Embedded Database Systems

GAYASHAN AMARASINGHE

GAYASHAN@CSE.MRT.AC.LK

Outline

Embedded Database Systems

- Definition
- Motivation
- Embedded database concerns
- Embedded database characteristics
- Embedded database architectures
  - Client-server vs embedded (lightly and deeply embedded)
- Examples

Commercial Embedded Databases (Oracle)

Mini Project

What is an embedded database system?

It is a database management system (DBMS) which is;

- Tightly integrated with an application software
- Transparent to the application end-user (hidden from the end-user)
- Requires little/no maintenance

Related terms: Backend DBs, In-memory DBs, Mobile DBs

Why do we need embedded databases?

- Extremely fast access
- Fault tolerance
- Persistence and reliability
- Fast to market
- Low Total Cost of Ownership (TCO – purchase price and the operational costs)
- No Database Administrator (DBA) required

Concerns of Embedded databases

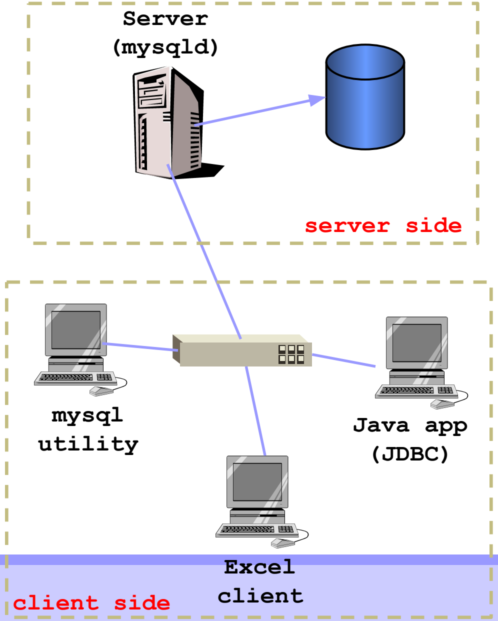
- APIs (SQL, proprietary and native APIs)
- Architectures (lightly embedded, deeply embedded)
- Storage modes (on-disk, in-memory, combined)
- Database modes (relational, object-oriented, entity-attribute-value, network, …)
- Target markets

Embedded database characteristics

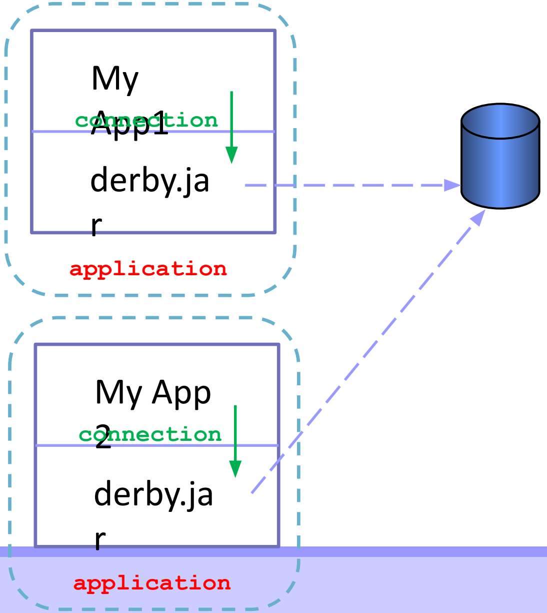
- Embedded in an application
- Can be incorporated in a scripting language
- Inexpensive
- May not scale well (depends on how it is implemented)
- Good transaction control
- Text search support may be minimal
- May not support SQL

