System Administration Guide

Version 2.0

CONFIDENTIAL INFORMATION

DISCLAIMER

TRADEMARKS

Document released: October 2009

Web address: http://www.exlibrisgroup.com

Table of Contents

Chapter 1 Introduction 7

About This Guide 7

Supported Platforms 8

Chapter 2 Overview of the Verde System Architecture and Directory Structure 9

Verde Server Architecture 9

Verde Directory Structure 11

Upper-Level Structure 11

Application Server and Batch Process Structure 12

Chapter 3 Verde UNIX Users 15

Chapter 4 Verde Tasks 17

Chapter 5 Verde Configuration and Log Files 19

Verde Configuration Files 19

Content of the global.properties File 19

Verde Log Files 22

Chapter 6 The Verde License 25

License Description 25

License Example 26

Chapter 7 Preventive Maintenance and Recovery 27

Overview 27

Ensuring Sufficient Database Space 28

Tablespace Free Space 28

Database Temporary Tables 29

Manual Cleanup 29

Working in Archive Log Mode 30

Backing Up Your System 30

Components to Back Up 30

Backup Strategy 32

Disk Cleanup 34

Application Cleanup 34

Oracle Log Cleanup 34

Reviewing the Oracle Alert Log 34
Recovery Policy 35
Summary of Periodic Maintenance Activities 36
Chapter 8 Oracle Overview 37
Oracle Structure 37
SQL Access to the Oracle Tables 37
Oracle Users in Verde 38
Passwords 39
Oracle Concepts 39
Storage 39
Users 40
Tables 40
Indexes 40
Triggers 40
Oracle Table Management – file_list 41
Oracle Utilities 42
Chapter 9 Oracle Management Utility (UTIL O) 43
Overview 43
Oracle Server (UTIL O/1) 44
Activate Oracle Server (UTIL O/1/1) 45
Close Oracle Server (UTIL O/1/2) 45
Show Running Oracle Server (UTIL O/1/3) 46
Show Oracle Server Status (UTIL O/1/4) 47
Oracle Listener (UTIL O/2) 47
Activate Oracle Listener (UTIL O/2/1) 47
Close Oracle Listener (UTIL O/2/2) 48
Show Running Oracle Listener (UTIL O/2/3) 48
Show Listener Status (UTIL O/2/4) 49
Show Listener Services (UTIL O/2/5) 50
Oracle Logs (UTIL O/3) 50
View Oracle ALERT LOG (UTIL O/3/1) 50
NLS (UTIL O/6) 51
Show NLS Parameters (UTIL O/6/1) 51

Archiving (UTIL O/7) 51
About Archiving 52
Turning Archiving On (UTIL O/7/1) 52
Turning Archiving Off (UTIL O/7/2) 52
Show Archiving Status (UTIL O/7/3) 53
Database Users (UTIL O/9) 53
List Database Users (UTIL O/9/1) 53
Create a New User (UTIL O/9/2) 54
Update Oracle Passwords of Verde Users (UTIL O/9/3) 56
SQL*Plus Session (UTIL O/10) 56
Database Verification Utility (UTIL O/12) 57
Run Database Verification Utility (UTIL O/12/1) 57
Find Corrupted Object (UTIL O/12/2) 58
Database Files (UTIL O/13) 59
List of Database Files (UTIL O/13/1) 60
Resize Oracle Datafile (UTIL O/13/2) 60
Add File to Tablespace (UTIL O/13/3) 61
Show Datafile Free Blocks by KBytes (UTIL O/13/4) 62
Show Datafile Free Blocks by BlockID (UTIL O/13/5) 62
Database Free/Used Space (UTIL O/14) 63
All Tablespaces Free Space Summary (UTIL O/14/1) 63
Number of Free Extents by Size in a Tablespace (UTIL O/14/2) 64
All Free Extents of Minimum Size in a Tablespace (UTIL O/14/3) 65
Space Used by a Library/Libraries in Each Tablespace (UTIL O/14/4) 65
Space Used by a Group of Libraries in Each Tablespace (UTIL O/14/5) 66
Coalesce Contiguous Free Extents (UTIL O/14/6) 66
Verde Tablespaces Total/Free/Used Space Report (UTIL O/14/8) 67
Clean Temporary Tablespace Free Storage (UTIL O/14/9) 67
Manage Database Links (UTIL O/16) 67
List Database Links (UTIL O/16/1) 68
Create Database Links (UTIL O/16/2) 68
Drop Database Link (UTIL O/16/3) 69

Database Tablespaces (UTIL O/17) 69
Create a Tablespace (UTIL O/17/1) *70*
List Tablespace Files (UTIL O/17/2) *71*
Check Tablespace for a Library Based on file_list Template (UTIL O/17/3) *71*
Show Tablespaces Definition (UTIL O/17/4) *71*
Show Tablespace Allocated/Free/Used Space (UTIL O/17/5) *72*
Oracle Statistics (UTIL O/18)..... 73
Performance Statistics (UTIL O/18/1) *74*
Rollback Segments Definitions (UTIL O/18/2) and Rollback Segments Dynamic Allocation (UTIL O/18/3) *74*
View Long Operations (UTIL O/18/4) *75*
IO Statistics (UTIL O/18/5) *76*
Sort Operations (UTIL O/18/6) *76*
Table Statistics (UTIL O/18/7) *76*
Shared Pool (UTIL O/19) 77
Show SGA Buffers (UTIL O/19/1) *77*
Flush Shared Pool (UTIL O/19/2) *78*
Multi Threaded Server (UTIL O/20) 79
Show MTS Parameters (UTIL O/20/1) *79*
Show Listener Services (UTIL O/20/2) *80*
Chapter 10 Cleanup Utility (UTIL C) 83
Deletion of Temporary Files (UTIL C/1) *83*
Deletion of Old Tasks Output Files (UTIL C/2) *83*
Truncate Oracle Text Errors (UTIL C/3) *83*

1

Introduction

This section includes:

- **About This Guide** on page 7
- **Supported Platforms** on page 8

About This Guide

This guide provides a detailed overview of the Verde system and its components, and instructions on how to perform system administrative tasks. In particular, it provides information about the following:

- the Verde system architecture and directory structure – See **Overview of the Verde System Architecture and Directory Structure** on page 9.
- Verde UNIX users – See **Verde UNIX Users** on page 15.
- Verde tasks – See **Verde Tasks** on page 17.
- Verde configuration files and log files – See **Verde Configuration and Log Files** on page 19.
- the Verde license – See **The Verde License** on page 25.
- preventive maintenance and recovery – See **Preventive Maintenance and Recovery** on page 27.
- Oracle maintenance and management – See the following chapters:
  - **Oracle Overview** on page 37
  - **Oracle Management Utility (UTIL O)** on page 43
  - **Cleanup Utility (UTIL C)** on page 83

Given its technical content, the primary audience for this guide is system administrators or staff with technical knowledge. Certain sections of the preventive maintenance and recovery chapter, as well as the Oracle maintenance and management chapters, are written for Oracle DBAs.

Supported Platforms

Verde can be installed on the following platforms:

- Sun Solaris
- Linux Red Hat

For a comprehensive list of system requirements, refer to the *Requirements for Verde Installation* document.

2

Overview of the Verde System Architecture and Directory Structure

This section includes:

- **Verde Server Architecture** on page **9**
- **Verde Directory Structure** on page **11**

Verde Server Architecture

The Verde server is composed of the following tiers:

- **Web** – Provides the interface with which the user interacts.
- **SOAP Gateway** – The gateway allows other application servers to carry out operations and retrieve information from Verde.
- **Application Logic** – The heart of Verde is the application services tier, which exposes a sets of APIs. The APIs provide library services for the different clients and application servers.

  The Application Logic tier contains the following:

  - **EJB Container** – Contains Verde's business logic
  - **JBPM Workflow Engine** – Contains a third-party engine that implements Business Process Management. Verde uses this engine to implement its managed workflows.
  - **Hibernate (Data Tier)** – This is a high-level database management layer. A logical server provides data services to the Application Logic objects. The Hibernate tier contains a group of objects that intermediate between the application and the database and translate an application request to a sequence of database commands.

    Having an intermediate level between the application and the DBMS ensures maximum flexibility of the DBMS logical and physical design.
- **Verde Database** – The Verde database runs using Oracle RDBMS.

- **Remote Job Manager** – This is a server that runs on the J2EE platform. It controls a collection of tasks running on the Remote Job Manager engine and carries out operations such as scheduling, cleaning, reporting, executing various business rules, and indexing.

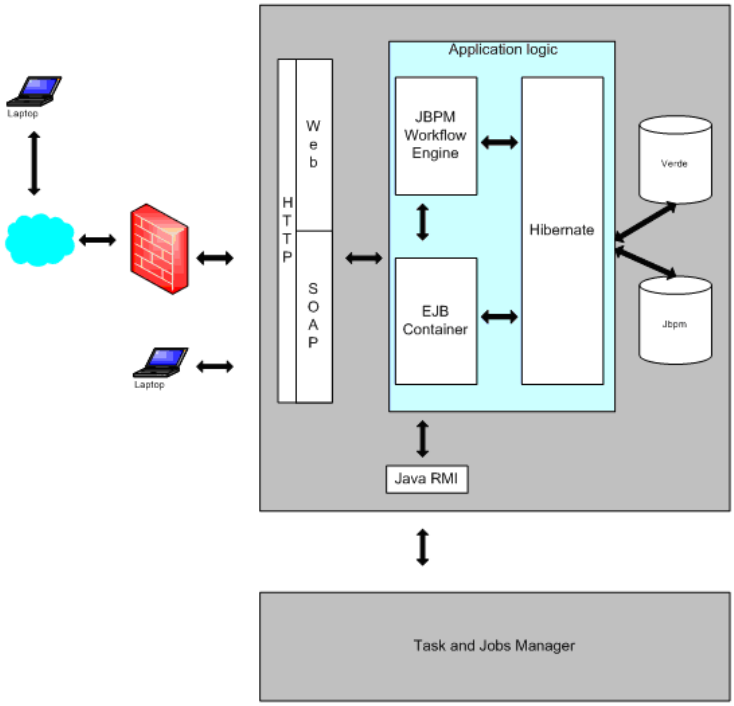

Figure 1: Verde Server Architecture

Verde Directory Structure

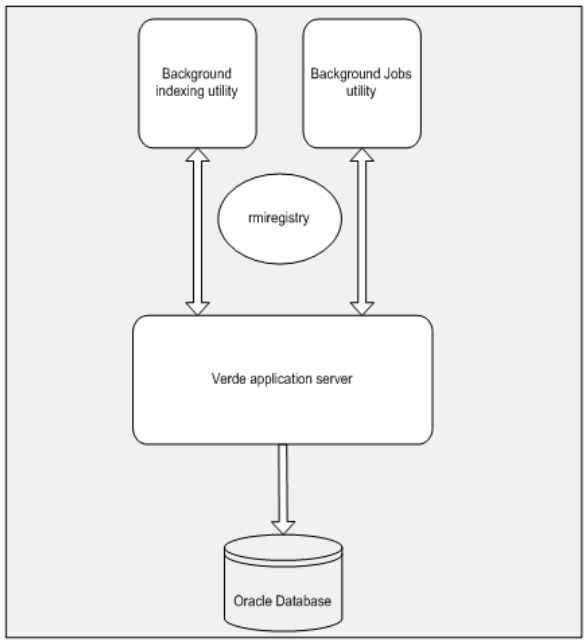

Figure 2: Verde Components - Processes Comprising the Verde Software

Upper-Level Structure

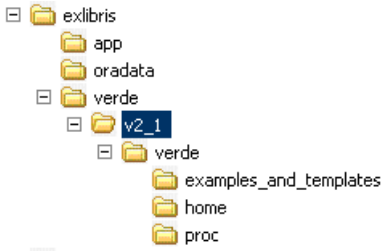

Figure 3: Directory Structure After Verde Has Been Installed

Table 1. Upper-Level Verde Directory Structure

Directory	Description
`/exlibris/app`	Contains Oracle software and resources
`/exlibris/oradata`	Contains the Oracle database
`/exlibris/verde/v2_1/verde`	Contains Verde's component files
`/exlibris/verde/v2_1/verde/proc`	Contains database utilities

Table 1. Upper-Level Verde Directory Structure

Directory	Description
`/exlibris/verde/v2_1/verde/home`	Contains the Verde application server and batch processes
`/exlibris/verde/v2_1/verde/examples_and_templates`	Contains templates and example files. Verde provides template files that can be used when preparing data for supported processes, such as loaders.

