

Oracle Database Lite 10gR3 Technical White Paper

*An Oracle White Paper
August 2008*

Oracle Database Lite 10gR3 Technical White Paper

Overview.....	3
Oracle Database Lite Components	3
Mobile Client.....	4
Oracle Lite Database	4
Mobile Server.....	5
Data Synchronization.....	5
Synchronization Process	6
Synchronization Conflicts	7
Synchronization Errors	7
Customizing Synchronization.....	8
Life Cycle Management.....	8
Application Management	9
User Management	9
System Management	9
Synchronization Management	9
Device Management	10
Mobile Development Kit	10
Supported Development Tools	10
Application Development Models	11
Mobile Database Workbench (MDW)	11
Packaging Wizard.....	12
Mobile Server Repository Diagnostic Tool (MSRDT).....	12
Application Deployment Configurations.....	13
Oracle is Your Partner of Choice	13

Oracle Database Lite 10gR3 Technical White Paper

OVERVIEW

Employees are increasingly working away from their desks and require access to corporate data normally found only on desktop devices connected to enterprise networks. Wireless connections for mobile users offer the promise of remote access to enterprise data but persistent wireless connections are not always possible, practical or desirable.

Wireless connections are not always possible because mobile workers may be in an environment that does not have wireless coverage. Often it is cost-prohibitive and impractical to use a persistent wireless connection because applications do not have a real-time data access requirement. Finally, a wireless connection may not be desirable because the very presence of a wireless connection could compromise an application or user's security.

Developers require an end-to-end infrastructure with application services that enables the development, delivery and operation of secure, personalized applications on mobile or embedded devices.

Oracle Database Lite Components

Oracle Database Lite (Database Lite) is a complete solution and includes more than a small footprint client database. Oracle Database Lite 10g is a full-featured, integrated infrastructure for mobile application development, deployment and management. It extends the grid environment by enabling mobile access to applications that rely on information maintained by the enterprise Oracle database in the grid.

The main components of Oracle Database Lite are 1) client stack (featuring the lightweight database) available on multiple platforms including various flavors of Windows 32-bit, Windows Mobile, Linux and Symbian OS 2) Mobile Server for synchronization and scalable deployment and management of applications, users, and devices and 3) developer tools that enable quick and simple application development. Figure 1 shows a typical application deployment architecture using Oracle Database Lite. The mobile server is generally hosted by an application server and the mobile server repository (which stores synchronization metadata) may reside in a separate database from the application data. Further, Oracle Database Lite supports RAC storage for application and repository data.

