Oracle® Rdb

Oracle SQL/Services Release 7.2.0.1 Release Notes

December 2006

This document contains release note information specific to Oracle SQL/Services, release 7.2.0.1 for OpenVMS Alpha and HP OpenVMS Industry Standard 64 for Integrity Servers operating systems. Also included in this document are release notes pertaining to OCI Services for Oracle Rdb release 7.2.0.1.

ORACLE

Oracle SQL/Services Release 7.2.0.1 Release Notes

Contents

2 Oracle SQL/Services: Software Errors Fixed

3 Oracle SQL/Services: Known Problems

4 OCI Services for Oracle Rdb: Release Notes

List of Tables

List of Examples

Send Us Your Comments

Oracle SQL/Services Release 7.2.0.1 Release Notes

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this document. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most?

If you find any errors or have any other suggestions for improvement, please indicate the document title and part number, and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail: nedc-doc_us@oracle.com
- FAX: 603.897.3825 Attn: Oracle Rdb
- Postal service:

  Oracle Corporation
  Oracle Rdb Documentation
  One Oracle Drive
  Nashua, NH 03062-2804
  USA

If you would like a reply, please give your name, address, telephone number, and (optionally) electronic mail address.

If you have problems with the software, please contact your local Oracle Support Services.

Preface

Oracle SQL/Services software is a client/server component of Oracle Rdb. Oracle SQL/Services lets you develop client application programs on a variety of desktop and mainframe systems so that you can access Oracle Rdb databases as well as other databases supported by the Oracle Rdb implementation of the SQL standard.

Oracle ODBC Driver for Rdb is available (in a separate kit on a separate CD-ROM) for the following client operating systems: Windows 95, Windows 98, Windows 2000, Windows NT Intel, Windows XP. The Oracle ODBC Driver for Rdb allows ODBC applications on these clients read and write access to Oracle Rdb databases using TCP/IP, DECnet, and Oracle Net.

This manual describes new and changed features; problems fixed in this release; and current problems, restrictions, and other notes.

Intended Audience

These release notes are intended for all users of Oracle SQL/Services and OCI Services for Oracle Rdb (formerly known as SQL*Net for Rdb) and should be read to supplement information contained in the *Oracle SQL/Services Installation Guide*, the *Oracle SQL/Services Server Configuration Guide*, and the *Guide to Using the Oracle SQL/Services Client API*.

To get the most out of this manual, you should be familiar with Oracle SQL/Services, data processing procedures, and basic database management concepts and terminology.

Operating System Information

Information about the versions of the operating system and related software that are compatible with this version of Oracle SQL/Services and OCI Services for Oracle Rdb is included in these release notes (see Section 1.1.1).

Structure

This manual contains five chapters

Chapter 1	Describes the new and changed features of Oracle SQL/Services.
Chapter 2	Describes known software errors fixed in release 7.2.0.1 of Oracle SQL/Services and prior releases.
Chapter 3	Describes problems, restrictions, and workarounds known to exist in Oracle SQL/Services release 7.2.0.1.
Chapter 4	Describes release notes that pertain to OCI Services for Oracle Rdb release 7.2.0.1 and prior releases.
Chapter 5	Describes new features and problems fixed in previous releases.

Related Manuals

For more information on Oracle Rdb and Oracle SQL/Services, see the following manuals in this documentation set, especially the following:

- *Oracle SQL/Services Installation Guide*
- *Oracle SQL/Services Server Configuration Guide*
- *Guide to Using the Oracle SQL/Services Client API*

The *Oracle SQL/Services Release Notes* and the *Oracle Rdb Release Notes* are provided as part of the software kit. Adobe Portable Document Format (.pdf) files for the release notes are available in SYS$HELP

The Oracle Rdb documentation is available on the OTN web site.

Conventions

In this manual, Oracle Rdb refers to Oracle Rdb for OpenVMS software.

HP OpenVMS Industry Standard 64 for Integrity Servers is often referred to as OpenVMS I64.

OpenVMS means both the OpenVMS Alpha and the OpenVMS I64 operating systems.

The SQL interface to Oracle Rdb is referred to as SQL. This interface is the Oracle Rdb implementation of the SQL standard adopted in 1999, in general referred to as the ANSI/ISO SQL standard or SQL:1999. See the *Oracle Rdb Release Notes* for additional information about this SQL standard.

Oracle ODBC Driver for Rdb software is referred to as the ODBC driver.

In examples, an implied carriage return occurs at the end of each line, unless otherwise noted. You must press the Return key at the end of a line of input.

Often in examples the prompts are not shown. Generally, they are shown where it is important to depict an interactive sequence exactly; otherwise, they are omitted in order to focus full attention on the statements or commands themselves.

The following conventions are also used in this manual:

[ ]	In text, brackets enclose optional information from which you can choose to use or not.
$	The dollar sign represents the DIGITAL Command Language prompt in OpenVMS.
>	The right angle bracket represents the MS-DOS command prompt. This symbol indicates that the MS-DOS command language interpreter is ready for input.
boldface text	Boldface type in text indicates a term defined in the text.

1

Oracle SQL/Services: New and Changed Features

This chapter describes the new features and technical changes to Oracle SQL/Services in release 7.2.0.1. In addition, it describes installation requirements, obsolete routines, structures, and features, and provides a summary of additions and changes to the documentation.

1.1 Installing Oracle SQL/Services Release 7.2.0.1

Refer to the *Oracle SQL/Services Installation Guide* for installation information.

Section 1.1.1 briefly highlights the major changes to the Oracle SQL/Services installation procedure, which are described in detail in the *Oracle SQL/Services Installation Guide*.

The Oracle SQL/Services release 7.2.0.1 installation kit contains the following files:

- SQLSRVV72010AM072.A – Oracle SQL/Services for OpenVMS Alpha
- SQLSRVV72010AM072.B – OCI Services for Oracle Rdb for OpenVMS Alpha
- SQLSRVV72010IM072.A – Oracle SQL/Services for OpenVMS I64
- SQLSRVV72010IM072.B– OCI Services for Oracle Rdb for OpenVMS I64
- SQSCLIA072.A – Oracle SQL/Services client kit for OpenVMS Alpha
- SQSCLII072.A – Oracle SQL/Services client kit for OpenVMS I64

1.1.1 Software Installation Requirement and Compatibility Information

The following information describes the Oracle SQL/Services hardware and software requirements and compatibility information for installing Oracle SQL/Services release 7.2.0.1.

Hardware Requirements

Oracle SQL/Services server platforms require hardware configurations and Ethernet LAN connectivity supported by the prerequisite software listed in Software Requirements.

Alpha EV56 Requirement

Oracle SQL/Services has been optimized for the Alpha EV56 platform. If running on an older platform, Oracle SQL/Services will still execute properly, but will run some operations in emulation mode and performance will be significantly slower.

DECnet Support for OCI Services for Oracle Rdb Connections

Oracle SQL/Services does not support the DECnet network protocol for OCI connections on Alpha or I64 platforms.

Optional Hardware

Oracle ODBC Driver for Rdb and Oracle SQL/Services client platforms require hardware configurations and Ethernet LAN connectivity supported by the prerequisite software listed in Software Requirements.

Software Requirements

The following software is required for the Oracle SQL/Services and OCI Services for Oracle Rdb OpenVMS Alpha server.

- OpenVMS Alpha operating system V7.3-2 or higher

  If you want to use the two-phase commit protocol, you must have the following patches installed:

  – For V7.3-2. install patch "DEC AXPVMS VMS732_DDTM-V0100"

  – For V8.2A, install patch "DEC AXPVMS VMS82A_DDTM-V0100"

  Use the following command to show the patch level that is currently on the system:

  ```
  $PRODUCT SHOW HISTORY "*_DDTM","*_TIE"
  ```

- Oracle Rdb for OpenVMS Alpha V7.0.8 or higher, V7.1.4 or higher, or V7.2 or higher
- One of the following network transport options:

  – Oracle Net

- TCP/IP transport - HP TCP/IP Services for OpenVMS or other HP TCP/IP Services for OpenVMS compliant transport on the host system.

Note: TCP/IP Services for OpenVMS Alpha V5.5 has been fully tested with Oracle SQL/Services. Other transports that comply with the HP TCP/IP Services for the OpenVMS interface may function correctly but have not been fully tested by Oracle Corporation.

The following software is required for the Oracle SQL/Services and OCI Services for Oracle Rdb OpenVMS I64 server.

- OpenVMS I64 operating system, V8.2-1 or higher

  If you want to use the two-phase commit protocol, you must have the following patches installed:

  - For V8.2I, install patch "HP I64VMS VMS82I_DDTM-V0100"
  - For V8.21I, install patches "HP I64VMS VMS821I_DDTM-V0100" and "HP I64VMS VMS821I_TIE-V0100"
  - For V8.3I, install patch HP I64VMS VMS83I_ADDENDUM-V0100

  Use the following command to show the patch level that is currently on the system:

```
$PRODUCT SHOW HISTORY "*_DDTM","*_TIE"
```

- Oracle Rdb for OpenVMS I64, V7.2 or higher
- One of the following network transport options:
  - DECnet transport - DECnet for OpenVMS I64, V8.2 or higher
  - Oracle Net
  - TCP/IP transport - HP TCP/IP Services for OpenVMS I64, V5.5-11 or other HP TCP/IP Services for OpenVMS compliant transport on the host system.

Note: TCP/IP Services for OpenVMS I64 V5.5-11 has been fully tested with Oracle SQL/Services. Other transports that comply with the HP TCP/IP Services for the OpenVMS interface may function correctly but have not been fully tested by Oracle Corporation.

Table 1–1 lists the desktop clients and network transports supported by Oracle SQL/Services on the OpenVMS platform.

Table 1–1 Oracle SQL/Services on OpenVMS Client/Network Support

Desktop	Client Transport Support		
Client Platform	**DECnet**	**TCP/IP**	**Oracle Net**
Oracle ODBC Driver for Rdb Clients			
Windows NT Intel/Windows XP	X	X	--
Windows 98/Windows 2000	X	X	--
Oracle SQL/Services Clients			
Windows NT/Windows ME/Windows XP	X	X	X
Windows 95/Windows 98/ Windows 2000	X	X	X
Tru64 UNIX	X	X	X
OpenVMS Alpha	X	X	X
Red Hat Linux	X	X	X

Desktop Client Software Requirements

This section describes the software required by each desktop client platform.

Oracle SQL/Services Clients

The following section describes the software required for each Oracle SQL/Services client and the supported transport.

Oracle SQL/Services Windows XP Client (TCP/IP Transport):

- Microsoft Windows XP Home Edition operating system, V5.1.2600
- Microsoft Windows XP Professional Edition operating system, V5.1.2600

Oracle SQL/Services Windows ME Client (TCP/IP Transport):

- Microsoft Windows ME operating system, V4.90.3000

Oracle SQL/Services Windows 2000 Client (TCP/IP Transport):

- Microsoft Windows 2000 operating system, V4.0

Oracle SQL/Services Windows 98 Client (TCP/IP Transport):

- Microsoft Windows 98 operating system, V4.10.1998

Oracle SQL/Services Windows NT Intel Client (TCP/IP Transport):

- Microsoft Windows NT operating system, V4.0

Oracle SQL/Services Windows NT Intel Client (DECnet Transport):

- Microsoft Windows NT operating system, V4.0
- PATHWORKS for Windows NT, V4.1B

Oracle SQL/Services Tru64 UNIX client (DECnet or TCP/IP Transports):

- Tru64 UNIX operating system, V3.2 or V4.0
- DECnet/OSI, V2.0, for Tru64 UNIX (required for DECnet applications only)

Oracle SQLServices Red Hat Linux client:

- Red Hat Enterprise Linux V2.1 or V3.0

1.2 Running the Oracle SQL/Services Configuration File Utility

If you have release 5.1, 6.0, or 6.1 Oracle SQL/Services configuration files that you want to convert to release 7.2.0.1 files, you must first convert the files to release 7.0 files, as shown in Section 1.2.2, and then convert release 7.0 files to release 7.2 files as shown in Section 1.2.1.

1.2.1 Converting Previous Configuration Files to Release 7.2.0.1

Follow these steps to convert release 7.0 Oracle SQL/Services configuration files to release 7.2.0.1 files:

1. Run the conversion utility:

```
$ @SYS$MANAGER:SQLSRV_EXT_CONFIG70TO71 sqlsrv_config_file70.dat -
$_ my_config.sqs
```

   where sqlsrv_config_file70.dat is the name of the SQL/Services release 7.0 configuration file that you want to convert and my_config.sqs is the SQL/Services manager script file created by this procedure that can be used to create the new configuration file.

   If you interrupt the execution of the command file, check to see if the logical SQLSRV_EXT_DISP_V70 has been assigned:

```
$ show logical SQLSRV_EXT_DISP_V70
```

If the logical has been assigned, deassign it with the following command before you run the command file again:

```
$ deassign/system SQLSRV_EXT_DISP_V70
```

2. Edit the SQL/Services manager script file (my_config.sqs) and make any corrections, such as modifying the log path for the dispatcher or the SQL release number.
3. Follow one of the following procedures to create a configuration file for release 7.2.0.1:
   a. Create a new configuration file:

```
$ SQLSRV_MANAGE72 :== $SYS$SYSTEM:SQLSRV_MANAGE72.EXE
$ SQLSRV_MANAGE72
SQLSRV> set config_file temp.dat;
The configuration file does not exist or could not be opened.
Create it now? (N)
SQLSRV> @MY_CONFIG.SQS;
SQLSRV> EXIT
$ COPY TEMP.DAT; SYS$MANAGER:SQLSRV_CONFIG_FILE72.DAT;
$ @SYS$STARTUP:SQLSRV$STARTUP72
```

   b. Or follow these steps to add the necessary dispatchers and services to the configuration file created by the installation:
      – Edit the my_config.sqs file to remove lines such as creating a server.
      – Run the following commands:

```
$ SQLSRV_MANAGE72 :== $SYS$SYSTEM:SQLSRV_MANAGE72.EXE
$ SQLSRV_MANAGE72
SQLSRV> @MY_CONFIG.SQS; -- This file must be modified before running
SQLSRV> EXIT
$ @SYS$STARTUP:SQLSRV$STARTUP72
```

1.2.2 Converting Previous Configuration Files to Release 7.0

In order to convert release 5.1, 6.0, or 6.1 Oracle SQL./Services configuration files to release 7.0 files, you must run the conversion utility, SYS$MANAGER:SQLSRV_CONVERT_CONFIG.EXE. This conversion utility converts any Oracle SQL/Services release 5.1, 6.0, or 6.1 configuration file into a SQLSRV_MANAGE script file that you can then run to add your previously defined classes to a release 7.0 configuration file as services. You can run the utility by following these steps:

- Run the image, taking defaults for the input and output file names. The defaults are:

SYS$STARTUP:SQLSRV$CONFIG.DAT

SYS$STARTUP:SQLSRV_CONFIG.SQS

- Define logical names to accept an input file name and produce an output file name from specific directories. For example:

```
$ DEFINE SQLSRV$CONFIG SYS$MANAGER:SQLSRV$CONFIG61.DAT
$ ! for the existing SQLSRV$CONFIG61.DAT
$ DEFINE SQLSRV_CONFIG MY_CONFIG_SCRIPT_FILE.SQS
$ ! for new MY_CONFIG_SCRIPT_FILE.SQS
$ RUN SYS$MANAGER:SQLSRV_CONVERT_CONFIG.EXE
```

- Define a foreign symbol using the arguments p1 and p2 as the input and output file names. For example:

```
$ CVT :== $SYS$MANAGER:SQLSRV_CONVERT_CONFIG.EXE
$ CVT SYS$MANAGER:SQLSRV$CONFIG61.DAT MY_CONFIG_SCRIPT_FILE.SQS
```

The correct Oracle Rdb version must be set for each service (CLASS in terminology used by releases prior to release 7.0). In previous versions of Oracle SQL/Services, the version was set by the login procedure of the account specified as the STARTUP account in the configuration file. In Oracle SQL/Services release 7.0 and higher, the version is an attribute (SQL VERSION) of the defined service. The conversion utility attempts to default to this version depending on the name of the STARTUP account; however, the default may not be correct. Before creating the script file entry, the conversion utility displays the selected default SQL VERSION for the service and allows you to take the default or enter a different version. The version may be either the string STANDARD or x.x where x is a decimal digit.

Once you have created the SQLSRV_MANAGE script file, you can run it to add the services to your release 7.0 configuration file. For example:

```
$ SQLSRV_MANAGE :== $SYS$SYSTEM:SQLSRV_MANAGE.EXE
$ SQLSRV_MANAGE -INPUT MY_CONFIG_SCRIPT_FILE.SQS
```

1.3 Documentation

Documentation for Oracle SQL/Services and OCI Services for Oracle Rdb is available in Adobe Acrobat (PDF) formats on MetaLink and OTN. Adobe Acrobat files ending with extension .PDF can be read with an Adobe Reader. Readers for many platforms are available without fee from the Adobe web site.

1.4 Summary of Oracle SQL/Services New and Changed Features for Release 7.2.0.1

The following sections describe new and changed features for Oracle SQL/Services release 7.2.0.1. See Chapter 4, "OCI Services for Oracle Rdb: Release Notes", for descriptions of new and changed features for OCI Services for Oracle Rdb.

1.4.1 New COPY SERVICE Command

The SQLSRV_MANAGE utility has been enhanced with a COPY SERVICE command for release 7.2.0.1. Use this command to copy a service definition that already exists in the configuration file.

The arguments for the COPY SERVICE command are the same as those for the ALTER SERVICE command. Refer to the *Oracle SQL/Services Server Configuration Guide* for a description of the arguments. You can use individual arguments to override values in the service definition you are copying.

If you add a new user with this command, the user name is added to the grant list. If you define a default connect user name, it is also added to the grant list.