Application Server and Batch Process Structure

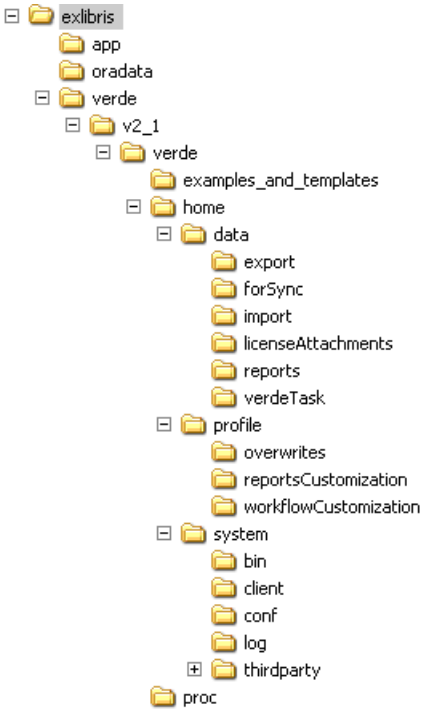

Figure 4: Application Server and Batch Process Structure

Table 2. Application Server and Batch Process Directory Structure

Directory	Description
`/exlibris/verde/v2_1/verde/home`	Contains the Verde application server and batch processes

Table 2. Application Server and Batch Process Directory Structure

Directory	Description
`/exlibris/verde/v2_1/verde/home/data`	Contains subdirectories in which the application stores data that is generated either during the application's activity or when the user uploads files such as license attachments. (For more information on license attachments, see the *Verde Staff User's Guide*.)
`/exlibris/verde/v2_1/verde/home/profile`	Contains configuration and template files with parameters that should be configured to match the installation site's values. All values are configured in one single file called `global.properties`. All template files are located in the `overwrites` directory. A dedicated process called `set_globals.sh`, which is located in `home/system/bin`, must then be executed to propagate the values as they have been configured in `global.properties`.
`/exlibris/verde/v2_1/verde/home/profile/reportsCustomization`	Contains directories for each instance in the system (by instance code). These directories contain configuration files for report output formats. (For more information about instances or output formats in Verde, see the *Verde Staff User's Guide*.)
`/exlibris/verde/v2_1/verde/home/system`	Contains the actual software
`/exlibris/verde/v2_1/verde/home/system/conf`	Contains the Verde application's configuration files, SQL scripts that are used to load/refresh initial data to the database, and Verde's license file (`license.xml`).
`/exlibris/verde/v2_1/verde/home/system/conf/license.xml`	This file specifies all the parameters that are agreed upon between the customer and Ex Libris and enforced by the Verde application. Any change that is made to the content of this file—even the smallest change possible—will result in a non-functioning Verde application.
`/exlibris/verde/v2_1/verde/home/system/bin`	Contains Verde utilities and programs, such as startup and shutdown, indexing, the Task Manager, data loader execution, and so forth.

Table 2. Application Server and Batch Process Directory Structure

Directory	Description
`/exlibris/verde/v2_1/verde/home/system/log`	Contains log files with log messages written by batch processes and utilities
`/exlibris/verde/v2_1/verde/home/system/client`	Contains software for running batch processes
`/exlibris/verde/v2_1/verde/home/system/thirdparty`	Contains resources and software for the JBoss application server, including the Verde application
`/exlibris/verde/v2_1/verde/home/system/thirdparty/openserver/server/default/log`	Contains log files with log messages written by the Tomcat Servlet Container and/or Verde

3

Verde UNIX Users

The Verde server requires the following UNIX users:

Table 3. UNIX Users

User	Home Directory	Description
`verde`	`/exlibris/verde/v2_1/verde`	Verde administrator
`oracle`	`/exlibris/app/oracle/product/920`	Oracle administrator (DBA)

The person who is in charge of all the repositories, the system administrator, can log in as `verde`. This enables the system administrator to modify the parameters and data of all repositories.

The Verde administrator (system administrator and/or system librarian) uses the `verde` account for various online utilities and command-line activity, including starting/stopping Verde processes. The `verde` user has access to and control over all the repositories in the system.

The Oracle DBA uses the `oracle` account for DBA activity outside the scope of the **UTIL O – Oracle Management**.

4

Verde Tasks

Verde comes with several administrative scheduled tasks, which are configured to run at certain intervals. System and network utilization have been taken into consideration in configuring these default intervals.

The following table describes each of the scheduled tasks:

Table 4. Verde Scheduled Tasks

Task	Interval	Description
Index optimization	Weekly, at midnight	Calls Oracle's indexing optimization process that improves the work of the database.
Resend e-mail	Every five minutes	Resends all e-mail messages that have been invoked by Verde's task/report functions without e-mail notification having been sent successfully.
Remove old job	Daily	Cleans up old and unused records from the job table, which functions as a built-in job queue in Verde.
Re-index failures	Hourly	Re-indexes e-products that have changed without their information having been re-indexed.
Output cleanup	Daily	Cleans up old output files generated by Verde's task/report functions. Files created 30 days or more before the current date are considered to be old.
Truncate index error	Daily	Truncates the Oracle text indexing errors table
Temporary index file cleanup	Daily	Deletes all old files created as temporary files during the indexing process. Old files are files that have already been loaded by Oracle text and have no use.

5

Verde Configuration and Log Files

This section includes:

- **Verde Configuration Files** on page **19**
- **Verde Log Files** on page **22**

Verde Configuration Files

Verde uses several configuration files to control its ongoing operations. Although these configuration files are located throughout the directory structure, Verde also includes a single file, `global.properties`, that contains all necessary configuration values and a utility that is responsible for uploading these values to all relevant configuration files. The `global.properties` file resides in the `/exlibris/verde/v2_1/verde/home/profile` directory.

Content of the global.properties File

```
# Profile identifications
profile.port.prefix - any number here will be used as a prefix
for all configured ports in the system


http.machine.address - The machine IP address where the
Database is installed
```

```
# data base connection parameters
dbconnection.dialect=net.sf.hibernate.dialect.OracleDialect
dbconnection.driver_class=oracle.jdbc.driver.OracleDriver
dbconnection.SID - Database SID
dbconnection.urlprefix=jdbc:oracle:thin:@
dbconnection.url=@@dbconnection.urlprefix@@@@http.machine.add
ress@@:1521:@@dbconnection.SID@@
dbconnection.username - Database user name
dbconnection.password - Database password


#bpm schema parameters
bpm.dbconnection.username=verde_bpm - JBpm schema user name
bpm.dbconnection.password=verde_bpm- JBpm schema password


# system type definitions

# define if Verde should have merge capabilities between
instances
# no value or "SEPARATE" mode indicates that the no sharing
allow between
# instances, "SHARE" mode means that there is some sharing
# level between instances information
verde.share.mode=
# define the shared instance or leave it empty if not relevant
# use "NONE" to define no shared instance
shared.instance=

# Ports

# deploy/jbossweb-tomcat.sar/server.xml
http.conector - http main port
http.redirect - http redirection port
http.ajp -

# deploy/conf/jboss-minimal.xml , deploy/conf/jboss-
service.xml
rmi.main - jboss jnp port
rmi.naming - jboss naming port
rmi.object - jboss listeners port
rmi.server.bind - jboss binding port
webservice.connector - jboss webservice connector port
jms.connector - jboss JMS port

# verde/home/profile/overwrites/conf/system_config.xml
rmi.registry.server - rmiregistry address for batch processes
rmi.registry.port - rmiregistry port
```

```
# Application parameters

# general
install.root.dir - root installation directory
# the sequence of this profile.
# if non empty then MUST be '-' prefix e.g. -1
verde.profile.sequence - will have value only if more then one
"system" is installed
verde.home- verde home directory
verde.data - where all sort of data (temporary) is
verde.import - where imported data is
verde.export - where exported data is
verde.profile - the current profile directory
verde.system - the current system directory
verde.bin - where all programs are
verde.proc.util - where all Database utilities


# index files
index.files.base.location - base location for temporary
indexing files
index.files.folder.volume - maximum number of files per
directory

# verde tasks files
task.files.base.location base location for Verde task report
files
task.files.folder.volume - maximum number of files per
directory


# loaders input files
sfx.import.files.dir - where SFX input file
acq.import.files.dir - where Acquisition input file
org.import.files.dir - where organization input file
marcit.import.files.dir - where MarcIt input file is

# URL path
resource.path - defines Verde task URL context prefix
task.report.path=@@resource.path@@report

workflowCustomization.path - directory where

task.report.emailPath=/verde/web/notSafeAction/download.do
# Verde processes charectaristics
process.indexingProcess.port=3001
process.taskManager.port=3002
process.local_sfx.port=3003
```

```
# mail configuration
sender.name=verdetaskReport - default sender name when Verde
is sending emails

# mail server
smtp.server= il-ex01.Corp.Exlibrisgroup.com - defines the
local SMTP server

# ftp server
ftp.user=ckbview - defines the FTP user name for LKB update
process
```

Verde Log Files

Verde log files are stored in two folders: one for client-side logs and the other for server-side logs. The client-side log folder can be accessed using the command `vclog`. The server-side log folder can be accessed using the command `vlog`.

All the log files use a **rolling-file** mechanism. This mechanism may work by size or by date. A **rolling by date** log file mechanism opens a new file for every logging date. This means that you will see files, such as `LKBUpdate.log.2009-02-19`, that contain data from the specified date. A **rolling by size** log file mechanism opens a new log file when the file reaches a certain size. In this case, the file name will usually include an additional digit, such as `server.log.1`.

The main log files contained in `vclog` are:

- `acqConvert.log` – Relates to Aleph interoperability and contains information from the converting stage of the acquisition import process
- `acqLoad.log` – Relates to Aleph interoperability and contains information from the loading stage of the acquisition import process
- `client.log` – The default client-side log file, which contains all the logs of various clients that do not have specific log files
- `cleanup.log` – The log of the cleanup task
- `collectorMatchesTask.log` – The log of the task that builds sets for the Transfer Local Data screen
- `eProductActivation.log` – The synchronization log (shows actions from the Verde side)
- `indexing.log` – Contains data regarding the indexing process
- `LKBUpdate.log` – Contains information about the KnowledgeBase update process
- `LKBupdatePreprocessing.log` – Shows what parameters were defined for the LKB Update
- `localSfxLoad.log` – Contains information about the localization process

- `tasks.log` – Contains detailed data regarding tasks such as tasks, loaders, reports, and so forth
- `TaskManagerLog.txt` – Monitors the state of the Task Manager
- `utilIndexing.log` – Contains information regarding the use of UTIL V/12 and the re-indexing of the database. For information on this utility, refer to the *Verde Staff User's Guide*.
- `workflow.log` – Contains all logs relating to the workflows

The main log files contained in `vlog` are:

- `server.log` – The default server-side log file
- `localhost_access_log` – Contains the HTTP request log records

6

The Verde License

This section includes:

- **License Description** on page **25**
- **License Example** on page **26**

License Description

The Verde license is described in a file deployed with Verde. Its role is to enforce the Verde license terms.

The file is read and validated during system initialization and system operation.

The main properties of the file are:

- `<issued-by>` – The entity that issued the license
- `<issue-date>` – The date on which the license was issued
- `<days-to-expiration>` – The number of days until license expiration (starting from the issue date)
- `<max-instances>` – The maximum number of instances to be created in Verde
- `<max-users>` – The maximum number of users that can be defined in Verde
- `<concurrent-users-allowed>` – The maximum number of concurrent users allowed in the system
- `<session-timeout>` – The session timeout, in minutes

CAUTION:
If this file is edited, the system will not be able to validate it.

License Example

```
<!-- DO NOT EDIT THIS FILE -->
<license>
  <issued-by>exlibris</issued-by>
  <!-- issued date must be in format yyyy/MM/dd -->
  <issue-date>2008/11/28</issue-date>
  <!-- a nagative value will represents non expired lisence --
>
  <days-to-expiration>10000</days-to-expiration>
  <max-instances>15</max-instances>
  <max-users>100</max-users>
  <concurrent-users-allowed>50</concurrent-users-allowed>
  <session-timeout>30</session-timeout>

  </license>
```

7

Preventive Maintenance and Recovery

This section includes:

- **Overview** on page **27**
- **Ensuring Sufficient Database Space** on page **28**
- **Working in Archive Log Mode** on page **30**
- **Backing Up Your System** on page **30**
- **Disk Cleanup** on page **34**
- **Reviewing the Oracle Alert Log** on page **34**
- **Recovery Policy** on page **35**
- **Summary of Periodic Maintenance Activities** on page **36**

Overview

This chapter discusses the maintenance activities that are necessary to ensure that the Verde application runs smoothly. It is recommended that the system administrator and/or the DBA onsite read this chapter thoroughly before performing the tasks described.

A summary of the various maintenance activities can be found in **Summary of Periodic Maintenance Activities** on page **36**. The following is a list of the activities that are described in detail in this chapter:

- Ensuring sufficient database space – See **Ensuring Sufficient Database Space** on page **28**.
- Using archive mode and ensuring sufficient disk space for archiving – See **Working in Archive Log Mode** on page **30**.
- Backing up all system/application components – See **Backing Up Your System** on page **30**.
- Performing disk cleanup – See **Disk Cleanup** on page **34**.

- Reviewing Oracle alert logs – See **Reviewing the Oracle Alert Log** on page **34**.