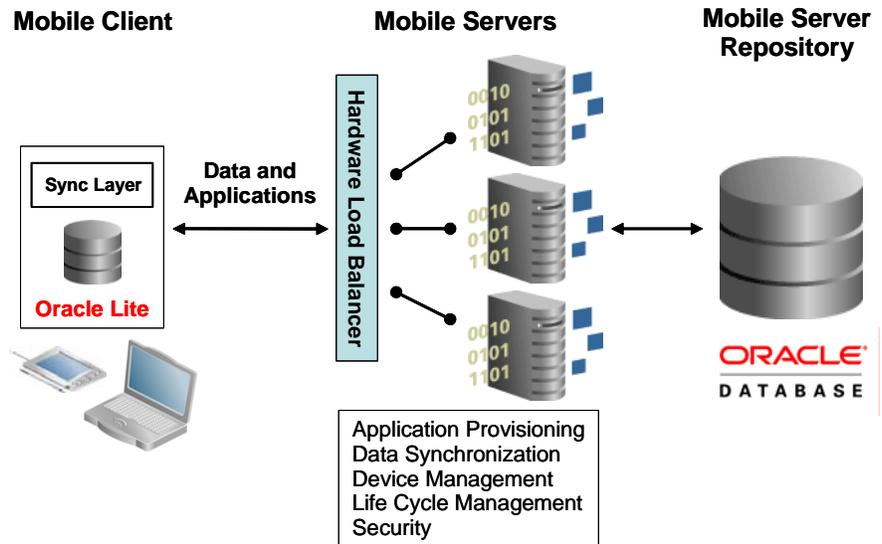


Figure 1. Oracle Database Lite Deployment Architecture

MOBILE CLIENT

The Oracle Database Lite client includes various components that operate in concert to facilitate a seamless user experience and easy remote administration. The client includes a small-footprint database (and associated utilities) optimized for handhelds, laptops and small business environments, supporting single or multi-user deployments. The client also provides rich data synchronization support: a synchronization agent that synchronizes automatically in the background, a GUI application that allows manual invocation of synchronization (mSync) and APIs that can be invoked by your application. The device agent on the client allows administrators to remotely manage the device by sending commands or querying system and application data. Finally, an update utility supports application life-cycle management by allowing new versions of the application to be downloaded by mobile workers in the field.

Users provision the Oracle Lite client runtime environment by 'boot strapping' a self-extracting *setup.exe* file from the Mobile Server. This file installs all the client runtime software needed including the Lite Database, mSync client, autosync client, device client, and various database utilities.

Once the client device is provisioned, the user may invoke the mSync client, which will perform a first time synchronization that deploys the authenticated users' applications, appropriate databases and DSN configurations.

Oracle Lite Database

The Oracle Lite database is not an abbreviated version of the Oracle database but rather was designed from the ground up to have a small footprint and be used with

mobile applications in small and embedded devices (e.g., laptop computers, handheld computers, PDAs, and lightweight information appliances). It provides a secure, relational data store that is self-administering and self-tuning without requiring DBA involvement.

The Oracle Lite database contains a subset of data stored in the Oracle database. This subset is stored in snapshot tables (defined using publication items) in the Oracle Lite database. Together with publication items, other client side database objects (e.g., stored procedures) are captured in a publication. Users can be subscribed to various publications. Unlike a base table, a snapshot keeps track of changes made to it in a change log. Users can make changes in the Oracle Lite database while the device is disconnected, and can later synchronize with the Oracle database server.

Oracle Lite Database runs on Windows 2000/2003/XP/Vista, Windows Mobile 5 & 6/CE/Pocket PC, Symbian OS, Linux and embedded Linux.

The Oracle Lite database features:

- Small Flexible object kernel
- SQL 92 and ACID Compliance
- Java/C++/.Net Stored Procedures and Triggers
- Built-in change capture for synchronization
- Zero Administration
- 4 GB maximum database size (separate 16TB BLOB support)
- Multi-user version (Branch Office) supports up to 64 simultaneous connections
- Compression support
- Data load/unload utility
- Query tuning support
- ODBC/JDBC/ADO.Net support
- Strong 128-bit AES encryption support

MOBILE SERVER

The Oracle Database Lite Mobile Server is required in the middle tier to allow mobile clients to synchronize data with the enterprise database. In addition, it provides life cycle management functionality for deployment and management of applications, users, and devices.

Data Synchronization

During synchronization, client changes are uploaded to the Mobile Server to be applied to the enterprise database, and the client-specific server side updates are

downloaded. Synchronization can be accomplished manually either by the user invoking the mobile synchronization client, mSync, or the application invoking the synchronization APIs. In scenarios lacking a communication network or requiring network silence (e.g. military applications), manual synchronization can also be accomplished by using the File-based synchronization feature that involves generation and physical transportation of the synchronization files between client and server.

Alternatively, synchronization can be invoked automatically when developer-defined rules are triggered on the client or server end if pre-defined conditions are met and a trigger event occurs. Developers can specify various data-centric and platform-centric rules. For example, such rules can trigger synchronization when the number of client-side or server-side changes exceeds a threshold or when the client device battery strength falls below a predefined level. Automatic synchronization executes seamlessly and does not require an exclusive database lock, thereby allowing the application to continue accessing the local database and allowing the user to remain productive.

Synchronization Process

Oracle Database Lite features an asynchronous synchronization mechanism that allows it to scale to support lots of users and meet the needs of large or growing enterprises. In a synchronous synchronization system, a client connects to the server to synchronize, and remains connected while its changes are applied to the enterprise database. Such a design inherently limits the number of users that can be supported due to limits on concurrent connections. In contrast, an asynchronous synchronization mechanism minimizes the client connection time by relying on queues to collect user changes that are uploaded and providing parallel queues to download server side updates to clients (see Fig. 2).