The following example copies the SA_MCS72 service definition to one named SA_MCS72_NEW and stores the new service definition in the configuration file. It replaces the previous values for the OWNER, SQL_VERSION, and IDLE_EXECUTOR_TIMEOUT arguments.

```
SQLSRV> show service sa_mcs72 full;
Service SA_MCS72
    State: UNKNOWN
    Owner: smith
    Owner Password: <not specified>
    Protocol: OCI clients
    Default Connect Username: <not specified>
    Default Connect Password: <not specified>
    SQL version: 7.2
    Autostart: off
    Process init: DISK1:[SMITH]proc_init.com
    Attach: ATTACH 'filename DISK1:[SMITH]mf_personnel'
    Schema: <not specified>
    Reuse: SESSION
    Database Authorization: CONNECT USERNAME
    dbsrc file: <not specified>
    SQL init file: DISK1:[SMITH]SA_MCS72.SQL
    Appl Transaction Usage: SERIAL
```

```
    Idle User Timeout: <none>
    Idle Exec Timeout: 1800 seconds
    Min Executors: 1
    Max Executors: 10
    Clients Per Executor: 1
    Active Clients: 0

Access to service SA_MCS72
    Granted to users:
        PUBLIC PRIVILEGED_USER 'AAA' 'smith'

SQLSRV> copy service sa_mcs71_new from_service sa_mcs71
_SQLSRV> owner 'new_owner'
_SQLSRV> sql version 7.2
_SQLSRV> Idle Executor Timeout 200
_SQLSRV> ;
SQLSRV> show service sa_mcs72_new full;
Service SA_MCS72_NEW
    State: UNKNOWN
    Owner: new_owner
    Owner Password: <not specified>
    Protocol: OCI clients
    Default Connect Username: <not specified>
    Default Connect Password: <not specified>
    SQL version: 7.2
    Autostart: off
    Process init: DISK1:[SMITH]proc_init.com
    Attach: ATTACH 'filename DISK1:[SMITH]mf_personnel'
    Schema: <not specified>
    Reuse: SESSION
    Database Authorization: CONNECT USERNAME
    dbsrc file: <not specified>
    SQL init file: DISK1:[SMITH]SA_MCS72.SQL
    Appl Transaction Usage: SERIAL
    Idle User Timeout: <none>
    Idle Exec Timeout: 200 seconds
    Min Executors: 1
    Max Executors: 10
    Clients Per Executor: 1
    Active Clients: 0

Access to service SA_MCS72_NEW
        Granted to users:
            PUBLIC PRIVILEGED_USER 'new_owner' 'AAA' 'smith'
```

1.4.2 Enhanced Logging in SQL/Services Log Files

Enhancement request: 5388589

SQL/Services log files did not identify the SQL/Services version, current time stamp, hardware type or operating system version. This identifying information is often useful to track problems.

With SQL/Services release 7.2.0.1 and later, the following header will now appear in SQL/Services executor, dispatcher and monitor log files to record that information.

```
$! ------------------------------------------------------------------------
$!
$! Oracle SQL/Services V7.2-01 executor log file
$! Wed Jul 19 11:41:27 2006
$!
$! ------------------------------------------------------------------------
$!
$! This is a AlphaServer 4X00 5/533 4MB running VMS V7.3-2
$!
```

1.4.3 Enhanced Error Reporting Creating & Opening Process Command Procedures

Enhancement request: 5388540

When an error occurred creating or opening a command procedure used to create monitor, dispatcher and executor processes, SQL/Services logged the error returned by the CC creat builtin function. This error was not the specific VMS error and was not always useful in diagnosing problems.

With SQL/Services release 7.2 and later, the OpenVMS error will also be logged. The following is a sample extract from a log file, containing such an error condition.

```
---EVENT BEG: EVENT_LOG ----------------------- Mon Jul 17 16:58:45.010 2006---
%SQLSRV-I-EVENT_LOG, event logged at line 1122 in file DBS_PROCESS_VMS.C;5
%SQLSRV-E-SM_FOPEN_ERR, Error opening DISK2:[JONES]SQS_NODE_SVC040000171.COM;
%SQLSRV-E-ERROR_TEXT, Error text: permission denied
%RMS-E-PRV, insufficient privilege or file protection violation
---EVENT END: EVENT_LOG -----------------------
```

1.4.4 Enhanced SQLSRV_MANAGE Show Version Command

Beginning with release 7.2.0.1, the SQLSRV_MANAGE SHOW VERSION command displays the full version number, rather than just the first three digits. For example:

```
SQLSRV> show version;
Version: V7.2-01
```

2

Oracle SQL/Services: Software Errors Fixed

This chapter describes problems with Oracle SQL/Services software that are fixed in this release.

2.1 Oracle SQL/Services Errors Fixed in Release 7.2.0.1

The following known problems found in the Oracle SQL/Services OpenVMS server have been fixed for this release.

2.1.1 Problem Using Persona Feature with JDBC Dispatchers

TAR: 15930012.6

The JDBC Dispatcher did not inherit IMPERSONATE privilege and this sometimes caused use of the persona feature to fail. SQL/Services will now start the JDBC dispatcher with IMPERSONATE privilege.

This problem has been corrected in release 7.2.0.1.

2.1.2 Poor Performance From OCI Queries

Bugs: 3259208, 4770496, 5144164

Oracle SQL/Services release 7.1.5.8 introduced a problem that could cause queries from an OCI source to sometimes take an extraordinarily long time to complete. Examination of the SQL/Services processes would show that they were idle even though the client had not received a response from SQL/Services.

This problem was introduced by changes in the underlying Oracle NET libraries employed by SQL/Services. There were instances where a network message would arrive but notification of that arrival was not being delivered to the SQL/Services dispatcher. When

that occurred, SQL/Services would only see that a new network message had arrived when it did periodic polling.

There is no workaround for this issue.

This problem has been corrected in release 7.2.0.1. Notification of network message arrival is now done immediately.

2.1.3 Misleading Dispatcher Logging Entries Removed

Bug: 5148550

When an Oracle Net connection was successfully disconnected, the following information was entered into the dispatcher log. This would occur for any executor using the SQLNET protocol, such as an OCI Services connection. Since the logging was done for every connection, it tended to make dispatcher log files large.

Dispatcher log entry:

```
---EVENT BEG: EVENT_LOG ----------------------- Fri Mar 31 09:40:31.5602006---
%SQLSRV-I-EVENT_LOG, event logged at line 2496 in file COM_TNS.C;1
%SQLSRV-E-TNSFAILURE, Oracle Net TNS nsrecv() service has failed
%SQLSRV-E-TNSEXTENDED, Oracle Net TNS error codes: primary (12537) secondary
(12560)

---EVENT END: EVENT_LOG ----------------------- Fri Mar 31 09:40:31.5602006---
```

This message (12537) is actually an informational message from Oracle TNS. There is no need to log the message. It appears to be reporting a problem, rather than a success condition, causing confusion. It has now been removed from dispatcher logging in Oracle SQL/Services release 7.2.0.1.

2.1.4 SQLSRV$MOD*.EXE Files Removed From the SQL/Services Kit

Bug: 5222605

SQLSRV$MOD images are part of the Oracle Rdb SQL component. Due to a past problem, some corrected SQLSRV$MOD images were shipped on the SQL/Services kit to be installed if needed to supercede older images. The SQL/Services startup and shutdown procedures installed and deinstalled these images.

Because the need for these images no longer exists, they have been removed from the SQL/Services kit and procedures in Oracle SQL/Services release 7.2.0.1.

2.1.5 Failure to Start 2PC Using OCI Universal Services

In releases 7.1.6, 7.1.6.1, and 7.2, attempting to access an OCI universal service using two-phase commit failed to start a two-phase commit transaction and therefore reverted to a one-phase commit transaction.

This problem has been corrected in release 7.2.0.1.

2.1.6 Occasional Access Violations During OCI Bugcheck Dumps

Occasionally, an access violation would occur during the process of writing an OCI bugcheck dump file.

This problem has been corrected in release 7.2.0.1.

2.1.7 SQL/Services Installation Procedure Fixes SQL Version Specified

During SQL/Services installation, the user is asked to specify the SQL version for the generic service. If the user specified more than 2 digits in that version number, the SQL/Services installation would fail.

This problem has been corrected in release 7.2.0.1. The installation procedure now truncates the version to 2 digits, as required.

2.1.8 Monitor and Dispatcher Processes in CPU Loop

The SQL/Services monitor and dispatcher processes may sometimes get into a deadlock condition where both processes are in a CPU loop. If SQL/Services is configured with services that have a bad SQL init file with the minimum executor parameter set to greater than 0, starting up the SQL/Services server can, in rare occasions, cause the SQL/Services monitor and dispatcher processes to be in a CPU loop waiting for a mutex.

This problem has been corrected in release 7.2.0.1.

3

Oracle SQL/Services: Known Problems

This chapter describes problems and restrictions relating to Oracle SQL/Services release 7.2.0.1.

3.1 Oracle SQL/Services Manager GUI Known Problems and Restrictions

The following information describes Oracle SQL/Services Manager graphical user interface (GUI) restrictions and known problems:

- Only one copy of the Oracle SQL/Services Manager GUI can run on a PC.

  Only one copy of the Oracle SQL/Services Manager GUI can be run on a given PC at any one time.

3.2 Oracle SQL/Services Release 7.2.0.1 Server Known Problems and Restrictions

The following sections describe Oracle SQL/Services release 7.2.0.1 server restrictions and known problems.

3.2.1 Support for OpenVMS VAX and Standard Kits

Oracle SQL/Services release 7.2.0.1 and OCI Services for Oracle Rdb release 7.2.0.1 are not supported on OpenVMS VAX.

Also, standard kit installation is not supported with this version. Only multiversion kits are available.

3.2.2 Concealed Attributes are Required for Rooted Directory Logicals

When Oracle SQL/Services starts a new monitor, dispatcher or executor process, it uses the SET DEFAULT DCL command to set the initial default disk and directory for the process. In addition, when a new client connects to a universal service with database authorization set to connect user, Oracle SQL/Services calls the SYS$SETDDIR OpenVMS system service to set the default disk and directory for the executor process. To set default to a disk and directory combination that includes a rooted directory logical name, the OpenVMS operating system requires that the rooted directory logical name be defined with the CONCEALED attribute.

Consider a rooted directory logical name ALL_USERS used to reference user directories in the following example:

```
Root top-level user directory:    $1$DKA100:[USERS]
Specific user directory:          $1$DKA100:[USERS.FRED]
```

In this example, the ALL_USERS rooted directory logical name must be defined as follows (the /EXECUTIVE switch may also be used for greater security):

```
$ DEFINE/SYSTEM ALL_USERS $1$DKA100:[USERS.]/TRANSLATION_ATTRIBUTE=CONCEALED
```

The default disk and directory for user FRED can then be specified as follows:

```
ALL_USERS:[FRED]
```

If a rooted directory logical name is not defined with the CONCEALED attribute, then the SET DEFAULT DCL command executed during monitor or dispatcher process creation fails as follows if the monitor is started from an account that specifies the rooted directory logical name. Likewise, the SET DEFAULT DCL command executed during executor process creation will also fail in the same way if the service owner user name account specifies the rooted directory logical name.

```
$ DEFINE SYS$LOGIN ALL_USERS:[FRED]
$ SET DEFAULT SYS$LOGIN
%DCL-W-DIRECT, invalid directory syntax - check brackets and other delimiters
```

In addition, if a rooted directory logical name specified for a client account is not defined with the CONCEALED attribute, then an executor process will bugcheck and exit with the following error message in the executor log if the user connects to a universal service with database authorization set to connect user:

```
----------EVENT BEGIN:  EVENT_LOG at Wed Sep 24 1997
14:05:33.914----------%SQLSRV-I-EVENT_LOG, event logged at line 1636 in file
DBS_PROT_VMS.C;1
Error setting VMS process user name
```

```
%RMS-F-DIR, error in directory name
----------EVENT END  :  EVENT_LOG at Wed Sep 24 1997 14:05:33.930----------
```

See the *Guide to OpenVMS File Applications* in the OpenVMS documentation set for more information on how to define and use rooted directory logical names.

3.2.3 Do Not Kill Oracle SQL/Services Processes

Under certain circumstances, the entire Oracle SQL/Services server shuts down if an Oracle SQL/Services dispatcher or executor is abnormally terminated. You should never use the DCL STOP/ID command on OpenVMS systems to stop an Oracle SQL/Services dispatcher or executor process. The Oracle SQL/Services system management command SHUTDOWN DISPATCHER or SHUTDOWN SERVICE should be used to stop dispatchers and executors. If an executor does not terminate after issuing the SHUTDOWN SERVICE command, or if you do not want to shut down the entire service, the Oracle SQL/Services system management command, KILL EXECUTOR, should be used instead.

Note that the RMU Close command can have the same effect as the STOP/ID or kill command by terminating Oracle SQL/Services executors attached to the database being closed. Before using the RMU Close command, ensure that no Oracle SQL/Services executors currently have the database open. Any executors you find that do have the database open should be terminated with either the Oracle SQL/Services system management command SHUTDOWN SERVICE or the KILL EXECUTOR command.

3.2.4 Do Not Shut Down or Restart the SQLSRV_MANAGE System Management Service

If you shut down or restart the SQLSRV_MANAGE system management service using either the SQLSRV_MANAGE utility or the Oracle SQL/Services Manager GUI, then subsequent attempts to connect to the server are rejected and you render the server unmanageable. If you do accidentally shut down or restart the SQLSRV_MANAGE service, then you must find and kill the Oracle SQL/Services monitor process, then restart the server.

3.2.5 Management Utilities Allow Multiple Dispatchers With the Same Port IDs

Oracle SQL/Services allows you to define multiple dispatchers, each listening on different network ports. Currently, the SQLSRV_MANAGE and the Oracle SQL/Services Manager GUI do not ensure that multiple dispatchers do not use the same port numbers or names. If multiple dispatchers are defined to use the same ports, the second dispatcher to be started fails.

3.2.6 Problem With RMU Dispatcher

You cannot configure the Oracle RMU dispatcher for Oracle SQL/Services with alternate network ports. Therefore, the RMU dispatchers for Oracle SQL/Services releases 7.1 and 7.2 use the same network ports as the RMU dispatcher for Oracle SQL/Services release 7.2.0.1.

For this reason, you cannot have an RMU dispatcher running for more than one release of Oracle SQL/Services. If the Oracle SQL/Services release 7.2.0.1 multiversion installation detects an existing version of Oracle SQL/Services release 7.0, the installation procedure creates the RMU dispatcher with the "autostart off" attribute in the release 7.2.0.1 configuration. If an existing release 7.2.0.1 configuration is found, the installation procedure will not change the RMU dispatcher, so you should alter the RMU dispatcher to disable autostart.

3.2.7 Database Service Attached to Remote Database Does Not Know If Database is Closed

It is possible for Oracle SQL/Services database services to be preattached to a remote database. For example, the payroll service defined below attaches to the database "payroll" on node "REMOTE".

```
create service payroll autostart on
        reuse session
        sql version 7.0
        attach 'filename REMOTE::payroll'
        owner 'payrollaccnt'
        database authorization service owner
        min_executors 5
        max_executors 5;
```

If the payroll database on node REMOTE is closed, the Oracle SQL/Services payroll service has no way of knowing that the database has been closed. The payroll service continues to run, even though it is no longer attached to the database. The service is useless and must be shut down and restarted after the database is reopened. Any clients attached to the service while it is in this state get a SQLCODE of –1 with the following errors when they attempt to access the database:

```
%RDB-F-IO_ERROR, input or output error
-SYSTEM-F-LINKABORT, network partner aborted logical link
```

All Oracle SQL/Services services that are preattached to a remote database should be shut down before the database is closed. If this is not possible, there is a workaround for database services defined to attach to Oracle Rdb V6.1 or higher databases. Rather than

defining session reusable database services, you can define a transaction reusable database service with CLIENTS_PER_EXECUTOR set to 1.

```
create service payroll autostart on
        reuse session
        sql version 7.0
        attach 'filename REMOTE::payroll'
        owner 'payrollaccnt'
        database authorization service owner
        min_executors 5
        max_executors 5
        clients_per_executor 1;
```

The service definition previously shown gives you essentially the same behavior as the previous session reusable database service. However, Oracle SQL/Services executes a "get diagnostics ? = transaction_active" statement to detect the end of a transaction for transaction reusable services. Because this requires a call to the Oracle Rdb engine, it fails and Oracle SQL/Services bugchecks and shuts down the executor. If this brings the executor count below the MIN_EXECUTORS value defined for the service, the Oracle SQL/Services monitor attempts to create a new executor process. If the monitor fails to start a new executor process after two attempts, it shuts down the service. Note that this workaround generates executor bugcheck dumps that need to be cleaned up.

3.2.8 Process Startup Fails Due to Errors in Systemwide OpenVMS Login Procedure

All processes in the Oracle SQL/Services server environment on OpenVMS are created running the SYS$SYSTEM:loginout image with a process-specific command procedure as SYS$INPUT. Because the loginout image is used to create the process, the systemwide login procedure will be executed by the loginout image during process creation. If this procedure fails for some reason, then the Oracle SQL/Services process will fail to start. By default, any DCL command or image that completes with a failure status with a severity level of error or fatal can cause the procedure to fail unless it is handled using the DCL ON or SET NOON commands.

All Oracle SQL/Services processes start by executing the following DCL commands during process creation:

```
$ DELETE/SYMBOL/ALL
$ VRFY_SAVE = F$VERIFY(1)
$ DELETE <disk>:[directory]SQS_<node>_<component>.COM;
$ DEFINE SQS$DBSERVER TRUE
$ DEFINE SYS$LOGIN "<disk>:[directory]"
$ SET DEFAULT SYS$LOGIN
$ DEFINE SYS$SCRATCH "<disk>:[directory]"
```

If an Oracle SQL/Services process fails before executing these commands, please review the systemwide login procedure to determine the reason for the failure.

3.2.9 DBSRC_FILE Service Attribute is Not Supported on OpenVMS

The Oracle SQL/Services DBSRC_FILE service attribute is not currently supported on OpenVMS. To specify values in either the RDB$CLIENT_DEFAULTS.DAT file or the RDB$SERVER_DEFAULTS.DAT file, you must define a logical name that specifies the directory that contains one or both of the these files. To specify the directory on a per-process basis in the Oracle SQL/Services environment, you must create an executor process initialization command procedure that defines the RDB$USER_DEFAULTS logical name, then specify the file name of the procedure in the PROCESS_INITIALIZATION service attribute. Alternatively, you can define the RDB$SYSTEM_DEFAULTS logical as a system logical name or the RDB$GROUP_DEFAULTS logical name as a group logical name. See the *Oracle Rdb7 SQL Reference Manual* for more information.

3.2.10 Installation Clarification: How to Determine Ports to Which to Direct Clients

As of Oracle SQL/Services V7.0, it is possible to run multiple versions of Oracle SQL/Services at the same time on the same system. In order to do this, each Oracle SQL/Services server must use unique ports.