Ensuring Sufficient Database Space

For Verde and the Oracle database to function properly, you must ensure that sufficient database space exists.

This section describes:

- **Tablespace Free Space**
- **Database Temporary Tables**

Tablespace Free Space

For the Oracle database to function properly, there must be free space in the various tablespaces, which are logical storage units made up of physical datafiles. Use UTIL O to view the amount of free and utilized space. Make sure that the free space of each tablespace equals the larger of the following: at least 10 percent of the tablespace or the minimum size listed below.

The following are the important tablespaces used by Verde:

- TS0 – Free space should be at least 2 G (and resized in 2 G chunks).
- TS1 – Free space should be at least 1 G (and resized in 1 G chunks).
- TSLOB – Free space should be at least 500 MB (and resized in 500 MB chunks).

If you need to add datafiles to a tablespace, use UTIL O. For information on UTIL O, see **Oracle Management Utility (UTIL O)** on page **43**.

A typical maintenance check may include the following steps:

1. Check the free space that exists in the tablespace, using UTIL O. To manage the tablespace, choose option **14** (Database Free/Used Space) and then **1** (All Tablespaces Free Space Summary). This enables you to view how much free tablespace (TOTAL_FREE_SPACE column) exists in the TS0, TS1, and TSLOB tablespaces (others are less relevant for Verde).
2. Add space to TS1. Go back to the UTIL O menu and choose option **13** (Database Files). You can then choose either option **1** (List of Database Files) or option **3** (Add File to Tablespace). Choose option 3 to add a datafile to TS1.
   a. Provide the `VERDE_DBA` user name/password. (They are identical and should be entered with a slash delimiter.)
   b. Enter the tablespace name, `TS1`.

c Specify a new datafile name. The system provides you with a list of files that exist in TS1 and asks you to provide the name for the new file (whereby a physical file is created on the disk to be used for tablespace purposes). Usually, the files are numbered using a specific convention. You should use the same convention, changing only the last number in the sequence. For example, if the last file in the list is `/exlibris2/oradata/verde4/verde4_ts1_05.dbf`, you would specify `/exlibris2/oradata/verde4/verde4_ts1_06.dbf` as the new file name.

d Enter the file size (in MB). For TS1, type `1000`. (You would type `2000` for TS0 and `500` for TSLOB.)

e Confirm your settings.

The additional space is allocated by Oracle to the tablespace. Note that this process can take several minutes.

Database Temporary Tables

The Verde application creates and uses Oracle tables for temporary data. These tables need to be cleaned up periodically using the special cleanup utility. This utility is activated automatically by the system. For more information on this utility, see **Cleanup Utility (UTIL C)** on page **83**.

Manual Cleanup

If you have encountered missing database space problem, we recommend removing the unnecessary files manually. Possible options include the following:

- vclog - Old client log files
- vlog - Old server log files

NOTE:
Deleting old client and server log files leave no files for future analysis, which may affect the level of support. If there is enough disk spice, we recommend not to remove them. If the disk must be cleaned up, we suggest keeping old files for at least 30 days.

- Backup files – Mostly DB backups (usually under /exibris/oradata), sometimes software backups (usually under ${VERDE_BASE}/..)
- forSync – Especially after massive updates , such as re-indexing the whole DB or KB updates. A server restart is needed if the whole directory contents are removed.
- vimport - All the KB update input files that have already been applied can be deleted (they are not needed and will not be downloaded from the FTP server anymore). Other large input files (named tmp*) can be deleted also.

- /exlibris/ftp_from_exlibris - This directory contains SW downloaded from the FTP server for Verde installation. The content of this directory can be removed once the customer confirms the installation.

Working in Archive Log Mode

In Oracle, all transactions made to the database are saved in special files called redo logs, which function in a cyclical manner. When all redo logs are full, the first one is reused and its original contents are overwritten. Archive log mode is a mechanism designed to preserve all the content of the redo logs. When you work in archive log mode, all redo log files are saved to a designated directory and their content can be used in recovery scenarios.

To ensure the smooth operation of the system, enough disk space must be available at all times for the archived redo logs. Make sure that the archive directory is on a disk with enough space for several days of work. The archived redo logs can be deleted only after they are backed up. By deleting the backed up archive files, space is freed for the new ones being created.

It is crucial to activate the archive mechanism prior to switching to production. If archive log mode was deactivated for some reason, a full database backup must be performed immediately upon archive log mode reactivation.

Backing Up Your System

Sufficient backup of your Verde application components is crucial for recovery scenarios. It is recommended that you use third-party tools and/or custom-made scripts to handle these backups.

This section describes:

- **Components to Back Up**
- **Backup Strategy**

Components to Back Up

The following application components require backup:

- **Database** – Backing up the database datafiles is also known as physical backup. There are two types of database backup:
  - **Cold backup** – Involves backing up all the database files to tape/disk while the database is down. Take the list of database files to back up from the database data dictionary before shutting down the database.

Because the database (and Verde) is down while the backup is being performed, no library activity can be stored during this time.

Note that complete recovery of the database to the time of the backup can always be performed. To restore the database to the point in time prior to the failure, all archive files that were generated from the time of the backup to the time of the failure must be available.

- **Hot backup** – Involves backing up all database files (except redo log files) to tape/disk while the database continues to run. Hot backups can be performed only if the database is in archive log mode. Take the list of database files to back up from the database data dictionary. Because the database (and Verde) continues to operate normally while the backup is being performed, normal library activity can take place during this time. However, large batch jobs should not be running during the course of the hot backup.

  Note that recovery from hot backups can be performed only if archive files exist. Assuming all archive files are available and are in sequence, the recovery will be performed to the point in time prior to the failure.

- **Archive files** – Archived redo log files are backed up to tape/disk. When recovering from hot backups, use archive files to enable the recovery. When recovering from cold backups, use archive files to minimize data loss and restore the database up to the point in time prior to the failure.
- **Data** – Backing up the data of an Oracle database is also known as logical backup. Oracle table content is extracted to disk and is backed up to tape/disk. You can perform this backup using the Oracle export utility or via special export utilities provided in your version of Verde.
- **Software** – There are two types of software that must be backed up:
  - **Verde software** – The Verde application software must be backed up to tape/disk. All Verde installations reside in the `/exlibris/verde` directory. Each installation is in a specific directory named `v2_i`, where `i` refers to the installation number. For example, on a machine with two Verde installations, the file system will have two directories: `/exlibris/verde/v2_1` and `/exlibris/verde/v2_2`. The backup of `/exlibris/verde/v2_i` contains Verde software, temporary files, and user configuration files.
  - **Oracle software** – The Oracle application software must be backed up to tape/disk.
- **Site configuration** – The file structure of the libraries, including exported data, must be backed up. Back up the `/exlibris/verde` directory that contains global configuration for all site-specific libraries. Note that a backup can be performed with our without export. For more information, see **Backup Strategy** on page **32**.

Backup Strategy

Once you have an understanding of the components, as well as how and how often they are modified, you can set up a backup plan. With the exception of the Oracle database, the components are basically directories and files. The more frequently they are backed up, the more up-to-date any recovered data will be in the event of a crash. This will reduce the chance of data loss to a minimum.

As mentioned in the previous section, there are two types of backup: physical and logical. Physical backup refers to backing up the database files. Logical backup refers to backing up the data extracted from the database tables. Physical backup can be performed using one of two methods: cold or hot backup. Cold backup is performed while the database is down. Hot backup is performed while the database is running. A hot backup can be performed only when the database is running in archive log mode.

Cold backup has an advantage over hot backup in that a database can be recovered from a cold backup to its state at the time of the backup, with no need for additional files. If there are archived redo logs from after the point in time at which the cold backup was performed, they can be applied. By applying these archived redo logs, the database can be brought up to date with minimum data loss, or none at all. The hot backup must be restored together with the archived redo logs in order to synchronize the database. Recovery from a hot backup itself, without archived redo logs, is not possible.

It is recommended that you use the following backup policy:

- Cold backup – Unless downtime is a major issue, this can run daily.
- Hot backup – For any day on which a cold backup is not run
- Archive – Run daily
- Application configuration – Run daily
- Export – As frequently as possible
- Verde application – Once every two months and after each upgrade or patch
- Oracle application – Once every two months and after each upgrade or patch

Backup Strategy Guidelines

In general, the more frequently you perform backups, the less likely you are to experience data loss. As described in the previous sections, there is a difference between backing up the database as files (that is, physical backup) and backing up the extracted data from the database tables (that is, logical backup). In addition, there are directories/files that are not related to the database that require backup as well (for example, the library structures).

The ideal backup strategy would involve running a cold backup of the database daily, as well as backing up the archived redo logs, the site configuration, and

the Verde software on a daily basis. This would ensure that if there is a need for recovery, recovery can be performed from the previous night's backup.

Sites that cannot afford (due to the downtime involved) to run a cold backup every night, should run a hot backup each night on which a cold backup cannot be run. This will also enable recovery from the previous night's backup.

Sites that cannot run a full backup every night (neither cold nor hot) should make every effort to minimize the time intervals between full backups. For these sites, the role of the archived redo logs is critical in restoring a full backup performed several days before and reapplying transactions to bring the database up to date.

IMPORTANT:
To perform hot backups you must have archived redo logs, regardless of the frequency with which the hot backup is performed.

The ability to restore an up-to-date file of Verde software and system information depends on the frequency with which the backup is performed. Note that specific user-configurable files reside in the `/exlibris/verde/v2_i/verde/home/profile` directory and are generally the files that are most frequently modified. The `/exlibris/verde/v2_i/verde/home/data` directory contains the following subdirectories: `licenseAttachments`, `import`, `reports`, and `verdeTask`. These directories contain important files that are either generated by Verde or uploaded by users. Backups of these directories should be performed more often than backups of other directories.

In addition to performing a backup, you must read the backup tapes to check their validity. Run a listing of a full backup tape at least once a week. Aside from verifying that the tape is valid as far as the media is concerned, check the listing and make sure that all expected directories/files were backed up. Do not rely solely on the backup mechanism.

Backup Strategy Examples

This section contains a chart with examples of backup strategies and their abbreviations. Make sure that you are familiar with all the components and that you devise a comprehensive backup methodology.

- C.A.S.E. – Cold + Archived redo logs + Site configuration + Export
- H.A.S.E. – Hot + Archived redo logs + Site configuration + Export
- A.S.E. – Archived redo logs + Site configuration + Export
- T.V. – Tape Validity check

- B.I. – Backup Integrity check

Table 5. Backup Strategies

Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday	Weekly
C.A.S.E.	C.A.S.E.	C.A.S.E.	C.A.S.E.	C.A.S.E.			T.V. + B.I.
C.A.S.E.	H.A.S.E.	C.A.S.E.	H.A.S.E.	C.A.S.E.			T.V. + B.I.
A.S.E.	H.A.S.E.	A.S.E.	A.S.E.	C.A.S.E.			T.V. + B.I.

Disk Cleanup

File systems tend to fill up with temporary files, logs, and various other material that can be deleted periodically. The system administrator should ensure that the file system does not reach a very high or full capacity.

Cleanup should be performed after backup and not beforehand.

Application Cleanup

Cleanup is performed by UTIL C. All related utilities are run automatically. For information on UTIL C, see **Cleanup Utility (UTIL C)** on page **83**.

Oracle Log Cleanup

Trace files and the alert log are generated by Oracle in the `$ORACLE_BASE/admin/$ORACLE_SID` directory. The following subdirectories can be found in this directory and may be cleaned up periodically:

- `bdump` – background process traces and the alert log
- `cdump` – core dumps
- `udump` – user traces

Reviewing the Oracle Alert Log

When one of the server or background Oracle processes detects an error, it dumps information about the error into a trace file.

In addition, each database has an `alert_<sid>.log` file. The database alert file is a chronological log of messages and errors. Messages include information about administrative operations performed on the database, tablespaces, and rollback segments, as well as errors (such as the lack of database space).

The trace files and alert log are located in the `$ORACLE_BASE/admin/$ORACLE_SID` directory. UTIL O can be used to review the alert log. For information on UTIL O, see **Oracle Management Utility (UTIL O)** on page **43**.

Recovery Policy

Performing recovery from crash scenarios is very complex and requires deep knowledge and extensive expertise. Recovery, therefore, should be performed only by a well-trained DBA. In addition, it is recommended that the DBA consult with Ex Libris Support before performing database recovery. The following are the reasons for this approach:

- Any given problem may have more than one solution. If you consult Ex Libris, you can expect to receive a wide range of solutions.
- Ex Libris has global support for Oracle and has a close relationship with Oracle. Ex Libris discusses and analyzes every crash scenario with Oracle before presenting the solution to the customer.
- Ex Libris has experienced DBAs who can offer suitable solutions, taking into account different perspectives of the application.

