 meter 16-bit SCSI cable (internal)**	**BN21K-10**	4	4
11	*Order one for each KZPSA-BB ordered on the second and third systems.* **3 meter 16-bit SCSI cable (internal)**	**BN21K-03**	0	8
12	*Order one for each KZPSA-BB ordered.* **SCSI Y cable, 68-pin**	**BN21W-0B**	4	12
13	*Order two for each shared SCSI bus.* **SCSI Differential terminator**	**H879-AA**	8	8
14	**StorageWorks Array Controller with 32-MB cache**	**HSZ50-AF**	8	8
15	*Order one for each HSZ50-AF* **SCSI Tri-link connector**	**H885-AA**	8	8
16	*Order one for each HSZ50-AF pair.* **SCSI Cable**	**BN21L-0B**	4	4
17	**PCI 10/100-Mbit Fast Ethernet controller**	**DE500-AA**	1	3
18	**100 MB Memory Channel Hub**	**CCMHA-AA**	0	1
19	**PCI to Memory Channel controller**	**CCMAA-BA**	0	3
20	**Memory Channel cable**	**BC12N-10**	0	3
21	*Select one high resolution color monitor:* **15-in Flat-square with 0.28 dot pitch** **17-in Trinitron aperture grille, 0.26mm** **21-in Diamondtron aperture grille, 0.30mm**	**VRC15-WA** **VRTX7-WA** **VRC21-WA**	1	3

PeopleSoft *HiTest AppSet* Oracle, TruCluster, DIGITAL UNIX, AlphaServer 4100 HiTest Foundation				
For documentation and updates: http://cosmo.tay.dec.com *and* http://www.partner.digital.com:9003 *For hardcopy of this Suite's HiTest Notes, order* EK-HPSUA-HN.				
Line Item	**Description**	**Part Number**	**Tested Range**	
			Min	**Max**
Foundation Software				
NOTE: *Order the exact versions and revisions of the software shown below. Paper documentation can be ordered separately.*				
--	**UNIX for AlphaServer V 4.0B**	Included with item 2.		
22	**TruCluster Production Server for DIGITAL UNIX V1.4 license + doc**	**QB-3RLAG-AA**	0	3
23	**DIGITAL UNIX V4.0B Alpha CD-ROM**	**QA-MT4AA-H8**	1	1
24	**StorageWorks HSOF Software - UNIX**	**QB-5JCAB-SA**	4	4
25	**Micro Focus COBOL V4.0**		1	1
	Oracle7 Server for DIGITAL UNIX *Oracle contact: 1-800-ORACLE1 Email: infodec@us.oracle.com*			
26	**Oracle7 Server (RDBMS), Release 7.3.2.3.0**		1	1
27	**Oracle Server Manager, Release 2.3.2.0.0**		1	1
28	**Oracle Server Manager (Motif), Release 2.3.2.0.0**		1	1
29	**Oracle Common Libraries and Utilities, Release 7.3.2.3.0**		1	1
30	**Parallel Server Option, Release 7.3.2.3.0**		1	1
31	**Parallel Query Option, Release 7.3.2.3.0**		1	1
32	**PL/SQL, Release 2.3.2.3.0**		1	1
33	**SQL*Plus, Release 3.3.2.0.0**		1	1
34	**Remote Operations, Release 1.3.2.0.0**		1	1
35	**Toolkit 2.1 Base, Release 2.1.4.14.1**		1	1
36	**Toolkit 2.1 Extension, Release 2.1.4.14.1**		1	1
37	**SLAX Parser, Release 7.3.2.1.0**		1	1
38	**Oracle Trace, Release 4.0.0**		1	1
39	**Precomp, Release 7.3.2.1.0**		1	1
40	**Pro*C, Release 2.2.2.0.0**		1	1
41	**Oracle WebServer, Release 1.0.2.0.0**		1	1
42	**Multimedia API's, Release 2.0.5.4.0**		1	1
43	**Oracle Help, Release 2.1.1.0.0**		1	1
44	**Oracle7 Server / Digital UNIX V4.0 compatibility Patch # 424307**		1	1
45	**Oracle Networking**			
46	**SQL*Net (V2), Release 2.3.2.1.0**		1	1
47	**Oracle Protocol Adapter for TCP/IP, Release 2.3.2.1.0**		1	1

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

Table 2-2: Component Revision Levels

Hardware Component	Hardware	Firmware	Software
SRM console	--	V3.0-10	--
SCSI host adapter (KZPDA-AA)			--
FWD SCSI controller (KZPSA-BB)	P01	A10	--
Array controller (HSZ40-CF)	A01	HSOF V30Z-2	--
4.3 GB disks (RZ29B-VW)		0016	--
400-MHz CPU (KN304-BA)	B04	--	--
Memory (MS330-FA)	A03	--	--
Fast Ethernet adapter (DE500-AA)			--
Software Component	**Version/ Revision**	**Patch Level**	
DIGITAL UNIX	4.0B	Rev564 40BAS00001	
TruCluster for DIGITAL UNIX	1.4		
DIGITAL UNIX Driver for Memory Channel	1.4		
EISA System Config Kit	V1.9		
PeopleSoft Financials	5.01		
Oracle7	7.3.2.3	Upgrade from 7.3.2.1 and added Oracle DIGITAL UNIX V4.0 patch 424307	
SQL*Net	2.3.2.1.0		
Oracle Pro ‘C’	2.2.2.0.0		
Micro Focus COBOL	4.0		

3
DIGITAL HiTest 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.

Preparing the System

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

There are specific hardware rules and restrictions that govern TruCluster Production Server configurations:

- Set the bus_probe_algorithm variable to new.
  ```
  >>> set bus_probe_algorithm new
  >>> init
  ```
- Set the boot_reset variable to on.
  ```
  >>> set boot_reset on
  ```
- Ensure the KZPSA adapters have different SCSI ID numbers (refer to Table 3-1). The four controllers in the first 4100 were all set to SCSI ID 7. For the maximum configuration Template, the controllers in the second 4100 were set to SCSI ID 6 and the controllers in the third 4100 were set to SCSI ID 5.
- Enable fast SCSI bus speed for each KZPSA adapter (refer to Table 3-1).
- Connect each KZPSA adapter to a shared SCSI bus with a “Y” cable.
- Internal KZPSA SCSI termination was disabled. The internal termination resistors (Z1, Z2, Z3, Z4, and Z5) on each KZPSA-BB were removed.
- Memory Channel PCI adapters were jumpered for standard mode.
- The Memory Channel link cables were screwed in tight to ensure proper grounding.
- Memory Channel Hub power is applied before the member systems are booted.

Table 3-1: SCSI Controller Configuration

Step	Action	Result

1	`>>> show pk*`	`pka0_fast 1`
		`pka0_host_id 7`
		`pka0_termpwr 1`
		`pkb0_fast 1`
		`pkb0_host_id 7`
		`pkb0_termpwr 1`
		`pkc0_fast 1`
		`pkc0_host_id 7`
		`pkc0_termpwr 1`
		`pkd0_fast 1`
		`pkd0_host_id 7`
		`pkd0_termpwr 1`
		`pke0_fast 1`
		`pke0_host_id 7`
		`pke0_termpwr 1`
		If any values need to be changed, refer to the next steps.

2	`>>> set pkb0_host_id 6`
	`>>> set pkc0_host_id 6`
	`>>> set pkd0_host_id 6`
	`>>> set pke0_host_id 6`
		Changes the SCSI ID of the KZPSA-BB controllers from 7 to 6. This was done on the second AlphaServer 4100 system.

3	`>>> set pkb0_host_id 5`
	`>>> set pkc0_host_id 5`
	`>>> set pkd0_host_id 5`
	`>>> set pke0_host_id 5`
		Changes the SCSI ID of the KZPSA-BB controllers from 7 to 5. This was done on the third AlphaServer 4100 system.

4	`>>> set pka0_fast 1`
	`>>> set pkb0_fast 1`
	`>>> set pkc0_fast 1`
	`>>> set pkd0_fast 1`
	`>>> set pke0_fast 1`
		Fast SCSI mode is enabled when the value is set to 1 (0 = slow mode). Set this value for the controllers on all three systems.

Disk Storage Configuration

The StorageWorks HSZ50 Array Controller Utility for DIGITAL UNIX was used to configure the HSZ50-AF controllers and disks as follows:

1. The disks attached to the HSZ50 controllers were configured as RAID-5 sets. There are 16 RAIDsets (each RAIDset consists of four 4.3 GB disks).
2. The chunk size for the Oracle7 database was computed and set to 16 (based on a 8K db_block_size and having 1 db_blocks_per_chunk).

$$\frac{(db\_block\_size)*(\#\_of\_db\_blocks\_per\_chunk)}{512} = chunk\_size$$

$$\frac{(8192)*(1)}{512} = 16 \text{ disk blocks}$$

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

The DIGITAL UNIX 4.0B operating system was installed with all kernel options. All DIGITAL UNIX subsets and the TruCluster software were installed for TruCluster Production Server operation.

Swap Space

The swap mode was set to *immediate*. Two swap spaces were created for each system:

- Primary on rz40g
- Secondary on rz44c

UNIX Kernel Parameters (/etc/sysconfigtab)

Table 3-2 lists the parameter settings used for the UNIX kernel. The settings are based on a 4 GB memory configuration, suggested in *Oracle7 Server for Digital UNIX Installation and Configuration Guide*.

