

UW-Extension Oracle 10g Quick Facts

January 4, 2008

1. Oracle 10g Database Instances

	SID	Version
Development	D1010X	10.1.0.5
Test	T1010X	10.1.0.5
Production	P1010X	10.1.0.5

2. TNSNAMES.ORA Entries required for the Oracle 10g database instances

The Database Services Unit maintains a current TNSNAMES.ORA file in the following directory on the INFOSYS LAN: Infosys1b\sys\data2:infosys\share\oracle. A copy of this file can also be requested from Database Services by sending an e-mail to oracle.dba@uwex.edu .

3. Oracle Client Software

It is recommended that the Oracle 10g client software be installed. In most cases, older clients will work correctly. However, there are some situations in which incompatibility errors will occur. For example, if the DBMS_OUTPUT.PUT_LINE function is called when connected to a 10g database using an older client, the error 'PLS-00553: Character set not recognized' might be generated.

In addition, it is recommended that upgrades are made to the latest version of SQL*Navigator or TOAD. Older versions of these products may result in anomalies under certain conditions.

4. ColdFusion Data Sources for the Oracle 10g Database Instances

The following table summarizes the ColdFusion data sources that have been predefined on each of the UWEX ColdFusion servers for the UWEX Oracle 10g database instances:

<i>ColdFusion Server</i>	<i>Database Instance</i>		
	<i>D1010X</i>	<i>T1010X</i>	<i>P1010X</i>
<i>CFMX Development</i>	D1010X ₁ ORA_1010X ₁	T1010X ₂	P1010X ₂
<i>CFMX Test</i>	--	T1010X ₁ ORA_1010X ₁	P1010X ₂
<i>CFMX Production</i>	--	--	P1010X ₁ ORA_1010X ₁

¹ Permits select, insert, update and delete statements and stored procedure calls

² Permits select statements

5. New Password Rules

Starting with Oracle 10g, Database Services is enforcing stricter rules regarding passwords. The following

list summarizes the password rules for Oracle 10g database accounts:

- May not contain the username.
- Must be at least 6 characters long.
- May not be a simple dictionary word.
- Must contain at least one letter and one digit.
- May not contain any punctuation marks or spaces.
- May not begin with a number.
- Must be different from the previous password by at least 3 characters.

The new rules will uniformly prevent reuse of similar passwords and extremely simple passwords.

6. **‘ORA-28221: Replace not specified’ error when running ALTER USER command**

When writing your own code to change a password for an Oracle 8i database account, the following code would work:

```
connect account_name/oldpassword@d1010x; alter user account_name identified by  
newpassword;
```

When this code is run to change the password for an Oracle 10g database account, the following error is returned:

```
ORA-28221: Replace not specified
```

As a result of implementing new rules regarding password complexity, the following alter user syntax must be used:

```
alter user account_name identified by newpassword replace oldpassword;
```

7. **In ColdFusion, number previously displayed as integers without a decimal point are displayed with a precision of 1.**

The most likely cause for this situation is that there are tables with number columns defined with neither precision nor scale specified (i.e. number rather than number(n,m) or number(n)). Oracle 10g handles numbers slightly differently than previous Oracle releases. As a result, ColdFusion displays number field defined as number as if it was defined as number(n,1). To correct this problem, the number columns without precision and scale specified should be redefined to specify precision and scale.

8. **Character set Changes**

The Oracle 10g database instances have been configured to use a multi-byte character set (AL32UTF8). The previous Oracle 8i instances were configured for a single byte character set (Western European character set - WE8ISO8859P1). As a result of this character set change, some characters that could be stored in a single byte in the Oracle 8i instances may require 2 or 3 bytes storage in the Oracle 10g instances. The characters requiring multiple bytes are, generally, foreign language characters (for example, \ddot{w} , \ddot{e} , ñ or ŀ), special symbols (for example, © or Σ) or hidden characters (for example, result of cutting/pasting from MS Word document).

If you do store these types of characters, then some coding changes may be necessary to restrict the number of characters that can be inserted into a text (CHAR or VARCHAR2) field. The Oracle 10g

database instance, by default, uses BYTE semantics. With BYTE semantics, the length of a text (CHAR or VARCHAR2) field defines the number of bytes allowed for the text. For a multi-byte character set (eg. AL32UTF8 used in the 10g instances), each character may use 1, 2 or 3 bytes of storage. Thus, the length of the CHAR or VARCHAR2 field specifies the maximum number of bytes of storage that can be used for the field which may or may not equal the number of characters. Depending on the application, the coding changes vary. Database Services is available to assist with analyzing individual situations and identifying the appropriate solution.

9. DBMS_CRYPTO and Sensitive Data

Oracle 10g contains a new package, DBMS_CRYPTO, to assist with obscuring sensitive data. The DBMS_CRYPTO package replaces the DBMS_OBFUSCATION package. An example of using DBMS_CRYPTO is available from the Database Services Unit.

While the account owner is responsible for identifying and protecting sensitive data, the Database Services Unit will notify account owners if they notice any data that might be considered sensitive during the 8i to 10g migration.

10. Required UTL_FILE Coding Change

When using the UTL_FILE FOPEN function/procedure in Oracle 10g, the LOCATION parameter has been changed from an explicit directory path to the name of a directory object defined in the database.

In the Oracle 8i instances, a call to the UTL_FILE.FOPEN procedure looked something like the following:

```
F1 := UTL_FILE.FOPEN('/usr/oracle/mydir', 'myfile', 'R');
```

In the Oracle 10g instances, the same call to UTL_FILE.FOPEN procedure must be changed to something like the following:

```
F1 := UTL_FILE.FOPEN('mydirname', 'myfile', 'R');
```

where mydirname is the name of a directory object that has been defined by the Database Services Unit.

11. Package Specification and Body Procedure/Function Parameter Defaults must be identical

A package body that compiled in the previous Oracle versions generates an error PLS-00593: default value of parameter ... during compilation. In Oracle 10g, the default values of parameters in a package body must match exactly those in the package specification.

In Oracle 8i instance, this was allowed:

```
Package Specification: Function examplefunction (parameterone IN Integer) Return Integer;  
Package Body: Function examplefunction (parameterone IN Integer default 0) Return Integer
```

In Oracle 10g, the above code results in "PLS-00593: default value of parameter "parameterone" in body must match that of spec" when trying to compile the package body. The solution is to add the default to the package specification or remove it from the package body.

12. "BIN" objects in USER_OBJECTS

Users who do a SELECT on the USER_OBJECTS view in 10g may notice objects listed that have names starting with "BIN", followed by a string of seemingly random characters. These objects are a normal part of Oracle 10g functionality, related to tables that have been dropped, and not a cause for concern. If you would rather not see these objects in the listing, please contact Database Services for assistance.