In the event of a database crash:

- Contact Ex Libris Support by CRM/PRB and by phone/e-mail.
- If the crash is detected while the database is up, do not shut down the database. This may cause an irreversible problem. In addition, although the database is corrupt, some critical information that is essential for recovery may be retrieved.
- It is extremely important to back up the corrupt scenario “as is,” while the database is down, to enable multiple solution attempts.
- Discuss the crash and its status with Ex Libris personnel. Make sure to review what backups were performed, the symptoms of the crash, and any log or additional information that is relevant to the occurrence.
- Send the Oracle alert log and trace files to Ex Libris Support.
- Prepare media that may be necessary for recovery (for example, backup tapes).

Summary of Periodic Maintenance Activities

Table 6. Periodic Maintenance Activities

Activity	Recommended Time Interval	Method
Clean file system space	Weekly or more frequently (as needed)	Online, using UTIL C or by script after a backup
Ensure free database space	Weekly	UTIL O/14
Back up database	Varies per site. See **Backup Strategy** on page **32** for recommendations.	Third-party tool or customer script
Back up site configuration	Varies per site. See **Backup Strategy** on page **32** for recommendations.	Third-party tool or customer script
Back up software	Once a month, or after each upgrade	Third-party tool or customer script
Review backup media	Weekly	Third-party tool or customer script

8

Oracle Overview

This section includes:

- **Oracle Structure** on page **37**
- **Oracle Concepts** on page **39**
- **Oracle Table Management – file_list** on page **41**
- **Oracle Utilities** on page **42**

Oracle Structure

Verde is based on the Oracle 9i RDBMS (Relational Database Management System).

A typical Verde installation includes a number of Oracle objects that are built in the Verde schema. Verde uses the ORACLE*TEXT package and includes some ORACLE*TEXT objects in the CTX schema.

SQL Access to the Oracle Tables

In Verde, SQL *Plus can be used to access Verde's Oracle tables.

```
ram11-16(1) USM50>>sqlplus verde

Enter password:
SQL*Plus: Release 9.2.0.3.0 - Production on Mon Oct 27
09:52:03 2003

Copyright (c) 1982, 2002, Oracle Corporation.  All rights
reserved.
```

```
Connected to:
Oracle9i Enterprise Edition Release 9.2.0.3.0 - 64bit
Production
With the Partitioning, OLAP and Oracle Data Mining options
JServer Release 9.2.0.3.0 - Production

SQL-VERDE> desc task;
 Name                                      Null?    Type
 ----------------------------------------- -------- ---------
-------------------
 ID                                        NOT NULL NUMBER(19)
 TASKTYPE                                  NOT NULL
VARCHAR2(255)
 NAME                                           VARCHAR2(255)
 IMPLCLASS                                      VARCHAR2(255)
 S_YEAR                                            NUMBER(10)
 S_MONTH                                           NUMBER(10)
 S_DAY                                             NUMBER(10)
 S_HOUR                                            NUMBER(10)
 S_MINUTES                                         NUMBER(10)
 E_YEAR                                            NUMBER(10)
 E_MONTH                                           NUMBER(10)
 E_DAY                                             NUMBER(10)
 E_HOUR                                            NUMBER(10)
 E_MINUTES                                         NUMBER(10)
 INTERVAL                                          NUMBER(10)
 RUNONHOLIDAYS                                      NUMBER(1)
 LASTRUNTIME                                       NUMBER(19)
 SCHED_RUNMONTHLY                                   NUMBER(1)
 SCHED_DAYOFMONTH                                  NUMBER(10)
 SCHED_HOUR                                        NUMBER(10)
 SCHED_MINUTES                                     NUMBER(10)
 SCHED_RUNONSUNDAY                                  NUMBER(1)
 SCHED_RUNONMONDAY                                  NUMBER(1)
 SCHED_RUNONTUESDAY                                 NUMBER(1)
 SCHED_RUNONWEDNESDAY                               NUMBER(1)
 SCHED_RUNONTHURSDAY                                NUMBER(1)
 SCHED_RUNONFRIDAY                                  NUMBER(1)
 SCHED_RUNONSATURDAY                                NUMBER(1)
 SCHED_SCHEDULEDTYPE                              VARCHAR2(30)
```

Oracle Users in Verde

There are several Oracle users used by the Verde application that are not related to a specific library:

- `VERDE` — The Verde application server connects to the Oracle databases via a designated Oracle user named `VERDE` (default password: `VERDE`). The `VERDE` user can select, insert, update, and delete data from the tables of the Verde schema.
- `VERDE_ADMIN` — The administrative user `VERDE_ADMIN` has additional privileges to those of the `VERDE` user and can also create, drop, and alter

Oracle tables, indexes, users, triggers, and so forth. The VERDE_ADMIN Oracle user is used for these actions in all Verde procedures.

- VERDE_DBA — The VERDE_DBA administrative Oracle user is the most privileged Oracle administrative user. This user is used by Verde utilities to start up, shut down, and perform other DBA operations.
- VERDE_RO — The VERDE_RO (Verde Read Only) Oracle user is the least privileged Oracle user and can view data only. This user was created for customers who want to be able to view data.

Passwords

Verde contains an encrypted file with the passwords of the Oracle users used by Verde (such as VERDE, VERDE_ADMIN, VERDE_DBA). This means that when you decide to change the password of an Oracle user, the password must be changed both in the Oracle database and in the Verde password file.

Oracle Concepts

The following Oracle concepts are discussed in this section:

- **Storage**
- **Users**
- **Tables**
- **Indexes**
- **Triggers**

Storage

An Oracle database consists of several logical units named tablespaces. Each tablespace consists of one or more physical datafiles, which can be stored on one or more disks. For example:

Table 7. Tablespaces Used in Verde

Tablespace Name	Usage	Physical File
system	Oracle system tables	/exlibris/oradata/verde1/system01.dbf
temp	Temporary space (for sorting, index creation, and so forth)	/exlibris2/oradata/verde1/temp01.dbf

Table 7. Tablespaces Used in Verde

Tablespace Name	Usage	Physical File
ts0	Verde tables	/exlibris2/oradata/verde1/ts0_0.dbf
ts1	Verde indexes	/exlibris/oradata/verde1/ts1_0.dbf

Each Oracle table and index is mapped to a tablespace. In Verde, this mapping is done via a configuration file named `file_list`. For more information, see **Oracle Table Management – file_list** on page **41**.

Users

In an Oracle database, users can be defined and identified by user names. A user has the following tablespaces:

- Default tablespace – Specifies where objects (tables and indexes) are built by default (unless otherwise specified).
- Temporary tablespace – Provides storage for SQL statements that require disk space to sort or summarize data.

Tables

A table is an Oracle object that contains rows of data. A row is composed of columns. Each table is mapped to a tablespace. For each table, Oracle allocates initial space and extended space, according to the specifications in the `CREATE TABLE` command.

Table mapping to a tablespace, and its initial space allocation are controlled by the `file_list` configuration file. The size of additional extent allocation also appears in the `file_list` file for reasons of backward compatibility. For more information, see **Oracle Table Management – file_list** on page **41**.

Indexes

An Oracle index is an Oracle object (B-tree) that contains pointers (rowid) to a specific row in a table.

Triggers

A database trigger is a stored PL/SQL block that is associated with a table. Oracle automatically executes a trigger when a specified SQL statement is issued on the table. The trigger can be executed before or after the SQL statement is issued on the table. Once the trigger has been created, it can be enabled (will be executed automatically in the case of a specific event) or disabled (is defined but will not be executed).

Oracle Table Management – file_list

The Oracle tablespaces used for the Verde application must be locally managed tablespaces.

There are two types of extent allocation when using locally managed tablespaces.

- **Auto Allocate** – Oracle takes full control, automatically allocating extents as needed and taking into account the initial allocation of the table/index as supplied in the `CREATE TABLE/INDEX` command. In a standard Verde installation, `tso` and `ts1` are created as locally managed tablespaces with the auto allocate allocation type.
- **Uniform** – When creating the tablespace, the DBA determines the uniform extent size for all the extents in the tablespace. Each extent is this size. The DBA determines which table is assigned to which tablespace, depending on the table size. All the extents of a table created in a locally managed tablespace with uniform size are of the same size. This size is the uniform size defined for the tablespace, regardless of the extents definition that may have been given in the `CREATE TABLE` command. In this way, there is no fragmentation and the utilization is optimal. For example, when creating a tablespace with a uniform extent size of 10 MB and a table that is 50 MB, 5 extents are used.

  When working with locally managed tablespaces, the word **LOCAL** appears in the **EXT_MGMT** (extent management) column in UTIL O/17/4 – Show Tablespace Definitions.

  For example:

```
TS_NAME      EXT_MGMT    ALLOC_TYP    INIT_EXT    NEXT_EXT  TYPE  STAT
----------   ----------  ---------  ----------  ----------  ----  ----
LOG          LOCAL       SYSTEM          65536              PERM  ONL
SYSAUX       LOCAL       SYSTEM          65536              PERM  ONL
SYSTEM       LOCAL       SYSTEM          65536              PERM  ONL
TEMP         LOCAL       UNIFORM       1048576     1048576  TEMP  ONL
TS0          LOCAL       SYSTEM          65536              PERM  ONL
TS1          LOCAL       SYSTEM          65536              PERM  ONL
TSLOB        LOCAL       UNIFORM       8388608     8388608  PERM  ONL
UNDOTBS1     LOCAL       SYSTEM          65536              UNDO  ONL
USERS        LOCAL       UNIFORM         40960       40960  PERM  ONL
```

In the **ALLOC_TYP** column, you may see the following values:

- `SYSTEM` = auto allocate
- `UNIFORM` = uniform

Oracle Utilities

There are various Verde online utilities that can be run on the Verde schema. Each utility is used for a different task. For example, the UTIL O – Oracle Management utility deals with the various Oracle objects (tablespaces, listener, files, and so forth).

Access to the Verde utilities is activated by the `util` command from the command-line prompt. The utilities main screen appears when this command is executed:

```
  c   Cleanup
  o - Oracle utilites
  v - Verde administration
  s - Verde Statuses
  e - Exit

Please select [exit]:
```

9

Oracle Management Utility (UTIL O)

This section includes:

- **Overview** on page **43**
- **Oracle Server (UTIL O/1)** on page **44**
- **Oracle Listener (UTIL O/2)** on page **47**
- **Oracle Logs (UTIL O/3)** on page **50**
- **NLS (UTIL O/6)** on page **51**
- **Archiving (UTIL O/7)** on page **51**
- **Database Users (UTIL O/9)** on page **53**
- **SQL*Plus Session (UTIL O/10)** on page **56**
- **Database Verification Utility (UTIL O/12)** on page **57**
- **Database Files (UTIL O/13)** on page **59**
- **Database Free/Used Space (UTIL O/14)** on page **63**
- **Manage Database Links (UTIL O/16)** on page **67**
- **Database Tablespaces (UTIL O/17)** on page **69**
- **Oracle Statistics (UTIL O/18)** on page **73**
- **Shared Pool (UTIL O/19)** on page **77**
- **Multi Threaded Server (UTIL O/20)** on page **79**

Overview

The Oracle Management utility (UTIL O) manages the database operations, such as activating/deactivating the Oracle server, creating Oracle users, and so forth.

When you enter o on the Verde utilities main screen, the following menu appears:

```
    O. Managing ORACLE
    ------------------
         0. Exit Procedure
         1. Oracle Server
         2. Oracle Listener
         3. Oracle Logs
         4.
         5.
         6. Nls
         7. Archiving
         8.
         9. Database Users
        10. SQL*Plus Session
        11.
        12. Database Verification Utility
        13. Database Files
        14. Database Free/Used Space
        15.
        16. Database Links
        17. Database Tablespaces
        18. Oracle Statistics
        19. Shared Pool
        20. Multi Threaded Server
Please select [exit]:
```

NOTE:
The Oracle Listener must be running on the database server in order for Verde to work properly.

Oracle Server (UTIL O/1)

To display the Oracle Server menu, enter option 1 from the Managing Oracle menu.

```
O.1.  Oracle Server
     -------------------
         0.  Exit Procedure
         1.  Activate Oracle Server
         2.  Close Oracle Server
         3.  Show Running Oracle Server
         4.  Show Oracle Server Status
Please select [exit]:
```

Activate Oracle Server (UTIL O/1/1)

NOTE:
This requires the VERDE_DBA user name and password.

In order for Verde to interact with Oracle, the Oracle server and Oracle Listener must be running. They can be started automatically at boot time (this is determined during installation), and can also be controlled by the Verde Oracle Management utilities under UTIL O.

To activate the Oracle server:

1 Enter option 1 on the Oracle Server menu. The following prompt displays:

```
To continue you will need to enter VERDE_DBA username/password.
Username/password: verde_dba/<verde_dba password>
```

2 Enter the VERDE_DBA user name and password.

Close Oracle Server (UTIL O/1/2)

NOTE:
The requires the VERDE_DBA user name and password.

This utility shuts down the Oracle server immediately by activating the Oracle shutdown immediate option. All the clients connected to the server are logged out immediately.

