20 - Lecture - Software architecture

The big picture: software architecture

Client-server architecture

```
    browser       <--------------------------> web server
    nc            <--------------------------> mdb-lookup-server
    TCPEchoClient <--------------------------> TCPEchoServer
```

The evolution of mdb-lookup-server

1) local command line: mdb-lookup

2) client-server architecture

   mdb-lookup-server-nc.sh
   mdb-lookup-server-nc-1.c
   mdb-lookup-server-nc-2.c
   mdb-lookup-server

3) web-based client-server architecture

   browser <--> our own web server <--> mdb-lookup-server

   broswer <--> apache web server / our module <--> mdb-lookup-server

4) web application server architecture

   browser <--> apache web server / PHP <--> RDBMS

   - with MySQL RDBMS, it’s called "LAMP" architecture

   browser <--> tomcat / Servlet & JSP <--> RDBMS

   browser <--> Java EE application server / EJB <--> large-scale RDBMS

   - JEE app servers: IBM WebSphere or JBoss
   - large scale RDBMS: Oracle

   or some combo:

   - apache web server together with tomcat
   - jboss includes embedded version of tomcat

   resources:

   - http://faq.javaranch.com/java/WebVsApplicationServer
   - CS W4111 DATABASE SYSTEMS
   - CS W4156 ADVANCED SOFTWARE ENGINEERING

The 3-tier architecture

- presentation tier
- application tier
- data tier

Database & SQL (optional material - not covered in class)

Client-server RDBMS systems:

- Oracle
- Sybase
- IBM DB2
- MySQL
- PostgreSQL

SQLite

From www.sqlite.org:

> SQLite is a software library that implements a self-contained, serverless, zero-configuration, transactional SQL database engine.
>
> SQLite is the most widely deployed SQL database engine in the world. It is used in countless desktop computer applications as well as consumer electronic devices including cellphones, PDAs, and MP3 players. The source code for SQLite is in the public domain.

Example session:

```
$ sqlite3 ex1
SQLite version 3.3.10
Enter ".help" for instructions
sqlite> create table tbl1(one varchar(10), two smallint);
sqlite> insert into tbl1 values('hello!',10);
sqlite> insert into tbl1 values('goodbye', 20);
sqlite> select * from tbl1;
hello!|10
goodbye|20
sqlite>
```