Note that each Oracle SQL/Services server uses multiple sets of ports; one set for the server for system management, and one set for each defined dispatcher. A Show Server command displays the set of ports used for system management:

```
SQLSRV> show server;
    Server Version:      7.0
    Server Platform:     OpenVMS Alpha
    Max Shared Mem Size: 2000 Kb
    Config file:         SYS$SYSROOT:[SYSMGR]SQLSRV_CONFIG_FILE70.DAT;1
    Log path:            SYS$MANAGER:
    Dump path:           SYS$MANAGER:
    Proc start time:     <none>
    Proc shut time:      <none>
     Network Ports:                                        (State)   (Protocol)
       DECnet  object   SQLSRV_SERVER             Running  Native
       TCP/IP  port     2199                      Running  Native
    Current shared memory usage:
       Allocation unit:      65536 bytes
       Total memory:       2031616 bytes ( 31 units)
       Free memory:        1835008 bytes ( 28 units)
```

```
       Partly allocated:    131072 bytes (  2 units)
    Log File:            SYS$SYSROOT:[SYSMGR]SQS_ORASQS_SQLSRV_MON_0070.LOG;
    Dump File:           SYS$SYSROOT:[SYSMGR]SQS_ORASQS_SQLSRV_70.DMP;
```

During installation, a SQLSRV_DISP dispatcher is defined to handle requests from Oracle SQL/Services clients. These include Oracle ODBC Driver for Rdb clients, Query Performance Tuner, and the Rdb Web Agent. The SQLSRV_DISP dispatcher listens on a specific set of network ports shown as follows under Network Ports. It is also defined to listen for Oracle SQL/Services protocol messages shown as follows under Protocol. A Show Dispatcher command displays the set of ports and protocol used for client communication.

```
SQLSRV> show dispatcher SQLSRV_DISP;
Dispatcher SQLSRV_DISP
        State:                      RUNNING
        Autostart:                  on
        Max connects:               100 clients
        Idle User Timeout:          <none>
        Max client buffer size:     5000 bytes
        Network Ports:                              (State)   (Protocol)
            IPX/SPX port     0x84b1                        Running    SQL/Services
            DECnet  object   81                             Running    SQL/Services
            TCP/IP  port     118                            Running    SQL/Services
        Log File:          SYS$SYSROOT:[SYSMGR]SQS_ORASQS_SQLSRV_DIS00370.LOG;
        Dump File:         SYS$SYSROOT:[SYSMGR]SQS_ORASQS_SQLSRV_DIS003.DMP;
```

During installation, an RMU_DISP dispatcher is defined to handle requests from RMU clients. The RMU_DISP listens on a specific set of network ports shown as follows under Network Ports. It is also defined to listen for Native protocol messages shown as follows under Protocol. Note that RMU clients send Native protocol messages. A Show Dispatcher command displays the set of ports and protocol used for RMU client communication.

```
SQLSRV> show dispatcher RMU_DISP;
Dispatcher RMU_DISP
    State:                      RUNNING
    Autostart:                  on
    Max connects:               100 clients
    Idle User Timeout:          <none>
    Max client buffer size:     5000 bytes
     Network Ports:                              (State)   (Protocol)
      DECnet  object   RMU_DISP                  Running   Native
      TCP/IP  port     1571                      Running   Native
    Log File:       SYS$SYSROOT:[SYSMGR]SQS_ORASQS_RMU_DISP00870.LOG;
    Dump File:      SYS$SYSROOT:[SYSMGR]SQS_ORASQS_RMU_DISP008.DMP;
```

The Oracle SQL/Services Installation Guide discusses directing Oracle SQL/Services clients to the correct dispatcher for the desired version of Oracle SQL/Services. However, it neglects to mention how to determine the port number to which to direct the client. Oracle SQL/Services clients must communicate with Oracle SQL/Services dispatchers (those defined with Protocol SQL/Services). Do a Show Dispatcher command to find the desired dispatcher, and direct the client to the ports defined for that dispatcher.

3.2.11 How to Re-create Your RMU_DISP Dispatcher and RMU_SERVICE Service

During Oracle SQL/Services installation, an RMU_DISP dispatcher and an RMU_ SERVICE service are created. These components listen for and execute requests from RMU clients.

The RMU_DISP dispatcher listens on a specific set of network ports as shown in the following example under Network Ports. RMU clients cannot be configured to use alternate ports, so it is important that the network ports for the RMU_DISP dispatcher are not changed.

The RMU_DISP dispatcher is also defined to listen for Native protocol messages as shown in the following example under Protocol. Note that RMU clients send Native protocol messages. It is important that the Protocol for the RMU_DISP dispatcher remain as Native.

A Show Dispatcher command displays the set of network ports and protocol used for RMU client communication.

```
SQLSRV> show dispatcher RMU_DISP;
Dispatcher RMU_DISP
    State:                      RUNNING
    Autostart:                  on
    Max connects:               100 clients
    Idle User Timeout:          <none>
    Max client buffer size:     5000 bytes
    Network Ports:                                (State)   (Protocol)
      DECnet  object   RMU_DISP                    Running   Native
      TCP/IP  port     1571                        Running   Native
    Log File:          SYS$SYSROOT:[SYSMGR]SQS_ORASQS_RMU_DISP00870.LOG;
    Dump File:         SYS$SYSROOT:[SYSMGR]SQS_ORASQS_RMU_DISP008.DMP;
```

The RMU_SERVICE service executes requests from RMU clients. The Service Protocol determines the type of request a service can execute. It is important that the Protocol for the RMU_SERVICE service remain as RMU.

A Show Service Full command displays the protocol used for the RMU service.

```
SQLSRV> show service rmu_service full;
```

```
Service RMU_SERVICE
    State:                    RUNNING
    Owner:                    RMU$SRV
    Protocol:                 RMU
    Default Connect Username: <not specified>
    SQL version:              7.0
    Autostart:                on
    Process init:             <not specified>
    Attach:                   <not specified>
    Schema:                   <not specified>
    Reuse:                    SESSION
    Database Authorization:   CONNECT USERNAME
    dbsrc file:               <not specified>
    SQL init file:            <not specified>
    Appl Transaction Usage:   SERIAL
    Idle User Timeout:        <none>
    Idle Exec Timeout:        120 seconds
    Min Executors:            4
    Max Executors:            100
    Running Executors:        4
    Clients Per Executor:     1
    Active Clients:           0

Access to service RMU_SERVICE
   Granted to users:
       PUBLIC  PRIVILEGED_USER 'RMU$SRV'
```

Oracle SQL/Services did not document the protocol attributes of the Create and Alter Dispatcher and Service commands in V7.0. There is no way to specify the dispatcher protocol or service protocol using the Oracle SQL/Services Manager GUI in V7.0.

If you lose or alter your RMU_DISP dispatcher or RMU_SERVICE service definitions, they can be recreated by invoking the SYS$MANAGER:SQLSRV_ CREATE[72].COM command procedure. This procedure invokes the SYS$MANAGER:SQLSRV_ CREATE[72].SQS system management script to create the RMU_DISP dispatcher and the RMU_SERVICE service as follows:

```
--
-- Create the Oracle RMU dispatcher.
--
create dispatcher rmu_disp
    autostart on
    network_port decnet object rmu_disp protocol native
    network_port tcpip port_id 1571 protocol native;
--
```

```
-- Create the Oracle RMU service for 7.2.
--
create service rmu_service
    protocol rmu
    autostart on
    sql version 7.2
    owner 'RMU$SRV'
    database authorization connect username
    min_executors 4
    max_executors 100
    idle_executor_timeout 120;
--
-- Grant access to the Oracle RMU service to all users. This
-- allows any client that supplies a valid user name and
-- password to access this service.
--
grant use on service rmu_service to public;
```

3.2.12 Implicit Attach Using the SQL$DATABASE Logical Name Not Supported

Oracle SQL/Services does not support the use of the SQL$DATABASE logical name on OpenVMS to implicitly attach to a database. For example, if you define the SQL$DATABASE logical name, a client application must still issue an explicit SQL ATTACH statement. For example, use ATTACH 'FILENAME SQL$DATABASE', to attach to the database. If a client application connected to a universal service issues a DML statement before attaching to a database, then the executor will return a status code of -1, with an associated "%SQL-F-NODEFDB, There is no default database" error message.

3.2.13 Suggested Maximum Executors of At Least Two for a Service

Many popular desktop tools make two connections to the Oracle SQL/Services server to do their work. For example, MS Access makes one connection initially and returns the list of tables. When the first request to reference a table is made, MS Access makes another connection to the Oracle SQL/Services server. If no executor is available, MS Access returns an error and suggests that you have a problem with your disk or network.

Oracle Corporation recommends that you configure maximum executors of at least 2.

3.2.14 Changing MIN and MAX_EXECUTORS for Transaction Reusable Service

When the values for the MIN_EXECUTORS and MAX_EXECUTORS arguments for a transaction reusable service are increased using the ALTER SERVICE command, more executors are made available; however, when the values for the MIN_EXECUTORS and

MAX_EXECUTORS arguments are decreased, the values are not changed dynamically. You must perform a SHUTDOWN SERVICE command followed by a START SERVICE command to make fewer executors available.

3.2.15 Problems That Exist for NO_SERVICE and SVCNOTRUN Error Returns

Clients may see the NO_SERVICE error returned when the service exists, but has not been started.

Clients may see the SVCNOTRUN (service not running) error when, in fact, the service does not even exist.

3.2.16 Some Error Messages Are Missing Object Names

Some error messages from SQLSRV_MANAGE are intended to display the object name that is the source of the error. However, the name is lost and no name is displayed.

3.2.17 Call SQLSRV_CLOSE_CURSOR() Before Using COMMIT or ROLLBACK

Within SQL, executing a COMMIT or ROLLBACK statement implies that all open cursors are closed unless you are using the Oracle Rdb Hold Cursors feature; this assumption is not true for Oracle SQL/Services. Because Oracle SQL/Services does not parse the SQL statements it passes, it does not know when a commit or rollback operation is executed. Instead, Oracle SQL/Services requires that the SQLSRV_CLOSE_CURSOR( ) call be issued to release the cursor-related data structures prior to a commit or roll back operation.

To reuse the same cursor name, you must close that cursor before executing a COMMIT or ROLLBACK statement.

3.2.18 Executors Do Not Execute LOGIN.COM Procedures for Clients

When a client connects to a server, the Oracle SQL/Services executor does not execute the LOGIN.COM DCL command procedure located in the client user name's default directory. Therefore, client applications should not use logical names defined in LOGIN.COM login procedures. Process logical names for Oracle SQL/Services executors can be defined only by a service's process initialization file.

3.3 Oracle SQL/Services Release 7.2.0.1 Client Known Problems and Restrictions

The following information describes Oracle SQL/Services release 7.2.0.1 client known problems and restrictions.

3.3.1 Oracle SQL/Services OpenVMS Client is Now Compiled With HP C

The Oracle SQL/Services client shared image for OpenVMS is now compiled using HP C. The options file provided by Oracle SQL/Services for linking client applications has changed. It used to include SYS$LIBRARY:VAXCRTL$API/SHARE. It now includes SYS$LIBRARY:SQLSRV$API/SHARE.

If you want to relink a client application that was compiled with VAX C, you must create an options file that specifies SYS$LIBRARY:VAXCRTL/SHARE and link against this new options file as well as SYS$LIBRARY:SQLSRV$API.OPT.

3.3.2 Use a Jacket Header File When Calling the Oracle SQL/Services API From C++

The Oracle SQL/Services header files, sqlsrv.h, sqlsrvca.h, and sqlsrvda.h, do not provide built-in support for use with the C++ programming language. However, by providing a jacket header file, you may call the Oracle SQL/Services API from C++ as you would from C. To include the Oracle SQL/Services header files in a C++ application, create the following header file, called sqlsrv.hxx, and #include it in your application program:

```
//
// Define VMS if compiling on OpenVMS to pick up the $ versions of
// the service names.
//
#ifdef __VMS
#ifndef VMS
#define VMS
#endif
#endif

//
// Include the headers files using C, not C++. No need to include
// sqlsrvca.h or sqlsrvda.h unless the application directly accesses
// the SQLCA and SQLDA structures.
//
extern "C"
{

#include <sqlsrv.h>
```

```
// #include <sqlsrvca.h>
// #include <sqlsrvda.h>

}
```

3.3.3 Problem Using Statement With No Parameter Markers in Batched Execution

If an application executes a prepared statement using the SQLSRV_EXE_BATCH flag, but the statement does not contain any parameter markers, the statement is incorrectly executed as if the SQLSRV_EXE_W_DATA flag had been specified. That is, the Oracle SQL/Services client API immediately sends an execute request message to the server to execute the statement. At this point, subsequent calls to any API routine, including sqlsrv_execute_in_out and sqlsrv_execute, all fail with SQLSRV_INTERR (-2011) or SQLSRV_MULTI_ACT (-2016) errors. Once the client API has entered this error state, only the sqlsrv_abort routine functions correctly. Therefore, client applications must not execute SQL statements that do not contain parameter markers using batched execution.

3.3.4 Incorrect Error Message is Returned if a Client Cancels Batched Execution

If an application calls sqlsrv_execute_in_out or sqlsrv_execute with the execute flag set to SQLSRV_EXE_WO_DATA before calling SQLSRV_EXECUTE_IN_OUT or SQLSRV_EXECUTE with the execute flag set to SQLSRV_EXE_BATCH, the client API incorrectly sends an execute request message to the server with no statement ID. Upon receipt of this message, the server returns an SQLSRV_INVSTMID (-2008) error back to the client with the following error message:

```
%SQLSRV-F-INVSTMID, Invalid statement id: 0
```

In this situation, the SQLSRV_INVSTMID error may be ignored.

3.3.5 PATHWORKS for DOS and Windows 95

In order to use the DECnet transport as a transport for Windows 95, use the following procedures:

Install the Windows 95 Release of PATHWORKS for DOS and Windows Version 1.0A.

Version 1.0A is the required version for the DECnet transport. Any Oracle SQL/Services client application that tries running against Version 1.0A will receive the following error:

```
SQLSRV_DLL_ADDR_ERR (-2046).
```

Note: An additional patch is required to the PATHWORKS V1.0A kit. Call the Customer Support Center at Hewlett-Packard Development Company to obtain this patch. The Oracle ODBC Driver for Rdb will return the following error SQLSRV_DLL_ADDR_ERR (-2046). This means pwsock32.dll, a PATHWORKS dynamic link library, is missing a required routine. The name of the missing PATHWORKS routine is called SktEndNodeEnt.

3.3.6 Disconnect Does Not Abort Running Transaction for Transaction Reusable Services

On long-running queries, users may expect that by rebooting the PC the query will be terminated. This is not the case for transaction reusable services. The query will continue until it is ready to send a response to the client. For session reusable services, the query will terminate.

3.3.7 Repeat Count on SQLSRV_FETCH_MANY Must be Less Than or Equal to 65535

Because the REPEAT_COUNT parameter to SQLSRV_FETCH_MANY is a 16-bit integer, the maximum number of rows a client can specify on SQLSRV_FETCH_MANY is 65535. If a larger number is specified, no error is detected. Rather, the repeat count wraps around and a smaller repeat count is used. For example, if a repeat count of 65536 is specified, the value in the 16-bit repeat count parameter is 0.

3.3.8 SQSAPIW.INI and SQSAPI32.INI Example is Misleading

The .INI files provided for the Windows platforms specify two commented out sections entitled [RDBSRV]. The first section specifies the transport to be used to communicate with node RDBSRV:

```
;[RDBSRV]
;Transport=DECnet
;Transport=TCP/IP
```

The second section specifies the ports to use to communicate with node RDBSRV:

```
;[RDBSRV]
;TCPIPPortNumber=1040
;DECnetObject=SQLSRV
```

This gives the impression that it is valid to specify two separate sections for each server node name specified in the .INI file. However, it is only valid to have one section per server

node name. All server node specific attributes must be specified together in the same section as follows:

```
;[RDBSRV]
;Transport=DECnet
;Transport=TCP/IP
;TCPIPPortNumber=1040
;DECnetObject=SQLSRV
```

Results are unpredictable if two sections are found for a target server node.

3.3.9 Avoid Using Cursor Names Starting with "SQLSRV_"

In designing your applications, avoid using cursor names starting with the prefix "SQLSRV_"; this prefix is reserved and used by the Oracle SQL/Services product.

3.3.10 Oracle SQL/Services Compatibility Issue with the Order of Include Files

With V4.1 and higher versions of Oracle SQL/Services, direct access to SQLDA and SQLCA structures is supported but is not recommended by Oracle Corporation. If direct access is used, the order of the Oracle SQL/Services include files must be as follows:

```
#include <sqlsrvca.h>
#include <sqlsrvda.h>
#include <sqlsrv.h>
```

Compile errors will result if the include files are not in this order.

3.3.11 Allocating Space for SQLSRV_VARCHAR and SQLSRV_VARBYTE Data Types

Be sure to specify the correct length for the SQLSRV_VARCHAR and SQLSRV_ VARBYTE data types in your API applications. Oracle SQL/Services does not issue an error message when the size of the data fields for SQLSRV_VARCHAR and SQLSRV_ VARBYTE data types exceeds the size of the SQLLEN field in the SQLDA data structure. See the *Guide to Using the Oracle SQL/Services Client API* for information on allocating space for the SQLSRV_VARBYTE data type and all other data types.

3.4 Oracle SQL/Services Documentation Errors or Omissions

The following information describes Oracle SQL/Services documentation errors or omissions.

- The *Guide to Using the Oracle SQL/Services Client API* does not describe changes to size and format of integer and floating-point data types

  Beginning with Oracle SQL/Services V5.1, the size and format of some integer and floating-point data types is changed as follows:

  – Trailing zeros occur in fixed-point numeric data types with SCALE FACTOR.

    Trailing zeros are now included after the decimal point up to the number of digits specified by the SCALE FACTOR. In versions of Oracle SQL/Services previous to V5.1, at most one trailing zero was included where the value was a whole number.

    The following examples illustrate the changes using a field defined as INTEGER(3):

```
V5.1 and     Versions previous
higher       to V5.1
--------     -----------------
  1.000        1.0
 23.400       23.4
567.890      567.89
```

  – Trailing zeros occur in floating-point data types. Trailing zeros are now included in the fraction, and leading zeros are included in the exponent, up to the maximum precision available, for fields assigned the REAL and DOUBLE PRECISION data types.

```
                                          Versions previous
Data Type           V5.1 and higher       to V5.1
-----------------   ----------------------  -----------------
REAL                1.2340000E+01           1.234E+1
DOUBLE PRECISION    5.678900000000000E+001  5.6789E+1
```

  – Size of TINYINT and REAL data types is changed.

    The maximum size of the TINYINT and REAL data types is changed to correctly reflect the precision of the respective data types.

    The following table shows the maximum lengths of the data types now and in previous versions:

```
                    V5.1 and    Versions previous
Data type           higher      to V5.1
----------          --------    -----------------
TINYINT              4           6
REAL                15          24
```

- The *Guide to Using the Oracle SQL/Services Client API*

  The Guide does not describe that the SQLSRV_ASSOCIATE( ) service returns SQL error code -1028 when connecting to a database service if the user has not been granted the right to attach to the database.

  When a user connects to a database service, the SQLSRV_ASSOCIATE( ) service completes with the SQL error code -1028, SQL_NO_PRIV, if the user has been granted access to the Oracle SQL/Services service, but has not been granted the right to attach to the database. A record of the failure is written to the executor process's log file. Note that the SQLSRV_ASSOCIATE( ) service completes with the Oracle SQL/Services error code -2034, SQLSRV_GETACCINF if the user has not been granted access to the Oracle SQL/Services service.