To close the Oracle server:

1 Enter option 2 on the Oracle Server menu. The following prompt displays:

```
Do you want to restart Oracle Server after closing? yes/[no]
```

2 Enter yes if you want to restart the server immediately after shutdown.

NOTE:
If you enter no, the server shuts down and does not restart. In order to restart it later on, you will need to select UTIL O/1/1 – Activate Oracle Server.

The following prompt displays:

```
To restart Oracle server enter VERDE_DBA username/password.
Username/password: verde_dba/<verde_dba password>
```

3 Enter the VERDE_DBA user name and password.

Show Running Oracle Server (UTIL O/1/3)

This utility displays the background processes, dispatchers, and shared servers used by your Oracle instance (database).

The following is an example of an Oracle server that is up and running:

```
oracle  5200     1  0   Sep 25 ?        0:00 ora_arc1_verde1
oracle  5162     1  0   Sep 25 ?        0:02 ora_pmon_verde1
oracle  5194     1  0   Sep 25 ?        0:00 ora_d006_verde1
oracle  5166     1  0   Sep 25 ?        1:47 ora_lgwr_verde1
oracle  5186     1  0   Sep 25 ?        0:00 ora_d002_verde1
oracle  5164     1  0   Sep 25 ?        0:16 ora_dbw0_verde1
oracle  5184     1  0   Sep 25 ?        0:00 ora_d001_verde1
oracle  5188     1  0   Sep 25 ?        0:00 ora_d003_verde1
oracle  5170     1  0   Sep 25 ?        1:34 ora_smon_verde1
oracle  5172     1  0   Sep 25 ?        0:00 ora_reco_verde1
oracle  5180     1  0   Sep 25 ?        0:00 ora_s003_verde1
oracle  5176     1  0   Sep 25 ?        0:00 ora_s001_verde1
oracle  5168     1  0   Sep 25 ?        0:23 ora_ckpt_verde1
oracle  5178     1  0   Sep 25 ?        0:00 ora_s002_verde1
oracle  5196     1  0   Sep 25 ?        0:00 ora_d007_verde1
oracle  5190     1  0   Sep 25 ?        0:00 ora_d004_verde1
oracle  5198     1  0   Sep 25 ?        0:00 ora_arc0_verde1
oracle  5192     1  0   Sep 25 ?        0:00 ora_d005_verde1
oracle  5174     1  0   Sep 25 ?        0:23 ora_s000_verde1
oracle  5182     1  0   Sep 25 ?        0:00 ora_d000_verde1
```

Note that the processes that appear on your server may differ slightly from the lines presented in the example.

If these lines do not appear, the Oracle server may be activated using UTIL O/1/1 – Activate Oracle Server.

Show Oracle Server Status (UTIL O/1/4)

This utility displays the status of the Oracle server. For example:

```
Connected.
INSTANCE_N  HOST_NAME   VERSION     STARTUP_TI  STATUS    LOGINS
----------  ----------  ----------  ----------  --------  ----------
verde1      keret       9.2.0.2.0   22-SEP-03   OPEN      ALLOWED

BANNER
----------------------------------------------------------------
Oracle9i Enterprise Edition Release 9.2.0.2.0 - 64bit Production
PL/SQL Release 9.2.0.2.0 - Production
CORE    9.2.0.2.0       Production
TNS for Solaris: Version 9.2.0.2.0 - Production
NLSRTL Version 9.2.0.2.0 - Production
```

Oracle Listener (UTIL O/2)

To display the Oracle Listener menu, enter option 2 from the Managing Oracle menu.

```
O.2.  Oracle Listener
     ---------------------
         0.  Exit Procedure
         1.  Activate Oracle Listener
         2.  Close Oracle Listener
         3.  Show Running Oracle Listener
         4.  Show Listener Status
         5.  Show Listener Services
Please select [exit]:
```

When a user process makes a connection request using a connect string, the Oracle Listener process examines the request and connects it to a server process. Both the Oracle server and the Oracle Listener must be running. They can be started automatically at boot time (this is determined during installation), and can also be controlled by the Verde Oracle Management utilities under UTIL O.

Activate Oracle Listener (UTIL O/2/1)

NOTE:
This requires a UNIX Oracle software owner password.

To activate the Oracle Listener:

1 Enter option `1` on the Oracle Listener menu. The following prompt displays:

```
To continue you will need to enter Oracle's password.
Password:
```

2 Enter the Oracle password.

Close Oracle Listener (UTIL O/2/2)

NOTE:
This requires an Oracle software owner password.

To close the Oracle Listener:

1 Enter option `2` on the Oracle Listener menu. The following prompt displays:

```
Do you want to restart Oracle Listener after closing? yes/[no]
```

2 Enter `yes` or `no`. The following prompt displays:

```
To close Oracle Listener enter Oracle's password.
Password:
```

3 Enter the Oracle password.

Show Running Oracle Listener (UTIL O/2/3)

This utility displays the active process for the Oracle Listener. For example:

```
oracle  3408   1  0   Sep 22 ? 0:00 /exlibris/app/oracle/product/920/
bin/tnslsnr LISTENER -inherit
```

Show Listener Status (UTIL O/2/4)

This utility displays the status of the Oracle Listener. For example:

```
LSNRCTL for Solaris: Version 9.2.0.2.0 - Production on 23-SEP-2003
11:15:30

Copyright (c) 1991, 2002, Oracle Corporation.  All rights reserved.

Connecting to (DESCRIPTION=(address=(protocol=ipc)(key=extproc)))
STATUS of the LISTENER
------------------------
Alias                     LISTENER
Version                   TNSLSNR for Solaris: Version 9.2.0.2.0 -
Production
Start Date                22-SEP-2003 09:23:48
Uptime                    1 days 1 hr. 51 min. 42 sec
Trace Level               off
Security                  OFF
SNMP                      OFF
Listener Parameter File   /exlibris/app/oracle/product/920/network/
admin/listene
r.ora
Listener Log File         /exlibris/app/oracle/product/920/network/log/
listener.
log
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=extproc)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=verde1)))
Service "verde1_services" has 1 instance(s).
 Instance "verde1", status READY, has 9 handler(s) for this service...
The command completed successfully
```

Show Listener Services (UTIL O/2/5)

This utility displays the Oracle Listener services that are running. For example:

```
LSNRCTL for Solaris: Version 9.2.0.4.0 - Production on 04-NOV-2003
10:17:15

Copyright (c) 1991, 2002, Oracle Corporation.  All rights reserved.

Connecting to (ADDRESS=(PROTOCOL=tcp)(PORT=1521))
Services Summary...
Service "verde1.exlibris.int.il" has 1 instance(s).
  Instance "verde1", status READY, has 9 handler(s) for this service...
    Handler(s):
      "D007" established:0 refused:0 current:0 max:972 state:ready
         DISPATCHER <machine: ram14, pid: 27080>
         (ADDRESS=(PROTOCOL=ipc)(KEY=#27080.1))
      "D006" established:0 refused:0 current:0 max:972 state:ready
         DISPATCHER <machine: ram14, pid: 27078>
         (ADDRESS=(PROTOCOL=ipc)(KEY=#27078.1))
      "D005" established:0 refused:0 current:0 max:972 state:ready
         DISPATCHER <machine: ram14, pid: 27076>
         (ADDRESS=(PROTOCOL=ipc)(KEY=#27076.1))
      "D004" established:0 refused:0 current:0 max:972 state:ready
         DISPATCHER <machine: ram14, pid: 27074>
         (ADDRESS=(PROTOCOL=ipc)(KEY=#27074.1))
```

Oracle Logs (UTIL O/3)

To display the Oracle Logs menu, enter option `3` from the Managing Oracle menu.

```
O.3. Oracle Logs
     ----------------
         0.  Exit Procedure
         1.  View Oracle ALERT LOG
Please select [exit]:
```

View Oracle ALERT LOG (UTIL O/3/1)

You are prompted for the number of lines to display from the Oracle alert log. The displayed lines are the most recent ones.

NLS (UTIL O/6)

To display the NLS menu, enter option 6 from the Managing Oracle menu.

```
 O.6.  NLS
     ---------
         0.  Exit Procedure
         1.  Show NLS Parameters
Please select [exit]:
```

Show NLS Parameters (UTIL O/6/1)

Verde uses the UTF-8 character set. This utility shows the NLS (National Language Support) definition of the database. For example:

```
PARAMETER                                VALUE
=========================================================
NLS_LANGUAGE                             AMERICAN
NLS_TERRITORY                            AMERICA
NLS_CURRENCY                             $
NLS_ISO_CURRENCY                         AMERICA
NLS_NUMERIC_CHARACTERS                   .,
NLS_CALENDAR                             GREGORIAN
NLS_DATE_FORMAT                          DD-MON-RR
NLS_DATE_LANGUAGE                        AMERICAN
NLS_CHARACTERSET                         UTF8
NLS_SORT                                 BINARY
NLS_TIME_FORMAT                          HH.MI.SSXFF AM
NLS_TIMESTAMP_FORMAT                     DD-MON-RR HH.MI.SSXFF AM
NLS_TIME_TZ_FORMAT                       HH.MI.SSXFF AM TZR
NLS_TIMESTAMP_TZ_FORMAT                  DD-MON-RR HH.MI.SSXFF AM TZR
NLS_DUAL_CURRENCY                        $
NLS_NCHAR_CHARACTERSET                   UTF8
NLS_COMP                                 BINARY
NLS_LENGTH_SEMANTICS                     BYTE
NLS_NCHAR_CONV_EXCP                      FALSE
```

Archiving (UTIL O/7)

To display the Archiving menu, enter option 7 from the Managing Oracle menu.

```
 O.7. Archiving
     --------------
         0.  Exit Procedure
         1.  Turning Archiving On
         2.  Turning Archiving Off
         3.  Show Archiving Status
Please select [exit]:
```

About Archiving

Verde backup and recovery procedures are based on Oracle. In order to have the complete ability to recover data up to the time of failure, the Oracle database should be in archive log mode. This ensures full recovery up to the time of failure. Hot backup cannot be done if the database is not in archive log mode. For more information on the components to be backed up as well as the backup strategy to be used, see **Backing Up Your System** on page **30**.

There are some preliminary actions that need to be performed before UTIL O/7 can be used. For further information, refer to the *Oracle Backup* and *Ex Libris Backup Package* documents.

NOTE:
Changing the archiving mode shuts down the database and restarts it in archive log mode.

When you change the archive log mode, the following occurs:

1. Verde processes (servers and batch procedures) are stopped (using the `verde_shutdown` script in `$VERDE_BIN`).
2. The Oracle database is shut down.
3. The Oracle database is started up.
4. Verde is restarted (using the `verde_startup` script in `$VERDE_BIN`).

NOTE:
Performing a full cold backup after switching to archive log mode is mandatory because otherwise, there is a gap in archive log files which prevents recovery.

Turning Archiving On (UTIL O/7/1)

This utility shuts down the database and restarts it in archive log mode. It requires the `VERDE_DBA` user name and password.

The process involved is described above.

Turning Archiving Off (UTIL O/7/2)

This utility shuts down the database and restarts it in non-archive log (NOARCHIVELOG) mode. It requires the `VERDE_DBA` user name and password.

The process involved is described above.

Show Archiving Status (UTIL O/7/3)

This utility displays the archiving status and requires the VERDE_DBA user name and password.

After entering the user name and password, you can see a display such as the following if archiving is off:

```
SQL*Plus: Release 9.2.0.2.0 - Production on Tue Sep 23 11:25:42 2003

Copyright (c) 1982, 2002, Oracle Corporation.  All rights reserved.

SQL> Connected.
SQL> SQL> Database log mode            No Archive Mode
Automatic archival             Enabled
Archive destination            /exlibris/oradata/verde1/arch
Oldest online log sequence     313
Current log sequence           317
SQL> Disconnected from Oracle9i Enterprise Edition Release 9.2.0.2.0 -
64bit Pro
duction
With the Partitioning, OLAP and Oracle Data Mining options
JServer Release 9.2.0.2.0 - Production
```

When archiving is on, you can see that the database log mode is set to **archive mode** and **automatic archival** is **enabled**.

In a production database, the database log mode should always be set to **archive mode**.

Database Users (UTIL O/9)

To display the Database Users menu, enter option 9 from the Managing Oracle menu.

```
O.9. Database Users
    -------------------
       0. Exit Procedure
       1. List Database Users
       2. Create a New User
       3. Update Password for User
Please select [exit]:
```

List Database Users (UTIL O/9/1)

This utility shows the list of all the users that exist in the database.

For example

```
The Database verde1 Contains the Following Users:
=================================================
SYS
SYSTEM
OUTLN
DBSNMP
VERDE
VERDE_ADMIN
CTXSYS
PERFSTAT
VERDE_BACKUP
VERDE_DBA
```

Create a New User (UTIL O/9/2)

This utility creates a new user and gives it a default password, which is the same as the user name.

NOTE:
If the name of the user already exists, all of the tables and data belonging to this user are dropped, and the user is created with all of its tables empty.

