

# Oracle9i Application Server

Client CD-ROM Release Notes

Release 2 (9.0.2) for Windows

April 2002

**Part No. A97356-01**

These *Release Notes* contain important last minute information not included in the online documentation library.

This document summarizes the differences between Oracle9iAS Database Client and its documented functionality.

**See Also:** *Oracle9i Application Server Release Notes*

## 1 Certification and System Requirements

This section describes Oracle9iAS Database Client system and accessibility requirements.

### 1.1 Windows 95 Support

Oracle9i Server and Oracle9i Client software is not supported on Windows 95. However, you can continue to use an Oracle8i Client or Oracle8 Client on Windows 95 to connect to an Oracle9i Database.

## 2 Documentation

This section describes Oracle9iAS Database Client documentation requirements.

### 2.1 Corrections to the Oracle9iAS Client CD (9.0.2) Documentation

Oracle9i is not supported on Windows 95. Ignore occurrences of support for Windows 95 in the Oracle9i documentation library.

**ORACLE®**

Copyright © 2002 Oracle Corporation.  
All Rights Reserved.

Oracle is a registered trademark, and SQL\*Plus, Oracle9i, Oracle8i, and Oracle8 is a trademark or registered trademark of Oracle Corporation. Other names may be trademarks of their respective owners.

## 2.2 Accessing Documentation on the CD-ROM

Some manuals referenced in this document are not on the Oracle9iAS Documentation CD-ROM. You can view them at:

<http://tahiti.oracle.com>

## 3 Installation Issues

This section describes Oracle9iAS Database Client installation issues.

- When mounting a CD-ROM on a Novell drive to a Windows NT client and running Oracle Universal Installer, the following error may appear:

The Java Runtime Environment was not found at *location* hence the Oracle Universal Installer cannot be run.

Because the Novell server does not support long filenames, the directory path for JRE\_LOCATION is not recognized (some of the directory names have more than eight characters). Therefore, this configuration is not supported. Try mounting the CD-ROM locally or on a Windows NT Server.

- On systems with a combination of Windows NT 4.0 Service Pack 5.0 and an ATI display driver, a memory dump occurs when the Oracle Universal Installer window is moved around the screen. Selecting a smaller display for the screen may reduce the frequency of this occurrence. This problem is a display driver conflict and no Windows NT patch is available at this time.
- When installing Oracle9i on dual boot systems, the software uses the same physical space on both operating systems. Therefore, any tasks done on one operating system are also done on the other operating system, including deinstallation.

This situation can be avoided by performing the following steps:

1. Install Oracle9i into the first environment.
  2. Before installing in the second environment, provide a different location than the default when Oracle Universal Installer prompts you on where to create the Oracle Universal Installer inventory. The second environment inventory is independent of the first, and products can be installed or deinstalled separately on both systems.
- For installations with a response file, the path to the response file must be the full path on the computer. Oracle Universal Installer does not properly handle relative paths.
  - Due to a JRE bug a command prompt window appears whenever a configuration tool is launched. Leave this window open until the configuration tool finishes. Closing the window kills the process and causes the configuration tool to fail.

## 4 Product Related Issues

This section describes Oracle9iAS Database Client product related issues.

### 4.1 Character Sets

This section contains these topics:

- [AL24UTFSS Character Set](#)
- [Accessing Object Types and Collections through JDBC](#)

#### 4.1.1 AL24UTFSS Character Set

Oracle9i Database Client 9.0.1.2 does not support the Unicode character set AL24UTFSS introduced in Oracle7. This character set was based on the Unicode standard 1.1, which is now obsolete.

Oracle9i Database Client 9.0.1.2 supports the Unicode database character sets AL32UTF8 and UTF8. These database character sets include the Unicode enhancements based on the Unicode standard 3.0.

To migrate the existing AL24UTFSS database, upgrade your database character set to UTF8 before upgrading to Oracle9i. Oracle Corporation recommends that you use the Character Set Scanner for data analysis before attempting to migrate your existing database character set.

#### 4.1.2 Accessing Object Types and Collections through JDBC

The Oracle JDBC class files, `classes12.zip` and `classes111.zip`, provide character set support for the thin and Oracle Call Interface (OCI) drivers. The files contain all the necessary classes to provide complete character set support for all Oracle character sets for CHAR and NCHAR datatypes not retrieved or inserted as part of an Oracle object or collection type. See "Oracle Character Datatypes Support" of *Oracle9i JDBC Developer's Guide and Reference* for a description of CHAR and NCHAR datatypes.

However, in the case of the CHAR and VARCHAR data portion of Oracle objects and collections, the thin and OCI drivers require `nls_charset12.zip` for JDK 1.2.x and 1.3.x or `nls_charset11.zip` for JDK 1.1.x for most Oracle character sets (except US7ASCII, WE8DEC, WE8ISO8859P1, and UTF8). These two `nls_charset*.zip` files are included in the Oracle JDBC driver installation. To obtain this support, you must add the appropriate `nls_charset*.zip` file to your CLASSPATH or the logical equivalent, depending on the application.