4

OCI Services for Oracle Rdb: Release Notes

This chapter highlights release notes that pertain to OCI Services for Oracle Rdb (formerly known as SQL*Net for Rdb) for release 7.2.0.1. It contains information about installation, new and changed features, known problems, software fixes, and documentation changes.

4.1 Software Requirements

OCI Services for Oracle Rdb release 7.2.0.1 requires OpenVMS Alpha Version 7.3-2 or higher, or OpenVMS I64 Version 8.2-1 or higher software.

4.2 Installing OCI Services for Oracle Rdb

The installation for OCI Services for Oracle Rdb is part of the installation for Oracle SQL/Services release 7.2.0.1. Refer to the following documentation for information on installing OCI Services for Oracle Rdb.

- SYS$HELP:SQLSRV072_INSTALL_GUIDE.PDF

  This document is the *Oracle Rdb SQL/Services Installation Guide.* Most of the information required to install OCI Services for Oracle Rdb is in this guide and supersedes the *Guide to SQL*Net for Rdb7.*

4.2.1 Problem Reporting

If an error occurs while you are using OCI Services for Oracle Rdb and you believe that the error is caused by a problem with this Oracle product, contact your Oracle support representative for assistance.

When you experience a reproducible problem, it is important to provide as much detailed information as possible. Use the ALTER SESSION LOG FULL, HEADER [,TIMESTAMP] statement or define SQLNET_DEBUG_FLAGS "HTF" to collect detailed information about

the current OCI Services for Oracle Rdb session. By providing the logged information with your problem report, you supply important data that can help solve the problem. See *Oracle SQL/Services Server Configuration Guide* for more information about using the ALTER SESSION LOG statement and defining the SQLNET_DEBUG_FLAGS logical.

4.3 Upgrading Oracle Rdb Release 7.0 Database to Higher Oracle Rdb Release

Oracle recommends that you convert your database from Oracle Rdb release 7.0 to a higher release using the Oracle Rdb CONVERT functionality, and then upgrade the converted database using the RDB_NATCONNnn command file.

4.4 New and Changed Features for OCI Services for Oracle Rdb Release 7.2.0.1

The following sections describe new or changed features for OCI Services for Oracle Rdb release 7.2.0.1.

4.4.1 New NLS Parameters

The following NLS parameters were added in release 7.2.0.1: NLS_COMP, NLS_ LENGTH_SEMANTICS, and NLS_NCHAR_CONV_EXCP. They will be initialized to default values in V$NLS_PARAMETERS at connection time. The values are as follows: NLS_COMP is set to BINARY, NLS_LENGTH_SEMANTICS to BYTE, and NLS_ NCHAR_CONV_EXCP to FALSE.

4.4.2 Greater Precision in Timestamp for Logging

Beginning with release 7.2.0.1, if you define the logical SQLNET_DEBUG_FLAGS to be HT, the resulting timestamp will show two decimal places of microseconds. The timestamp will be of the following format: YYYY-MM-DD HH:MM:SS.mm.

4.4.3 Data Dictionary support for Oracle 10g application releases

In release 7.2.0.1, the RDB_NATCONNnn.COM database prepare and upgrade functions were enhanced to add several new tables and views in support of Oracle 10g releases of Explorer, Developer and Discoverer.

4.4.4 Changes for Oracle Explorer

Several changes have been made in release 7.2.0.1 that were necessitated by Oracle Explorer release 10g connecting to an OCI Service. The most visible change is that if the user does a "select * from v$version" statement, or an OCI Version call, the first row returned contains the Oracle compatible version information. Oracle Explorer uses that information to identify the Oracle version it is connecting to, and it was confused by the Oracle Rdb version of 7.1 or 7.2. New default information was included in the views ALL_TABLES and ALL_TAB_COLUMNS. A new view ALL_REFS is created with no rows, and a new table SYSTEM_PRIVILEGE_MAP is created containing several privilege definitions that map to functionality allowed by Oracle Rdb. This table is not used by OCI Services for Oracle Rdb or by Oracle Rdb, and the entries do not imply any privileges granted or available.

4.4.5 Changes for Oracle JDBC Release 10.2 Thin Driver

Bug: 5064467

A Java application that uses the Oracle JDBC release 10.2 thin driver to access an Rdb database through the OCI interface would get a "/ by zero" Java exception during the following method call.

```
.prepareStatement( )
```

OCI Services for Oracle Rdb release 7.2.0.1 has been enhanced to support the modified OCI calls used by the Oracle JDBC release 10.2 thin driver.

4.4.6 New Argument for MODIFY_USER Command

In release 7.2.0.1, a new optional argument has been added to the MODIFY_USER function in the RDB_NATCONNnn.COM command procedure. This argument allows users to specify the database in which the user password is updated. The new command format for the MODIFY_USER function is:

```
$ @SYS$LIBRARY:RDB_NATCONN72 MODIFY_USER username newpassword -
$_ oldpassword database
```

where username, oldpassword, and database are optional. Quotation marks ("") can be used as a placeholder for all optional arguments. If a value is not entered for database, the command file reads the file pointed to by the SQLSRV_NATCONN_DBS logical and updates the user's password in each database where the user has previously been entered. The following examples show the use of this new argument:

```
$ @SYS$LIBRARY:RDB_NATCONN72 MODIFY_USER JohnDoe ChangedPwd OldPwd -
$_ Mf_Personnel.rdb
```

```
$ @SYS$LIBRARY:RDB_NATCONN72 MODIFY_USER "" MyNewPwd "" Mf_Personnel.rdb
```

4.4.7 New Datatype: New Formats for Oracle Rowids

Some newer versions of Oracle clients use a new format for sending and receiving rowids and dbkeys. This new format, DTYRDD in Oracle datatype descriptions, is implemented as needed, depending on the version of the client. This feature was made available in releases 7.1.6.1 and 7.2.0.1.

4.4.8 Security Enhancements

Because of security inconsistencies and problems preparing databases with defined default collating sequences, there have been many changes to the Prepare and Upgrade functions in release 7.1.6.1 and 7.2.0.1. There are many new domains named ORA_VCn, and many tables and views are redefined to use these domains. Also, privilege checking will be done by the stored procedures ORA_CREATE_USER and ORA_DROP_USER, so the security requirements are the same whether you use RDB_NATCONNnn.COM or invoke the stored procedures directly. You can add or drop your own username in the database without any privileges, but you must have BYPASS, SECURITY, or SYSPRV privilege to add or drop another user.

4.4.9 New Security Policy

A new and enhanced security policy is in place for release 7.1.6.1 for all ADD_USER, REMOVE_USER, SHOW_USER, and MODIFY_USER functions. The following list explains the changes:

Privileged User

A privileged user is defined as an OpenVMS user who has either SYSPRV, SECURITY, or BYPASS privilege in the account's Authorized Privilege settings. The user's current process privileges and Default Privilege settings are not used.

ADD_USER

The ADD_USER command now supports the no password ("") format for both privileged and non-privileged users. For privileged users the <username> parameter can be any valid OpenVMS user on the system. For a non-privileged user, the <username> must be the same user who is running the RDB_NATCONNnn command procedure

When a new password is supplied, a privileged user issuing the command will cause the password of the OpenVMS account for the user being added to the database to be

updated. A non-privileged user must type in a password that matches the current OpenVMS account password of the user.

REMOVE_USER

A non-privileged user can now remove his username from a NATCONN prepared Rdb database. In previous releases, only privileged users could issue the REMOVE_USER command.

SHOW_USER

In prior releases, only a privileged user could issue the SHOW_USER command. Now, non-privileged users can show all users in any NATCONN prepared Rdb databases.

MODIFY_USER

Both privileged and non-privileged users can update the password of a NATCONN prepared Rdb database user. The difference is that non-privileged users can only update their own user password. For non-privileged users, the new password must conform to the password policy set up on the OpenVMS system (i.e. password length, character requirements, and password history). For privileged users, there is no restriction on the new password.

In both cases, if the old password was supplied, it must match the current account password. If security auditing is turned on, each old password mismatch will trigger an intrusion count and subsequently may lead to a user lockout condition. If the old password was not supplied, the new password for the non-privileged user must match his current OpenVMS account password. This will allow nonprivileged users to synchronize their database and VMS passwords. If the user is privileged, the OpenVMS account password will be changed to the new password.

4.5 OCI Services for Oracle Rdb Problems Fixed for Release 7.2.0.1

This section highlights software errors relating to OCI Services for Oracle Rdb release 7.2.0.1 that have been fixed.

4.5.1 Problem Describing Column With Name Longer Than 30 Characters

Bug: 5632639

Describing a table in SQL*Plus with a column whose name was longer than 30 characters would display an ORA-03113 error and the following error was logged in the OCI executor log file.

```
Assertion failed: "find_mblock(current_mgroup,mblock) == current_mgroup
```

This problem has been corrected in OCI Services for Oracle Rdb release 7.2.0.1. The following error is now correctly displayed in SQL*Plus.

```
ORA-00972: identifier is too long
```

4.5.2 Modified Transaction Control to Better Fit XA Model

With the addition of support for XA 2pc transactions in release 7.1.6, OCI Services for Oracle Rdb and SQL/Services sometimes have a need to mix 2pc and non-2pc transactions. In some cases, these transactions could potentially collide, leading to problems. Transaction control has now been modified to avoid such problems.

One example is when a severe error occurs causing SQL/Services to shut down an OCI executor. In some cases, a database recovery could still be in progress when the SQL/Services shut down occurs. SQL/Services would attempt to rollback and disconnect from the database, causing a bugcheck dump because the database was still locked by the recovery process.

OCI Services for Oracle Rdb and SQL/Services have been modified in release 7.2.0.1 to correct these problems.

4.5.3 ADD_USER Failed When Database Default Character Set Was ISOLATINGREEK

Bug: 5333023

The ADD_USER function, invoked via SYS$LIBRARY:RDB_NATCONNnn.COM, failed when a database had a default character set of ISOLATINGREEK or any character set other than the default DEC_MCS.

The error displayed was:

```
Reason, - no privilege to perform operation on database <db_name>
```

But, the actual problem was:

```
SQL-E-INCCSASS, Incompatible character set assignment between ...
```

This problem has been corrected in release 7.2.0.1.

4.5.4 SELECT INTO Commands Are Stripped of INTO When Passed to Rdb

Bug: 5253380

The INTO clause of a SELECT ... INTO SQL statement was being removed by OCI Services for Oracle Rdb before passing the statement to Rdb. This caused the statement to be processed by opening a cursor, fetching the row, and closing the cursor, rather than just executing the SELECT INTO statement. This caused execution of the statement to be unacceptably slow.

This problem has been corrected in release 7.2.0.1.

4.5.5 ADD_USER Command Does Not Work for Non-Privileged Users

Bug: 5264258

In OCI Services for Oracle Rdb release 7.2, it was documented that users without privileges can add or update their own user name and password information in the USER$ table of a database prepared for OCI Services for Oracle Rdb. The RDB_NATCONNnn.COM utility appeared to have completed with no error, but the user name or password was not added or updated.

This problem has been fixed in release 7.2.0.1. Non-privileged users can now add, remove, and update their own user name and password information using the utility in SYS$LIBRARY:RDB_NATCONNnn. Because these fixes have a new version of the dictionary stored procedures, you must perform an UPGRADE operation on the OCI Services for Oracle Rdb database.

For customers who are calling these stored procedures directly from a SQL script or an application with embedded SQL, the last argument (,:ret_val) is no longer required and should be removed. Beginning with release 7.2.0.1, the call status is returned via the SQLCA.SQLCODE and SQLSTATE variables. All programs should test the return value for errors. A (-1042) SQLCA.SQLCODE and an "O1031" SQLSTATE string indicate that a non-privileged user is trying to make changes for another user. The following example is an excerpt from a .SC application:

```
    char SQLSTATE(6);

    /* Call the stored procedure ORA_CREATE_USER to add/change user/pwd.*/
    EXEC SQL CALL ORA_CREATE_USER(:name, :pass);

    /* Check the return status from the ora_user_password call. */
    if ((SQLCA.SQLCODE == -1042) && (strcmp(SQLSTATE, "O1031") == 0))
       status = CUPP$_E_NOPRIV;  /* failure to change different user w/o privs.
*/
```

4.5.6 Queries with TO_NUMBER() Function Calls Are Slow

Bug: 5027052

Some complex queries that contain calls to Oracle functions TO_NUMBER, TO_CHAR, TO_DATE, DATE_ROUND, or DATE_TRUNC perform much slower on an Oracle Rdb database prepared for OCI Services for Oracle Rdb than on a native Oracle database or a similar query without the function call. This is more apparent on a query that generates hundreds of thousands of the function calls.

This problem has been fixed in release 7.2.0.1 by new optimizing logic. Queries that used to take 90 seconds to complete now take 15 seconds. If you are running on OpenVMS I64, the improvement is even more dramatic. This improvement is especially true for queries that resulted in a large number of calls to the functions from an OCI client such as SQL*Plus.

4.5.7 Random Error Message When SQLNET_DEBUG_FLAGS is HT

When SQLNET_DEBUG_FLAGS was set to HT, random error messages would print in the log during the connect, although the connect completed successfully. The error messages were of the form:

```
ERROR: ORA-26426: Message 26426 not found; product=NATCONN; facility=ORA.
```

This problem has been fixed in release 7.2.0.1.

4.5.8 Query Hangs with a Variable Comparison Using Oracle 10G SQL*Plus

When you used Oracle 10G SQL*Plus, queries using a variable rather than a literal in a comparison would hang. The following is an example of the type of query that would fail:

```
declare x char(3); begin x := 'abc';
select count(*) from table1@rdb1 where column1 = x;
```

This problem has been fixed in release 7.2.0.1.

4.5.9 Failure Upgrading Database After Upgrading to Release 7.1.6 Update03

If a database had been upgraded to release 7.1.6 Update03, attempting to upgrade it to release 7.1.6.1 would fail with the following error:

```
%RDB-E-NO-DUP, index field value already exists; duplicates not allowed for
   ORA_OBJECTS_NAME.
```

You would have to issue a DROP command on the database and then prepare it in order to upgrade to release 7.1.6.1. This problem has been fixed in release 7.2.0.1.

4.5.10 Reference to Obsolete Procedure in Error Message in Log

In releases prior to 7.2.0.1, when a user who was not in the USER$ table in a database tried to connect to the database, an incorrect error message was generated. The reference to the obsolete procedure ORA_SET_PASSWORD has now been replaced by the current procedure ORA_CREATE_USER.

4.5.11 Returning ROWID in an Insert Statement Caused Error ORA-00900

If users included RETURNING ROWID or RETURNING DBKEY in an INSERT statement, the statement would fail with the error:

```
ORA-00900: invalid SQL statement
```

and the executor log would show the error:

```
%SQL-F-SYNTAX-ERR, Syntax error
```

This happened because OCI Services for Oracle Rdb added the clause RETURNING DBKEY INTO :ORA_DBKEY to every INSERT statement. The resulting statements would have two RETURNING DBKEY clauses, and would cause SQL syntax errors. Starting with release 7.2.0.1, OCI Services for Oracle Rdb scans the statement for a RETURNING DBKEY or RETURNING ROWID before it adds the clause to the statement. There still may be some cases, especially if the RETURNING clause includes several elements, where OCI Services for Oracle Rdb will not recognize that a RETURNING DBKEY clause is already in the statement and will try to add it, causing an error. The executor log will show the generated SQL statement, so the user can correct it. It is recommended that the word DBKEY or ROWID be the first in the list of elements to be returned in a RETURNING clause of an INSERT statement. This will help OCI Services for Oracle Rdb recognize that it is already there.

4.5.12 Declare Transaction in SQL Init File Being Overridden

In releases of OCI Services for Oracle Rdb prior to 7.2.0.1, if a DECLARE TRANSACTION statement was executed in the SQL initialization file of a service, it would be overridden by a DECLARE TRANSACTION statement executed later by OCI Services for Oracle Rdb. Toward the end of the connection setup, OCI Services for Oracle Rdb would execute a DECLARE TRANSACTION statement to set the default transaction characteristics to be close to Oracle default transaction characteristics. This would supersede any DECLARE TRANSACTION statement in the SQL initialization file. Starting with release 7.2.0.1, OCI Services for Oracle Rdb recognizes that a DECLARE TRANSACTION statement has been executed and will not execute another one.

4.5.13 Problem with Master/Detail Records

Bug: 5531638

This problem could appear in several different ways. There were a few different errors or, sometimes, the detail records were displayed but were the wrong detail records for the master. Sometimes Forms would display the error:

```
Unable to perform query
```

and sometimes the following error would be seen in the executor log:

```
SQL-F-ILLDATLEN, An invalid SQLLEN(0) was found for a date
```

This problem has been fixed in release 7.2.0.1.

4.5.14 Prefetch in a Pro*C Program Using WHERE CURRENT OF CURSOR

Bug: 5547621

In a prior release, a problem with prefetch (bug 4651271) was fixed to work with Oracle 9.2.0.4. There was a bug in Oracle 9.2.0.4 with prefetching (bug 3512385). When the Oracle bug was fixed, in release 9.2.0.5, the OCI Services fix no longer worked. For this release, there is a fix that will work with Oracle 9.2.0.5 and later. If you are running Oracle 9.2.0.4 as a client, you must define the new logical SQLNET_9204_PREFETCH, which will cause OCI Services to use the earlier fix.

This problem has been fixed in release 7.2.0.1.

4.6 Known Problems and Restrictions

This section highlights problems and restrictions relating to OCI Services for Oracle Rdb and includes workarounds where appropriate.

4.6.1 Support for OpenVMS VAX and Standard Kits

Oracle SQL/Services release 7.2.0.1 and OCI Services for Oracle Rdb release 7.2.0.1 are not supported on OpenVMS VAX.

Standard kit installation is not supported with this version. Only multiversion kits are available.