To create a new user:

1 Enter option `2` on the Database Users menu. The following prompt displays:

```
Enter User Name to Create New User:
```

2 Enter the new user name. The following prompt displays:

```
enter yes to create oracle user <new user name>
```

3 Enter `yes` to continue. The following prompt displays:

```
default password is <new user name>
if user <new user name> exists all data will be erased!!!
enter no to reconfirm
```

4 Enter `no` to reconfirm. The following displays:

```
source create_ora_user_b newabc
create_ora_user_b newabc

SQL*Plus: Release 9.2.0.2.0 - Production on Tue Sep 23
11:32:22 2003
```

```
Copyright (c) 1982, 2002, Oracle Corporation.  All rights
reserved.
Enter user-name:
Connected to:
Oracle9i Enterprise Edition Release 9.2.0.2.0 - 64bit
Production
With the Partitioning, OLAP and Oracle Data Mining options
JServer Release 9.2.0.2.0 - Production

SQL-VERDE_ADMIN> EXIT
Disconnected from Oracle9i Enterprise Edition Release
9.2.0.2.0 - 64bit Producti
on
With the Partitioning, OLAP and Oracle Data Mining options
JServer Release 9.2.0.2.0 - Production

SQL*Plus: Release 9.2.0.2.0 - Production on Tue Sep 23
11:32:23 2003

Copyright (c) 1982, 2002, Oracle Corporation.  All rights
reserved.

Enter user-name:
Connected to:
Oracle9i Enterprise Edition Release 9.2.0.2.0 - 64bit
Production
With the Partitioning, OLAP and Oracle Data Mining options
JServer Release 9.2.0.2.0 - Production
SQL-VERDE_ADMIN> DROP USER newabc CASCADE
          *
ERROR at line 1:
ORA-01918: user 'NEWABC' does not exist

User created.

Grant succeeded.

Disconnected from Oracle9i Enterprise Edition Release
9.2.0.2.0 - 64bit Product
on
With the Partitioning, OLAP and Oracle Data Mining options
JServer Release 9.2.0.2.0 - Production
```

5 Press ENTER to return to the menu.

Update Oracle Passwords of Verde Users (UTIL O/9/3)

This utility enables you to change the password of an existing Verde user.

To update the password of a Verde user:

1. Shut down Verde using UTIL V/2.
2. Enter option `3` on the Database Users menu. The following prompt displays:

```
Enter user name for password updating:
```

3. Enter the user name to update. The following prompt displays:

```
Enter old password:
```

4. Enter the current password. The following prompt displays:

```
After updating VERDE password Verde application will be
restarted - including set_globals script

Are you sure you want to proceed? (Y/[N]):
```

5. Enter `y` to confirm. The following prompt displays:

```
Enter new password:
```

6. Enter the new Verde user password and retype it when prompted.
7. Log off from the current UNIX session.
8. Log on to the current UNIX session.
9. Start Verde using UTIL V/1.

SQL*Plus Session (UTIL O/10)

In Oracle 9i, SQL*Plus is used to perform sysdba operations. This utility runs `sqlplus/nolog`. You can then connect as sysdba using the sysdba user, `VERDE_DBA`.

This utility starts an Oracle SQL*Plus session as sysdba, as follows:

```
***** DO: connect verde_dba/<verde_dba password>_passwd as
sysdba

SQL*Plus: Release 9.2.0.2.0 - Production on Tue Sep 23
11:36:03 2003

Copyright (c) 1982, 2002, Oracle Corporation.  All rights
reserved.

SQL> connect verde_dba as sysdba
Enter password: <verde_dba password>
Connected.
SQL>
```

Type `exit` to end your SQL session.

Database Verification Utility (UTIL O/12)

To display the Database Verification Utility menu, enter option `12` from the Managing Oracle menu.

```
 O.12. Database Verification Utility
    ----------------------------------
       0. Exit Procedure
       1. Run Database Verification Utility
       2. Find Corrupted Object
Please select [exit]:
```

Run Database Verification Utility (UTIL O/12/1)

NOTE:
This requires an Oracle password.

This procedure verifies that all the Oracle datafiles are fully readable and accessible. It is advisable to run it periodically for all database files.

To run the database verification utility:

1 Enter option `1` on the Database Verification Utility menu. The following prompt displays:

```
Select one of the oracle files:
```

2 Enter the Oracle file, such as `/exlibris/oradata/verde1/verde1_ts0_01.dbf`. The following prompt displays:

```
Enter database block size [8192]:
```

3 Enter the database block size or press `ENTER` to use the default. The following prompt displays:

```
To continue you will need to enter oracle's password.
Password:
```

4 Enter the Oracle password. The following displays:

```
Sun Microsystems Inc.   SunOS 5.9       Generic May 2002
Execute oracle cshrc

DBVERIFY: Release 9.2.0.3.0 - Production on Tue Nov 4 10:28:14
2003

Copyright (c) 1982, 2002, Oracle Corporation.  All rights
reserved.

DBVERIFY - Verification starting : FILE = /exlibris3/oradata/
verde1/verde1_ts0_01.dbf
..........................................................
........
..........................................................
..........................................................
.................................
DBVERIFY - Verification complete

Total Pages Examined         : 254720
Total Pages Processed (Data) : 180916
Total Pages Failing   (Data) : 0
Total Pages Processed (Index): 0
Total Pages Failing   (Index): 0
Total Pages Processed (Other): 5819
Total Pages Processed (Seg)  : 0
Total Pages Failing   (Seg)  : 0
Total Pages Empty            : 67985
Total Pages Marked Corrupt   : 0
Total Pages Influx           : 0
```

5 Press `ENTER` to return to the menu.

Find Corrupted Object (UTIL O/12/2)

If the Run Database Verification utility (UTIL O/12/1) indicates that corrupt blocks were found, use this utility to identify the Oracle objects that reside in the corrupted blocks.

NOTE:
This routine should be activated by a DBA only.

To identify the Oracle objects that reside in the corrupted blocks:

1 Enter option `2` on the Database Verification Utility menu. The following prompt displays:

```
Enter datafile name:
```

2 Enter the datafile name. The following prompt displays:

```
Enter block number:
```

3 Enter the block number.

Database Files (UTIL O/13)

To display the Database Files menu, enter option `13` from the Managing Oracle menu.

```
O.13. Database Files
     --------------------
         0.  Exit Procedure
         1.  List of Database Files
         2.  Resize Oracle Datafile
         3.  Add File to Tablespace
         4.  Show Datafile Free Blocks by KBytes
         5.  Show Datafile Free Blocks by BlockID
Please select [exit]:
```

List of Database Files (UTIL O/13/1)

This utility lists the Oracle datafiles and their sizes.

For example:

```
The Database verde1 contains the following files:
=================================================

T        NAME                                          SIZE K  F
-------  --------------------------------------------  -------  ---
DRSYS    /exlibris3/oradata/verde1/verde1_drsys_01.dbf    86016  7
SYSTEM   /exlibris3/oradata/verde1/verde1_system_01.dbf  266240  1
TOOLS    /exlibris3/oradata/verde1/verde1_tools_01.dbf   204800  3
TS0      /exlibris3/oradata/verde1/verde1_ts0_01.dbf     525312  4
TS0      /exlibris2/oradata/verde1/verde1_ts0_02.dbf    2048000  8
TS1      /exlibris3/oradata/verde1/verde1_ts1_01.dbf     525312  5
TS1      /exlibris2/oradata/verde1/verde1_ts1_02.dbf    1228800  9
TSLOB    /exlibris3/oradata/verde1/verde1_tslob_01.dbf    51200  6
TSLOB    /exlibris3/oradata/verde1/verde1_tslob_02.dbf    30720  10
UNDOTS   /exlibris3/oradata/verde1/verde1_undots_01.dbf  524288  2

T        NAME                                          SIZE K  F
-------  --------------------------------------------  -------  ---
TEMP     /exlibris3/oradata/verde1/verde1_temp_01.dbf   1024000  1
TEMP     /exlibris3/oradata/verde1/verde1_temp_02.dbf     61440  2
```

Resize Oracle Datafile (UTIL O/13/2)

NOTE:
This requires the VERDE_DBA user name and password.

This utility is used to enlarge or reduce the size of an Oracle datafile. You are prompted for the datafile name and the new size.

To resize an Oracle datafile:

1 Enter option 2 on the Database Files menu. The following prompt displays:

```
To resize a database file enter VERDE_DBA username/password.
username/password:
```

2 Enter the VERDE_DBA user name and password. The following prompt displays:

```
Enter Tablespace name:
```

3 Enter the tablespace name.

4 Enter the name of the datafile to resize (specify full path).

5 Enter the new size in MB.

6 Enter `y` to confirm.

Add File to Tablespace (UTIL O/13/3)

NOTE:
This requires the `VERDE_DBA` user name and password.

Tablespaces are composed of one or more datafiles. When a tablespace does not have enough free space, it needs to be enlarged. This may be done by adding new files or by resizing existing files (see **Database Tablespaces (UTIL O/17)** on page **69**).

To add a file to a tablespace:

1 Enter option `3` on the Database Files menu. The following prompt displays:

```
To add a file to a tablespace enter VERDE_DBA username/
password.
username/password:
```

2 Enter the `VERDE_DBA` user name and password. The following prompt displays:

```
Enter Tablespace name:
```

3 Enter the tablespace name. For example, the following prompt displays if `TS1` is specified:

```
Tablespace TS1 consists of the following files:

/exlibris/oradata/verde1/ts1_0.dbf
/exlibris/oradata/verde1/ts1_1.dbf
/exlibris/oradata/verde1/ts1_2.dbf

Enter new file name:
```

4 Enter the name of the datafile to add (specify full path).

5 Enter the file's size (in MB).

6 Enter `y` to confirm.

Show Datafile Free Blocks by KBytes (UTIL O/13/4)

This utility displays the Free Blocks Report. For example:

```
Free Blocks Report by Kbytes
========================================

TABLES  F    BLOCK_ID  KBYTES   NAME
------  ---  --------  ------   ----------------------------------
TS1     9    65161     707520   /exlibris2/oradata/verde1/verde1_ts
                                1_02.dbf

UNDOTS  2    2953      500672   /exlibris3/oradata/verde1/verde1_un
                                dots_01.dbf

TS0     8    227465    228288   /exlibris2/oradata/verde1/verde1_ts
                                0_02.dbf

TOOLS   3    2869      181856   /exlibris3/oradata/verde1/verde1_to
                                ols_01.dbf

TABLES  F    BLOCK_ID  KBYTES   NAME
------  ---  --------  ------   ----------------------------------
TS0     8    209673    110592   /exlibris2/oradata/verde1/verde1_ts
                                0_02.dbf

DRSYS   7    617       81088    /exlibris3/oradata/verde1/verde1_dr
```

Show Datafile Free Blocks by BlockID (UTIL O/13/5)

This utility displays the free blocks for a specified tablespace and datafile.

To show the free blocks for a datafile:

1. Enter option 5 on the Database Files menu. The following prompt displays:

```
Enter Tablespace name:
```

2. Enter the tablespace name. The following prompt displays:

```
Datafile Number:
```

NOTE:
To determine the value of the datafile number, use the Show Datafile Free Blocks by KBytes menu (UTIL O/13/4).

3 Enter the datafile number. A report such as the following displays:

```
BLOCK_ID       BYTES
----------  ----------
    249865    39780352
    248729      917504
```

4 Press ENTER to return to the menu.

Database Free/Used Space (UTIL O/14)

This utility provides information about the tablespaces' free space.

To display the Database Free/Used Space menu, enter option 14 from the Managing Oracle menu.

```
O.14. DataBase Free/Used Space
       ------------------------
       0. Exit procedure
       1. All Tablespaces free space summary
       2. Number of Free extents by size in a Tablespace
       3. All Free extents of min size in a Tablespace
       4. Space Used by a Library/Libraries in each Tablespace
       5. Space Used by a Group of Libraries in each Tablespace
       6. Coalesce Contiguous Free Extents
       7.
       8. Verde Tablespaces Total/Free/Used space report
       9. Clean Temporary Tablespace Free Storage
Please select [exit]:
```

All Tablespaces Free Space Summary (UTIL O/14/1)

This utility displays details about the database free space in the Oracle DBA_FREE_SPACE table. It is important to review this report periodically in order to prepare additional resources for the database.

This report contains the following columns:

- TABLESPACE_NAME – The tablespace's name.
- TOTAL_FREE_SPACE – The total amount of free space in the tablespace (in megabytes).
- MAX_EXTENT – The size of the largest contiguous extent of the tablespace (in megabytes).
- NUM_FREE_EXTENTS – The number of free extents in the tablespace.

NOTE:
If a tablespace has no free space left, it does not appear in this report.

The following is an example of this report:

```
TABLESPACE_NAME  TOTAL_FREE_SPACE    MAX_EXTENT NUM_FREE_EXTENTS
---------------  -----------------   ----------  ----------------
SYSTEM                    28.273343   28.2733435                 1
TEMP                      200.09429   40.0141679                33
TOOLS                    17.7753893   17.6972214                 2
TS0                      425.632244   118.549496               754
TS1                      444.236214   147.565435               118
USERS                     4.9949313    4.9949313                 1
```

Number of Free Extents by Size in a Tablespace (UTIL O/14/2)

This utility lists the number of extents per size (truncated in megabytes) in the tablespace.