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

Table 3-2: Kernel Parameters

Parameter	Value
vm parameters:	
vm-maxvas	4292967296
vm-mapentries	400
ubc-minpercent	1
ubc-maxpercent	2
new-wire-method	1
rt parameters:	
aio-max-num	1024
aio-max-percent	2
proc:	
max-proc-per-user	1024
max-threads-per-user	1024
max-per-proc-data-size	4292967296
max-per-proc-address-space	4292967296
ipc parameters:	
shm-max	2139095040 *** 8 MB aligned address
shm-seg	32
msg-max	8192
msg-mnb	16384
msg-mni	1024
msg-tql	4096
sem-aem	16384
sem-mni	200
num-of-sems	200
sem-msl	200
sem-opm	200
sem-ume	200
sem-vmx	32767

TruCluster Production Server

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

1. The TruCluster Production Server Software Version 1.4 software license (TCR-UA) was registered.
2. The Associated Products Volume 2 CD-ROM was loaded and mounted to `/mnt`.
3. The `setld -l /mnt/TCR140` command was used to load the TruCluster Production Server Software subsets. The installation procedure starts after the subsets were loaded.
4. All mandatory and optional subsets were installed.
5. The IP name and address for the cluster interconnect (Memory Channel) was entered.
6. The system is in ASE 0 and the ASE logger is enabled.
7. The kernel was automatically rebuilt.
8. The shared SCSI buses are identified. There are four in this Template.
9. The new kernel was moved to the root file system.
10. The systems were rebooted and `asemgr` was used to add the ASE members.

Distributed Raw Disk Service

The shared storage on which the common database resides is provided by the Distributed Raw Device (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 expected growth with expansion for the future. The `asemgr` facility is provided to assist in the assignment of physical storage (device/partition level) to DRD entity and node location of a DRD services.

Consider the following when setting up the DRD service:

- Use RAID controllers. This Suite had the disks set up as RAID-5 sets at the hardware level using the HSZ50 array controllers. RAID-5 provides disk striping with distributed parity, but provides only the capacity of *n*-1 devices, where *n* = Number of RAID set members. For example, four 4.3 GB drives have a 12.9 GB RAID-5 capacity.
- Plan to place data that is accessed by different instances on different DRD services. This reduces memory channel traffic and also reduces Oracle PCM lock requirements.
- Use a symbol link to assign meaningful names to all the DRD services. This improves manageability.

Distributed Raw Disk (DRD) services were set up for three cluster members (depot4, depot5, and depot6) using `asemgr`. Table 3-3 lists the full DRD service configuration. The example following the table shows how the first DRD service was created.

Table 3-3: DRD Service Configuration

DRD Service Name	ASE Member	DRD Device Special File	Underlying Storage
DRD1	mcdepot6	/dev/rdrd/drd110001 /dev/rdrd/drd110002	/dev/rrz8b /dev/rrz8c
REDO1	mcdepot4	/dev/rdrd/drd110003 /dev/rdrd/drd110004 /dev/rdrd/drd110005	/dev/rrz19b /dev/rrz19c /dev/rrz19d
APT1	mcdepot5	/dev/rdrd/drd110006	/dev/rrz16c
API1	mcdepot5	/dev/rdrd/drd110007 /dev/rdrd/drd110016	/dev/rrz25c /dev/rrz25a
ART1	mcdepot6	/dev/rdrd/drd110008 /dev/rdrd/drd110017	/dev/rrz33c /dev/rrz33a
ARI1	mcdepot6	/dev/rdrd/drd110009	/dev/rrz9c
AMT1	mcdepot4	/dev/rdrd/drd110010 /dev/rdrd/drd110015	/dev/rrz17c /dev/rrz17a
AMI1	mcdepot4	/dev/rdrd/drd110011	/dev/rrz26c
GLT1	mcdepot4	/dev/rdrd/drd110012	/dev/rrz34c
GLI1	mcdepot4	/dev/rdrd/drd110013	/dev/rrz10c
PSINDEX1	mcdepot5	/dev/rdrd/drd110014	/dev/rrz18c
RBS1	mcdepot4	/dev/rdrd/drd110018	/dev/rrz11c
RBS2	mcdepot5	/dev/rdrd/drd110019	/dev/rrz32c
RBS3	mcdepot6	/dev/rdrd/drd110020	/dev/rrz24c
REDO2	mcdepot5	/dev/rdrd/drd110021 /dev/rdrd/drd110022 /dev/rdrd/drd110023	/dev/rrz27b /dev/rrz27c /dev/rrz27d
REDO3	mcdepot6	/dev/rdrd/drd110024 /dev/rdrd/drd110025 /dev/rdrd/drd110026	/dev/rrz35b /dev/rrz35c /dev/rrz35d

The following example goes through the process of creating the first DRD service:

```
# asemgr

TruCluster Production Server (ASE)

    ASE Main Menu
    a)  Managing the ASE         -->
    m)  Managing ASE Services    -->
    s)  Obtaining ASE Status     -->

    x)  Exit                              ?)  Help
Enter your choice: m

    Managing ASE Services
    c)  Service Configuration    -->
    r)  Relocate a service
   on)  Set a service on line
  off)  Set a service off line
  res)  Restart a service
    s)  Display the status of a service
    a)  Advanced Utilities       -->

Exit to the Main Menu            ?)  Help

Enter your choice [x]: c
```

```
    Service Configuration
    a)  Add a new service
    m)  Modify a service
    d)  Delete a service
    s)  Display the status of a service

    x)  Exit to Managing ASE Services    ?)  Help

Enter your choice [x]: a

    Adding a service
Select the type of service:

    1)  NFS service
    2)  Disk service
    3)  User-defined service
    4)  DRD service

    x)  Exit to Service Configuration    ?)  Help

Enter your choice [1]: 4

You are now adding a new DRD disk service to your ASE.

A DRD disk service is comprised of any number of DRDs which can be
created from a single raw disk partition or LSM volume which will be
accessible from all members in the cluster.


Note: If using a raw disk partition please be sure that the character
      device special file exists on all members which are in this ASE.

                    DRD Service Name

The name of a DRD disk service must be a unique service name within this ASE.


Enter the DRD disk service name ('q' to quit): drd1


You will now be prompted to enter a list of devices comprising
the DRD service, enter Return when you have completed the list.


Enter an existing character device special file for one of the following:

    a physical device (ie /dev/rrz1c)
    a LSM volume (ie /dev/rvol/dg/vol01)
    To end the list, press the Return key at the prompt.


Enter character device special file: /dev/rrz8b


Enter an existing character device special file for one of the following:


    a physical device (ie /dev/rrz1c)
    a LSM volume (ie /dev/rvol/dg/vol01)
    To end the list, press the Return key at the prompt.

Enter character device special file: /dev/rrz8c

Enter an existing character device special file for one of the following:

    a physical device (ie /dev/rrz1c)
    a LSM volume (ie /dev/rvol/dg/vol01)
    To end the list, press the Return key at the prompt.
```

```
Enter character device special file:


DRD Device Special File:    /dev/rdrd/drd110001
Underlying Storage:         /dev/rrz8b

DRD Device Special File:    /dev/rdrd/drd110002
Underlying Storage:         /dev/rrz8c


NOTE: In order to access the DRD devices in this service from cluster
      members outside of this ASE execute the following on each cluster
      node which is not a member of this ASE:
                drd_mknod -f drd110001
                drd_mknod -f drd110002

    Selecting an Automatic Service Placement (ASP) Policy

Select the policy you want ASE to use when choosing a member
to run this service:

    b)  Balanced Service Distribution
    f)  Favor Members
    r)  Restrict to Favored Members


    x)  Exit to Service Configuration    ?)  Help

Enter your choice [b]:

    Selecting an Automatic Service Placement (ASP) Policy

Do you want ASE to consider relocating this service to another member
if one becomes available while this service is running (y/n/?): y

Enter 'y' to add Service 'drd1' (y/n): y
Adding service...
Starting service...
Service drd1 successfully added...

    Service Configuration

    a)  Add a new service
    m)  Modify a service
    d)  Delete a service
    s)  Display the status of a service

    x)  Exit to Managing ASE Services    ?)  Help

Enter your choice [x]: s


    Service Status

Select the service whose status you want to display:

    1)  drd1 on mcdepot4

    x)  Exit to previous menu            ?)  Help


Enter your choice [x]: 1

    Status for DRD service `drd1`


Status:             Relocate:  Placement Policy:       Favored Member(s):
on mcdepot4         yes        Balance Services        None
```

```
    Storage configuration for DRD service `drd1`

DRD Device Special File:    /dev/rdrd/drd110001
Underlying Storage:         /dev/rrz8b

DRD Device Special File:    /dev/rdrd/drd110002
Underlying Storage:         /dev/rrz8c


NOTE: In order to access the DRD devices in this service from cluster
      members outside of this ASE execute the following on each cluster
      node which is not a member of this ASE:
                drd_mknod -f drd110001
                drd_mknod -f drd110002

Press 'Return' to continue:
    Service Status

Select the service whose status you want to display:
    1)  drd1 on mcdepot4

    x)  Exit to previous menu           ?)  Help

Enter your choice [x]: x

    Service Configuration
    a)  Add a new service
    m)  Modify a service
    d)  Delete a service
    s)  Display the status of a service

    x)  Exit to Managing ASE Services   ?)  Help


Enter your choice [x]:

    Managing ASE Services
    c)  Service Configuration    -->
    r)  Relocate a service
   on)  Set a service on line
  off)  Set a service off line
  res)  Restart a service
    s)  Display the status of a service
    a)  Advanced Utilities       -->
    x)  Exit to the Main Menu           ?)  Help

Enter your choice [x]:

    TruCluster Production Server (ASE)

    ASE Main Menu
    a)  Managing the ASE         -->
    m)  Managing ASE Services    -->
    s)  Obtaining ASE Status     -->

    x)  Exit                            ?)  Help

Enter your choice: x


# exit
#
```

Applications

Oracle

Oracle software and DIGITAL UNIX are installed on separate disk to eliminate I/O contention. Oracle7 Server, Version 7.3.2.1, the Parallel Query Option, PL/SQL, SQL*Plus, and SQL*Net were installed using Oracle Installer from the Oracle7 Server CD.

The following Oracle upgrade and patches were installed:

- Oracle 7 Server Version 7.3.2.3
- Oracle DIGITAL UNIX V 4.0 patch 424307

Oracle Parallel Server Considerations

The implementation of a shared database utilizing the Oracle Parallel Server (OPS) on a DIGITAL UNIX TruCluster requires some special considerations to enable the proper and efficient use of the application. This section is designed to make the reader aware of these considerations and provide an overview of how they were utilized.

OPS uses shared memory to hold lock information in the memory image log and to cache database blocks. OPS has the lock information in the distributed lock manager (DLM). This allows multiple member nodes to see the current lock state.