4.6.2 Restrictions and Limitations

The restrictions and limitations in Table 4–1 apply to this release of OCI Services for Oracle Rdb:

Table 4–1 OCI Services for Oracle Rdb Restrictions and Limitations

Category	Description
Character sets	The Oracle US7ASCII and DEC_MCS character sets are supported by default. See the *Oracle SQL/Services Server Release 7.1.6 Configuration Guide* for information about using other character sets.
Dates	Oracle dates can pre-date the OpenVMS 17-NOV-1858 date. Such dates are not supported by Oracle Rdb or OCI Services for Oracle Rdb.
Transaction reusable services	Oracle SQL/Services transaction reusable services are not supported. Only session reusable services are supported.
Database access by service owner	Oracle SQL/Services services with database access by service owners are not supported. Only services with database access by connect user are supported.
Grant use on OCI services	Grant use on OCI Services for Oracle Rdb is not supported. All users have access to OCI Services for Oracle Rdb. However, the database is still protected because access to the database must be through connect (client) username.
Multischema databases	OCI Services for Oracle Rdb does not support multischema Oracle Rdb databases. However, OCI Services for Oracle Rdb adds an emulation layer that provides a multischema environment that is similar to what you get with Oracle.
Error mapping is not exact	The Oracle error codes do not always map well to Oracle Rdb message vectors. As a result, OCI Services for Oracle Rdb might issue error code 32800. Error code 32800 is a generic server error code that does not convey any information about the error. The text accompanying the message provides more information about handling this error.
SQL dialect	OCI Services for Oracle Rdb sets the Oracle Level1 or Oracle Level2 dialect and requires the dialect to perform its tasks. Changing the dialect may result in unexpected OCI Services for Oracle Rdb failures.
Multisession server is not supported	The Oracle server can support multiple sessions from multiple client applications in one server process. OCI Services for Oracle Rdb cannot because of the locking differences between the two database products.
Cursors	The maximum number of open cursors is currently 998.
ROWID support	ROWID support exists only if there are less than 2.1 billion pages in a logical area. Logical areas greater than 2.1 billion pages are not likely to occur unless you implement VLDB with a single-file database.

Table 4–1 (Cont.) OCI Services for Oracle Rdb Restrictions and Limitations

Category	Description
Piecewise inserts	Piecewise inserts (specific to Oracle SQL) are not supported in this release.
Oracle Rdb does not differentiate between RAW (binary) and VARCHAR (text) data	If you use the VARCHAR data type in place of the RAW data type, your application works with one exception. OCI Services for Oracle Rdb does not convert the binary data to text data if you select the data using the VARCHAR2 data type. You must explicitly convert the data using the HEXTORAW or RAWTOHEX SQL function. Explicitly converting the data works with both Oracle Rdb and Oracle RDBMS.
Determining a NOT NULL constraint through a describe call	If the statement SELECT col1 FROM tab1 were parsed and described in OCI, the Oracle server returns a message that col1 either does or does not have a NOT NULL constraint. OCI Services for Oracle Rdb does not return this information in the describe call. You can, however, obtain this information from the metadata tables. This restriction causes applications like SQL*Plus to always describe a column as not having a NOT NULL constraint when it in fact may have a NOT NULL constraint.

4.6.3 Alterations to Dispatcher Network Port Resets Protocol to SQLSRV

The Oracle SQL/Services dispatcher used to serve Oracle clients must specify an OCI protocol. The installation procedure creates a default dispatcher, OCI_DISP, that you can use to serve OCI clients, or you can optionally create your own OCI dispatcher.

If you use the SQLSRV_MANAGE command line utility or the Oracle SQL/Services Manager GUI to create a dispatcher or to change any network port attributes, you must specify the OCI protocol on the command line. For example, if you alter an existing dispatcher and modify the NETWORK PORT, you also must specify PROTOCOL OCI. Otherwise, the protocol automatically changes from OCI to SQL/Services. The following example shows a command line that specifies an OCI protocol:

Example 4–1 Specifying the OCI Protocol with SQLSRV_MANAGE72

```
$ MCR SQLSRV_MANAGE72
SQLSRV> alter disp oci_disp
_SQLSRV> network_port sqlnet
_SQLSRV> listener "oci_listener"
_SQLSRV> protocol oci;
SQLSRV>
```

If you try to access an Oracle SQL/Services dispatcher defined with the SQL/Services protocol from an OCI client, the connect attempt fails and the following errors are logged in the dispatcher log file:

Example 4–2 Errors Logged to the Dispatcher Log File

```
------------EVENT BEGIN:  EVENT_LOG at Tue Feb 11 1997 16:15:42.456-------------
%SQLSRV-I-EVENT_LOG, event logged at line 838 in file SRVUTL.C;1
%SQLSRV-F-INTERR, Internal error -2007 in Oracle SQL/Services dispatcher
    version V7.1-5 at line 1403 in module SRVPRSMS
%SQLSRV-E-ERROR_TEXT, Error text: invalid packet ID tag 1 or data type 4
    message data 02030401 flag 0
------------EVENT END  :  EVENT_LOG at Tue Feb 11 1997 16:15:42.502-------------

------------EVENT BEGIN:  EVENT_LOG at Tue Feb 11 1997 16:15:42.529-------------
%SQLSRV-I-EVENT_LOG, event logged at line 1062 in file MSG_COM_SQS.C;1
%SQLSRV-E-SQSMSGERROR, Oracle SQL/Services MSG-layer error, client: unknown
%SQLSRV-E-SQSBADPKTHDR, Bad Oracle SQL/Services packet header
------------EVENT END  :  EVENT_LOG at Tue Feb 11 1997 16:15:42.544-------------

------------EVENT BEGIN:  EVENT_LOG at Tue Feb 11 1997 16:15:42.572-------------
%SQLSRV-I-EVENT_LOG, event logged at line 284 in file DISP.C;1
%SQLSRV-W-EXCEPTION_RAISE, Exception raised: DBS_CANCEL_CONNECT
%SQLSRV-I-CONNECTNAME, Connect : CONNECT_0000001
%SQLSRV-I-CONNECTSTATE, Connect state: 1
%SQLSRV-I-NODENAME, Node : unknown
------------EVENT END  :  EVENT_LOG at Tue Feb 11 1997 16:15:42.573-------------
```

To correct this problem, use the SQLSRV_MANAGE command line utility or the Oracle SQL/Services Manager GUI to:

1. Alter your dispatcher and specify PROTOCOL OCI.
2. Stop and restart your Oracle SQL/Services dispatcher so the change can take effect.

4.6.4 OCI Error When Attempting to Fetch a Binary ROWID

The OCI FETCH routine returns the ORA-03106 error when you attempt to fetch a binary ROWID (data type DTYRID) with an OpenVMS Alpha server. The FETCH routine does not work currently on the OpenVMS Alpha server.

To work around this problem, fetch a ROWID in text. This method works well for all servers, including OpenVMS Alpha servers.

4.6.5 SYSDATE Function is Available from a Dblink Connection

The SYSDATE function is available from a dblink connection. When used in this way, SYSDATE is evaluated by the Oracle Rdb Server, which is not standard Oracle semantics. Please be aware of this difference and certain that you want to use SYSDATE in this way.

4.6.6 2PC Transactions Forced To Be ReadOnly

When an OCI service is set up with a non-privileged service owner such as SQLSRV$DEFLT, two-phase commit transactions from OCI clients like Oracle SQL*Plus may be restricted to ReadOnly. This problem can impact OCI client applications that use Oracle database links to access the Oracle Rdb database.

Because all OCI two-phase commit transactions are considered by Oracle Rdb as distributed transactions, the service owner must have DISTRIBTRAN access on the database service. Oracle recommends that DISTRIBTRAN access be granted to all users on databases that may participate in a two-phase commit transaction using an OCI database service, as shown in the following example:

```
SQL> Grant DISTRIBTRAN On Database Alias RDB$DBHANDLE to PUBLIC;
```

5

Previous Releases: New Features and Fixed Problems

This chapter describes the new features and technical changes to previous releases of Oracle SQL/Services and OCI Services for Oracle Rdb. It also describes problems that were fixed in these products.

5.1 New and Changed General Features in Previous Releases of Oracle SQL/Services

This section highlights new and changed general features that were added in previous releases.

5.1.1 Oracle SQL/Services Available on OpenVMS I64

Oracle SQL/Services release 7.2 is available on the HP OpenVMS Industry Standard 64 for Integrity Servers operating system.

5.1.2 New Logical SQLSRV$MAX_EXECUTOR_FAILURES

In Oracle SQL/Services releases prior to release 7.1.6, an executor would fail if two startup failures occurred. If there were no other active executors for the service at the time of the second failure, the service would also be shut down.

For example, if a database was closed for maintenance, any user who attempted to connect to that database through a service would get an executor startup failure. After two such failed users through an executor of that service, the executor would be shut down.

In Oracle SQL/Services release 7.1.6, support for a new system logical was added to allow users to control the maximum number of executor startup failures allowed before an

executor fails. The logical SQLSRV$MAX_EXECUTOR_FAILURES can be defined to change the maximum failures from the default of 2 to any positive integer value. In this way, a user can control how often executors and services shut down during routine database maintenance.

By defining the following with Oracle SQL/Services release 7.1.6 or later, a user will ensure that an executor will not be terminated until at least 10 failures have occurred.

```
$ define SQLSRV$MAX_EXECUTOR_FAILURES 10
```

5.1.3 Logical SQLSRV$LOG_CONNECTIONS

Beginning with release 7.1.6, you can define the logical SQLSRV$LOG_CONNECTIONS to "NO" so that successful connections are not logged to dispatcher log files, and the size of the dispatcher log files is reduced. If the logical is undefined or assigned to any other value, the successful connections are logged. Because this logical is evaluated when a dispatcher is started, the dispatcher must be restarted if the logical is changed.

5.1.4 Oracle Server Two-Phase Commit Support

Beginning with release 7.1.6, Oracle SQL/Services provides two-phase commit support with the following capabilities:

- Oracle Rdb databases can fully participate in Oracle RDBMS-managed distributed transactions.
- Multiple Oracle server DBLINKs to Rdb databases can participate in a transaction.
- The DECdtm XA Gateway provides an interface between the Oracle distributed transaction protocol and DECdtm distributed transaction protocol.

Perform the following steps to enable the two-phase commit protocol:

- Define the following logical in your service process initialization file:

  ```
  $ DEFINE RDB$DDTM_XG_INFO gateway-name
  ```

  where gateway-name is the name specified in the CREATE_LOG/GATEWAY_NAME command in XGCP, the XA Gateway control program.

- Add the following command to the SQL initialization file used by the CREATE SERVICE and ALTER SERVICE commands:

  ```
  ALTER SESSION SET TX_MODE NOWARN_1PC
  ```

Note: Two-phase commit functionality works on all versions of OpenVMS beginning with version 7.3-2 and higher, if the required patch kit is installed. Refer to the Software Requirements section for more information.

5.1.5 Linux Client Available

Beginning with release 7.1.6, Oracle SQL/Services supports the Linux API client as well as Windows, HP Tru64 Unix, and OpenVMS clients. For prerequisite and installation information for the Linux client, read the Linux_client_readme.txt file included with the Linux kit, found in sqslinuxcli.tar. The Linux client kit is available on the CD for the Oracle Rdb client kits, as well as on MetaLink and OTN.

5.1.6 Enhancements to SQLSRV$DEINSTALL_DELETE Procedure

Beginning with release 7.1.6, the SQLSRV$DEINSTALL_DELETE.COM procedure no longer removes the SQLNET4RDB identifier. If the command procedure is executed on a cluster, it now displays information about all SQL/Services monitors that are currently executing on the cluster prior to asking if the user would like to continue, as shown in the following example:

```
$ @SYS$MANAGER:SQLSRV$DEINSTALL_DELETE

Oracle SQL/Services versions currently installed on your system

1 Version 7.1 (Multiversion)
0 Quit

Enter Choice to deinstall (0...1) : 1

You are about to deinstall Oracle SQL/Services 7.1

This procedure will delete SYS$MANAGER:SQLSRV$SHUTDOWN71.COM.
If the Oracle SQL/Services 7.1 monitor is running on any
other node on your cluster besides the node SYS1,
you will have to manually stop the monitor on each of these other
nodes after this procedure has finished.

The following monitor(s) are currently executing on your cluster.

OpenVMS V7.3-2 on node SYS2 4-JUN-2004 14:34:23.61 Uptime 10 01:23:29
```

```
Pid Process Name State Pri I/O CPU Page flts Pages
20400126 sqlsrv_mon_0071 HIB 6 2020 0 00:00:00.79 399 523

OpenVMS V7.3-2 on node SYS3 4-JUN-2004 14:34:23.63 Uptime 19 20:39:17
Pid Process Name State Pri I/O CPU Page flts Pages
2080025D sqlsrv_mon_0071 HIB 6 599 0 00:00:25.98 1321 3030

Enter Y(ES) to continue to deinstall Oracle SQL/Services 7.1 :
```

5.1.7 Logical SQLSRV_EXEC_LOG

A new logical name was added in release 7.1.5.6 to allow you to disable the service log for all services:

```
$DEFINE/SYSTEM SQLSRV_EXEC_LOG NOLOG
```

This logical must be defined before a service is started. Oracle Corporation recommends that you do not disable the service log because it is needed if a problem occurs.

5.1.8 Configuration File Version Check

Beginning with release 7.1.5.6, if you try to use a configuration file created by Oracle SQL/Services prior to release 7.1.5.4, you will get a bugcheck dump and an error message will be written to the monitor log file.

5.1.9 New and Changed SQLSRV_MANAGE Features

The following new capabilities and logical names have been added:

5.1.9.1 Command Line Recall Function

Beginning in release 7.1.5.2, you can use the up arrow and down arrow keys at the SQLSRV> prompt to recall any of the last 20 commands that were entered.

5.1.9.2 EXTRACT Command

The EXTRACT command was added in release 7.1.5.6. It extracts definitions that are written to an Oracle SQL/Services command script. The script can be used to re-create servers, dispatchers, and services.

FORMAT

```
EXTRACT keyword [option]
```

Keyword can be one of the following:

- Server

  Extracts the definition for the server. There is no option for this keyword.

- Disp [disp_name]

  If the disp_name is omitted, definitions for all dispatchers are extracted. If the disp_name is specified, the definition for just that dispatcher is extracted.

- Service [service_name]

  If the service_name is omitted, definitions for all services are extracted. If the service_name is specified, the definition for just that service is extracted.

EXAMPLES

The following example extracts the definition for the service OCI_AAA and displays the information on the output device.

```
$ SQLSRV_MANAGE71
SQLSRV> extract service OCI_AAA;
Create Service OCI_AAA
    Owner                    'AAA'
    Protocol                 OCI
    SQL version              7.1
    Autostart                off
    Process_initialization  'DBD_USER6:[JONES]INIT_OCI_ENG70.COM'
    ATTACH 'filename DBD_USER6:[JONES]mf_personnel'
    Reuse scope is           SESSION
    Database Authorization   CONNECT USERNAME
    SQL_init_file             'DBD_USER6:[JONES]init.sql'
    Application Transaction Usage   SERIAL
    Idle Executor Timeout    1800
    Min Executors             1
    Max Executors             10
    Clients Per Executor      1
    ;
Grant use on service OCI_AAA
    To  'AAA'
    ;
SQLSRV>
```

The following example extracts definitions for the entire dispatcher definition and writes them to an output file.

```
$ SQLSRV_MANAGE71 -OUTPUT A.SQL
SQLSRV> extract disp;
SQLSRV> exit

$TYPE A.SQL
Create Dispatcher SQLSRV_DISP
    Autostart                 off
    Max connects              100
    Idle User Timeout         0
    network_port DECnet object   81           protocol SQLServices
    network_port tcpip  port_id  118          protocol SQLServices
    Log path            'SYS$MANAGER:'
    Dump path           'SYS$MANAGER:'
    ;
```

5.1.9.3 ALTER DISPATCHER and CREATE DISPATCHER

In release 7.1.5, the ALTER DISPATCHER and CREATE DISPATCHER commands were enhanced to include LOG PATH and DUMP PATH arguments. These arguments allow you to specify a directory name for log and bugcheck dump files, for example:

```
SQLSRV> ALTER DISPATCHER SQLSRV_DISP1 LOG PATH
       'USER1:[SQLSRV_TEST1.AAA]'
_SQLSRV> DUMP PATH 'USER1:[SQLSRV_TEST2.BBB]';
%SQLSRV-S-ALTER_RESTART, Restart object to have altered settings take effect
```

If you specify NOLOG instead of a directory name for the LOG PATH qualifier, no log file is written, for example:

```
SQLSRV> CREATE DISPATCHER SQLSRV_DISP1 LOG PATH 'NOLOG'
_SQLSRV> DUMP PATH 'SYS$MANAGER';
```

Oracle recommends that you create a log file for troubleshooting purposes unless you have a problem with excessive I/O entries in the log file. The default directory for both qualifiers is SYS$MANAGER.

The LOG PATH and DUMP PATH arguments are available when you use the SQLSRV_MANAGE command utility, but not when you use the SQLSRV_MANAGE GUI tool.

If values are assigned to existing logicals SQLSRV_DISP_LOGPATH and SQLSRV_DISP_DUMPPATH, they override log path and dump path values specified by the LOG PATH and DUMP PATH arguments.

The SHOW DISPATCHER command has been enhanced to display the values for Log path and Dump path:

```
SQLSRV> SHOW DISP SQLSRV_DISP;
Dispatcher SQLSRV_DISP
    .
    .
    .
Log Path:           USER1:[SQLSRV_TEST1]
Dump path:          USER1:[SQLSRV_TEST1]
     .
     .
     .
** This Dispatcher will be updated as follows when it is restarted **
    Log path:              USER1:[SQLSRV_TEST1.AAA]
    Dump path:             USER1:[SQLSRV_TEST2.BBB]
```

In release 7.1.5, a new logical, SQLSRV_DISP_LOGPATH, was added. It allows you to specify the location of the dispatcher log directory as the following example shows:

```
$ DEFINE/SYSTEM/EXEC SQLSRV_DISP_LOGPATH DKA100:[USER1.LOG]
```

The SQLSRV_DISP_LOGPATH logical name has to be defined as a system logical name. If you do not define the logical name, the default directory for dispatcher log files is the SYS$MANAGER directory.

Once you define the SQLSRV_DISP_LOGPATH logical name, you have to restart the dispatcher.

In release 7.1.5, a new logical, SQLSRV_DISP_DUMPPATH, was added. It allows you to specify the location of the dump file directory as the following example shows:

```
$ DEFINE/SYSTEM/EXEC SQLSRV_DISP_DUMPPATH DKA100:[USER1.DUMP]
```

The SQLSRV_DISP_DUMPPATH logical name has to be defined as a system logical name. If you do not define the logical name, the default directory for dump files is the SYS$MANAGER directory.

Once you define the SQLSRV_DISP_DUMPPATH logical name, you have to restart the dispatcher.

5.1.9.4 ALTER DISPATCHER and CREATE DISPATCHER

In release 7.1.5, the argument PROTOCOL [NATIVE | OCI | SQLSERVICES] was added to the ALTER DISPATCHER and CREATE DISPATCHER commands.

5.1.9.5 ALTER SERVICE and CREATE SERVICE

In release 7.1.5, three new arguments were added to the ALTER SERVICE and CREATE SERVICE commands:

- PROTOCOL [OCI | RMU | SQLSERVICES]
- DEFAULT_CONNECT_PASSWORD <password-string>
- OWNER_PASSWORD <password-string>

5.1.9.6 GRANT USE ON SERVICE

In release 7.1.5, a new keyword, GROUP[S] <group-name>, was added to the GRANT USE ON SERVICE command.