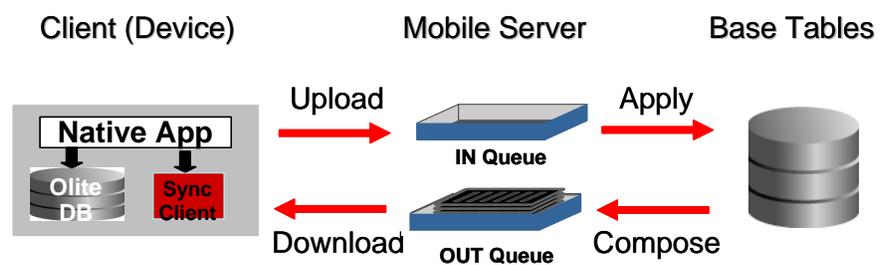


Figure 2. Oracle Database Lite Synchronization Process

Simultaneously, a background process called the Message Generator and Processor (MGP) *asynchronously* collects the changes uploaded by all clients and applies them to database tables. It then composes new data, ready to be downloaded to each client during the next synchronization, based on predefined subscriptions. The MGP manages both the in and out queues.

Oracle Lite synchronization is a two-step asynchronous process:

1. Sync moves updated rows from the client database to the **In Queue** and from the **Out Queue** to the client database...
2. **MGP** (Message Generator Processor) applies In Queue changes to base tables and **Composes** the changes to the base tables to client Out Queues.

Oracle Lite asynchronous synchronization architecture emphasizes:

- Performance
- Data compression
- Scalability (supports thousands of users)
- Transport independence (wired or wireless)
- Conflict detection and customizable resolution
- Checkpointing (allows disrupted synchronization session to resume where it left off)
- Non-blocking synchronization (no user is blocked by another user)
- Multi-threaded architecture
- Custom synchronization invocation
- Supports any TCP/IP enabled network
- Open transport APIs to support any wireless network (any other reliable transport adaptor can be plugged in)

Synchronization Conflicts

Various conflicts can occur during synchronization. For example, a client may update the same row as the server, create rows with the same primary key values, or delete the same row that the server updates. Oracle Lite will detect such conflicts and provides multiple mechanisms for resolving them.

Developers can choose to use the built-in, default conflict handling rules that allow either the client changes to override server changes (Client Wins) or vice versa (Server Wins). In the latter case, even though server changes take precedence over client changes, client changes are cached in an error queue to allow an administrator the opportunity to review and apply them if needed. Developers may also choose to resolve synchronization errors programmatically.

Synchronization Errors

Client changes may result in errors such as nullity violations or foreign key constraint violations during synchronization. The Mobile Server does not automatically resolve synchronization errors. Instead, it moves the transaction operations into the Mobile Server error queue. Later, Mobile Server database

administrators can purge them from the error queue or change and re-execute them.

Customizing Synchronization

MyCompose API

A developer defined Java class extended from the abstract MyCompose class allows a developer to programmatically customize the compose phase of synchronization. MyCompose is a programming interface option available to developers to control MGP's DML operations. This is especially useful if a DML operation is very complex. Compose may be applied to specific clients using programming selection techniques or to prioritize client compose.

Queue-Based Synchronization

Queue-based synchronization is ideally suited for scenarios that do not require complete synchronization but require something somewhere in between such as data collection on the client. With data collection, there is no need to detect conflicts, client state information, or server side updates. Therefore, there is no need to add the additional overhead normally associated with the default logic used by Database Lite for fast refresh or complete refresh publication items during MGP processing.

Application developers may manage the synchronization process programmatically by using the Queue-based publication item API. This can be considered the most basic form of synchronization, since there is no synchronization logic created with it. The synchronization logic is left entirely in the hands of the developer who must manage the queues and implement PL/SQL packages to provide the DML operations that manage the enterprise database tables.