To show the number of free extents in a tablespace:

1 Enter option `2` on the Database Free/Used Space menu. The following prompt displays:

```
Enter Tablespace name:
```

2 Enter the tablespace name. For example, the following displays if `TS0` is specified:

```
SIZE IN MB  NUM OF EXTENTS
----------  --------------
       879               1
         0               4

Standard input: END
```

In the above example, there are five free extents. Four of them are less than 1 MB and one of them is 879 MB.

3 Press `ENTER` to return to the menu.

All Free Extents of Minimum Size in a Tablespace (UTIL O/14/3)

This utility lists the exact size (in megabytes) of all free extents that are larger than a given size.

To show the number of free extents per minimum size:

1 Enter option `3` on the Database Free/Used Space menu. The following prompt displays:

```
Enter Tablespace name:
```

2 Enter the tablespace name. The following prompt displays:

```
Enter Min size (MB) of free extent [0=ALL]:
```

3 Enter the minimum size (MB). For example, the following prompt displays if `0` is specified:

```
EXTENT_SIZE
-----------
 879.428397
 .156335878
 .117251908
 .117251908
 .117251908
Standard input: END
```

Since the minimum size entered is 0, this example lists the exact sizes of all the free extents in tablespace `TS0`.

4 Press `ENTER` to return to the menu.

Space Used by a Library/Libraries in Each Tablespace (UTIL O/14/4)

This utility shows, for each library, the amount of space that the library occupies in each tablespace. If a truncated library name is used, all the libraries starting with the given characters are listed and the occupied space is listed for each one of them.

To show the space used by libraries in each tablespace:

1 Enter option `4` on the Database Free/Used Space menu. The following prompt displays:

```
Enter Library name (full or truncated, e.g. usm):
```

2 Enter the library name (full or truncated). For example, the following displays if `verde` is specified:

```
OWNER           TABLESPACE_NAME                  SIZE_MB
------------------------------------------------------------
VERDE                  TS0                    4099.87713
VERDE                  TS1                    290.159389
VERDE                  TSLOB                  32.0175878
```

3 Press `ENTER` to return to the menu.

Space Used by a Group of Libraries in Each Tablespace (UTIL O/14/5)

This utility shows the total amount of space that all the libraries whose names start with the given characters occupy in each tablespace.

To show the space used by a group of libraries in each tablespace:

1 Enter option `5` on the Database Free/Used Space menu. The following prompt displays:

```
Enter first 3 characters of Library code (e.g. usm):
```

2 Enter the first three characters of the library code. For example, the following displays if `usm` is specified:

```
TABLESPACE_NAME                   SIZE_MB
------------------------------  ----------
TS0                            1514.76959
TS1                            848.466076
```

3 Press `ENTER` to return to the menu.

Coalesce Contiguous Free Extents (UTIL O/14/6)

This utility is no longer needed when using a locally managed tablespace. It remains only for backward compatibility and will be removed in future versions.

NOTE:
This utility requires the `VERDE_DBA` user name and password.

Database free space may be composed of extents of various sizes. It is worthwhile to use this procedure to coalesce the contiguous free extents in an attempt to create larger free extents. Perform this procedure periodically.

To coalesce contiguous free extents:

1 Enter option 6 on the Database Free/Used Space menu. The following prompt displays:

```
To Coalesce Tablespaces enter VERDE_DBA username/password.
username/password:
```

2 Enter the VERDE_DBA user name and password .

NOTE:
The procedure coalesces extents only for tablespaces TS0 and TS1.

Verde Tablespaces Total/Free/Used Space Report (UTIL O/14/8)

This utility provides details about the database free and used space in the Oracle tablespaces that is in use for the Verde application, as well as the size used by each Verde library (in MB) in each tablespace. For example:

```
NAME                           TOTAL SIZE M
------------------------------ ------------
TS0                                    2513
TS1                                    1713
TSLOB                                    80

NAME                           TOTAL FREE M
------------------------------ ------------
TS0                                     414
TS1                                     723
TSLOB                                    72

NAME                           TOTAL USED M
------------------------------ ------------
TS0                                    2098
TS1                                     989
```

Clean Temporary Tablespace Free Storage (UTIL O/14/9)

In rare cases, the temporary tablespace does not free non-used pages quickly enough. This utility allows you to free these pages manually.

Manage Database Links (UTIL O/16)

The Manage Database Links utilities are used to manage one logical database throughout the network from more than one physical database. They are used

when there are some Oracle tables that are on one or more separate servers, and not on the Verde application server.

To display the Manage Database Links menu, enter option `16` from the Managing Oracle menu.

```
 O.16. Manage Database Links
    --------------------------
       0. Exit Procedure
       1. List Database Links
       2. Create Database Link
       3. Drop Database Link
Please select [exit]:
```

List Database Links (UTIL O/16/1)

This utility lists the existing database links. An empty list is returned if no links exist.

Create Database Links (UTIL O/16/2)

This utility creates a new database link.

NOTE:
Database links are used when working with tables on a remote database.

To create a new database link:

1. Enter option `2` on the Manage Database Links menu. The following prompt displays:

```
Enter Database Link Name (only one word):
```

2. Enter the name of the new database link. The following prompt displays:

```
Enter oracle TNS service name for remote database:
```

3. Enter the name of the Oracle TNS service (`<hostname>.<SID>`) as defined in the `$ORACLE_HOME/network/admin/tnsnames.ora` file.

   If the network service is defined, the following prompt displays:

```
Enter username to remote system [verde]
```

NOTE:
If the network service is not defined in the configuration file, an error message appears and you are not able to create the new link.

4 Enter the user name. The following prompt displays:

```
Enter password to remote system [verde passwd]
```

5 Enter the password. The database link is created and a message such as the following appears (in this example, the TNS service name is `ram40.verde1` and the user name is `verde`):

```
Now creating a private database link to remote user verde,
If the remote database's verde password is changed
in the remote location, then
this database link should be recreated!

drop database link ram40.verde1
                   *
ERROR at line 1:
ORA-02024: database link not found

Database link created.
```

Note that this utility drops the link and then creates it. Therefore, if this is the first time a link is created, the following error message can be ignored:

```
ERROR at line 1:
ORA-02024: database link not found
```

Drop Database Link (UTIL O/16/3)

This utility is used to drop a database link when it is no longer needed. You are prompted for the name of the database link to drop.

Database Tablespaces (UTIL O/17)

To display the Database Tablespaces menu, enter option `17` from the Managing Oracle menu.

```
O.17. Manage Database Tablespaces
       ---------------------------
      0. Exit procedure
      1. Create a Tablespace
      2. List Tablespace Files
      3. Check Tablespace for a library based on file_list template
      4. Show Tablespaces Definition
      5. Show Tablespace Allocated/Free/Used Space
```

Create a Tablespace (UTIL O/17/1)

One rarely needs to create a tablespace, since all necessary tablespaces are created during system installation. This utility is used if there is a need for an additional tablespace. You can read about tablespaces and their types in **Oracle Table Management – file_list** on page **41**.

To create a tablespace:

1 Enter option `1` on the Database Tablespaces menu. The following prompt displays:

```
To Create a new Tablespace, Enter VERDE_DBA username/
password.
username/password:
```

1 Enter the `VERDE_DBA` user name and password.The following prompt displays:

```
Enter Tablespace name:
```

2 Enter the new tablespace name. The following prompt displays:

```
Enter new file name (full path):
```

3 Enter the new datafile, including the full path. The following prompt displays:

```
Enter new file size (MB):
```

4 Enter the file size (MB). The following prompt displays:

```
============================================================
Tablespaces can be created with a UNIFORM size for all extents
or with allocation type AUTOALLOCATE which means
Oracle will decide how to define extents
Util o 17 4 can be used to see current definitions
for existing tablespaces
============================================================
Tablespace Allocation Type : [AUTO/UNIFORM]
```

5 Enter the allocation type (`uniform` or `auto`). The following prompt displays:

```
UNIFORM SIZE : [128K/1M/4M/128M/1920M]
```

NOTE:
If you select `auto`, Oracle decides how to define extents. Use UTIL O/17/4 to view the current definitions for existing tablespaces.

6 Specify one of the uniform sizes: `128K`, `1M`, `4M`, `128M`, or `1920M`. Lines similar to the following display:

```
Tablespace:  TEST
File:        /exlibris/oradata/verde1/test_01.dbf
File size:    1000MB
Allocation : UNIFORM SIZE 4m
confirm (y/[n]):
```

7 Enter `y` to confirm.

List Tablespace Files (UTIL O/17/2)

This utility enables you to view a list of the files in a tablespace.

To list a tablespace's files:

1 Enter option `2` on the Database Tablespaces menu. The following prompt displays:

```
Enter Tablespace name:
```

2 Enter the tablespace name. Lines similar to the following display:

```
Tablespace TS1 consists of the following files:

NAME                                                SIZE K  F
--------------------------------------------------  ----------  ---
/exlibris/oradata/verde1/verde1_ts1_01.dbf            525312  5
/exlibris/oradata/verde1/verde1_ts1_02.dbf           1228800  9
```

3 Press `ENTER` to return to the menu.

Check Tablespace for a Library Based on file_list Template (UTIL O/17/3)

This utility verifies the existence of tablespaces for a library based on the library's type and size.

Show Tablespaces Definition (UTIL O/17/4)

This utility shows the following for each tablespace:

- the types of extent management
- segment allocation tablespace (for permanent or temporary objects or for undo segments)
- the tablespace status (online or offline)

For example:

```
TS_NAME    EXT_MGMT    ALLOC_TYP    INIT_EXT    NEXT_EXT  TYPE  STAT
------  ----------  ---------  ----------  ----------  ----  ----
DRSYS      LOCAL       SYSTEM         65536              PERM  ONL
SYSTEM     LOCAL       SYSTEM         65536              PERM  ONL
TEMP       LOCAL       UNIFORM      1048576     1048576  TEMP  ONL
TOOLS      LOCAL       UNIFORM        32768       32768  PERM  ONL
TS0        LOCAL       SYSTEM         65536              PERM  ONL
TS1        LOCAL       SYSTEM         65536              PERM  ONL
TSLOB      LOCAL       UNIFORM      8388608     8388608  PERM  ONL
UNDOTS     LOCAL       SYSTEM         65536              UNDO  ONL
```

Show Tablespace Allocated/Free/Used Space (UTIL O/17/5)

This utility shows a given tablespace's total tablespace size, amount of free space, and amount of used space.

To display space details for a tablespace:

1 Enter option `5` on the Database Tablespaces menu. The following prompt displays:

```
Enter Tablespace name:
```

2 Enter the tablespace name. Lines such as the following display:

```
Enter Tablespace name : TS1
Tablespace TS1 :

_______________________________
TOTAL SIZE M
------------
        1713

TOTAL FREE M
------------
         723

TOTAL USED M
------------
         989
```

3 Press `ENTER` to return to the menu.

Oracle Statistics (UTIL O/18)

This utility displays Oracle statistics. To display the Oracle Statistics menu, enter option `18` from the Managing Oracle menu.

```
 O.18. Oracle Statistics
    ----------------------
       0. Exit Procedure
       1. Performance Statistics
       2. Rollback Segments Definitions
       3. Rollback Segments Dynamic Allocation
       4. View Long Operations
       5. IO Statistics
       6. Sort Operations
       7. Table Statistics
Please select [exit]:
```

Performance Statistics (UTIL O/18/1)

Option **1** on the Oracle Statistics menu allows you to display the performance statistics. For example:

```
opened cursors current
       254
db block gets
    281077
consistent gets
   3854704
NAME
     VALUE
physical reads
     36747
physical writes
     17228
DBWR checkpoints
         2
NAME
     VALUE
redo log space requests
         2
sorts (memory)
      7063
sorts (disk)
         4
=====================================================
DATA DICTIONARY CACHE (shared_pool_size)
      GETS      MISSES
RATIO
     79930        3224
95.97%
=====================================================
LIBRARY CACHE (shared_pool_size)
EXECUTIONS      MISSES
LIBCACHEPROZ
    265828         321
99.88%
```

Rollback Segments Definitions (UTIL O/18/2) and Rollback Segments Dynamic Allocation (UTIL O/18/3)

In Oracle 9i, there is a new feature called Automatic Undo Management (AUM), which simplifies and automates the management of undo segments. When AUM is used, the rollback segments are defined and managed by Oracle and called UNDO segments.

NOTE:
UTIL O/18/2 and UTIL O/18/3 are still available for backward compatibility.