This new keyword appears in the <grant-element> syntax and supports granting the USE privilege descriptor to a group and permits access to the specified service by users who are members of that group.

5.1.9.7 REVOKE USE ON SERVICE

In release 7.1.5, a new keyword, GROUP[S] <group-name>, was added to the REVOKE USE ON SERVICE command.

This new keyword appears in the <grant-element> syntax and supports revoking the USE privilege descriptor from a group to remove access to the specified service by users who are members of that group.

5.1.9.8 SHOW

In release 7.1.5, all SHOW commands were updated to display the new arguments and keywords that have been added to each command.

5.1.10 New and Changed Client Features

The following describes new and changed client features:

- Deprecated clients no longer ship with Oracle Rdb clients

  All deprecated Oracle SQL/Services clients no longer ship on the Oracle SQL/Services software kit.

- Windows 98, Windows 2000, Windows ME, and Windows XP

  Windows 98, Windows 2000, Windows ME, and Windows XP are now supported operating systems for Oracle SQL/Services release 7.1.6.

5.1.11 New and Changed Server Management Features

The following describes new and changed server management features:

- Windows 98, Windows 2000, Windows ME, and Windows XP

  Windows 98, Windows 2000, Windows ME, and Windows XP are now supported operating systems for Oracle SQL/Services Manager.

5.2 Oracle SQL/Services Errors Fixed in Prior Releases

The following known problems found in the Oracle SQL/Services OpenVMS server in previous releases have been fixed.

5.2.1 PROCESS_INIT Defined as Keyword LOGIN Could Fail

Bug: 4664833

Starting an executor with PROCESS_INIT defined as keyword LOGIN would sometimes fail with garbage in the login command file name.

This problem was corrected in release 7.1.6.1.

5.2.2 Monitor Aborts When Connection Cancelled

If you attempted to use the monitor port for an OCI connection and then cancelled the process, the monitor would abort.

This problem was fixed in release 7.2.

5.2.3 Shared Memory Not Released with Continuous Start and Shutdown of Server

Free shared memory would decrease to zero if services were started and stopped continuously. This problem was fixed in release 7.1.6.

5.2.4 Erroneous RDB$_NO_PRIV Errors Using OCI Services

In releases prior to 7.1.5.9.1, users would sometimes get erroneous RDB$_NO_PRIV messages connecting to an OCI service. This problem was fixed in Oracle SQL/Services and SQL*Net for Rdb releases 7.1.5.9 and 7.1.5.9.1, and Oracle Rdb release 7.0.7.2.

5.2.5 Monitor Crashes with "bind mon: unexpected monitor state"

Under rare timing conditions it was possible for the SQL/Services monitor to crash with a bugcheck stack trace similar to the following:

```
%DBS-F-BUGCHECK: in DBS01:[SQLSRV_BUILD.NOV_05_2003.DBS_SRC]CMD.C;1 at line
11003 bind mon: unexpected monitor state
```

This problem would occur when multiple SQLSRV_MANAGE clients were sending "system management" requests to the SQL/Services monitor. If one of those clients sent a request that required many buffers of data for the monitor to respond, and that client was slow in reading the messages sent by the monitor, causing all available buffers for that client to be consumed, the monitor would process messages from other clients until the slow client had read its buffers. If the monitor had stalled on a slow client process, and a new client connection "CONNECT SERVER command" arrived while the monitor was stalled on the slow process, a bugcheck dump would occur and the monitor process would fail.

The bugcheck would occur because the routine that handled new connection requests assumed that there could not be a current client connection active when processing a new connection request. When this routine saw that condition it would force a bugcheck and monitor termination. However, this condition is possible and acceptable and thus there is no reason to bugcheck.

This bugcheck exception was removed in release 7.1.5.9.

The incidence of this problem can be reduced if only one SQLSRV_MANAGE process is run at one time.

5.2.6 OCI Universal Services Not Correctly Impersonating the Connect User

In Oracle SQL/Services release 7.1.5.8, OCI universal services with database authorization defined as CONNECT USER were not correctly impersonating the connect user. Therefore, Rdb external functions would be executed under the service owner, rather than the connect user account.

This problem was corrected in release 7.1.5.9.

5.2.7 Poor Performance From OCI Queries

SQL/Services Release 7.1.5.8 introduced a problem that could cause queries from an OCI source to sometimes take an extraordinarily long time to complete. Examination of the SQL/Services processes would show that they were idle even though the client had not received a response from SQL/Services.

This problem was introduced by changes in the underlying OracleNet libraries employed by SQL/Services. There were instances where a network message would arrive but notification of that arrival was not being delivered to the SQL/Services dispatcher. When that occurred SQL/Services would only see that a new network message had arrived when it did periodic polling.

There was no workaround for this issue.

Notification of network message arrival is now done immediately in SQL/Services releases 7.1.6.2 and 7.2.0.1.0.

5.2.8 Upgrade of Oracle SQLNET Libraries

Note to Reviewers: release 7.1.6.2 says OracleNet release 9.2.0.4; this note says 9.2.0.2

Releases of Oracle SQL/Services prior to 7.1.5.8 included Oracle SQLNET release 2.1.5 libraries. Oracle SQL/Services release 7.1.5.8 has been upgraded to include Oracle Net release 9.2.0.2 libraries on OpenVMS Alpha. Any restrictions or problems experienced using Oracle SQL/Services that were due to Oracle SQLNET release 2.1.5 library problems, which have subsequently been fixed in Oracle Net release 9.2.0.2, have now been solved.

As a result of this upgrade, the restriction of a maximum of 30 concurrent connections for an OCI service has now been lifted for Oracle SQL/Services release 7.1.5.8 on OpenVMS Alpha.

5.2.9 Memory Leak of Monitor Process

The memory leaks of the monitor process with the following SQLSRV_MANAGE commands were fixed in release 7.1.5.8:

CONNECT SERVER

DISCONNECT SERVER

SHOW commands

EXTRACT commands

DROP commands

ALTER commands

5.2.10 Executing External Routines from Universal OCI Services

With Oracle SQL/Services releases prior to release 7.1.5.8, universal OCI services would sometimes require that the connect user have SYSPRV or GRPPRV in order to successfully execute an external routine.

This problem has been fixed in Oracle SQL/Services release 7.1.5.8.

5.2.11 Support for Multiline Statements in SQL Initialization File

Prior to SQL/Services release 7.1.5.8, SQL statements in the SQL initialization file were required to be on one line, and were restricted to a length of 512 characters. This restriction has been removed. Multiline statements are now supported in the SQL initialization file.

A hyphen must be used as a continuation character at the end of a line in the initialization file to indicate that the SQL statement continues on the following line. The limit of length for one line in the initialization file is 512 characters, so if the SQL statement exceeds 512 characters, you must use additional lines.

5.2.12 Increased ENQLM Quota

Beginning with OpenVMS Version 7.1, any process that has its ENQLM set to 32767 is granted almost unlimited ENQLM. Beginning with release 7.1.5.6, SQL/Services processes have this capability. If the service owner has its ENQLM set to 32767, the service process can have ENQLM of 16776959.

5.2.13 Severity of Error Changed

Beginning with release 7.1.5.6, severity of the error SQLSRV-E-FLOBEXECUNAVAIL has been changed from -E- to -I- to reflect its correct status as an informational message.

5.2.14 SERVICE_NAME Now Allowed in TNSNAMES.ORA

Beginning with release 7.1.5.6, the logical SERVICE_NAME has been allowed in TNSNAMES.ORA, for example:

```
cra_univ1 =
      (DESCRIPTION =
           .
           .
           .
```

```
        (CONNECT_DATA =
           (SERVICE_NAME = oci_eng_univ_cra
            .
            .
            .
)
```

5.2.15 Installation Procedure Overwrote Configuration File

In versions of SQL/Services prior to release 7.1.5.4, the installation procedure overwrote the configuration file during installation.

This problem has been fixed if you are upgrading from multiversion to multiversion, which is the only version that has been supported since SQL/Services release 7.1.5.

If you are upgrading from a release 7.1.* standard version to release 7.1.5.4 or later, the installation still overwrites the configuration file. To work around this problem: follow these steps prior to running the installation procedure:

- Rename the configuraton file or move it to another location.
- Run the SYS$MANAGER:SQLSRV$DEINSTALL_DELETE.COM command file (the routine that overwrites the configuration file).
- If you are upgrading from release 7.1 or later, move the configuration file back to its original location or rename back to its original name and append ’71’ to the end of the filename.
- Run VMSINSTAL to finish the upgrade to 7.1.5.6.

  Oracle Corporation recommends that you copy the configuration file (SYS$MANAGER:SQLSRV_CONFIG_FILE*nn*.DAT) to have as a backup before installing a new release of Oracle SQL/Services.

5.2.16 Preattached Database Service with Default_connect_username Could Bugcheck

In versions prior to 7.1.5.4, if you created a preattached database service and included a default_connect_username, the dispatch could bugcheck with an access violation when you attempted to connect.

5.2.17 Impersonation Is Supported

PERSONA support was a new feature in Oracle Rdb release 7.1. This feature enables the full impersonation of an OpenVMS user. As of release 7.1.5.2, impersonation is now supported in Oracle SQL/Services.

5.2.18 Logicals Now Defined for ORA_NLS, ORA_NLS32, and ORA_NLS33

In versions of SQL/Services prior to release 7.1.5.2, users who wanted to define a language other than AMERICAN had to define the logical names ORA_NLS, ORA_NLS32, and ORA_NLS33 to point to message files and other NLS files.

SQLSRV_MON71.EXE now contains definitions for all the ORA_NLS*nn* logical names and users no longer need to define them.

5.3 New and Changed Features for Previous OCI Services for Oracle Rdb Releases

This section highlights new and changed features for previous OCI Services for Oracle Rdb releases.

5.3.1 SQLNET_RECO_USER No Longer Required

The logical SQLNET_RECO_USER, which used to be required for users of distributed transactions, is no longer required in release 7.1.6.1 if you are using the XA Gateway to DDTM. The recovery processes do not attach to the database, so they no longer need a username and password.

5.3.2 Improved Logging

Many changes were made to OCI Services for Oracle Rdb logging in release 7.1.6.1, mostly to make the logging information more concise and the log files smaller. When the SQLNET_DEBUG_FLAGS logical is defined as HT, the timestamp for each item logged is at the beginning of the log information and no longer on a separate line. Logging of SQLDAs and GTADAs is also more concise.

5.3.3 OCI Services for Oracle Rdb Available on OpenVMS I64

OCI Services for Oracle Rdb release 7.2 is available for the HP OpenVMS Industry Standard 64 for Integrity Servers operating system.

5.3.4 Support for New ALTER SESSION SET CONSTRAINTS Syntax

OCI Services for Oracle Rdb release 7.1.6 supports the ALTER SESSION SET CONSTRAINTS syntax added in Oracle Rdb release 7.1.2. Refer to the Oracle Rdb documentation for details on this new feature.

5.3.5 Date-Time Data Type Enhancements

The following date-time data types were implemented in release 7.1.6:

- TIME data type
  If structured date datatypes have been turned on, columns can be defined as type TIME.
- INTERVAL DAY-TO-SECOND data type
  If structured date datatypes have been turned on, columns can be defined as type INTERVAL DAY <size> TO SECOND <size>.
- INTERVAL YEAR-TO-MONTH data type
  If structured date datatypes have been turned on, columns can be defined as type:
  - INTERVAL YEAR<size> TO MONTH <size>
  - INTERVAL YEAR<size>
  - INTERVAL MONTH <size>

Two logical names have been added in release 7.1.6:

- SQLNET_TIMESTAMP_DATE_TYPE
  You can define this logical to 'Y' or 'y' in order to use the TIMESTAMP datatype.
- SQLNET_STRUCTURED_DATE_TYPES
  You can define this logical to 'Y' or 'y' in order to use the structured date data types.

5.3.6 ALTER SESSION Enhancements

The ALTER SESSION statement accepts the following new syntax beginning with release 7.1.6:

- ALTER SESSION SET SQLNET_TIMESTAMP_DATE {ON | OFF}
  Allows the interactive ability to turn on and off the usage of TIMESTAMP data types. The result is the same as setting the SQLNET_TIMESTAMP_DATE_TYPE logical.
- ALTER SESSION SET SQLNET_STRUCTURED_DATE_TYPES {ON | OFF}
  Allows the interactive ability to turn on and off the usage of structured date types. The result is the same as setting the SQLNET_STRUCTURED_DATE_TYPES logical.

- ALTER SESSION SET SQLNET_DEBUG_FLAGS {values}
  Allows the interactive ability to set the debug flag values for debug logging purposes. The result is the same as setting the SQLNET_DEBUG_FLAGS logical.

5.3.7 Emulate Oracle release 9.2.0.4

OCI Services for Oracle Rdb release 7.1.6 emulates Oracle release 9.2.0.4 on OpenVMS Alpha.

5.3.8 Dictionary Prepare/Upgrade/Drop Program Replaces Scripts

In release 7.1.6 and later of OCI Services for Oracle Rdb, the RDB_NATCONN71_ PREPARE/UPGRADE/DROP scripts have been replaced by a program, RDB$NATCONN_ DICnn. This program prepares a database that has not been previously prepared, upgrades a database previously prepared at release 1.0.2.5 and later to the current version, and drops all NATCONN objects created by any version of the scripts. The RDB$NATCONN_DICnn program is invoked by the RDB$NATCONNnn.COM command file. For information on using the command file, see the *Oracle SQL/Services Installation Guide*.

5.3.9 More Efficient Dictionary Queries

This release of OCI Services for Oracle Rdb has changed the way many of the dictionary queries are done, both to Oracle Rdb tables and to the NATCONN tables. This may be especially apparent in faster connect/attach times.

5.3.10 Hidden Objects in OCI Services for Oracle Rdb

The OCI Services for Oracle Rdb data dictionary program, RDB$NATCONN_DICnn.EXE, makes use of a feature, first available in Oracle Rdb release 7.1.3, that defines all the NATCONN objects created by the program as hidden objects. They will not appear in the output of a SHOW TABLES command; if you want to see them you must enter the command SHOW SYSTEM TABLES (or SHOW SYSTEM MODULES, etc).

If you are running a version of Oracle Rdb before release 7.1.3, the behavior will be the same as it is currently; that is, tables, modules, etc., will appear as user objects in the database, rather than as system objects. If you run the upgrade option rather than the prepare option, you will get a mixture of hidden and not hidden objects, depending on what version you are upgrading from. In that case, you must enter a SHOW TABLES and a SHOW SYSTEM TABLES command to see all the NATCONN tables. Oracle recommends that you drop all NATCONN objects from your database, and use the prepare option. You can save usernames and passwords from your USER$ table in a new table, then prepare the database, and insert the usernames and passwords back into the new USER$ table.

5.3.11 New Tables in Oracle Data Dictionary

The following three tables and views are created by the OCI Services for Oracle Rdb data dictionary program: DBA_USERS, DBA_DB_LINKS, and DBA_SYS_PRIVS. These tables and views did not get created in versions of OCI Services for Oracle Rdb prior to release 7.1.6.

5.3.12 Enhanced RDB_NATCONNnn.COM File

Beginning with release 7.1.6, the RDB_NATCONNnn file can be used to modify, drop, and show users as well as add them. See the *Oracle SQL/Services Server Configuration Guide* for information about how to use these capabilities.

5.3.13 Optional Validation of Connecting Program Name

New functionality in release 7.1.6 allows sites to restrict which programs are allowed to access each database through OCI Services for Oracle Rdb. When the database is prepared or upgraded for OCI Services for Oracle Rdb, a new table, ORA_VALID_PROGRAMS, is created. It has two columns, USERNAME and PROGRAM. These columns are used in a LIKE comparison to validate the user and program that are connecting. To activate this functionality you must define a new logical SQLNET_VALIDATE_PROGRAM as Y or y in the process initialization file for the service. The table ORA_VALID_PROGRAMS allows select access to public but insert, update, and delete only to SQLNET4RDB. Therefore, a user must have the SQLNET4RDB identifier or SYSPRIV or BYPASS privilege to insert rows into the table. If the logical is defined, OCI Services for Oracle Rdb checks at connection time that there is an entry in the ORA_VALID_PROGRAMS table that matches the user and program that are connecting. Entries in the table must use the syntax of a LIKE comparison; that means that an entry of '%' in the USERNAME column would allow any user. An entry of %SQLPLUS% in the PROGRAM column would allow SQL*Plus from any platform. Both columns of ORA_VALID_PROGRAMS must contain data for each row. An entry of '%' in both columns would allow any user from any program to connect; not defining the logical SQLNET_VALIDATE_PROGRAM has the same effect. All other validation and security checking is still done; this will NOT allow anyone access to the database without all required privileges. It can only restrict usage by some or all users to a particular program or set of programs.

The logical SQLNET_VALIDATE_PROGRAM can be defined as Y or y to turn on program validation, so that the program name is validated against the program name and username entries in the ORA_VALID_PROGRAMS table. This definition allows clients that do not send their program name to connect. In order to disallow null program names, define the logical as "NONULL" or "nonull". The program name check is case sensitive, so it may be

necessary to include an entry for %SQLPLUS% and %sqlplus% in the ORA_VALID_PROGRAMS table.

5.3.14 New Users Visible Using OEM Console

In release 7.1.6 and later, if you add users of OCI Services for Oracle Rdb to an Oracle Rdb release 7.1 or later database by using the ADD_USER function of the RDB_NATCONNnn command procedure, they are now visible through the OEM console. A CREATE USER SQL command is executed by the ADD_USER function, which adds new users to the RDB$PROFILE table and makes them visible when you use the OEM console.

5.3.15 Milliseconds Supported for TIMESTAMP and INTERVAL Data Types

Milliseconds are supported in release 7.1.6 and later if structured date types are turned on. You must issue one of the following commands:

```
ALTER SESSION SET STRUCTURED_DATE_TYPES ON
```

or

```
DEFINE SQLNET_STRUCTURED_DATE_TYPES Y
```

You must also issue the command:

```
ALTER SESSION SET NLS_TIMESTAMP_FORMAT = "yyyy-mm-dd hh24:mi:ss.ff"
```

5.3.16 New Error Message for Unregistered Usernames

In release 7.1.5.3 and later, a new error message, ORA-1017, is returned at logon time if the username is not registered in the database. This message replaces the old message, ORA-1010. A message is put in the log file that says the username is not registered and that RDB_NATCOMM.COM should be run to add the username to the database. This message can be ignored if the application successfully connects.

5.3.17 Thin JDBC Access to Oracle Rdb Databases

Starting with SQL*Net for Rdb release 7.1.5, access to Oracle Rdb databases is supported through thin JDBC.