Embedded database architectures

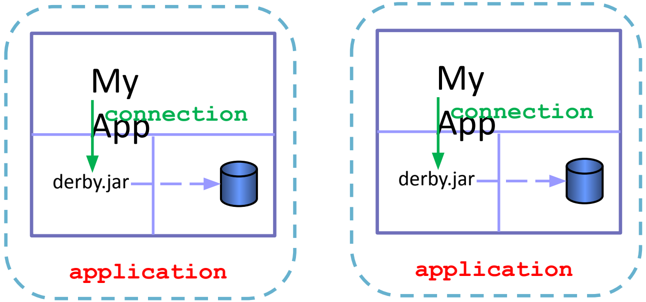
CLIENT-SERVER



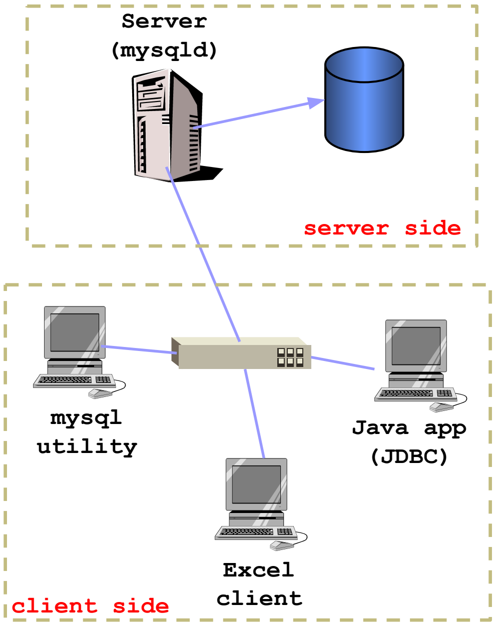
LIGHTLY EMBEDDED



DEEPLY EMBEDDED

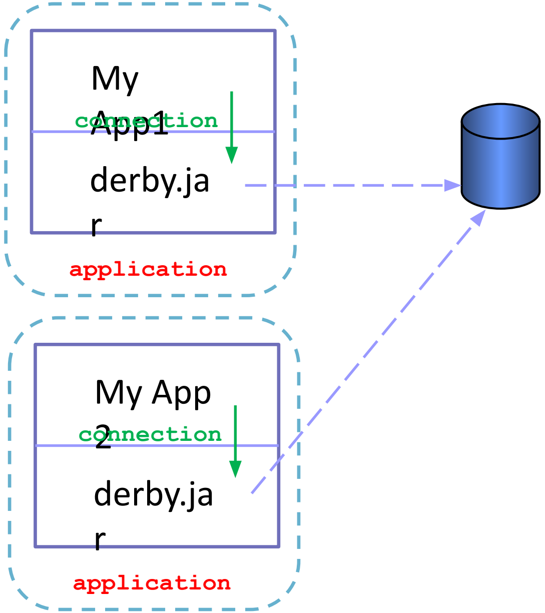


Client Server architecture (traditional)



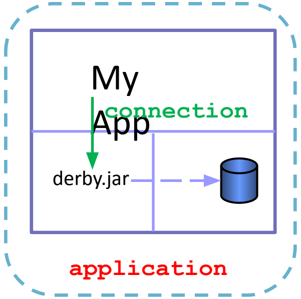
- Database server is a separate process
- Running on a host
- Clients can run on any machine
- Many different programs may be clients
- Supports standard APIs

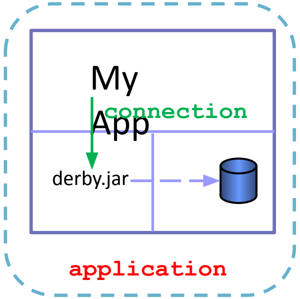
Lightly Embedded Database architecture



- Database engine installed as part of the application installation on the same machine
- Small footprint
- Full features (mostly)
- Other applications maybe able to access the same database
- Multi device support may be available

Deeply Embedded Database architecture





- Application runs the database ‘inside’ it
- No database engine installation required
- Uses a library that has database capabilities
- Popular with mobile applications
- Can be easily distributed with the application

Example: Hypersonic SQL

- lightweight, *fast* database written in Java
- database can be stored in memory or on disk.
- embed in Java app - no separate server
  - don't need to install database server or disk-based database
- can also run-in client-server mode
- useful for development and "demo" systems
- **`http://hsqldb.org`**