DRD ownership is critical to OPS. Oracle users on each node must own the DRDs.

Rollback segments

The creation of one tablespace per instance (node) is not a requirement, but was used here to allow each instance private rollbacks segments that would produce only local DRD access. This slightly more optimal IO 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, rollbacks segments are created in the tablespace and assigned to the instance using Oracle initialization parameter ROLLBACK_SEGMENTS (in INIT.ORA), 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, three control files were distributed across three separate DRDs, that are served by different cluster nodes and utilize three unique storage controllers.

Oracle Initialization Parameters

Implementation of an Oracle Parallel Service requires some parameters be identical across all instances and this is desired behavior for some other parameters. 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 include in the instance specific parameter file, through use of the oracle IFILE parameter. In general the majority of our initialization parameters were consistent across the instances, but a few such as ROLLBACK_SEGMENTS, THREAD, and INSTANCE_NUMBER were uniquely defined in the instance specific parameter file.

Table 3-4 shows the optimizations made to the Oracle7 server and database for PeopleSoft Online Financials.

Table 3-4: Oracle Parameter Optimization

Parameter	Value	Comment
db_block_size	8192	
_init_sql_file	(?/dbs/sql.bsq)	Contains storage parameters specifically for 32K block size. Although we set this parameter, it is no longer required as of Oracle7 Version 7.3.
db_block_buffers	16000	Approximately 125 MB (131,072,000 bytes) of buffer cache based on the db_block_size of 8192. This number should be maximized to provide the highest cache hit ratio without adversely affecting the memory requirements of other Oracle and system processes.
async_write	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	32768	Memory area allocated to each process/thread performing sorts. This parameter should be maximized during intensive sort operations, such as index creation on very large tables. However, this is highly sensitive to the parallel degree of the table(s) being sorted, as each thread will allocate this amount of space.
sort_direct_writes	auto	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	32768	The correct default value for this parameter is documented in the 7.3.2 README file. 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	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. This parameter is set sufficiently large enough to accommodate smaller dimension tables.
ccf_io_size	4194304	Determines the number of bytes per write when creating a contiguous file.
db_file_multiblock_ read_count	16	Due to an Oracle limit of 128 KB per I/O, this parameter was set to 16, which is the maximum based on the formula max_io_size/db_block_size (128KB/8KB).
parallel_max_servers	5	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 2 threads will be created for each degree, therefore this parameter should be set to twice the degree for full benefit on index creation.

Parameter	Value	Comment
shared_pool_size	419430400	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. Otherwise, if cache misses are prevalent, then more memory should be allocated and the open_cursors parameter may need to be increased.
log_checkpoint_interval	100000000	Set to sufficiently large number to disable time based checkpoints forcing checkpoints only to occur on log switches.
gc_db_locks	191021	This is the total number of Parallel Cache Management (PCM) locks covering the datablocks in the SGAs of a Parallel Server.
gc_rollback_segments	41	The maximum number of rollback segments system wide. This parameter is set to the total number of rollback segments acquired by all instances in a parallel server.

Enabling Process Limits for the Oracle DBA User Account

The Oracle DBA User Account Process Settings were edited to enable use of larger parameter process values that are required for the Oracle 64-bit Option, according to *Oracle7 Server for Digital UNIX Installation and Configuration Guide*, "Appendix C: 64-bit Option". An Oracle DBA user account, using the UNIX C Shell, should have these settings in their .login file.

.login file commands:

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

Tablespace Configurations

The tablespaces for Asset Management (AM), Accounts Payable (AP), Accounts Receivable (AR), General Ledger (GL), temporary segments, and indices are created on different DRD services. The partitioning of user data and indices is based upon business function and distribution of the workload across all cluster members.

Separating table and indices helps performance by allowing concurrent reads of indices and their associated table data while minimizing I/O contention. Placing the indices in their own tablespaces allows the assignment of PCM locks specific to data files in that tablespace. This helps to reduce possible block pinging.

Table 3-5 lists the tablespace name, symbolic link name and size of the Oracle database files.

Table 3-5: Tablespace Configurations

Tablespace	Datafile Name (link)	DRD Number	Size (MB)
SYSTEM	/oracle/links/SYSTEM	1	110
PSRBS	/oracle/links/RBS_1	18	1000
PSRBS2	/oracle/links/RBS_2	19	1000
PSRBS3	/oracle/links/RBS_3	20	1000
PSTEMP	/oracle/links/TEMP1	2	4000
GLT1	/oracle/links/GLT1	12	1996
AMT1	/oracle/links/AMT1	10	3000
ART1	/oracle/links/ART1	8	1996
APT1	/oracle/links/APT1	6	1996
GLI1	/oracle/links/GLI1	13	3000
AMI1	/oracle/links/AMI1	11	1996
ARI1	/oracle/links/ARI1	9	3500
API1	/oracle/links/API1	7	1996
PSINDEX1	/oracle/links/PSINDEX1	14	2500
		Total Tablespace:	29,090

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 instances redo log to the database to provide consistency across the shared database. Hence, the placement of the redo logs for all three 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 in using local DRDs when configuring 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 two additional instances is 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;
alter database add logfile thread 3
          '/oracle/links/LOGA_3' size 50M reuse,
          '/oracle/links/LOGB_3' size 50M reuse,
          '/oracle/links/LOGC_3' size 50M reuse;
alter database enable public thread 3;
```

The redo logs are listed in Table 3-6.

Table 3-6: Redo Logs

File name (link)	Size (MB)	DRD Number
/oracle/links/LOGA_1	50	DRD# 3
/oracle/links/LOGB_1	50	DRD# 4
/oracle/links/LOGC_1	50	DRD# 5
/oracle/links/LOGA_2	50	DRD# 21
/oracle/links/LOGB_2	50	DRD# 22
/oracle/links/LOGC_2	50	DRD# 23
/oracle/links/LOGA_3	50	DRD# 24
/oracle/links/LOGB_3	50	DRD# 25
/oracle/links/LOGC_3	50	DRD# 26

Create Database Script

The script below was used to create the database

```
REM * Set terminal output and command echoing on; log output of this script.
REM *

set termout on
set echo on
spool bld01_ps01.lis
connect internal
startup nomount pfile=$ORACLE_HOME/dbs/initPSF1_0.ora

create database PSFT
maxdatafiles  1000
maxinstances  3
maxlogfiles   40
maxlogmembers 4
character set "US7ASCII"
datafile
        '/oracle/links/SYSTEM'  size   110m
logfile
        '/oracle/links/LOGA_1'                  size   50m,
        '/oracle/links/LOGB_1'                  size   50m,
        '/oracle/links/LOGC_1'                  size   50m;

REM - Creates data dictionary views.

@$ORACLE_HOME/rdbms/admin/catalog.sql

REM - Scripts for procedural option

@$ORACLE_HOME/rdbms/admin/catproc.sql

REM - Grant public access to all views used by the char-mode SQLDBA.

@$ORACLE_HOME/rdbms/admin/utlmontr.sql

spool create2
set echo on

REM * Create additional rollback segment in SYSTEM before creating tablespace.
REM *
REM create rollback segment r00 tablespace system
REM storage (initial 16k next 16k minextents 2 maxextents 20);
REM * Use ALTER ROLLBACK SEGMENT ONLINE to put r00 online without shutting
REM * down and restarting the database.
REM *

rem elter rollback segment r00 online;
```

```
REM * Create a tablespace for rollback segments.
REM * Rollback segment configuration guidelines:
REM *   1 rollback segments for every 4 concurrent xactions.
REM *   No more than 50 rollback segments.
REM *   All rollback segments the same size.
REM *   Between 2 and 4 homogeneously-sized extents per rollback segment.
REM * Attempt to keep rollback segments to 4 extents.
REM *
create tablespace PSRBS datafile
        '/oracle/links/RBS_1'    size   1000M
default storage (
        initial          1M
        next             1M
        pctincrease      0
);
create tablespace PSRBS2 datafile
        '/oracle/links/RBS_2'    size   1000M
default storage (
        initial          1M
        next             1M
        pctincrease      0
);
create tablespace PSRBS3 datafile
        '/oracle/links/RBS_3'    size   1000M
default storage (
        initial          1M
        next             1M
        pctincrease      0
);

REM *
create tablespace PSTEMP datafile
        '/oracle/links/TEMP1'    size   4000M
default storage (
        initial          1M
        next             1M
        pctincrease      0
);
REM *
```

Create Tablespace Script

The script below was used to create the tablespaces.

```
REM * Set terminal output and command echoing on; log output of this script.
REM *

set termout on
set echo on
spool $ORACLE_SID\2c.lst

REM * The database should already be started up at this point with:
REM * pfile=/oracle/admin/in501/pfile/initin501.ora

connect internal
set echo off

create tablespace glt datafile
        '/oracle/links/GLT1'    size   1996m
default storage (
        initial      10m
        next         10m
        pctincrease  1
);

create tablespace amt datafile
        '/oracle/links/AMT1'    size   3000M
default storage (
        initial      10m
        next         10m
```

```
        pctincrease  1
);
create tablespace art datafile
        '/oracle/links/ART1'    size   1996m
default storage (
        initial      10m
        next         10m
        pctincrease  1
);
create tablespace apt datafile
        '/oracle/links/APT1'    size   1996m
default storage (
        initial      10m
        next         10m
        pctincrease  1
);
create tablespace gli datafile
        '/oracle/links/GLI1'    size   3000M
default storage (
        initial      10m
        next         10m
        pctincrease  1
);
create tablespace ami datafile
        '/oracle/links/AMI1'    size   1996m
default storage (
        initial      10m
        next         10m
        pctincrease  1
);
create tablespace ari datafile
        '/oracle/links/ARI1'    size   3500m
default storage (
        initial      10m
        next         10m
        pctincrease  1
);
create tablespace api datafile
        '/oracle/links/API1'    size   1996m
default storage (
        initial      10m
        next         10m
        pctincrease  1
);
create tablespace psindex datafile
        '/oracle/links/PSINDEX1'    size   2500m
default storage (
        initial      10m
        next         10m
        pctincrease  1
);
```

Oracle Data Import

The data used as the starting point for this testing was imported into the database using the PeopleSoft Financials 5.01 Benchmark utility. The selection of this option from the menu uncompresses a tar file and imports the data into the existing database. This compressed import file, provided with the benchmark kit, is 994 MB and required just under 6 hours to completely process. The steps we used to perform the import are as follows:

```
$ cd $PS_HOME
$ ./psdbsetup
PeopleSoft Financials 5.01 Benchmark Kit utility
       Oracle Database Server machine Setup

     Server: depot4      SID: PSF1      User: oracle
$PS_HOME: /fs501/fs501           PS Process Scheduler:   Down
                                 Oracle Job Schedulers:  1
Menu of choices:

  1) Install DBMS Server Content into /fs501/fs501
  2) Compile DBMS Server COBOL
  3) Build Database
  4) Refresh Database
  5) Start Process Scheduler
  6) Monitor Currently Executing Benchmark
  7) View Benchmark Timings
  9) Exit

Please enter your choice ==> 3
Do you wish to proceed with building the database?
All tables will be truncated, dropped, and imported (y/n): y
Coalescing Free Space ...
Creating users ...
Enter password for user SYS: <password><cr>
```

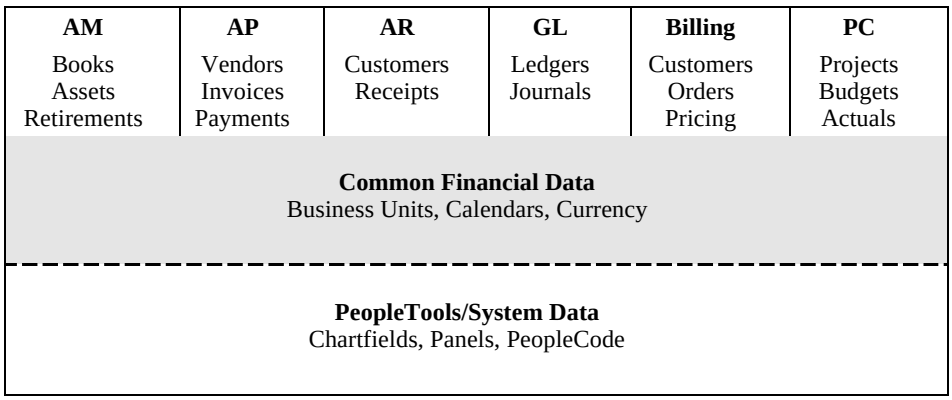
After the database build completed (about 6 hours with this HiTest System), the log showed that some additional operations, outside of the database import, were performed. If the database is imported without using the PeopleSoft Financials 5.01 Benchmark Kit utility, these additional operations would have to be performed manually.

PeopleSoft Financials System Configuration

The Financial System Configuration used for this HiTest Template consists of Common Structure, Asset Management (AM), Accounts Payable (AP), Accounts Receivable (AR), and General Ledger (GL).

Figure 3-1 shows a logical view of a PeopleSoft Financial database.

Figure 3-1: Logical View of a PeopleSoft Financial Database



Common Structure

The following common structure components are defined:

Business Units	Three business units are utilized for the majority of the Financial Systems tests. **BU** **Usage** DEC Small volume, integrated testing BEA Business unit setup and overall control testing Z002 Feature testing associated with volumes. Other business units exist in the database as examples, test areas, and to help simulate total database volumes.
Tableset	DEC shared control data established in the "MFG" business unit.
Calendar	Model 01, standard monthly calendar.
Chartfield Structure	Standard PS delivered structure, Account, Deptid, Product, Project (affiliate, currency code, statistics code).

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 section describes the test environment, including:

- Tools and scripts used to perform the tests
- Test configuration
- Test process
- Test results

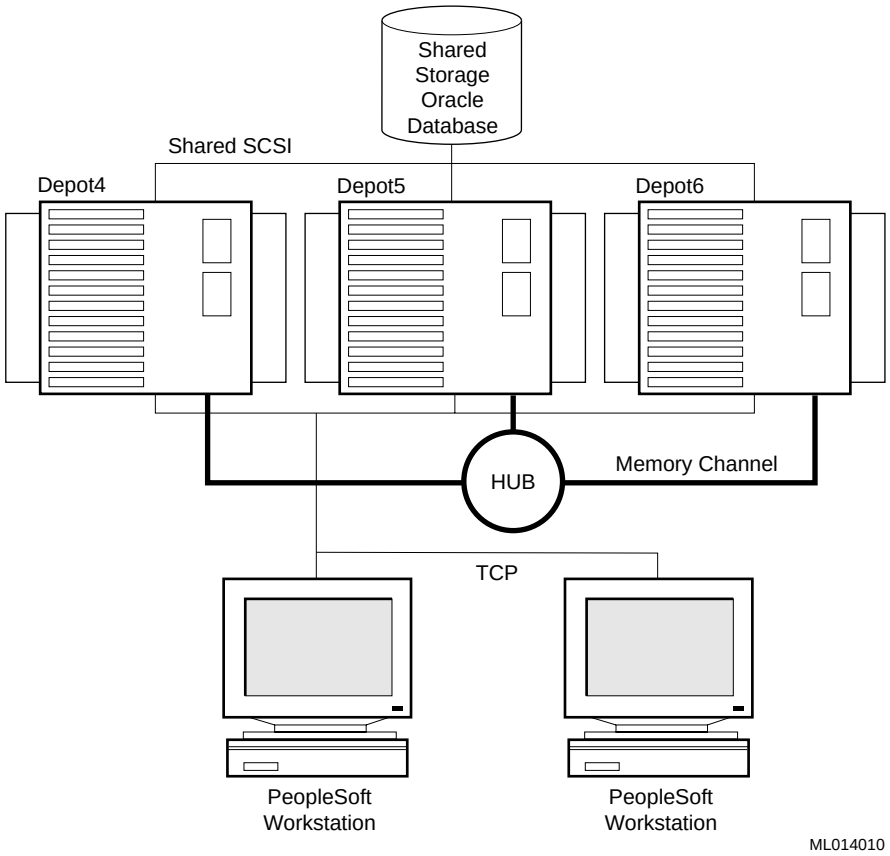
Overview of Results

Interoperability testing successfully was performed on the PeopleSoft Oracle TruCluster DIGITAL UNIX AlphaServer 4100 HiTest Suite. Within the PeopleSoft Financials, the Asset Management, Payable, Receivable, and General Ledger modules were tested. The tests verified that the business processes functioned as expected.

Test Environment

Figure 4-1 shows the PeopleSoft Financials Database test environment.

Figure 4-1: Test Environment



Test Tools

The tools used for interoperability testing were two PeopleSoft workstations - PCs running Windows 95 with SQA Robot version 5, SQL*Net version 2.2, Microsoft Word Version 6.0, and Microsoft Excel version 5.0.

Test Configuration

The test load was generated from database creation, benchmark kit load, and client system input (Windows 95 PC).

Minimum Configuration

The minimum configuration includes one AlphaServer 4100 system with 21.5 GB of local storage and 4 GB of memory. The system is connected to database storage (16 RAID-5 sets with four 4.3-GB disks in each set).

Maximum Configuration

The maximum configuration includes three AlphaServer 4100 systems, each with 21.5 GB of local storage and 4 GB of memory. All three systems are connected to the same amount of database storage (shared) as the minimum configuration.

Test Process and Results

Interoperability testing of the PeopleSoft Financials with Oracle, TruCluster, DIGITAL UNIX, and AlphaServer 4100 systems was divided into the following sections:

- PeopleSoft Asset Management (AM)
- PeopleSoft Payable (AP)
- PeopleSoft Receivable (AR)
- PeopleSoft General Ledge (GL)
- PeopleSoft Billing (Future)
- PeopleSoft Project Costing (PC) (Future)

NOTE

For all the above modules, online functionality was tested with limited batch processing.

PeopleSoft Asset Management (AM)

The purpose of the PeopleSoft Asset Management test was to verify the business processes function as expected in the HiTest Template. The business processes checked, include: business definitions, tracking and reporting structures (corporate, tax), depreciation processing, asset transactions, closing, and reporting. The testing does not include chartfield customization. Table 4-1 lists the Process, Function, Task, and test engineer comments.

Table 4-1: Asset Management Business Process Testing

Process	Function	Task	Comments
Business rules definition	Define general options	Create Detail Calendar Define installation options Define Books Define Operator Preferences	Expanded Model 01 calendar N/A Corporate and federal tax Updated to DEC business unit
Asset Management definition	Define Business Units	Add a Business Unit Specify processing options Assign a SetID	DEC Process batch later DEC
	Define AM Controls	Category codes Class codes GI interface definition Profiles	Used Model Used Model “MFG” structure Used Model
Asset Maintenance	Asset adds	Online express add Online basic add Copy asset	To DEC, Z001 N/A
	Transfers	Department transfer Full transfer	Successfully completed Successfully completed
	Retirements	Full retirement	Successfully completed
Depreciation processing	Technique	Remaining Value Life to date	Successfully completed
	Methods	DB200/STL Table method DB200/STL Table method	Successfully completed
AM close processing (AMAEDIST)	Accounting entries	Adds Transfers Retirements	Successfully completed Successfully completed Batch process error
(AMCLOSE)	Depreciation expense	Current period Prior period	Process run successfully Process run successfully
SQR reporting	Net book value reports	Corporate Book Tax Book Current period acquisitions	Reports generated successfully Reports generated successfully Parameter error (see reports under “AM Reports” session)
Query	Create queries	Department acquisitions Retirements	Queries created successfully Queries created successfully
Query Reports	Create query reports	Run query to Excel	Excel query reports created successfully

PeopleSoft Asset Management Detail Test Results

Business Units: DEC & Z001

Simulated normal monthly "Asset Maintenance Process" was executed successfully.

The following transactional and functional tests were performed in the PeopleSoft Asset Management (AM) System (refer to Table 4-2).

- Asset Additions:
  1. Added over 21 historical assets using Express entry function. (Asset No. H0000001 - H0000021 - asset numbers were manually assigned)
  2. Added over 50 assets for the current period (Period 4) (Asset No. 00000044 - 00000093 - asset numbers were automatically assigned by the system)
- Asset Transfers: (Department level asset transfers)
  1. Transferred 5 historical assets from Information Services department to Administration Department. (Asset No. H0000001 - H0000005)
  2. Transferred 5 historical assets from Finance Department to Administration Department (Asset No. H0000010 - H00000014)
- Asset Retirements:
  1. Retired 5 historical assets Asset No. H0000010 - H0000015)

Table 4-2: Asset Add, Transfer, and Retirement

Bus Unit	Book Name	Trans Type	In/Out	Count Asset ID	Sum Total Cost
DEC	CORP	ADD		53	2,676,000.00
DEC	CORP	RET		5	-41,500.00
DEC	CORP	TRF	I	10	339,200.00
DEC	CORP	TRF	O	10	-339,200.00
DEC	FEDERAL	ADD		53	2,676,000.00
DEC	FEDERAL	RET		5	-41,500.00
DEC	FEDERAL	TRF	I	10	339,200.00
DEC	FEDERAL	TRF	O	10	-339,200.00

Depreciation (AMDPCALC)

Depreciation calculations for both historical and current assets were run successfully.

Z002 - simulate conversion load

Asset base volume - Over ½ million assets were loaded, 5000 of which were depreciated. Refer to Table 4-3 and Table 4-4 for statistical information.

Table 4-3: Asset Processing - Conversion Load

Unit	Book Name	Count Asset ID	Sum Total Cost
Z001	AMT	50,003	850,017,000.00
Z001	CORP	50,004	850,020,000.00
Z001	FEDERAL	50,004	850,020,000.00
Z002	AMT	50,003	850,017,000.00
Z002	CORP	50,011	850,031,110.00
Z002	FEDERAL	50,011	850,031,110.00
Z003	AMT	50,001	850,017,000.00
Z003	CORP	50,001	850,017,000.00
Z003	FEDERAL	50,001	850,017,000.00
Z004	AMT	50,001	850,017,000.00
Z004	CORP	50,001	850,017,000.00
Z004	FEDERAL	50,001	850,017,000.00
Z005	AMT	50,001	850,017,000.00
Z005	CORP	50,001	850,017,000.00
Z005	FEDERAL	50,001	850,017,000.00
Z006	AMT	50,001	850,017,000.00
Z006	CORP	50,001	850,017,000.00
Z006	FEDERAL	50,001	850,017,000.00
Z007	AMT	50,001	850,017,000.00
Z007	CORP	50,001	850,017,000.00
Z007	FEDERAL	50,001	850,017,000.00
Z008	AMT	50,001	850,017,000.00
Z008	CORP	50,001	850,017,000.00
Z008	FEDERAL	50,001	850,017,000.00
Z009	AMT	50,001	850,017,000.00
Z009	CORP	50,001	850,017,000.00
Z009	FEDERAL	50,001	850,017,000.00
Z010	AMT	50,001	850,017,000.00
Z010	CORP	50,001	850,017,000.00
Z010	FEDERAL	50,001	850,017,000.00

Table 4-4: Depreciation Processing - Conversion Load

Unit	Book Name	Count Asset ID	Sum Depr Amt
Z002	CORP	5,106	86,680,110.01

General Ledger Distribution Process (AMAEDIST & AMCOLSE)

Created G/L interface transactions for Asset additions and Depreciation expenses (refer to Table 4-5).

Table 4-5: Asset Management Feeder to GL for Period 4/1997 (Summary Information)

Bus Unit	Trans Type	Acct	Count Asset ID	Sum Amount
DEC	ADD	153000	24	1,008,000.00
DEC	ADD	154000	29	1,668,000.00
DEC	ADD	674000	53	-2,676,000.00
DEC	DPR	163000	10	-1,317.85
DEC	DPR	164000	9	-14,930.56
DEC	DPR	681300	10	1,317.85
DEC	DPR	681400	9	14,930.56
DEC	TRF	153000	10	0.00
DEC	TRF	154000	10	0.00
DEC	TRF	163000	10	0.00
DEC	TRF	164000	10	0.00
				0.00

Asset Management - Reports

Net Book Value Report

The following report is produced with “As of date” parameter which shows the asset/assets status at any point in time.

PeopleSoft AM

Report ID: AMDP2110 **NET BOOK VALUE DETAIL BY CATEGORY** Page No. 1
Bus. Unit: DEC -- DEC esac partner engineering Run Date 03/18/1997
Book: CORP -- Corporate Run Time 14:36:46
Category: FF -- Furniture & Fixtures
As of Year 1997 Period 4

Dept	Prod	Proj	Asset Id	Description	Cost Type	Cost Balance	Current Depr	YTD Depr	LTD Depr	Net Book Value
Total for Project ID						1,008,000.00	0.00	0.00	0.00	1,008,000.00
Total for Product						1,008,000.00	0.00	0.00	0.00	1,008,000.00
Total for Department 11001						1,008,000.00	0.00	0.00	0.00	1,008,000.00
Total for Project ID						62,200.00	740.48	740.48	2,961.91	59,238.09
Total for Product						62,200.00	740.48	740.48	2,961.91	59,238.09
Total for Department 13000						62,200.00	740.48	740.48	2,961.91	59,238.09
Total for Project ID						7,000.00	83.33	83.33	333.33	6,666.67
Total for Product						7,000.00	83.33	83.33	333.33	6,666.67
Total for Department 14000						7,000.00	83.33	83.33	333.33	6,666.67
Total for Category FF						1,077,200.00	823.81	823.81	3,295.24	1,073,904.76

PeopleSoft AM

Report ID: AMDP2110 NET BOOK VALUE DETAIL BY CATEGORY Page No. 2
Bus. Unit: DEC -- DEC esac partner engineering Run Date 03/18/1997
Book: CORP -- Corporate Run Time 14:36:51
Category: OFFCE -- Office Equipment
As of Year 1997 Period 4

Dept	Prod	Proj	Asset Id	Description	Cost Type	Cost Balance	Current Depr	YTD Depr	LTD Depr	Net Book Value
Total for Project ID						41,000.00	0.00	0.00	0.00	41,000.00
Total for Product						41,000.00	0.00	0.00	0.00	41,000.00
Total for Department						41,000.00	0.00	0.00	0.00	41,000.00
Total for Project ID						160,000.00	5,555.56	22,222.23	62,222.23	97,777.77