5.3.18 Support for Hot Standby

Beginning with release 7.1.5, SQL*Net for Rdb supports Hot Standby databases.

5.3.19 Compatibility with Oracle Forms

If you use Developer/2000 prior to release 2.0 or prior to Oracle Forms release 5.0, you must set the following properties in your Oracle Forms module for compatibility with SQL*Net for Rdb:

- Set the Module property "Savepoint Mode" to FALSE
- Set the Data Block property "Locking Mode" to DELAYED

You can modify the properties using the Property Palette or use the Property Class RDB_PROPERTY_CLASS of the file RDB_PROP.FMB provided in the Oracle SQL/Services examples directory on your server.

Beginning with Developer/2000 release 2.0 or Oracle Forms Release 5.0, these properties will be handled automatically during run time by Developer/2000.

5.4 Software Errors Fixed in Previous OCI Services for Oracle Rdb Releases

This section highlights software errors fixed in previous OCI Services for Oracle Rdb releases.

5.4.1 SQL Statement with WHERE CURRENT OF CURSOR Clause Failure

Bug: 4651371

Because of a change to the format of the dbkey/rowid datatype, a SQL statement that included the WHERE CURRENT OF CURSOR clause would fail in one of two ways:

- The update operation would appear to succeed.There would be no error message, but the update would not be applied. This is because the UPDATE ... WHERE ROWID = :rowid command would not find a match for the given rowid so no update would be done.
- The following SQL run-time error would be returned:

```
SQL-02118 Invalid row for a WHERE CURRENT OF operation
```

  Cause: An attempt was made to reference a nonexistent row using the CURRENT OF clause in an UPDATE or DELETE statement. This happens when no FETCH has executed or when FETCH returns a "no data found" error that the program fails to trap.

This problem was corrected in release 7.1.6.1.

5.4.2 Problem Preparing a Database with Default Collating Sequence

Bug: 4889032

If a database was created with a default collating sequence, and you used the SYS$LIBRARY:RDB_NATCONNnn.COM PREPARE command to prepare it for use with OCI Services for Oracle Rdb, the process would fail with the following errors:

```
Rdb error: %RDB-E-NO_META_UPDATE, metadata update failed
-RDB-E-CONVERT_ERROR, invalid or unsupported data conversion
-RDMS-F-UNLIKECOLL, fields of unlike collating sequence may not be compared
```

This problem was corrected in release 7.1.6.1.

5.4.3 OCI Services Executor Process Could Go Into a CPU Bound Loop

Bug: 4889032

Periodically, OCI Services for Oracle Rdb executor processes would go into a CPU bound loop, after you upgraded from SQL/Services release 7.1.5.9.1 to 7.1.6.

This problem was corrected in release 7.1.6.1.

5.4.4 SQLDA Logged Excessively

Starting with OCI Services for Oracle Rdb release 7.1.6, the SQLDA was logged multiple times unintentionally.

This problem was corrected in release 7.1.6.1. The SQLDA will only be logged once.

5.4.5 Using a Single Quote in NLS_NUMERIC_CHARACTERS Causes an Error

Bug: 4180259

When you are connected to an Oracle Rdb database from SQL*Plus using OCI Services for Oracle Rdb, the following command results in a "SQL-F-UNTSTR, Unterminated string found" error.

```
ALTER SESSION SET NLS_NUMERIC_CHARACTERS = ",'";
```

This problem also occurs when you use any numeric character string literals that contain any single quote characters.

This problem was corrected in release 7.1.6. All valid nls_numeric_characters strings that work with SQL*Plus also work when you use OCI Services for Oracle Rdb.

5.4.6 RDB_NATCONNnn Does Not Exit as Expected

Bug: 4465288

In OCI Services for Oracle Rdb release 7.1.6, the command procedure RDB_NATCONNnn.COM did not exit at the completion of the operation, if you supplied all of the parameters.

This problem was corrected in release 7.1.6.

5.4.7 The MODIFY_USER Command Does Not Work

In release 7.1.6, if you modified a user password using the MODIFY_USER command in the RDB_NATCONNnn.COM command file, the process failed with an "Old password is incorrect ..." or "%DCL-W-MAXPARM, too many parameters..." error.

This problem was corrected in release 7.1.6.

5.4.8 RDB_NATCONNnn Always Updates OpenVMS Password

Bugs: 4864338, 4870313

In release 7.1.6, you could not add a user to an Oracle Rdb database prepared for OCI Services for Oracle Rdb without modifying the OpenVMS password of the user. The documented format, using the ADD_USER feature with a blank password, did not work.

This problem was corrected in release 7.1.6. If you do not specify a password, the ADD_USER feature now adds a new user into the database with a random password, and the OpenVMS password is not updated.

5.4.9 RDB_NATCONNnn Upgrade Fails with %COSI-E-RNF Error

Bugs: 5054528, 5016115

Upgrading an Oracle Rdb release 7.1 database (that had been prepared with SQL/Services release 7.1.5.9) to release 7.1.6 using the release 7.1.6 RDB_NATCONNnn command procedure would sometimes fail with a "Rdb error: %COSI-E-RNF, record not found" error. This failure could prevent the Oracle Rdb database from being converted to release 7.1.6. As a result, it was not usable on a system that was upgraded from SQL/Services release 7.1.5.9 to release 7.1.6. In addition, if you tried to perform the ADD_USER command, you would get an "%RDB-E_READONLY_TRANS, attempt to update during a read-only transaction" error.

This problem was corrected in release 7.1.6.

5.4.10 Cursor Name Cxxx Has Already Been Declared

Bug: 5009963

When a number of cursors were allocated and some of them were freed, you could get the error "Cursor Name Cxxx already declared" when you opened a new cursor. There were also errors when you tried to free some of the cursors that had been allocated.

This problem was corrected in release 7.1.6.

5.4.11 RDB_NATCONNnn Fails with COSI-E-RNF

Bug: 4759991

When you upgraded a database that had been prepared but had no users added to it, the upgrade would fail with a COSI-E-RNF error.

This problem was corrected in release 7.1.6.

5.4.12 Trimming Cursors Caused SYSTEM-ACCVIO

Bug: 4584207

If you used many cursors and tried to free some of the allocated cursors, you could get an access violation.

This problem was corrected in release 7.1.6.

5.4.13 ORA-02052 Error Updating Rdb Table with Trigger in Distributed Transaction

Bug: 4765574

If you used a distributed transaction, updates on Oracle Rdb tables that had triggers defined could fail with a transaction error. This error was caused because the trigger was not evaluated before the transaction was committed. With 2-pc support, constraints are now evaluated before issuing the commit.

This problem was corrected in release 7.1.6.

5.4.14 Inserting Blobs Larger than 100,000 Bytes Fails

Bug: 4768399

The default maximum size for segmented strings is 100,000 bytes. If you are creating blobs larger than 100,000 bytes, you must define the logical SQLNET_MAXLONGRAW to be the

size of the largest blob you are creating. The error reporting for this problem was enhanced to include the fix for the problem.

This problem was corrected in release 7.1.6.

5.4.15 Bugcheck at Attach

Bug: 5048788

Some newer versions of Oracle clients send a function of type 135 (Client ID Propagation), which OCI Services for Oracle Rdb did not handle correctly. This could cause a bugcheck and an access violation at attach.

This problem was corrected in release 7.1.6.

5.4.16 RDB_NATCONNnn Does Not Work for DB Owner without SYSPRV

Bug: 4864564, 4870232

If you had SELECT access to an RDB database prepared for OCI Services for Oracle Rdb, but did not have SYSPRV privilege, you could not perform an ADD_USER function to add yourself to the database. The same problem occurred with the REMOVE_USER function.

This problem was corrected in release 7.1.6. You will still need BYPASS, SYSPRV, or SECURITY privilege to perform these operations for users other than yourself.

5.4.17 SQL-F-FIELD_EXISTS Error When Preparing After Dropping

Bug: 5111196

This problem only happened when using update 3 of release 7.1.6. Many new domains were created when preparing a database for OCI Services for Oracle Rdb but were not dropped when doing a drop.

This problem was corrected in release 7.1.6.1.

5.4.18 ORA-01456 Connecting to OCI Services if Table DUAL Modified to Real Table

Bug:5074191

Because a row is inserted into table DUAL at connect time within a read only transaction, if table DUAL has been recreated as a real table rather than a temporary table, the insert fails with the error ORA-01456. With release 7.1.6, a row will be inserted into DUAL only if there are no rows in DUAL already, so if DUAL is a real table and there are no rows in it,

this error will still occur. But if DUAL is a real table with at least one row in it, the connection will succeed.

This problem was corrected in release 7.1.6.

5.4.19 OCI Service Hangs

Bug: 4898806

Users doing a SELECT ... FOR UPDATE OF operation could see a service hang using up to 100% of CPU. This was caused by a memory overrun that caused corruption in the chain of memory blocks, and there would be an endless loop trying to free memory.

This was fixed in release 7.1.6.1.

5.4.20 Invalid ROWID Messages

There were two cases where the ORA-01410 invalid ROWID error message occurred. The error occurred if you were updating data from a row selected in a form; when you clicked on the update button, the error message appeared. The error was also seen if you were running a procedure using Procedure Builder. In both cases, the declare cursor statement contained the clause "for update of" and the update statement contained the clause "where current of <cursor>". This problem has been fixed in release 7.1.6.

5.4.21 Unable to Connect to Multiple Databases Using a Universal Service

Various errors were displayed if you tried to connect to multiple databases in a universal service. Sometimes the Rdb error RDB-F-REQ_WRONG_DB was returned from the connect statement; sometimes it appeared that SQL*Plus had connected, but references to the second database failed. This problem has been fixed in release 7.1.6.

5.4.22 Error ORA-03106 Returned Instead of Expected ORA-01722 Error

When a character value was entered into a NUMBER field, error ORA-03106 was displayed instead of the correct error, ORA-01722. This has been corrected in release 7.1.6.

5.4.23 Additional Error Messages in Oracle Forms

Additional error messages made Oracle Rdb errors too long to fit in standard error windows in Oracle Forms. This problem has been fixed for the ORA-0001 (index field value already exists) and ORA-02290 (check constraint violated) errors in release 7.1.6.

5.4.24 TIMESTAMP Data Type Caused Error

Use of the TIMESTAMP data type resulted in an unimplemented or unreasonable conversion requested error. This has been corrected in release 7.1.6.

5.4.25 Error Using DESCRIBE Command

The DESCRIBE command failed when it was used on a table that was associated with a database link. This has been corrected in release 7.1.6.

5.4.26 Error Storing Null Value

Storing a null value in a column with the LIST OF BYTE VARYING data type caused a protocol violation. This has been corrected in release 7.1.6.

5.4.27 DOUBLE PRECISION Column Converted Incorrectly

A DOUBLE PRECISION column in an Rdb table was converted to NUMBER(53) instead of FLOAT(53) when it was accessed through OCI Services for Oracle Rdb. This has been corrected in release 7.1.6.

5.4.28 JDBC Errors

The following JDBC errors have been fixed in release 7.1.6:

- Protocol violation - An error occurred when a null value stored in a column with the LIST OF BYTE VARYING datatype was retrieved via the ResultSet.getBinaryStream method.
- Invalid datatype in OCI call - An error occurred when an attempt was made to store a value into a timestamp column via the PreparedStatement.setTimestamp method or when an attempt was made to store a null into a timestamp column via the PreparedStatement.setNull method.

5.4.29 Incorrect Oracle Errors Returned to OCI Clients

In releases prior to 7.1.5.9.1, if users connecting to an OCI universal service got an error during logon, instead of an Oracle error code Oracle SQL*Net for Rdb returned an Rdb error code to the OCI client.

This led to the Oracle client displaying the Oracle error text corresponding to the Rdb error code, which bore no relation to the actual error condition. The errors most often displayed were the following.

```
ORA-01028: internal two task error
ORA-00001: unique constraint violated
```

This problem has been fixed in Oracle SQL/Services / SQL*Net for Rdb release 7.1.5.9.1. The OCI client now displays the correct Oracle error.

5.4.30 Rdb Errors Unintentionally Suppressed

In releases prior to 7.1.5.9.1, if users connecting to an OCI database service got an error during logon, a "Database not setup..." message was added to the error stack, causing the error reported by Rdb to be pushed off the error stack and not be displayed. The following is an example of the resulting errors reported to the user.

```
ERROR: ORA-01031: insufficient privileges
Database not setup correctly for Oracle SQL*Net for Rdb.
For details, look in Oracle SQL/Services executor log file: ...
```

This problem has been fixed in Oracle SQL/Services / SQL*Net for Rdb release 7.1.5.9.1. The OCI client will now display the Rdb error.

5.4.31 Displayed Release Type Corrected

With releases prior to 7.1.5.8, Oracle SQL*Net for Rdb always displayed "Development" as the release type for Oracle Rdb OCI Server, regardless of the type of kit installed. This could be seen in the SQL*Plus welcome banner or by invoking the "SELECT * FROM V$VERSION" command.

This problem has been fixed in Oracle SQL*Net for Rdb . "Production" will now be displayed for production releases.

5.4.32 Detection of Repeated Intrusion Attempts

Code was added to SQL*Net for Rdb to detect repeated intrusion attempts in release 7.1.5.8. This code is activated by a user whose username and/or password are not in the USER$ table of the Rdb database. Such a user may be locked out of the system, even if he has a valid OpenVMS account, if he exceeds the parameters set at the site for intrusion detection and denial.

5.4.33 ADD_USER Failure

In releases prior to 7.1.5.8, if an ADD_USER function was performed after a DROP_USER function was performed to delete a user, the ADD_USER function would fail.

5.4.34 Prestarted Transactions Left Open After a Commit Is Issued

Performing a two-phase commit transaction with an RDBMS database and an Rdb database always leaves a pre-started transaction open on the Rdb database after a commit is issued. This potentially stalls an online RMU/BACKUP command unless the qualifier is specified. This has been fixed by a change to the ORA_TRANS module which will be executed by running the script RDB_NATCONN_UPGRADE[72].SQL. The module ORA_TRANS will be replaced with a fixed version.

This problem was corrected in release 7.1.5.8.

5.4.35 Using Dblink to Service Where NLS_LANG is Defined as Other Than Default

A change was made in release 7.1.5.8 to fix the problem of using a dblink to a service where NLS_LANG is defined as other than the default.

Example:

```
% sqlplus scott/tiger

SQL*Plus: Release 8.1.7.0.0 - Production on Thu Apr 10 15:32:11 2003
@ (c) Copyright 2000 Oracle Corporation.  All rights reserved.

Connected to:
Oracle8i Enterprise Edition Release 8.1.7.2.0 - Production
With the Partitioning option
JServer Release 8.1.7.2.0 - Production
SQL> select * from v$version@rdb71;
select * from v$version@rdb71
*
ERROR at line 1:
ORA-02068: following severe error from RDB71
ORA-03113: end-of-file on communication channel
Excerpt from the log file:
Client data for fetch item 4
Client data: column_name <NAME>
Client data: type 1, length 17, nul/ind 0, precision 0, scale 0 Client text
<SPECIAL FUNCTIONS> Rows fetched(1) are: 2
Function: CLOSE
gta: gtaclse: release3(1)
```

```
Function: OK2RPC
statement......(1): begin call v$nls_set_func(); end


Prescribe.
gta: gtaclse: release3(1)
%SYSTEM-F-ACCVIO, access violation, reason mask=00, virtual address=000000000
0000464, PC=00000000003EF0C0, PS=0000001B

  Improperly handled condition, image exit forced.
    Signal arguments:  Number = 0000000000000005
                        Name  = 000000000000000C
                                0000000000000000
                                0000000000000464
                                00000000003EF0C0
                                000000000000001B
```

5.4.36 Schemas Defined for Users in the USER$ Table

As of release 7.1.5.8, all users in the USER$ table, as well as all users who are creators of tables, have schemas defined for them.

5.4.37 Using a Bind Variable Twice in a SQL Statement

There was a problem when using a bind variable twice in a SQL statement; that problem has been fixed in release 7.1.5.8.

Following is an example of a query that would cause the error:

```
SQL> select sh.employee_id, sh.salary_amount
  2  from salary_history sh
  3  where sh.employee_id = :p_employee_id
  4  and sh.salary_amount = (select max (sh2.salary_amount) from salary_history
sh2 where sh2.employee_id = :p_employee_id); select sh.employee_id, sh.salary_
amount
*
ERROR at line 1:
ORA-03115: unsupported network datatype or representation
```

5.4.38 Testing for a Read-only Database

A change was made in the login code in release 7.1.5.8 to test for a read-only database in a different way that does not cause possible privilege errors. These errors would be seen in a

log file, although they didn't cause the connection to fail. This privilege error at check_read_only will no longer happen.

5.4.39 FILLM Quota Problem

When using a language other than the default (American), fillm quota would be used up by disconnecting and connecting to a service.

This problem was corrected in release 7.1.5.8.

5.4.40 RDB_NATCONNnn Problem Fixed

There was a problem when the command file RDB_NATCONNnn.COM was used in an environment where a sql initialization file (sqlini) was defined and contained an attach statement. Sqlini is now defined to be nl: in the command file, as of release 7.1.5.8.

Example:

```
SQLUSER71> define sql$database mf_personnel
SQLUSER71> define sqlini "dkd100:[sqluser71]sql.ini"
(NOTE: sqlini file contains attach 'filename sql$database';)
SQLUSER71> @sys$library:rdb_natconn
  Operation (prepare/upgrade/drop/adduser): adduser
  Database: sql$database
  Username: sqluser70
@    Password:
@    Password Verification:
  Username:
  This alias has already been declared.
  %SQL-F-DEFDBDEC, A database has already been declared with the default

alias
  %RDB-E-EXTFUN_FAIL, external routine failed to compile or execute successfully
@    -RDMS-E-RTN_ERROR, routine "ORA_ENCRYPT_PASSWORD" generated an
error
during
        execution
  -SYSTEM-F-ACCVIO, access violation, reason mask=00, virtual
address=000000000000
  0010, PC=00000000019B3DB8, PS=0000001B
```

5.4.41 Problem with Universal Service Fixed

When using a universal service, the username would be set to the service owner rather than the connecting user.

This was fixed for release 7.1.5.8.

5.4.42 Memory Leak Related to dblink Fixed

There was a memory leak when doing many inserts using a dblink. This was fixed for release 7.1.5.8.

5.4.43 Connection Problem Fixed

If an error occurred when a user connected to an executor, the next user who tried to connect would get the following error:

```
Assertion failed: "find_mblock(current_mgroup,mblock) == current_mgroup"
in file NATCONN$SRC_V071573:[CODE]GTME.C;1 at line 468.
```

This problem was fixed in release 7.1.5.8.