HyperSQL

Example: derby

- lightweight, pure Java database
- formerly "Cloudscape", donated to Apache foundation
- only 1 user can connect to database at a time
- embed in Java applications - no separate server
  - similar to HSQLDB
- can also run in client-server mode
- included with JavaEE as "Java DB"
- **`http://db.apache.org/derby`**

Example: SQLITE

- World's most widely distributed database
- written in C
- very small: 350KB binary
- used on Android
- 3rd party JDBC drivers:
  - http://code.google.com/p/sqlite-jdbc/
  - http://www.ch-werner.de/javasqlite/
  - http://www.xerial.org/trac/Xerial/wiki/SQLiteJDBC

Example: Berkeley DB

- *libraries* for embedded database using the OS's file system.
- No db manager, No network access, No query language.
- used as data tier for LDAP, sendmail, and many other apps
- very small and *fast* -- faster than any relational DB w/ manag
- **C** and **pure Java** version
  - language bindings for C++, Perl, Python, Ruby, and more
- bought by Oracle in 2006: http://www.oracle.com/database/berkeley-db/index.html
- still Open Source under the "Sleepycat Public License" and "Sleepycat Commercial License",
- not required to distribute the source code with your app.

Example: Interbase

- developed and marketed by Embarcadero Technologies
- can operate in both server and embedded modes
- supports database and column level encryption
- very small – server (40MB), client (400KB)
- supports SQL, JDBC, ODBC, …
- https://www.embarcadero.com/products/interbase

Commercial embedded databases (Oracle)

Oracle is a market leader in embedded databases – 23% market share

“Lights out” embedded databases – no database administrator (DBA) required

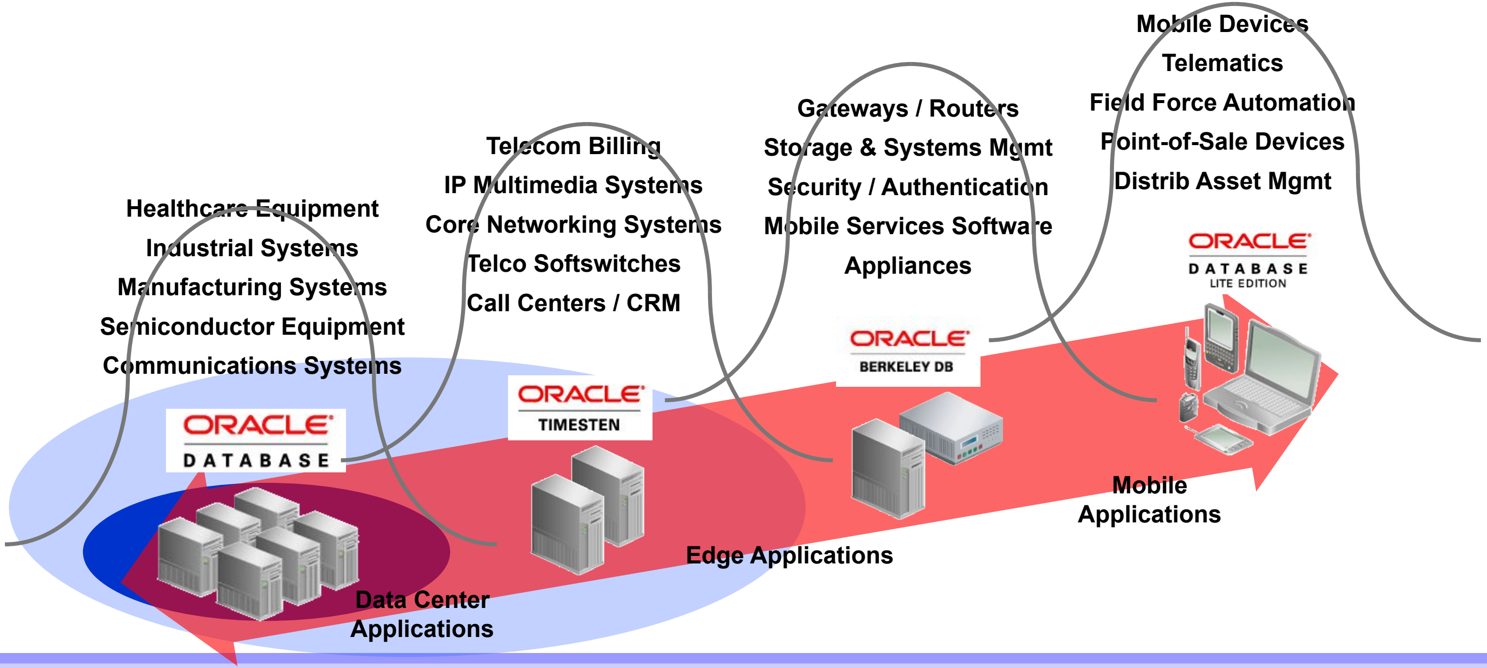
Small footprint (< MB)