When you enter option 2 on the Oracle Statistics menu, a message such as the following displays:

```
All Rollback Segments
Segm Name        Ownr  In TabSpace
File containing header of rbs
SYSTEM           Priv  SYSTEM
/exlibris/oradata/verde1/verde1_system_01.dbf
_SYSSMU1$        Publ  UNDOTS
/exlibris/oradata/verde1/verde1_undots_01.dbf
_SYSSMU2$        Publ  UNDOTS
.
.
.

Online Rollback Segments:
Name     NrEx  Size         Init  Next  PctI  MinE  MaxE Opt   Stat
                                                        size

SYSTEM      6  376K      114,688                1  ####       OnL
_SYSSMU     3  1144K     131,072                2  ####       OnL
_SYSSMU     3  1144K     131,072                2  ####       OnL
_SYSSMU     4  2168K     131,072                2  ####       OnL
_SYSSMU     3  1144K     131,072                2  ####       OnL
_SYSSMU     3  1144K     131,072                2  ####       OnL
_SYSSMU     3  1144K     131,072                2  ####       OnL
_SYSSMU     3  1144K     131,072                2  ####       OnL
_SYSSMU     3  1144K     131,072                2  ####       OnL
_SYSSMU     3  184K      131,072                2  ####       OnL
_SYSSMU     3  1144K     131,072
```

When you enter option 3 on the Oracle Statistics menu, a message such as the following displays:

```
NAME   EXT   RSSIZE      WRITES  SHRN    AVGSHR  WRAPS  CUREXT  WAITS
SYST     6   385024        7012     0         0      1       3      0
_SYS     4  2220032     1801856     1   1048576      8       2      0
_SYS     3  1171456     1500476     0         0      6       2      0
_SYS     4   253952     1624992     1    327680     28       2      0
_SYS     4   253952     1870178     2    294912     32       3      0
_SYS     4   253952     1642924     2    196608     29       2      0
_SYS     3  1171456     1426884     0         0      5       2      0
_SYS     3  1171456     1865062     1   1048576      5       2      0
_SYS     3  1171456     1462950     1   1048576      6       2      0
_SYS     3  1171456     1613910     0         0      6       2      0
_SYS     3  1171456     1851606     0         0      6       2      0
```

View Long Operations (UTIL O/18/4)

This utility displays Oracle long operations, if they occur in the system at the time at which the utility is run. The following information displays:

- SID – Session identifier
- OPNAME – Operation name

- TARGET – The object on which the operation is being performed
- DONE SO FAR – The percentage of work already done

Use **<CTRL> + C** to stop the display.

IO Statistics (UTIL O/18/5)

This utility displays the I/O statistics. The following information displays:

- BLOCK_GETS – Block gets for this session
- CONSISTENT_GETS – Consistent gets for this session
- PHYSICAL_READS – Physical reads for this session
- BLOCK_CHANGES – Block changes for this session
- CONSISTENT_CHANGES – Consistent changes for this session

Use **<CTRL> + C** to stop the display.

For example:

```
BLOCK_GETS CONSISTENT_GETS PHYSICAL_READS BLOCK_CHANGES CONSISTENT_CHANGES

5360720    16859217       153081         4975011        15065
5364774    16869224       153118         4978618        15065
5368592    16877862       153138         4981732        15065
5375440    16891538       153180         4986583        15065
5378493    16898409       153196         4988801        15065
```

Sort Operations (UTIL O/18/6)

This utility displays sort operations if they occur in the system when the utility is running.

Use **<CTRL> + C** to stop the display.

Table Statistics (UTIL O/18/7)

This utility rebuilds the Oracle system statistics for the ctxsys, system, and verde Oracle schemas.

Shared Pool (UTIL O/19)

To display the Shared Pool menu, enter option `19` from the Managing Oracle menu.

```
 O.19. Shared Pool
    -----------------
       0. Exit Procedure
       1. Show SGA Buffers
       2. Flush Shared Pool
Please select [exit]:
```

Show SGA Buffers (UTIL O/19/1)

This utility shows the various SGA buffers. For example:

```
NAME                          BYTES
--------------------------  ----------
fixed_sga                     730424
buffer_cache                167772160
log_buffer                    787456
errors                        13864
enqueue                       408944
KGK heap                      7000
KQR L PO                      1228856
KQR M PO                      514224
KQR S SO                      3880
sessions                      608400
sql area                      10699040

NAME                          BYTES
--------------------------  ----------
1M buffer                     2098176
KGLS heap                     2456504
PX subheap                    177176
parameters                    7448
free memory                 117650816
transaction                   399696
PL/SQL DIANA                  688568
FileOpenBlock                 1563248
PL/SQL MPCODE                 204712
library cache                 8620560
miscellaneous                 8793184
```

```
NAME                           BYTES
-------------------------  ----------
KGSK scheduler                108552
KGSKI schedule                 21008
MTTR advisory                  55312
PLS non-lib hp                  2088
joxs heap init                  4240
kgl simulator                1315752
sim memory hea                166272
table definiti                 15384
trigger defini                 55360
trigger inform                  1040
trigger source                  1216

NAME                           BYTES
MTTR advisory                  55312
PLS non-lib hp                  2088
joxs heap init                  4240
kgl simulator                1315752
sim memory hea                166272
table definiti                 15384
trigger defini                 55360
trigger inform                  1040
trigger source                  1216

NAME                           BYTES
-------------------------  ----------
Checkpoint queue              1026560
VIRTUAL CIRCUITS               470080
dictionary cache              3229952
KSXR receive buffers          1034000
message pool freequeue         940944
KSXR pending messages que      853952
event statistics per sess     2325600
fixed allocation callback         552
PX msg pool                   1966080
free memory                  65142784
```

Flush Shared Pool (UTIL O/19/2)

This utility flushes the shared pool. You are prompted for the `VERDE_DBA` user name and password.

Multi Threaded Server (UTIL O/20)

In a standard Oracle configuration, a separate server process is created on behalf of each user process. It is called a dedicated server process (or shadow process), because it acts only on behalf of the associated user process.

Oracle also supports the Shared Server Architecture (or Multi Threaded Server Architecture - MTS) in which there are several server processes, each serving several user processes.

In Verde, the MTS infrastructure exists in the database, but is implemented only in special cases, in coordination with Ex Libris.

To display the Multi Threaded Server menu, enter option `20` from the Managing Oracle menu.

```
O.20. Multi Threaded Server
     --------------------------
         0.  Exit Procedure
         1.  Show MTS Parameters
         2.  Show Listener Services
Please select [exit]:
```

Show MTS Parameters (UTIL O/20/1)

This utility shows the MTS parameters. For example:

```
VERDE_DBA/VERDE_DBA

SQL*Plus: Release 9.2.0.2.0 - Production on Tue Sep 23
12:51:25 2003

Copyright (c) 1982, 2002, Oracle Corporation.  All rights
reserved.

SQL> Connected.
SQL> SQL>
NAME                                 TYPE         VALUE
------------------------------------ -----------  --------
mts_circuits                         integer      225
mts_dispatchers                      string
(PROTOCOL=TCP)(DISPATCHERS=4),
 (PROTOCOL=IPC)(DISPATCHERS=4)
mts_listener_address                 string
mts_max_dispatchers                  integer      30
```

```
NAME                                TYPE         VALUE
----------------------------------  -----------  --------
mts_max_servers                     integer      30
mts_multiple_listeners              boolean      FALSE
mts_servers                         integer      4
mts_service                         string       verde1
mts_sessions                        integer

NAME                                TYPE         VALUE
----------------------------------  -----------  --------
220
SQL> Disconnected from Oracle9i Enterprise Edition Release
9.2.0.2.0 - 64bit Pro
duction
With the Partitioning, OLAP and Oracle Data Mining options
JServer Release 9.2.0.2.0 - Production
```

Show Listener Services (UTIL O/20/2)

This utility displays the listener services. For example:

```
 ============================

Service "meta3" has 1 instance(s).
  Instance "meta3", status UNKNOWN, has 1 handler(s) for this service...
    Handler(s):
Service "meta4" has 1 instance(s).
  Instance "meta4", status UNKNOWN, has 1 handler(s) for this service...
  Service "verde1" has 1 instance(s).
  Instance "verde1", status UNKNOWN, has 1 handler(s) for this service...
    Handler(s):
Service "verde1" has 1 instance(s).
  Instance "verde1", status UNKNOWN, has 1 handler(s) for this service...
    Handler(s):
Service "verde1.exlibris.int.il" has 1 instance(s).
  Instance "verde1", status READY, has 9 handler(s) for this service...
    Handler(s):
      "D007" established:0 refused:0 current:0 max:972 state:ready
         DISPATCHER <machine: keret, pid: 2899>
         (ADDRESS=(PROTOCOL=ipc)(KEY=#2899.2))
      "D006" established:0 refused:0 current:0 max:972 state:ready
         DISPATCHER <machine: keret, pid: 2897>
         (ADDRESS=(PROTOCOL=ipc)(KEY=#2897.1))
      "D005" established:0 refused:0 current:0 max:972 state:ready
         DISPATCHER <machine: keret, pid: 2895>
         (ADDRESS=(PROTOCOL=ipc)(KEY=#2895.1))
      "D004" established:0 refused:0 current:0 max:972 state:ready

--More--
```

```
  "D004" established:0 refused:0 current:0 max:972 state:ready
         DISPATCHER <machine: keret, pid: 2893>
         (ADDRESS=(PROTOCOL=ipc)(KEY=#2893.1))
      "D003" established:0 refused:0 current:0 max:972
state:ready
         DISPATCHER <machine: keret, pid: 2891>
         (ADDRESS=(PROTOCOL=tcp)(HOST=keret)(PORT=32796))
      "D002" established:0 refused:0 current:0 max:972
state:ready
         DISPATCHER <machine: keret, pid: 2884>
         (ADDRESS=(PROTOCOL=tcp)(HOST=keret)(PORT=32795))
      "D001" established:0 refused:0 current:0 max:972
state:ready
         DISPATCHER <machine: keret, pid: 2882>
         (ADDRESS=(PROTOCOL=tcp)(HOST=keret)(PORT=32794))
      "D000" established:0 refused:0 current:0 max:972
state:ready
         DISPATCHER <machine: keret, pid: 2880>
         (ADDRESS=(PROTOCOL=tcp)(HOST=keret)(PORT=32792))
Service "VERDE2" has 1 instance(s).
  Instance "VERDE2", status UNKNOWN, has 1 handler(s) for this
service...
    Handler(s):
Service "VERDE3" has 1 instance(s).
  Instance "VERDE3", status UNKNOWN, has 1 handler(s) for this
service...
    Handler(s):
Service "verde1.exlibris.int.il" has 1 instance(s).
  Instance "verde1", status READY, has 9 handler(s) for this
service...
--More-
Instance "verde1", status READY, has 9 handler(s) for this
service...
    Handler(s):
      "D007" established:0 refused:0 current:0 max:972
state:ready
         DISPATCHER <machine: keret, pid: 3267>
         (ADDRESS=(PROTOCOL=ipc)(KEY=#3267.1))
      "D006" established:0 refused:0 current:0 max:972
state:ready
         DISPATCHER <machine: keret, pid: 3265>
         (ADDRESS=(PROTOCOL=ipc)(KEY=#3265.1))
      "D005" established:0 refused:0 current:0 max:972
state:ready
         DISPATCHER <machine: keret, pid: 3263>
         (ADDRESS=(PROTOCOL=ipc)(KEY=#3263.1))
      "D004" established:0 refused:0 current:0 max:972
state:ready
         DISPATCHER <machine: keret, pid: 3261>
         (ADDRESS=(PROTOCOL=ipc)(KEY=#3261.1))
      "D003" established:0 refused:0 current:0 max:972
```

```
state:ready
         DISPATCHER <machine: keret, pid: 3259>
         (ADDRESS=(PROTOCOL=tcp)(HOST=keret)(PORT=32801))
      "D002" established:0 refused:0 current:0 max:972
state:ready
         DISPATCHER <machine: keret, pid: 3257>
         (ADDRESS=(PROTOCOL=tcp)(HOST=keret)(PORT=32800))
      "D001" established:0 refused:0 current:0 max:972
state:ready
         DISPATCHER <machine: keret, pid: 3255>
         (ADDRESS=(PROTOCOL=tcp)(HOST=keret)(PORT=32799))
```

10

Cleanup Utility (UTIL C)

The Cleanup utility (UTIL C) should be used for the periodic cleanup of Oracle temporary tables and scratch data from the disk.

The Cleanup menu contains the following options:

```
C. Cleanup
    ------------------
         0. Exit
         1. Deletion of temporary files
         2. Deletion of old tasks output
         3. Truncate Oracle text errors
Please select [exit]:
```

Deletion of Temporary Files (UTIL C/1)

Deletes temporary files created for Oracle text indexing from the scratch directory on the disk. Run this utility when indexing has been completed and results have been verified.

Deletion of Old Tasks Output Files (UTIL C/2)

Removes files described in the `verdetaskoutput` Oracle table from the scratch directory on the disk, according to the files' input dates. Run this utility periodically.

Truncate Oracle Text Errors (UTIL C/3)

Truncates the `DR$INDEX_ERROR` Oracle text table. Run this utility periodically.