Total for Product						160,000.00	5,555.56	22,222.23	62,222.23	97,777.77
Total for Department 11001						160,000.00	5,555.56	22,222.23	62,222.23	97,777.77
Total for Project ID						270,000.00	9,375.00	9,375.00	105,000.00	165,000.00
Total for Product						270,000.00	9,375.00	9,375.00	105,000.00	165,000.00
Total for Department 13000						270,000.00	9,375.00	9,375.00	105,000.00	165,000.00
Total for Project ID						1,628,000.00	0.00	0.00	0.00	1,628,000.00
Total for Product						1,628,000.00	0.00	0.00	0.00	1,628,000.00
Total for Department 14000						1,628,000.00	0.00	0.00	0.00	1,628,000.00
Total for Category OFFCE						2,099,000.00	14,930.56	31,597.23	167,222.23	1,931,777.77
Total for Book ID CORP						3,176,200.00	15,754.37	32,421.04	170,517.47	3,005,682.53
Total for Report						3,176,200.00	15,754.37	32,421.04	170,517.47	3,005,682.53

End of Report

Tax Depreciation Report

This report is generated for tax purposes as part of month-end or Year-end closing processes.

PeopleSoft AM

Report ID: AMTX3500 **TAX DEPRECIATION DETAIL** Page No. 1
Bus. Unit: DEC -- DEC esac partner engineering Run Date 03/18/1997
Book: CORP -- Corporate Run Time 14:42:35
Dept: --
As of Year 1997 Period 1 Thru Year 1998 Period 1

Asset ID	Description	Reg	Rec. Life	Est. Life	Meth/ Sched	Conv	In-Service	Sec.179 Exp	Basis Reduct.	Basis for Depr	Depr Deduction
Total for In Service Fiscal Year 0								0.00	0.00	1,000.00	0.00
Total for In Service Fiscal Year 1996								0.00	0.00	499,200.00	200,810.71
MACRS GDS = 0.00											
MACRS ADS = 0.00											
ACRS GDS = 0.00											
ACRS GDS = 0.00											
other/Sec168 = 783,727.35											
Total for In Service Fiscal Year 1997								0.00	0.00	2,676,000.00	582,916.64
Total for Regulation								0.00	0.00	3,176,200.00	783,727.35
Total for Book ID CORP								0.00	0.00	3,176,200.00	783,727.35
Total for Report								0.00	0.00	3,176,200.00	783,727.35

PeopleSoft Payable (AP)

The purpose of the PeopleSoft Payable test was to verify the business processes function as expected in the HiTest Template. Table 4-6 lists the Process, Function, Task, and test engineer comments and Table 4-7 lists the statistics.

Table 4-6: Payable Process Testing

Process	Function	Task	Comments
Vendor Maintenance	Vendor Add	On-line vendor add	Completed
Voucher Maintenance	Voucher Adds	On-line voucher add	Completed
Inquiry	Inquire Voucher	On-line voucher inquiry	Completed
	Inquire Payment Info	On-line payment inquiry	Completed

Table 4-7: Accounts Payable Database Statistics

SetID	Vendor Count	BUS Unit	Voucher Count	SetID	Payment Count
HC	29	H01	5	MFG	28,058
MFG	50,047	M02	1		
		M04A	64		
		M04B	10		
		M04C	10		
		M05A	10		
		M05B	10		
		M60	2		
		Z001	17,001		
		Z002	17,001		

PeopleSoft Receivable (AR)

The purpose of the PeopleSoft Receivable test was to verify the business processes function as expected in the HiTest Template. Table 4-8 lists the Process, Function, Task, and test engineer comments and Table 4-9 lists the statistics.

Table 4-8: Receivable Process Testing

Process	Function	Task	Comments
Customer Maintenance	Add Customer	On-line customer add	Completed
	Update Customer	On-line customer update	Completed
Payment maintenance	Apply Payment	On-line Payment application	Completed
Inquire	Inquire Customer item	On-line Customer item inquiry	Completed

Table 4-9: Accounts Receivable Database Statistics

SetID	Customer Count	Unit	Payment Count	Group Unit	Item Count
HC	27	M02	13	M02	31
MFG	25,033	M04	70	M04	324
		M05	1	M30	24
		M30	13	M60	24
		M60	12	M64	26
		M64	13	Z001	434,375
		Z001	25,000		

PeopleSoft General Ledge (GL)

The purpose of the PeopleSoft General Ledge test was to verify the business processes function as expected in the HiTest Template. Table 4-10 lists the Process, Function, Task, and test engineer comments and Table 4-11 lists the statistics.

Table 4-10: General Ledge Process Testing

Process	Function	Task	Comments
Business Rules Definition	Define General Options	Establish Account Types	Done
		Create Detail Calendar	Added 2 detail Calendars
		Define Installation Options	Done
		Add Journal Source	Added Journal sources to SetID “BEA”
		Define Operator Preferences	Added VP1,Actuals
		Add Unit of Measure	Added “WS” - Wingspan
		Add Statistics Code	Added a Statistics code “AIR”
General Ledger Definition	Define Business Units	Add a Business Unit	Added “BEAC1”
		Specify Processing Options	Done
		Assign a SetID	Added “BEA”
	Define Chartfield Values	Add Account	Added Accounts
		Add Interunit/Open Item Account	Added 166666
		Assign Interunit Account to Bus. Unit	Added 101000 to SetID to BU Z002 and “DEC”
		Add Department	Added department “ESAC”
		Add Product	Added “HINOTE”
		Add Statistical Account	Added account 700999
		Add Statistics Code	Added type “S”
Journal Processing	Create On-line Journal Entries	Add On-line Journal Entry	Successful
		Add On-line Entry w/Errors	Completed
		Add Statistical Journal	Completed
		Add Interunit Journal	Completed
		Edit Journals	COBOL process limitation
		Correct Edit Errors	COBOL process limitation
		Process Journal Post	COBOL process limitation
		Inquiry	Successful
	Perform Journal Inquiry	Review Actuals Journal	Completed
		Review Statistical Journal	Completed
	Unpost Journals	Unpost a Journal	COBOL process limitation
	Copy Journals	Copy a Journal	Successful
Financial Inquiries	Ledger Inquiry	Perform Ledger Inquiry	See ledger report, Table 4-13.
		Drill to Ledger Detail	Completed
		Perform Account Inquiry	Done.
		Drill to Account Detail	Completed
	Journal Inquiry	Perform Journal Line Inquiry	See journal report, Table 4-12.
	Open Item & Interunit Inquiry	Perform Open Item Inquiry	Completed
		Perform Interunit Inquiry	Completed

Table 4-11: General Ledger Database Statistics

Unit	Journal Line Count
DEC	39
F01	186
H01	2,232
M02	2,141
M04	4,569
M30	2,019
M60	2,019
M64	2,019
Z001	1,168,003
Z002	1,152,126
ZZ01	1,152,009
Total	3,487,362

Table 4-12: General Ledger Journal Entries Detail - Business Unit "DEC"

Unit	Journal	Year	Per	Ledger	Description	Debits	Credits
DEC	0000005196	1,997	3	ACTUALS	test new j/v	1,000.00	1,000.00
DEC	0000005191	1,997	3	ACTUALS	test 1	100,000.00	100,000.00
DEC	0000005188	1,997	3	ACTUALS	test Journal Entry - statistic	0.00	0.00
DEC	0000005189	0	0	ACTUALS	Test interunit journal entry	2,000.00	2,000.00
DEC	ALOPER6	1,996	6	ACTUALS	This journal loads the amounts	400,000.00	400,000.00
DEC	ALOPER7	1,996	7	ACTUALS	This journal loads the amounts	400,000.00	400,000.00
DEC	ALOPER8	1,996	8	ACTUALS	This journal loads the amounts	400,000.00	400,000.00
DEC	ALOPER9	1,996	9	ACTUALS	This journal loads the amounts	400,000.00	400,000.00
DEC	ALOPER10	1,996	10	ACTUALS	This journal loads the amounts	400,000.00	400,000.00
DEC	ALOPER11	1,996	11	ACTUALS	This journal loads the amounts	400,000.00	400,000.00
DEC	ALOPER12	1,996	12	ACTUALS	This journal loads the amounts	400,000.00	400,000.00
DEC	ALOPER1	1,996	1	ACTUALS	This journal loads the amounts	400,000.00	400,000.00
DEC	0000005195	1,997	3	ACTUALS	Test the server	1,000.00	1,000.00
DEC	0000005186	1,997	3	ACTUALS	Test Journal Entry - Base Curr	40,000.00	40,000.00
DEC	0000005194	1,997	3	ACTUALS		0.00	0.00
DEC	0000005192	1,997	3	ACTUALS	Test the process	100.00	100.00
DEC	0000005193	1,997	3	ACTUALS	dec test	100.00	0.00
DEC	ALOPER2	1,996	2	ACTUALS	This journal loads the amounts	400,000.00	400,000.00
DEC	ALOPER3	1,996	3	ACTUALS	This journal loads the amounts	400,000.00	400,000.00
DEC	ALOPER4	1,996	4	ACTUALS	This journal loads the amounts	400,000.00	400,000.00
DEC	ALOPER5	1,996	5	ACTUALS	This journal loads the amounts	400,000.00	400,000.00

Table 4-13: General Ledger - F01 Balances

Unit	Ledger	Acct	Sum Total Amt
F01	ACTUALS	100000	-110,000.00
F01	ACTUALS	111000	200,000.00
F01	ACTUALS	120000	200,000.00
F01	ACTUALS	121000	450,000.00
F01	ACTUALS	130000	66,000.00
F01	ACTUALS	141000	1,000,000.00
F01	ACTUALS	142000	1,665,000.00
F01	ACTUALS	150000	-40,000.00
F01	ACTUALS	160000	70,000.00
F01	ACTUALS	200000	-650,000.00
F01	ACTUALS	210000	-2,540,000.00
F01	ACTUALS	240000	-10,000.00
F01	ACTUALS	250000	-65,000.00
F01	ACTUALS	251000	-66,000.00
F01	ACTUALS	270000	-20,000.00
F01	ACTUALS	300000	-40,000.00
F01	ACTUALS	310000	-5,000.00
F01	ACTUALS	320000	-40,000.00
F01	ACTUALS	330000	-65,000.00
F01	ACTUALS	410000	-240,000.00
F01	ACTUALS	430000	-15,000.00
F01	ACTUALS	440000	-66,000.00
F01	ACTUALS	500000	150,000.00
F01	ACTUALS	510000	0.00
F01	ACTUALS	540000	65,000.00
F01	ACTUALS	542000	66,000.00
F01	ACTUALS	580000	40,000.00

5
System Limits and Performance Data

This section discusses any system limits that may have been determined as a result of the testing, along with information regarding the system performance during testing. Areas covered include:

- Transaction response time by business functions
- Performance issues with respect to the PeopleSoft import manager and sample query statistics

Transaction Response Time:

Table 5-1 reports the percentage breakdown of business functions tested and the average retrieve, average update, and retrieve per panel response times (in seconds) from the PeopleSoft Financial database.

The transaction response times were generated based upon 10 iterations of the panel tests from the PeopleSoft PC client (measured by SQA Robot test procedures).

Table 5-1: Transaction Response Time

Financials On-line Benchmark 5.01
Number of Processes: 10
DIGITAL ESAC Engineering

Process	Trans Name	Percent Total	Number Panels	Number Inserts	Number Updates	Number Selects	Avg Retrieve (sec)	Avg Update (sec)	Retrieve Per Panel
Asset Management									
Asset Add	am01	3.96%	6	11	1	36	0.41	5.77	0.07
Asset Transfer	am02	0.52%	4	10	1	43	3.12	4.78	0.78
Inquire Asset Basic Info.	am03	1.26%	13	1	1	34	3.11		0.24
Inquire Asset Depreciation	am04	0.84%	3	1	0	40	2.46		0.82
Inquire Asset Cost History	am06	0.84%	2	1	0	19	1.71		0.86
Payables									
Vendor Add	ap01	3.39%	9	7	0	24	4.76	1.51	0.53
Inquire Payment Info.	ap02	4.29%	1	1	0	15	2.13		2.13
Inquire Voucher	ap03	4.29%	1	1	1	16	1.73		1.73
Voucher Add	ap04	42.39%	16	4	0	50	4.07	7.02	0.25
Receivables									
Customer Add	ar01	0.58%	3	3	2	31	3.73	1.96	1.24
Customer Update	ar02	0.67%	3	2	5	22	2.86	2.44	0.95
Apply Payments	ar03	6.39%	2	3	15	66	6.05	1.04	3.03
Inquire Customer Item	ar04	13.24%	3	2	4	56	4.47	5.96	1.49
General Ledger									
Journal Add	gl01	10.59%	5	6	0	33	5.01	2.07	1
Inquire Ledger	gl03	4.23%	2	1	0	33	1.64		0.82
Inquire Journal	gl05	2.11%	2	1	0	23	1.78		0.89
Jrnl add Foreign Currency	gl06	0.41%	5	6	0	43	3.56	2.11	0.71
Nvision Reports									
Income Statements	nv01	Once	n/a				0.7		
Balance Sheet	nv02	Once	n/a				0.52		
Totals		100.00%		61	30	584			

Performance Issues Using the PeopleSoft Import Manager

Because the import manager runs on client workstation, PeopleTools executes the PeopleCode program to edit and process each row. This ensures data integrity, but is very slow compared to other database load tools.

One way to avoid this performance problem is to split the source table into small files and run the imports simultaneously from multiple available workstations.

Another strategy to ensure data and table integrity with the input data is to utilize the faster and more powerful tools which run on the DIGITAL UNIX Oracle Parallel Database server. For example, when using import manager to load the ps_open_trans table for the ‘CORP’

book, 1000 rows completed in 1 hour. SQL*Plus was then used to create the FEDERAL book entries from the previously loaded CORP book. This ensured the data and table integrity. SQL*Plus created a temporary table first, which was then used to load the next 1000 FEDERAL rows. This took approximately 2 minutes. Below are examples of the SQL statements:

```
> create table booktmp as select * from ps_open_trans
      where business_unit = 'DEC' and book = 'CORP'
> update booktmp set book = 'FEDERAL'  where business_unit = 'DEC'
> insert into ps_open_trans select * from booktmp
```

Sample Query statistics

Response was extremely quick when selecting queries (row selects, row counts, extracts and join queries) from PeopleTools, as shown in Table 5-2.

Table 5-2: Sample Query Statistics

Query Type	Query Scope	Response (seconds)
Selection of specific rows	7 rows selected from a total of 50,076 rows in the ps_vendor table.	2.58
Row count query	24 subgroups 500,164 total rows in the ps_asset table.	3.69
Combination of value select and row count	1 subgroup with 1,168,003 rows selected from a total of 11 subgroups with a total row count over 3,500,000 in the ps_jrnl_ln table.	26
Table join query	1 asset with 1 cost row and 8 depreciation rows in the ps_cost and ps_depreciation tables.	0.33

6
Problems and Resolutions

This section discusses any problems that may have been encountered during the testing, along with any required actions or workarounds.

NOTE

The PeopleSoft Financials benchmark kit that was used during testing currently focuses on only four of the Financial Products: AM, AP, AR and GL. Some of the problems encountered were due to the use of this kit. Those problems have the word Benchmark in the problem name.

The following problems were identified and resolved during testing:

Compiling PeopleSoft Financials Software

Problem Compiling the PeopleSoft software with Microfocus COBOL V4.0 results in an internal error for 13 of the 256 programs compiled as follows:

```
038-F Internal error.  Contact technical support
cob: error(s) in code generation: ARPDIBLD.int
```

Resolution This error, related to producing generated native code, does not appear to occur in the previous Microfocus COBOL version (V3.2). The error can be worked around by using the `.int` files of the same name that are produced. The problem is known to Microfocus and a fix should be included in the next product update.

Problem PeopleSoft makefiles and Oracle V7.3.2.3 libraries locations. Attempts to compile the PeopleSoft software on the DIGITAL UNIX platform would fail with the loader reporting unresolved symbols. This was caused by a change in the location of the Oracle V7.3.2.x release libraries.

Resolution PeopleSoft provided a new version of the make file to address this issue.

Oracle Software Groups and Ownership

Problem

If the Oracle software is to be shared to all nodes in the cluster, then make sure the oracle user accounts are the same, in particular the DBA group as this can cause the following error:

```
depot5> svrmgrl
Oracle Server Manager Release 2.3.2.0.0 - Production
Copyright (c) Oracle Corporation 1994, 1995. All rights reserved.
ORA-03113: end-of-file on communication channel
```

Resolution

The problem was that the dba group was 79 on the other nodes, but 26 on depot5. Either change the group number (/etc/group) and the Oracle users paswd entry to 79 (use vipw), or add the Oracle users name to the DBA entry in /etc/group. This can also be handled during account creation by making the Oracle account identical on the cluster nodes.

Problem

Make sure to properly set the protections on the DRDs on all nodes in the cluster. All devices in the /dev/rdrd directory on each node should be owned by Oracle.

Resolution

Execute the following command from root to set the protections:

```
chown oracle:dba /dev/rdrd/*
```

PeopleSoft Process Scheduler

Problem

We were unable to perform testing of the batch functionality on the UNIX database server. The reason was, the PeopleSoft Process Scheduler was unable to correctly logon to Oracle as the SYSADM user. Although, it was first suspected that this was related to the Microfocus COBOL version, this did not prove to be the case; a second version, provided by PeopleSoft and compiled on the previous version of Microfocus COBOL (V3.2), had the same problem.

Resolution

This issue and supporting data have been provided to PeopleSoft for resolution.

Benchmark Installation - Data

Problem

The application data provided is in a corrupted state for future processing. Some of this is from a logical standpoint (asset depreciation is greater than asset cost) and some of this corruption invalidates the PS table integrity which prevents future processing (assets without open_trans rows).

Resolution

Correcting this requires a couple of steps. First the existing data and tables were corrected via direct SQL insert, update and delete statements. Also, for ongoing processing, a more scientific data clean-up and refresh process must be established. This is explained in more detail under “On-Line Panel Testing - Data Refresh/Clean-Up,” below.

Utilized SQL statements to clean-up and reestablish the table integrity as well as inputting new data.

Benchmark Installation - System Components

Problem The nature of the financial systems require a significant amount of batch processing and reporting. The kit's focus was on-line, thus the requirements to process batch were not complete. The missing components are:

- SQR server processing
- COBOL compile and execution (client and server).
- Crystal executables

Resolution Upgrade the system packaging to ensure that the full functionality is provided.

Executed SQR processing via PS's Process Scheduler and directly from SQR. The inability to process COBOL limited GL and AP to on-line tests.

On-line Panel Testing - Timing (Benchmark Kit)

Problem The 'refresh' or clean-up job, UNDOFSADDS, is set to execute every 20 minutes and takes less than a minute to complete. This time dependency forces the SQA Robot process, FSONLINE, to wait until this job has run. At best this time requirement causes long delays. Because this clean-up is time driven, there is the risk that the testing and clean-up processing become out of sync thus corrupting the tests.

Resolution Change the clean-up job to be executed via a submit process which can be triggered by the testing process. Ideally the clean-up SQL statements could be incorporated into a PS function which is then incorporated into the normal PS processing loop.

Established a copy of the clean-up which was submitted manually at the proper testing points.

On-line Panel Testing - Data Refresh/Clean-Up (Benchmark Kit)

Problem UNDOFSADDS is a collection of SQL update and delete statements which resets the data of specific PS tables back to an original status. The "where" clauses of certain statements are too generic which causes other data to be affected

Resolution Update these statements to ensure that only the benchmark robot data is reset.

Example: Delete from `ps_open_trans` where `dttm_stamp` >= '01-JAN-96' effects all asset activity dated after Jan 1, 1996.

This statement should be updated to affect only the benchmark data, i.e. where business_unit = 'Z001' and asset_id = 'xxxx'.

Established data which would not conflict with the existing statements and utilized an updated version of the UNDOFSADDS with improved "where" clauses.

Batch processing - Batch types

Problem As mentioned in the system components above, the batch processing was limited to SQR client processes.

Resolution The inability to process COBOL limited GL and AP to on-line tests.

Batch processing - PS Process Scheduler

Problem When utilizing PS process scheduler to submit SQR's, in most cases the job would hit memory limitations on the client. However the exact same job would process successfully when submitted directly to the client via SQR.

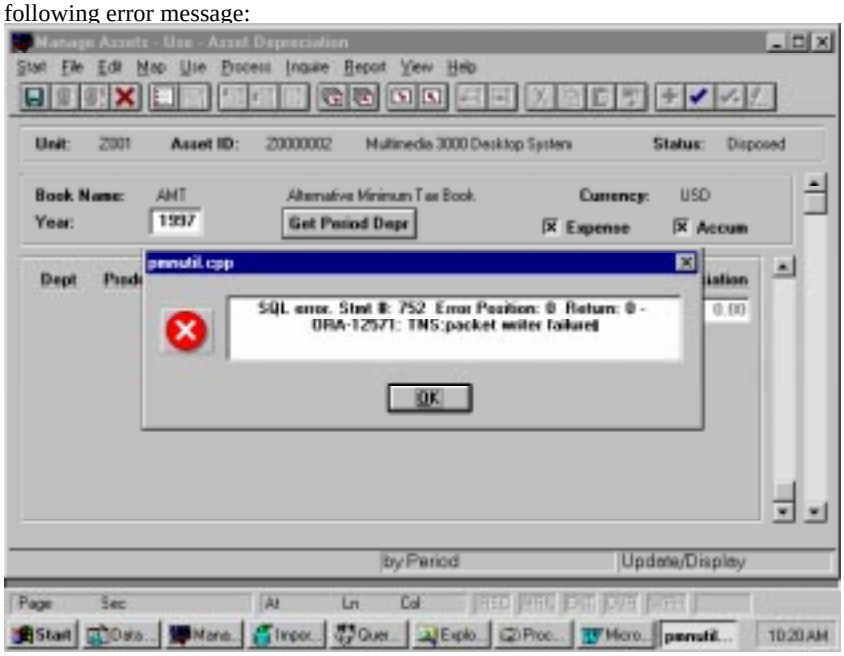
Resolution The long term solution is to do batch processing on the server which eliminates this processing.

Submitted the jobs directly from SQR, not the PS Process Scheduler

---

System/Functional Issues - Database disconnect

Problem Periodically the client would be disconnected from the database, generating the following error message:



Resolution Exiting and re-entering PS established a new database connection. We did not find enough consistency with the triggers to this disconnect to establish a permanent solution.

System/Functional Issues - Duplicate Open Transaction IDs

Problem While utilizing multiple workstations and logged into PS with the same operator id: If the second workstation begins an asset add transaction before the first workstation completes its asset add transaction, both assets generate the same open_trans_id number.

Resolution It appears to be a cosmetic issue only. Future processing is not prevented or corrupted.

System/Functional Issues - Retirements Processed by Program AMAEDIST

Problem The AMAEDIST program aborted while attempting to process RET transactions. Because the job was not submitted via Process Scheduler logs or debugging information is not provided.

Resolution This is probably a known and corrected PS issue because RET transactions do successfully process in client environments.

A
Detailed Hardware Configuration

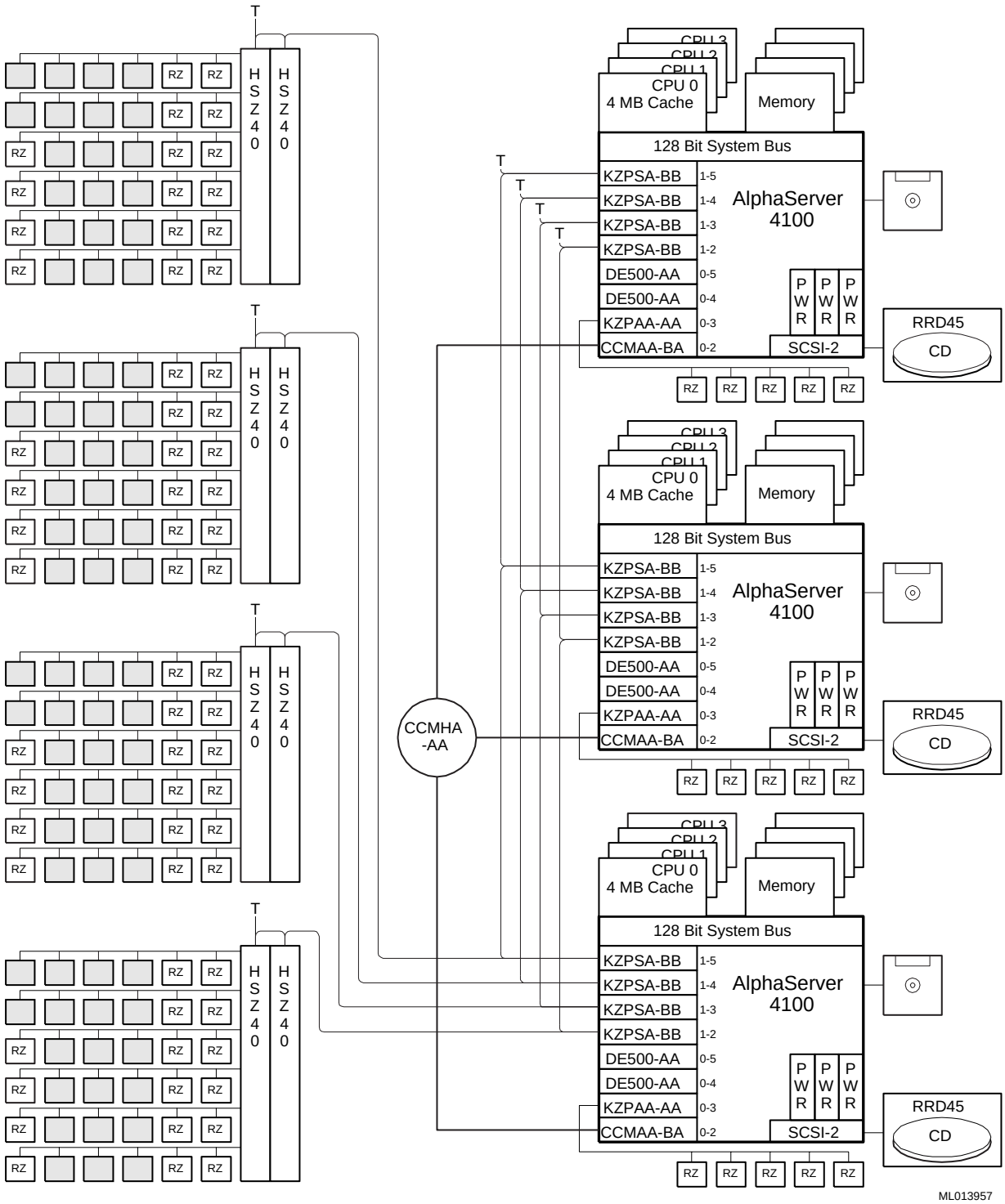
This appendix describes the minimum and maximum hardware configuration for the following:

- Hardware Overview (Golden Eggs Like)
- AlphaServer 4100 configurations, including
  - System motherboard
  - PCI backplanes
- Configuration Cabling
- Disk architecture

Overview (Golden Eggs Like)

Figure A-1 shows an overview of the entire HiTest Template

Figure A-1: Overview (Golden Eggs Like)



AlphaServer 4100 Motherboard

Figure A-2 and Table A-1 show the AlphaServer 4100 system motherboard and describe the minimum and maximum hardware configurations used in this HiTest Template.

Figure A-2: AlphaServer 4100 Motherboard

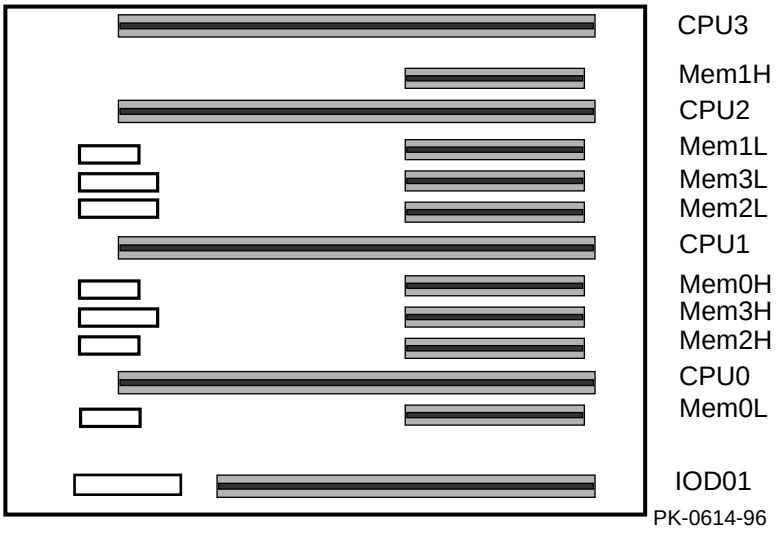


Table A-1: AlphaServer 4100 Usage (Minimum and Maximum Configurations)

Slot	Minimum Configuration Options	Maximum Configuration Options	Description
CPU3	KN304-BC	KN304-BC	400 MHz CPU 4 MB cache
Mem1H	MS320-FA	MS330-FA	Memory pair 1 (2 of 2)
CPU2	KN304-BC	KN304-BC	400 MHz CPU 4 MB cache
Mem1L	MS320-FA	MS330-FA	Memory pair 1 (1 of 2)
Mem3L	open	open	
Mem2L	MS320-FA	MS330-FA	Memory pair 2 (1 of 2)
CPU1	KN304-BC	KN304-BC	400 MHz CPU 4 MB cache
Mem0H	MS320-GA	MS330-GA	Memory pair 0 (2 of 2)
Mem3H	open	open	
Mem2H	MS320-FA	MS330-FA	Memory pair 2 (2 of 2)
CPU0	KN304-BC	KN304-BC	400 MHz CPU 4 MB cache
Mem0L	MS320-GA	MS330-GA	Memory pair 0 (1 of 2)
IOD01	Bridge	Bridge	System bus to PCI bus bridge module

AlphaServer 4100 PCI Slot Usage

Figure A-3 and Table A-2 show the PCI slot usage for the minimum and maximum configurations of this HiTest Template.

Figure A-3: AlphaServer 4100 PCI Slot Usage

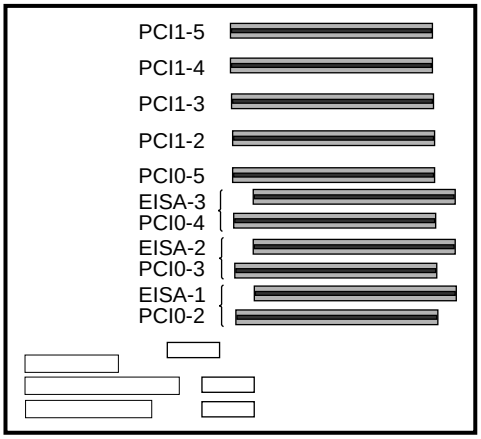


Table A-2: PCI Slot Usage (Minimum and Maximum Configurations)

Slots	Minimum Configuration Options	Maximum Configuration Options	Description
PCI1-5	KZPSA-BB	KZPSA-BB	FWD SCSI controller
PCI1-4	KZPSA-BB	KZPSA-BB	FWD SCSI controller
PCI1-3	KZPSA-BB	KZPSA-BB	FWD SCSI controller
PCI1-2	KZPSA-BB	KZPSA-BB	FWD SCSI controller
PCI0-5	DE500-AA	DE500-AA	Ethernet controller
EISA-3/ PCI0-4	DE500-AA	DE500-AA	Ethernet controller
EISA-2/ PCI0-3	KZPDA-AA	KZPDA-AA	SCSI host adapter
EISA-1/ PCI0-2	CCMAA-BA	CCMAA-BA	Memory Channel

Configuration Cabling

Table A-3 lists the major cables in the HiTest Template. Use this table in conjunction with the overview figure.

Table A-3: Configuration Cabling

Part Number	Qty	Description	From	To
BN21W-0B	12	SCSI Y cable	KZPSA-BB	Shared SCSI buses
BN21K-03	8	Shared SCSI buses	SCSI Y-cable	SCSI Y-cable
BN21K-10	4	Shared SCSI buses	SCSI Y-cable	HSZ40
BN21L-0B	4	Shared SCSI bus	HSZ40	HSZ40
BN21N-02	24	8/16-bit SCSI bus	HSZ40	StorageWorks shelves
BC12N-10	3	Memory channel link	CCMAA-BA	CCMHA-AA

Disk Architecture

Table A-4 lists the architecture of the disk subsystem for this HiTest System.

Table A-4: Disk Architecture

SCSI Bus	Logical Unit (LUN)	RAID Level	Member Drives
SCSI1	rrz8c	5	4 each 4.3 GB
	rrz9c	5	4 each 4.3 GB
	rrz10c	5	4 each 4.3 GB
	rrz11c	5	4 each 4.3 GB
SCSI2	rrz16c	5	4 each 4.3 GB
	rrz17c	5	4 each 4.3 GB
	rrz18c	5	4 each 4.3 GB
	rrz19c	5	4 each 4.3 GB
SCSI3	rrz24c	5	4 each 4.3 GB
	rrz25c	5	4 each 4.3 GB
	rrz126c	5	4 each 4.3 GB
	rrz27c	5	4 each 4.3 GB
SCSI4	rrz32c	5	4 each 4.3 GB
	rrz33c	5	4 each 4.3 GB
	rrz34c	5	4 each 4.3 GB
	rrz35c	5	4 each 4.3 GB