Aggressive pricing models

Examples:

- TimesTen
- Berkeley DB
- Database Lite

Where are they deployed?



TimesTen database

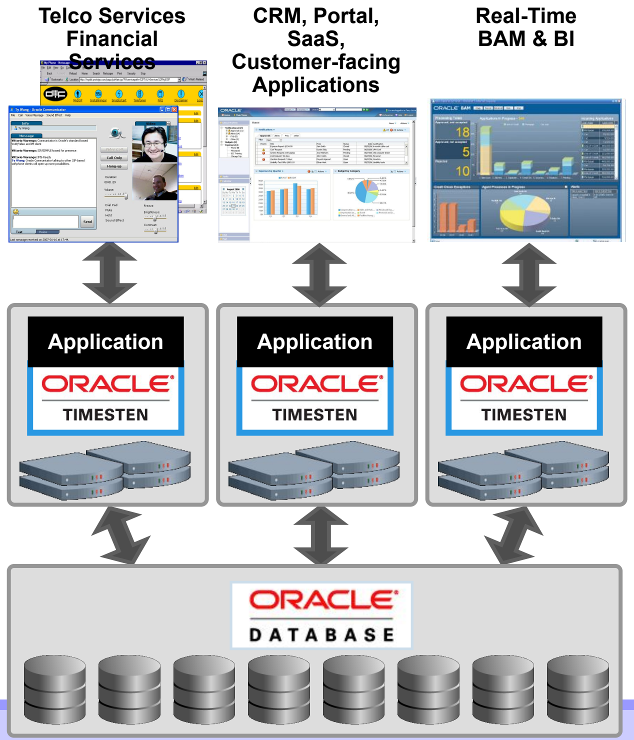
Memory optimized in-memory RDBMS for real time applications

Application-tier relational database

Delivers instant responsiveness and very high throughput

Operates as database of record or as a read/write cache for Oracle Database

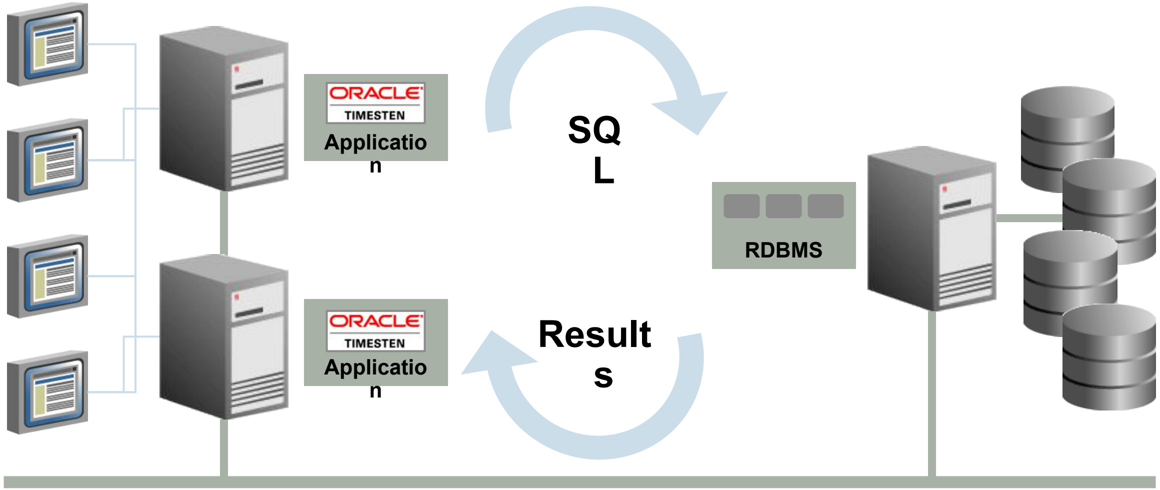
Provides replication for high availability and scalability