Life Cycle Management

Oracle Database Lite provides complete life cycle management for distributing, installing and managing software, data and files on mobile devices and remote systems. It enables a central site support team to manage distributions to an entire workforce from a single management console. Mobile Manager, a web-based administration interface, enables complete server-side management of all mobile applications, devices, users and Mobile Servers. With Mobile Manager, administrators have complete control to view, monitor and manage the mobile clients.

Mobile Manager supports:

- Application Management
- User Management
- System Management
- Synchronization Management

- Device Management

Application Management

Applications are packaged and published to the Mobile Server repository using Packaging Wizard (described later) where they are managed by the web-based administration tool Mobile Manager. The Mobile Manager allows uploading and removing applications from the Mobile Server repository and modifying application properties. Administrators can also use Mobile Manager to subscribe users and groups to applications, and specify user-specific data subsets by setting values for data-subsetting parameters. Finally, an administrator may use Mobile Manager to suspend/resume applications and specify application files for public usage.

User Management

User management via Mobile Manager allows the definition of new users, and groups. It also allows establishing roles for groups of users and application access permissions. User management may interface with external validation mechanisms such as an LDAP directory, e.g., Oracle Internet Directory (OID).

System Management

The Mobile Manager interface also allows management of all the Mobile Servers in a production deployment configuration. This includes detailed information about active user synchronization sessions. Trace properties may be dynamically set and allow an administrator to specify trace filters, log file locations and trace log size.

The Mobile Server configuration file, *webtogo.ora*, contains Mobile Server operational parameters relating to external authentication, file systems, cache, synchronization and MGP default settings. These settings can also be dynamically altered through the system management interface.

Synchronization Management

Synchronization management provides very detailed monitoring of users' synchronization results and includes details like start and finish times, upload and download durations, record counts, and a listing of publications, publication items and device tables.

The Synchronization Manager graphical interface allows administrators to monitor and manage the synchronization service and performance, administer the configuration, trace synchronization history, browse synchronization publication and subscription information, and monitor MGP (Message Generator and Processor) performance.

The transaction error queue lists transactions that resulted from synchronization conflicts or errors, which the administrator can correct by re-executing or purging them.

Device Management

As enterprises deploy a growing number of applications on small devices, managing these devices presents a crucial requirement for enterprise IT solutions. The Device Management (DM) system in Oracle Database Lite provides a solution to administer the deployed devices and remotely manage applications and data on these devices.

Device Management supports the following functionality:

- Remote inspection of client device hardware and operating system settings.
- Remote inspection and modification of application configuration settings.
- Client database information retrieval, validation and Sync with Oracle DB.
- Client device lock down, application removal, application data removal.
- Client software management.
- Device configuration files modification.

MOBILE DEVELOPMENT KIT

Oracle Database Lite provides critical support for rapid development and deployment of off-line mobile applications. Developers can install the Mobile Development Kit (MDK) from the Oracle Database Lite installation medium and install not just the client database and the Database Lite runtime binaries, but also utilities and GUI based tools (MDW and Packaging Wizard) that simplify describing, deploying and debugging a mobile application. Furthermore, the Oracle Database Lite mobile development kit contains code samples that accelerate the development of mobile applications.

The MDK provides access to the synchronization APIs as well as database access APIs. Oracle Database Lite supports application development in Java, C/C++ and .Net languages. Thus, synchronization APIs are available in all these languages. Similarly, Database Lite offers JDBC, ODBC, ADO.NET access to the Oracle Lite database.

Supported Development Tools

Oracle Database Lite supports the following development tools and IDEs:

- Oracle JDeveloper 10g
- Eclipse
- IntelliJ
- Microsoft Visual Studio .Net
- Sybase Powerbuilder

- Other ODBC/JDBC compliant IDEs

Application Development Models

Database Lite offers developers a choice of application development models. Using Java, .Net or C/C++, developers can build traditional local, standalone off-line applications (see Fig. 3).