---

---

**Note:** For most languages (including English and other Western European languages), the default character set on Windows is not US7ASCII, WE8DEC, WE8ISO8859P1, or UTF8. Therefore, `nls_charset*.zip` file is necessary on Windows for most languages.

---

---

**See Also:** *Oracle9i JDBC Developer's Guide and Reference* for more information on this topic

## 4.2 Oracle C++ Call Interface Methods Specific to Windows NT

The global methods for getting collections of `Refs` or setting collections of `Refs` from classes `Statement` and `ResultSet` have changed for Windows NT as follows:

- Use `getVectorOfRefs` in place of `getVector` on Windows NT
- Use `setVectorOfRefs` in place of `setVector` on Windows NT

The method names have been changed but the number of parameters and the types of the parameters remain the same as the original `getVector` and `setVector` methods for `Refs` on these classes.

- Applications on Windows NT platforms should be calling these new methods only for retrieving and inserting collections of `Refs`.
- Applications not running on Windows NT platforms have a choice of calling the currently existing `getVector` and `setVector` methods. However, Oracle Corporation recommends the use of the new methods for any vector operations with `Refs`.

### ResultSet Class: Fetching collection of Refs

```
void getVectorOfRefs(ResultSet *rs, unsigned int index,  
    OCCI_STD_NAMESPACE::vector<Ref<T> > &vect);
```

This method fetches a column value specified by the column index that is a collection of `Refs` from a result set.

The parameters are:

- `rs` - `ResultSet` object
- `index` - the column index of a column which is a collection of `Refs`
- `vect` - the vector into which the `Refs` are fetched

### Statement Class: Fetching collection of Refs

```
void getVectorOfRefs(Statement *stmt, unsigned int index,  
    OCCI_STD_NAMESPACE::vector<Ref<T> > &vect);
```

This method fetches a column value specified by the column index that is a collection of `Refs` from a statement. This is used in case of OUT binds and data manipulation language (DML) returning clauses. The parameters are:

- `stmt` - statement object
- `index` - the column index of a column that is a collection of `Refs`
- `vect` - the vector into which the `Refs` are fetched

#### **Statement Class: Inserting a collection of Refs**

```
template <class T>
void setVectorOfRefs(Statement *stmt, unsigned int paramIndex,
    const OCCI_STD_NAMESPACE::vector<Ref<T> > &vect,
    const OCCI_STD_NAMESPACE::string &sqltype);
```

This method inserts a collection of `Refs` into a column specified by the `index`. The parameters are:

- `stmt` - statement object
- `paramIndex` - the column index of a column that is a collection of `Refs`
- `vect` - the vector of `Refs` that are inserted into the column
- `sqltype` - the type name of the collection that was created in the database

The global methods for the fetching or inserting of collections of objects have been changed for Windows NT. The interface remains the same with respect to the method names and the number of parameters and the datatypes, but differs in the template parameter definition for Windows NT. Specifically, the template parameter for the template methods of `getVector` and `setVector` of objects (object pointers) on Windows NT have a `T` instead of a `T*` as shown in the following APIs.

Note that the usage of the methods does not differ across the platforms (users need not modify the call to these methods at all). On Windows NT, the template arguments passed as object pointers in the method call are specialized for the parameter `T` instead of `T*` on other platforms.

#### **class ResultSet: fetching a collection of objects**

```
#ifndef WIN32COMMON
    template <class T>
    void getVector( ResultSet *rs, unsigned int index,
        OCCI_STD_NAMESPACE::vector< T > &vect ) ;
#else
    template <class T>
    void getVector( ResultSet *rs, unsigned int index,
        OCCI_STD_NAMESPACE::vector< T* > &vect ) ;
#endif
```

This method fetches a collection of objects from a `ResultSet` for the column specified by the `index`.

The parameters are:

- `rs` - `resultSet` object
- `index` - column index
- `vect` - the vector into which the objects should be fetched

#### **class Statement: fetching a collection of objects**

```
#ifdef WIN32COMMON
    template <class T>
        void getVector( Statement *stmt, unsigned int index,
            OCCI_STD_NAMESPACE::vector< T > &vect) ;
#else
    template <class T>
        void getVector( Statement *stmt, unsigned int index,
            OCCI_STD_NAMESPACE::vector< T* > &vect) ;
#endif
```

This method fetches a collection of objects from a statement for the column specified by the `index`. This method is used in case of OUT binds and DML returning clauses. The parameters are:

- `stmt` - `statement` object
- `index` - column index
- `vect` - the vector into which the objects should be fetched

#### **class Statement: inserting a vector of objects**

```
#ifdef WIN32COMMON
    template <class T>
        void setVector( Statement *stmt, unsigned int paramIndex,
            const OCCI_STD_NAMESPACE::vector< T > &vect,
            const OCCI_STD_NAMESPACE::string &sqltype) ;
#else
    template <class T>
        void setVector( Statement *stmt, unsigned int paramIndex,
            const OCCI_STD_NAMESPACE::vector<T* > &vect,
            const OCCI_STD_NAMESPACE::string &sqltype) ;
#endif
```

This method inserts a collection of objects into a statement for the column specified by the `index`. The parameters are:

- `stmt` - `statement` object
- `paramIndex` - column index
- `vect` - the vector into which the objects should be fetched
- `sqltype` - the type name of the collection created in the database

**See Also:** *Oracle C++ Call Interface Programmer's Guide* for more information on Oracle C++ Call Interface APIs

### 4.3 Oracle Call Interface (OCI)

Oracle Corporation only ships an import library, `oci.lib`, for use with the Microsoft Compiler. Other compilers, for example, Borland, though likely compatible with the Oracle DLLs, are not tested and supported by Oracle for use with OCI.