https://www.oracle.com/database/technologies/related/timesten.html

TimesTen database

Combines database + cache



TimesTen database features

Base product

- Everything runs in-memory – efficiency
- Local disks for persistence and recovery
- Full read/write transactional RDBMS with shares and multi-user access

High availability

- Server pairs on hot-standby
- Replication

Can be used to as a cache to an external oracle database

Oracle 10g vs timesten

Database Characteristic	Oracle Database 10*g*	Oracle TimesTen In-Memory Database
Data Model	Relational – SQL	Relational – SQL
Target Applications	All	OLTP, some DSS
Optimization	Disk-centric	Memory-centric
Typical Deployment	Database Tier	Application Tier
Architecture	Client / Server	Direct Data Access
Response Time	Milliseconds	Microseconds
Data Capacity	Tens of Terabytes	Tens of Gigabytes
Scalability	Unlimited SMP/Cluster	Good SMP

Berkeley db

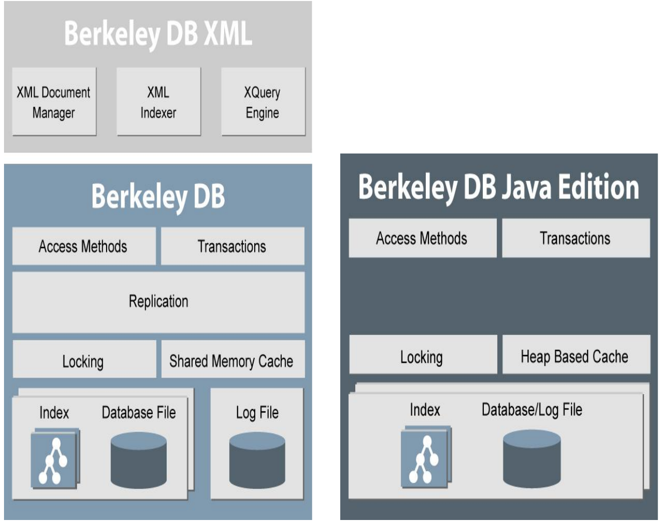
- High performance database engine
  - Runs directly in application’s address space
  - Application-native data storage
  - No SQL layer overhead

- Low total cost of ownership
  - High performance with less hardware
  - Embedded administration
  - Lower development cost: cheaper to buy vs. build

Berkeley db

All three Berkeley DB Products:

- Libraries linked to your application
- Simple, direct, indexed data storage
- Key-value pairs with simple, get-put style API
  - getDocument/putDocument for DB XML
- Operate in memory, on disk or both
- Programmatic administration API
- Low latency & high throughput
- ACID transactions and recovery
- Open source



Summary

Embedded databases are databases tightly coupled with applications

- with a small footprint
- requires very little maintenance
- transparent to the end user

Two architectures: lightly vs deeply embedded

Commercial embedded database examples (Oracle)

Embedded DB - Mini Project

Implement the data storage of an Android based expense manager application using an embedded database.

More details on the submission will be provided on Moodle.

Visit: https://github.com/GayashanNA/SimpleExpenseManager

References

Database System Concepts, Sixth Edition, Avi Silberschatz, Henry F. Korth, S. Sudarshan

Oracle embedded databases, https://www.slideshare.net/Prem02/oracle-embedded

Seltzer, Margo I., and Michael A. Olson. "Challenges in Embedded Database System Administration." USENIX Workshop on Embedded Systems. 1999.