In addition to this application development model, Oracle Database Lite also supports developing web-based applications that can be implemented using applets or Java servlets/JSPs but can execute in the absence of a network connection. In disconnected mode, a local application server serves these “web applications” to the client’s browser. The client application will continue to work and appear as if it is connected to the network. When the client reconnects to the Mobile Server, the Oracle Lite database is synchronized and the application is served from an enterprise application server, as any other normal web application.

A web-based application development model yields tremendous savings of development effort by allowing the same application to be used in the office where there is continuous network connectivity, as well as in the field, where the network connectivity may be sporadic. Furthermore, a web-based model insulates the application from significant future platform porting requirements. Finally, Oracle Database Lite uses the full-featured OC4J servlet container as the local application server. The OC4J container forms the core of the Oracle Application Server thereby allowing developers to leverage their existing investment in Oracle technology training.

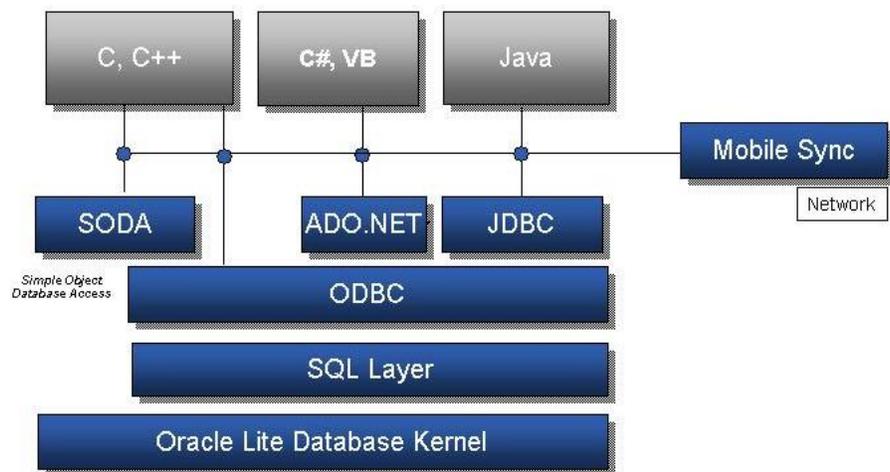


Figure 3. Oracle Lite Development Model

Mobile Database Workbench (MDW)

The Mobile Database Workbench (MDW) is a GUI tool for mobile application development that is installed with the MDK. The Mobile Database Workbench

uses a set of well-defined interview steps to guide mobile application developers through the process of building, testing and tuning their application schema definition prior to writing application code. Developers can use it to specify publication items, shared sequences, stored procedures/triggers and DDL statements to execute on the client database. In addition, developers can manipulate and reuse preexisting schema components.

Together, these client-side schema components are referred to as a publication. Application developers write their application against this publication specification.

Packaging Wizard

The Packaging Wizard is a standalone GUI tool installed with the MDK that allows developers to create, edit and publish an application to the Mobile Server.

A developer can use the Packaging Wizard to specify application metadata (e.g., name, location on mobile server, etc.), the collection of files (binaries, images, servlets/JSPs, etc.) that comprise the application and which client database schema (publication) the application uses. The Packaging Wizard then bundles (“packages”) all application components (executables, DLLs, images, etc.) into a self-extracting file used for deployment to mobile or lightweight environments.

This file can then be published to the Mobile Server using Packaging Wizard. After publishing the application, the administrator may control application access privileges for users and groups.

Mobile Server Repository Diagnostic Tool (MSRDT)

The Mobile Server Repository Diagnostic Tool can be used to verify the integrity of the Mobile Repository. MSRDT performs a consistency check on the repository and provides a detailed report of potential problems allowing administrators to identify corruptions in their repository and take corrective action.

APPLICATION DEPLOYMENT CONFIGURATIONS

Oracle Database Lite offers two primary deployment architectures. Applications can synchronize directly with the Mobile Server (as shown in Fig. 4) and can be written to access a local Oracle Lite Database.