### 4.4 Object Type Translator

- (Bug 1950643) Object Type Translator requires that the environment variable `NLS_LANG` be set to a non null value such as `us7ascii`. If this environment variable is set to a null value, Object Type Translator does not work.
- (Bug 1892470) If you are trying to invoke Object Type Translator from the command prompt and do not receive any output, make sure that the `CLASSPATH` includes the following.

```
%ORACLE_HOME%\sqlj\lib\runtime.zip;%ORACLE_
HOME%\sqlj\lib\translator.zip;%ORACLE_
HOME%\jdbc\lib\classes111.zip;%ORACLE_HOME%\jdbc\lib\nls_
charset11.zip
```

In addition, a JDK 1.1-compatible Java compiler (assumed to be `javac`) and the Java interpreter (assumed `java`) must be correctly installed.

### 4.5 Oracle9i Network, Directory, and Security Issues

This section contains these topics:

- [Windows Native Authentication](#)
- [Active Directory](#)

#### 4.5.1 Windows Native Authentication

Currently user database links are not supported with Windows Native Authentication.

#### 4.5.2 Active Directory

- On Windows NT and Windows 2000, the Oracle database service runs in the security context of the `LocalSystem` or a specific local or domain user. When using Active Directory, if the database service runs in the security context of `LocalSystem`, manually add the computer name in

which the database service is running to the access control entries on the OracleDBSecurity container object. Set read permissions on the OracleDBSecurity container object.

For example, if the database service `OracleServiceORCL` is running in the security context of `LocalSystem` in the computer `MYPC1`, then add `MYPC1` to the access control entries on the OracleDBSecurity container object with `READ` permissions on the OracleDBSecurity object.

- Oracle Enterprise Manager with patch `EM_90100_1835286.ZIP` is required when using Enterprise Security Manager for native authentication with Active Directory. This patch also allows Enterprise Login Assistant to store Oracle Wallets in the registry. The patch can be downloaded from Oracle *MetaLink*.

<http://metalink.oracle.com/>

- Upgrading Oracle Schema and Oracle Context in Active Directory (Bug 1993113) Oracle Net Configuration Assistant fails while upgrading the Oracle8i release of Oracle Schema and Oracle Context in Active Directory. The Oracle Schema and Oracle Context in Active Directory can be upgraded manually for Active Directory as follows:

1. Copy and paste the following lines into a file. Be sure to include a blank line after each of the three code segments that end with `orclProductVersion: 90000` or `orclVersion: 90000`.

```
dn: cn=BASE,cn=OracleSchemaVersion,cn=configuration,AD_Domain_DN
changetype: add
objectclass: orclSchemaVersion
cn: BASE
orclProductVersion: 90000
```

```
dn: cn=NET,cn=OracleSchemaVersion,cn=configuration,AD_Domain_DN
changetype: modify
replace: orclProductVersion
orclProductVersion: 90000
```

```
dn: cn=RDBMS,cn=OracleSchemaVersion,cn=configuration,AD_Domain_
DN
changetype: modify
replace: orclProductVersion
orclProductVersion: 90000
```

```
dn: cn=OracleContext,AD_Domain_DN
changetype: modify
replace: orclVersion
orclVersion: 90000
```

2. Replace *AD\_Domain\_DN* in four locations with the Distinguished Name (DN) of the Active Directory domain for your Active Directory server. For example:

```
dc=acme,dc=com
```

3. Execute the following command from the command prompt:

```
C:\> ldapmodify -Z -h AD_Host_Name -f filename
```

where:

*AD\_Host\_Name* is the host name of the Windows 2000 Domain Controller where your Active Directory is located and *filename* is the name of the file you created in steps 1 and 2.

---

---

**Note:** The -Z option must be uppercase.

---

---

4. Rerun Oracle Net Configuration Assistant.

## 4.6 SQL\*Plus

SQL\*Plus does not run and you cannot install the database successfully without the DLLs `wininet.dll` and `shlwapi.dll`.

If these DLLs are not present on the computer, you can get them from another computer, the MSDN Web site, or the Windows NT 4.0 installation CD-ROM. Copy the DLLs to `C:\WINNT\system32\` before installing Oracle9i.

**See Also:** These Microsoft Knowledge Base articles at the Microsoft Support Web site:

- Q196917
- Q174180
- Q184349

<http://search.support.microsoft.com/kb/c.asp>

## 4.7 Enterprise JavaBeans

Enterprise JavaBeans (EJB) are not supported with JDK release 1.1. EJB can only be used with JDK release 1.2.2.

