

# The mod\_plsql Environment

## Basic Software Prerequisites

If you're going to start developing PL/SQL-based web-deployed applications with Oracle's mod\_plsql HTTP server module, you'll need to understand some basics of PL/SQL Web Toolkit application configuration and security. Details of software installation and configuration can be found in the appropriate Oracle manuals. At the time this is being written, the manuals are available at <http://tahiti.oracle.com>. The author strongly recommends downloaded PDF formatted copies of at least the following manuals:

- <insert recommendations here>

Basic software prerequisites include installation of:

- Oracle Database 8.1.7 or higher, with the PL/SQL Web Toolkit packages installed.
- Oracle Application Server (AS) or HTTP Server, with the mod\_plsql server module installed
- A browser
- A database interface tool, such as SQL\*Plus or TOAD. The author recommends the freeware version of TOAD, available for download at [www.toadsoft.com](http://www.toadsoft.com).

All software components can be installed on the same machine for a development or learning environment. But for testing and production, you will minimally have three machines: a database server, an application server, and the client machine. An additional machine may be used for authentication services, such as Oracle's Single Sign-On.

## mod\_plsql Processing

The mod\_plsql server module enables users to execute PL/SQL procedures and packages stored in the database via standard URL syntax.

Each URL that accesses a PL/SQL procedure includes:

- The host machine name and optional port number
- A virtual path, which the Application Server maps to a Database Access Descriptor (DAD)
- The name of the PL/SQL procedure (perhaps qualified by a schema name and/or a package name, or perhaps referenced via a database synonym)
- The names of and values for any PL/SQL procedure required parameters

<insert example URL>

For each URL submitted by a user or a webpage link or form, the Application Server recognizes the virtual path (as defined in the DAD) and passes off the request to the mod\_plsql module. The mod\_plsql module then establishes (or reuses) a database connection based on database credentials stored in the DAD, optionally augmented by database account credentials supplied by the user via the browser's authentication dialog box.

The PL/SQL procedure must return HTML to the mod\_plsql module via calls to the PL/SQL Web Toolkit's `http` package.

Typically, the mod\_plsql module performs an automatic commit after the called procedure returns control. Transaction management and application state options are discussed in greater detail later in this series.

<insert illustration of application flow>

## Database Account Requirements

The PL/SQL Web Toolkit packages are generally installed in the database with PUBLIC execution privileges, and public synonyms and are thus available to all authenticated database users without any additional configuration. However, the mod\_plsql server module must be configured to authenticate URL requests for PL/SQL procedures. Once a user's credentials are authenticated, all database access is controlled via procedures on which target database schema has execute privileges.

## Minimum Developer Requirements — Basic mod\_plsql Authentication

The minimal database account requirements, suitable for learning and practice, and for some development environments, are:

- An Oracle application-owning database account with the following privileges:
  - CREATE SESSION
  - CREATE TABLE
  - CREATE VIEW
  - CREATE SEQUENCE
  - CREATE TRIGGER
  - CREATE PROCEDURE
  - Quota on an appropriate tablespace
- An Application Server Database Access Descriptor (DAD) that specifies:
  - The virtual path to include in the URL
  - The database service name
  - Authentication type set to Basic

<insert same DAD>

The DAD can optionally include the database account name and password, which allows public access to the database account's mod\_plsql-based procedures — not a good idea when using Basic authentication. Omitting the account name and password causes the browser to prompt for authentication credentials.

<insert screen shot of authentication>

Database account configuration and version control issues in a multi-developer environment are discussed in detail later in this series.

### Minimum End User Requirements — Basic mod\_plsql Authentication

The minimal database account requirements for application end users, using basic mod\_plsql authentication, are:

- An Oracle database account for each end-user with the following privileges:
  - CREATE SESSION
  - EXECUTE privilege on all appropriate PL/SQL procedures and packages owned by the application schema
  - Synonyms (PUBLIC or PRIVATE) for referencing the application's PL/SQL procedures, or, an AFTER LOGON trigger that issues an ALTER SESSION SET CURRENT\_SCHEMA command.
- An Application Server Database Access Descriptor (DAD) that specifies:
  - The virtual path to include in the URL
  - The database service name

Since the account name and password are not specified in the DAD, the browser will prompt the user for authentication credentials (i.e., her Oracle account name and password). Different end users will all use the same DAD, but will each require a separate Oracle account.

This approach has the benefits of simplicity for the application developer (no customer authentication code required) but quickly gets out of hand as new end users are added (or dropped) and each requires DBA intervention.

### Minimum End User Requirements — Custom mod\_plsql Authentication

The mod\_plsql module and PL/SQL Web Toolkit supports two types of custom authentication — Global and Application specific (referred to as Global OWA and Custom OWA, respectively).

Both flavors of customer authentication require the creation of a package named owa\_custom with a procedure named authorize — either a centralized package, or an application-specific package.

The minimum database account requirements for custom mod\_plsql authentication are:

- A single Oracle database account to be shared by all end-users with the following privileges:
  - CREATE SESSION
  - EXECUTE privilege on all appropriate PL/SQL procedures and packages owned by the application schema

- Synonyms (PUBLIC or PRIVATE) for referencing the application's PL/SQL procedures, or, an AFTER LOGON trigger that issues an ALTER SESSION SET CURRENT\_SCHEMA command.
- An Application Server Database Access Descriptor (DAD) that specifies:
  - The virtual path to include in the URL
  - The database service name, account name, and password
- A central or application-specific owa\_custom.authorize procedure.

Since full database credentials are specified in the DAD, mod\_plsql connects to the database without prompting the user, but then invokes the owa\_custom.authorize procedure to validate user credentials supplied by the end-user via the browser's authentication mechanism. It is up to the application developer to supply appropriate authentication logic in the owa\_custom.authorize procedure. This usually involves looking up the end-users browser-supplied credentials in an application table.

DRAFT