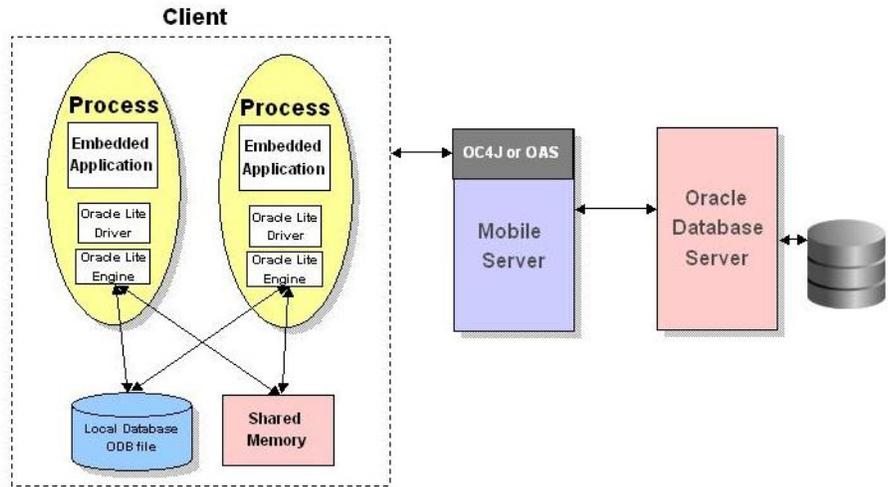


Figure 4. Oracle Lite Database Local Deployment

Alternatively, business needs may dictate a Branch Office deployment of Oracle Database Lite (see Fig. 5). In a Branch Office configuration, the Oracle Lite Database provides a mini-server that can support up to 64 concurrently connected clients via a listener. A Branch Office configuration allows the database to be remote, and allows it to behave like a departmental database server. Branch office deployments are typical in many retail scenarios. Each Branch Office Server can be centrally managed using Mobile Manager.

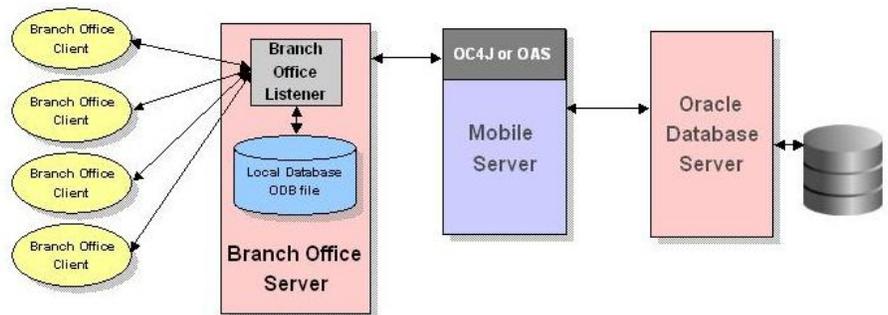


Figure 5. Oracle Lite Database Branch Office Deployment

ORACLE IS YOUR PARTNER OF CHOICE

Oracle is committed to serving the mobile computing needs of businesses worldwide with solutions and software platforms that offer the depth of feature, scalability, reliability and performance Oracle has built its reputation and business

on. Oracle is the partner of choice for extending your Oracle Database into the field.

As enterprises look to enhance their business with mobile requirements, the need for mobile infrastructure becomes evident. The ideal mobile infrastructure must (1) enable the development of your mobile strategy, (2) support the full range of devices and application platforms, and (3) leverage existing IT investments.

Oracle Database Lite is a complete, end-to-end solution for deploying offline mobile solutions offering:

- Device client stack that includes comprehensive device management and synchronization technology, and a secure, lightweight, SQL-enabled relational database.
- Scalable, high-performance Mobile Server that supports robust synchronization of thousands of clients to an Oracle database.
- An integrated management interface for managing your applications, users, devices and system.
- Development tools that support rapid application development.



Oracle Database Lite 10gR3 Technical White Paper
August 2008
Author: Karun Bakshi

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com

Copyright © 2007, Oracle. All rights reserved.

This document is provided for information purposes only and the contents hereof are subject to change without notice.

This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle, JD Edwards, and PeopleSoft are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.