5.4.44 Read Only Transactions Started During Connection

There was a problem in which SQL*Net for Rdb would start a READ ONLY transaction during connection. this caused problems for databases running with snapshots deferred. This problem has been fixed so that SQL*Net for Rdb now starts a transaction in an appropriate mode, as determined by Rdb.

This problem was fixed in release 7.1.5.8.

5.4.45 SQL*Plus Invocation Failed with some NLS_LANG Definitions

Some versions of SQL*Plus, when invoked with NLS_LANG defined to be something other than the default (AMERICAN_AMERICA), would cause later connection attempts by other client tools to fail. For example, the following error would be returned to a JAVA program:

```
 Exception thrown java.sql.SQLException: ORA-00900: invalid SQL statement ,8
E85wwÈ_F E  zE85
```

The log might also show a result similar to the following:

```
>>>>> NEW SESSION USER: scott PROGRAM: JDBC AT tue oct 07 17:14:37
<<<<<
```

```
>>>>> GTA compiled at Apr 25 2003 10:52:43
gtaschema:: schema: <NULL>, object: ORA_OBJECTS, action: REF_TABLE
statement......: 0"D?
```

This problem was fixed in release 7.1.5.8.

5.4.46 Insufficient Memory or Quota Problem

In V7.1.5.6. of SQL*Net for Rdb, there was a problem during cleanup, when a user disconnected, that prevented resources from being properly released. This problem could manifest itself as insufficient memory, or insufficient BYTLM or FILLM quotas.

This problem was corrected in release 7.1.5.7.

5.4.47 Problem Storing and Retrieving Long Raw Data

Using the thin JDBC driver with SQL*Net for Rdb prior to release 7.1.5.7 , there was a problem in both storing and retrieving long raw (image) data. These problems have been fixed in release 7.1.5.7,with the following restrictions:

- Image data must be defined as BinaryStream data, and should be stored using setBinaryStream and retrieved using getBinaryStream.
- For the thin JDBC driver, use of the 'blob' datatype is not yet supported. The logical SQLNET_BLOB must not be defined or must be defined as "N".
- The default maximum size of LONG RAW data is 100,000 bytes. If your long raw data is longer or significantly shorter than that, you should define the new logical SQLNET_MAXLONGRAW to the value of your longest long raw data. Long raw data must be stored in a single insert statement and retrieved in a single get statement. The SQLNET_MAXLONGRAW value is used to allocate the buffer to hold the data, so it must be large enough to hold the entire value of the image data.

5.4.48 Problem Retrieving Non-English Long Raw Data

In releases prior to 7.1.5.7, when using a language other than English, data defined as LONG RAW was not being fetched correctly. Image data was being converted as if it were character data in the case where NLS_LANG was defined to be other than English.

5.4.49 Problem Retrieving Long Raw Data Shorter Than 255 Bytes

In previous releases, image or LONG RAW data that was shorter than 255 bytes was not being fetched and returned correctly. Following is an example of a JAVA return showing the error:

```
D:\TAR\1141042\testThin901\TESTCASE>java TestLongRawRead
java.sql.SQLException: Io exception: Protocol violation at
oracle.jdbc.dbaccess.DBError.throwSqlException(DBError.java:180) at
oracle.jdbc.dbaccess.DBError.throwSqlException(DBError.java:222) at
oracle.jdbc.dbaccess.DBError.throwSqlException(DBError.java:335) at
oracle.jdbc.dbaccess.DBDataSetImpl.getStreamItem(DBDataSetImpl.
java:1157)
```

5.4.50 Problem Starting Read/Write Transactions

With previous releases, users were not able to start a read/write transaction when using a character set other than DEC_MCS. Following is the error the user would see:

```
SQL> insert into tab1 values (1);
insert into tab1 values (1)
*
ERROR at line 1:
ORA-01456: may not perform insert/delete/update operation inside a
READ ONLY transaction
```

The executor log file shows the error:

```
gtaschema:: schema: <NULL>, object: ORA_OBJECTS, action: REF_TABLE
Rdb operation..: EXECUTE IMMEDIATE - CHECK_READ_ONLY
Rdb error...(0): %SQL-F-INCCSCMP, Incompatible character set
comparision between NAME and <value expression
```

5.4.51 Change to OCI Services for Oracle Rdb Logging

In previous releases, OCI Services for Oracle Rdb turned on logging during user connection and logon. Starting with Release 7.1.6, OCI Services for Oracle Rdb will no longer turn on logging, so executor log files will not contain the same information as in prior versions. The behavior will be the same as prior versions if users turn on logging.

If logging is turned on and a SET SESSION AUTHORIZATION statement is being logged, as it always is for universal services, only "SET SESSION AUTHORIZATION" will be logged, not the rest of the statement which contains the username and password.

5.4.52 Persona Nopriv Error Using SQL*Plus and Other OCI Clients

Non-privileged users could not connect from SQL*Plus even though they were granted database access. The connect would succeed when 'persona support is disabled' but fail when 'persona support is enabled'.

The following example shows the problem:

```
UAF> show joe_nobody
Username: JOE_NOBODY                        Owner:
Account:                                    UIC:    [424,7] ([JOE_NOBODY])
CLI:      DCL                               Tables: DCLTABLES
Default:  DISK$DKD600:[JOE_NOBODY]
LGICMD:
Flags:
Primary days:   Mon Tue Wed Thu Fri
Secondary days:                     Sat Sun
No access restrictions
@ Expiration:            (none)    Pwdminimum:  6   Login Fails:     0
@ Pwdlifetime:           (none)    Pwdchange:  23-APR-2002 07:39
@ Last Login: 21-MAY-2002 11:39 (interactive),            (none)
(non-interactive)
Maxjobs:         0  Fillm:       100  Bytlm:        64000
Maxacctjobs:     0  Shrfillm:      0  Pbytlm:           0
Maxdetach:       0  BIOlm:       150  JTquota:       4096
Prclm:           8  DIOlm:        60  WSdef:         2000
Prio:            4  ASTlm:       250  WSquo:         4000
Queprio:         4  TQElm:        10  WSextent:     16384
CPU:        (none)  Enqlm:      2000  Pgflquo:      50000
Authorized Privileges:
  NETMBX       TMPMBX
Default Privileges:
  NETMBX       TMPMBX
Identifier                      Value           Attributes
  JOE                           %X80010015
  READ_ONLY                     %X80010016

SQL> show prote on database rdb$dbhandle
Protection on Alias RDB$DBHANDLE
    (IDENTIFIER=SQLNET4RDB,ACCESS=SELECT+INSERT+UPDATE+DELETE+SHOW+CREATE+ALTER+
      DROP+DBCTRL+OPERATOR+DBADM+SECURITY+DISTRIBTRAN)
    (IDENTIFIER=JOE,ACCESS=SELECT+UPDATE)
    (IDENTIFIER=READ_ONLY,ACCESS=SELECT)
    (IDENTIFIER=[*,*],ACCESS=NONE)
```

Error from SQL*Plus when connecting as joe_nobody (after the service has started successfully):

```
ERROR: ORA-01031: insufficient privileges

Error in executor log file:
Rdb operation..: EXECUTE IMMEDIATE - LOGIN2
```

```
Rdb error...(0): %RDB-E-NO_PRIV, privilege denied by database facility
```

Possible workarounds include giving the user more privileges or rights, or disabling persona 'security checking is external (persona support is disabled)'.

This problem was corrected in Release 7.1.5.6.

5.4.53 Connections to SQL*Net for Rdb Would Hang

Connections to SQL*Net for Rdb from Oracle version 8.1.7 would sometimes hang. This problem was corrected in release 7.1.5.6.

5.4.54 DESCRIBE of an Object Reported Datatype UNKNOWN

A DESCRIBE of an object, especially a procedure or function, often reported datatype UNKNOWN. The correct datatype is reported in release 7.1.5.6 and later.

5.4.55 NLS_CHARACTERSET Ignored in ALTER SESSION Command

If an ALTER SESSION command included the symbol NLS_CHARACTERSET, the command reported success but did not change the character set. The value of NLS_CHARACTERSET is determined at connection time and cannot be changed during the session. The command now returns the error ORA-12705 in release 7.1.5.6 and later.

5.4.56 Memory Leaks

Some memory leaks were fixed in release 7.1.5.6 through 7.1.5.9.1.

5.4.57 Fatal Error When Running Queries Longer Than 16K

If you ran queries that were longer than 16 K through SQL*Plus and SQL*Net for Rdb, the process was truncated, and a fatal error was generated:

```
ERROR at line 1:  ORA-09100: Message 9100 not found; No message file
for product=NATCONN, facility=GTW
%SQL-F-UNTERM_C_STR ,Missing null terminator for C string
```

This problem has been fixed for release 7.1.5.6.

5.4.58 Error Accessing Rdb Database Using Dblink

In previous releases, if you used Oracle 9.0.1 and accessed an Rdb database using a dblink, you could see the following error:

```
ORA-00600: internal error code, arguments: [qctstc2o1], [1], [0],
[31],[96],
[2], [2], []
```

The following two SQL statements are the types of statements that would see the error:

```
insert into ora_table (select *  from rdb_table@rdb.world);
create table ora_table as select  col1 from rdb_table@rdb.world;
```

This problem was fixed in release 7.1.5.5.

5.4.59 Running an Rdb Stored Procedure Using a Database Link

In versions prior to release 7.1.5.4, when you ran an Oracle Rdb stored procedure using a database link, it was possible to get the following error:

```
SQL> execute insert_col@rdb('abcde');
BEGIN insert_col@rdb('abcde'); END;
*
ERROR at line 1:
ORA-01401: inserted value too large for column
ORA-06512: at line 1
```

In the course of fixing this error, some changes were made to the parameter handling and conversion implementation. These changes can sometimes cause the following error to appear when the stored procedure is first run:

```
SQL> execute insert_col@rdb('abcde');
BEGIN insert_col@rdb('abcde'); END;
*
ERROR at line 1:
ORA-04068: existing state of packages has been discarded
ORA-04062: %s of %s has been changed
ORA-06512: at line 1
```

This error can be ignored; the procedure will run correctly when run again.

5.4.60 Protocol Mismatch Error

In versions prior to 7.1.5.4, you could see a protocol mismatch error when you did a select from a table that had an index, as shown in the following example:

```
SQL> select count(*) from test_optim@rdb;
```

```
select count(*) from test_optim@rdb
                     *
ERROR at line 1:
ORA-02072: distributed database network protocol mismatch
```

5.4.61 SQL*Plus Hangs Using SQL*Net for Rdb

If you had run the RDB_NATCONN_DBMSAPPL_PREPARE or the RDB_NATCONN_DBMSOUTPUT_PREPARE script on your Rdb database, SQL*Plus Version 9.0.1.0.1 would hang trying to connect using SQL*Net for Rdb. This no longer occurs as of release 7.1.5.4.

5.4.62 Thick JDBC Driver Returned Error

In versions prior to 7.1.5.4, it was possible that the thick JDBC driver would return an error on the second of two select statements:

```
***** ACCESS BAD *****
java.sql.SQLException: ORA-01010: Ungultige (=invalid)
OCI-Operation
     at
oracle.jdbc.dbaccess.DBError.throwSqlException(DBError.java:168)
     at
oracle.jdbc.oci8.OCIDBAccess.check_error(OCIDBAccess.java:1597)
     at
oracle.jdbc.oci8.OCIDBAccess.parseExecuteDescribe(OCIDBAccess.java:798)
```

The SQL*Net for Rdb log would show the following:

```
Rows fetched(1) are: 1
AL7 close [opncrs=0 GTOD_ALL->allopt=2048 err=1403
Function: Cancel All
Oracle error: 1010 for function: Cancel All
Function: Cursor close all
```

5.4.63 Select of Column Returned Data in Wrong Format

In versions of SQL*Net for Rdb prior to release 7.1.5.4, if you were using Rdb V7.0xx, a select of a column using TO_CHAR and a column alias could display the data as hex instead of characters:

```
sql> select to_char(1) from dual;
    shows the value 1
```

```
sql> select to_char(1) col_alias from dual;
    shows the value 31
```

5.4.64 Two-phase Commits Using Both Rdb and Oracle Databases

When several users were doing two-phase commits using a dblink including both an Rdb database and an Oracle database, they could see locks on the ORA_COMM_TRANS table that caused users to be blocked.

This problem was fixed in release 7.1.5.4.

5.4.65 Dblink Failures Fixed

The following dblink problems in SQL*Net for Rdb releases 7.1.5 and 7.1.5.1 were fixed in release 7.1.5.3:

- Taking an update lock on a table through dblink caused the connection to abort.

  The SQL statement that failed was:

  ```
  SELECT * FROM tab@dblink WHERE col='xxx' FOR UPDATE OF col2
  ```

  The error returned was:

  ```
  ERROR at line 1:
  ORA-03113: end-of-file on communication channel
  ```

- Creating a view through a dblink caused the connection to abort.

  The SQL statement that failed was:

  ```
  CREATE VIEW vu AS SELECT * FROM tab
  ```

  The error returned was:

  ```
  ERROR at line 1:
  ORA-03113: end-of-file on communication channel
  ```

- SQL functions such as USER and SYSDATE did not work correctly through a dblink.

  The SQL statement that failed was:

  ```
  SELECT * FROM tab@dblink WHERE col=USER
  ```

  The error returned was:

  ```
  ORA-00911: invalid character
  ```

```
%SQL-F-BAD-TOKEN, is not a valid SQL language element
```

5.4.66 Better Performance of SQL*Net for Rdb Startup

For releases prior to release 7.1.5.3, the startup time for a SQL*Net for Rdb session was very slow, especially for databases with many tables. For example, the following query was slow:

```
SELECT COUNT(*) from ORA_OBJECTS where NAME not in (SELECT TRIM (RDB$RELATION_
NAME from RDB$RELATIONS) and OTYPE = 1;
```

This query was intended to help keep the table, ORA_OBJECTS, synchronized with the tables that were actually in the database. The stored procedure, ORA_DELETE_ PHANTOMS, was called if the count in the query was greater than 0. Since the query was taking so much time, and almost always returning 0, it was decided to remove the query from the startup and remove the call to the stored procedure, ORA_DELETE_PHANTOMS. It is now recommended that the stored procedure, ORA_DELETE_PHANTOMS, be run periodically, especially if tables are being created or dropped from both SQL and SQL*Plus. The following example shows how to invoke the stored procedure:

```
SQLPLUS> begin ora_delete_phantoms(); end;
SQLPLUS> /
```

If the application did not require strict schema emulation, the workaround was to add the following line to the SQL initialization file for the service:

```
ALTER SESSION SET SCHEMA EMULATION RELAXED;
```

This would disable the schema emulation mechanism.

5.4.67 Change Maximum CHAR and VARCHAR Sizes

For Oracle8, the maximum length of CHAR and VARCHAR datatypes changed from what the maximum was for Oracle7. Maximum character length changed from 255 to 2000; maximum VARCHAR length changed from 2000 to 4000. This affects the datatype mappings for SQL*Net for Rdb, since CHAR and VARCHAR strings larger than the Oracle maximum are mapped to long or blob. This change was made in release 7.1.5.3.

5.4.68 Fixed Metadata Retrieval Using JDBC

Using JDBC, the metadata query, meta.getColumns, did not return data about any of the tables in the database except for GLOBAL_NAME.

The following example shows the JDBC code for the metadata query that was failing. Running this query for any table, other than GLOBAL_NAME, would produce no results. Following the code fragment is part of the result set for the ALL_TABLES table.

```
DatabaseMetaData meta = conn.getMetaData();
ResultSet c = null;
c = meta.getColumns(null, null, args[0], "%");
 while (c.next()) {

 System.out.println("\t" + c.getString(1) + ", " +
        c.getString(2) + ", " + c.getString(3) + ", " +
        c.getString(4) + ", " + c.getString(5) + ", " +
        c.getString(6));
        }
        c.close();
```

Result Set:

```
    null, SYS, ALL_TABLES, OWNER, 12, VARCHAR2
    null, SYS, ALL_TABLES, TABLE_NAME, 12, VARCHAR2
    null, SYS, ALL_TABLES, TABLESPACE_NAME, 12, VARCHAR2
    null, SYS, ALL_TABLES, CLUSTER_NAME, 12, VARCHAR2
    null, SYS, ALL_TABLES, IOT_NAME, 12, VARCHAR2
 ...
```

This problem was corrected in release 7.1.5.3.

5.4.69 Access Violations Fixed

The following access violation bugs were fixed in release 7.1.5.3 of SQL*Net for Rdb:

- In previous releases when you attempted to connect from Oracle SQL*Net, you received an ORA-00022 error message (invalid session id; access denied). This error no longer occurs and you can now connect from Oracle SQL*Net.
- The access violation that occurred when the special recovery user logged on has been fixed.

5.4.70 Using Bind Variable Twice in a Query

If you used a bind variable twice in a query, you might have seen the error ORA-01006.

This problem was corrected in release 7.1.5.3.

5.4.71 ANSI_DATE and SYSDATE Functions Work Correctly

In releases prior to 7.1.5.3, the use of ANSI_DATE and SYSDATE functions failed unless you used them in conjunction with CAST or the TO_DATE, or TO_CHAR functions.

The following SQL statements now work:

- SELECT * FROM TEST_DATE WHERE ANSI_DATE = '01-jan-1999';
- SELECT * FROM TEST_DATE WHERE ANSI_DATE = SYSDATE;
- UPDATE TEST_DATE SET ANSI_DATE = SYSDATE;
- INSERT INTO TEST_DATE (ANSI_DATE) VALUES (SYSDATE);
- DELETE FROM TEST_DATE WHERE ANSI_DATE = SYSDATE;
- UPDATE TEST_DATE SET VMS_DATE = TO_DATE(SYSDATE);
- SELECT * FROM TEST_DATE WHERE ANSI_DATE = TO_DATE ('01-JAN-1999') - 1;

5.4.72 Wrong Error Returned When Inserting a Duplicate Value into a Unique Index

In releases prior to 7.1.5.3, an incorrect error message was being returned when you attempted to insert a duplicate value into a unique index. The incorrect error message returned was:

```
ORA-09100: Target system returned following message
```

The ORA-09100 error message was followed by the RDB_E_NO_DUP error message.

The following error message is now returned which is the correct Oracle error:

```
ORA-00001, unique constraint (<table_name>.<index_name>) violated
```

5.4.73 Embedded SQL Program Using SET TRANSACTION Works Correctly

In releases prior to 7.1.5.3, an embedded SQL program that performed a SET TRANSACTION statement returned an SQL_F_BAD_TXN_STATE error message.

5.4.74 ALL/USER_TAB_COLUMNS Return Correct Scale & Precision

Beginning with release 7.1.5.3, the ALL_TAB_COLUMNS and USER_TAB_COLUMNS views created from the RDB_NATCONN_PREPARE.SQL script return the correct scale and precision for smallints and integers withdecimal points.