Class Announcements

- Midterm Thursday 10/27
  - Study guide to be posted soon

- Assignment 3 due Tuesday 10/25
E-commerce Marketing

• Internet provides marketers with new ways of identifying and communicating with customers.

• Long tail marketing:
  • Sell large number of unique items
  • Relatively few of each item sold

• Behavioral targeting: tracking online behavior of individuals on thousands of Web sites.

• Advertising formats include search engine marketing, display ads, rich media, and e-mail.
Architecture

by

David G. Messerschmitt
©Copyright David G. Messerschmitt, 2000. This material may be used, copied, and distributed freely for educational purposes as long as this copyright notice remains attached. It cannot be used for any commercial purpose without the written permission of the author.
What is Architecture?

How do you architect a solution?
A system is decomposed into interacting subsystems.

Each subsystem may have a similar internal decomposition.
Three elements of architecture

- Decomposition
- Functionality
- Interaction
- Organization
- Responsibility
- Cooperation

Slide adapted from slides for *Understanding Networked Applications* by David G Messerschmitt. Copyright 2000. See copyright notice.
System examples

Let’s quickly look at some system decomposition examples

- Quick tour of information technology systems
Time sharing

ASCII terminal
(no graphics)

Point-to-point wire
(no network)

Mainframe
(database and application server)
Two-tier client/server

Local-area network

Server/ Mainframe
Three-tier client/server

- Client
- Application server
- Enterprise data server
System integration

1. Architecture
2. subsystem implementation
3. system integration
   Bring together subsystems and make them achieve desired system functionality
   - Testing
   - Modifications often needed
Emergence

Subsystems are
- specialized
- have simple functionality

Higher-level system functionality arises from the interaction of subsystems

Called: Emergence

e.g. airplane flies, but subsystems can’t
Why system decomposition?

- Divide and conquer approach to containing complexity
- Reuse
- Consonant with industry structure (unless system is to be supplied by one company)
- Others?
Networked computing infrastructure

by

David G. Messerschmitt
Layering

- Elaboration or specialization

- Services

- Existing layers
Example of Layering: networking

- Physical
  - Signals
  - Bits
  - Frames
  - Packets
  - Messages
- Application
- Transport
Software Layering

- Application
- Middleware
- Operating System
Operating system functions

- Graphical user interface (client only)
- Hide details of equipment from the application
- Multitasking
- Resource management
  - Processing, memory, storage, etc
- etc
Middleware Functions

- Capabilities that can be shared by many applications, but that is not part of OS
  - Example: Database Management System (DBMS)
- Hide details of OS from application
  - Java Virtual Machine
- More purposes we’ll talk about later.
What’s a database?

Database

- File with specified structure
- Example: relational table
## A Database

<table>
<thead>
<tr>
<th>Year</th>
<th>City</th>
<th>Accommodation</th>
<th>Tourists</th>
</tr>
</thead>
<tbody>
<tr>
<td>2002</td>
<td>Oakley</td>
<td>Bed&amp;Breakfast</td>
<td>14</td>
</tr>
<tr>
<td>2002</td>
<td>Oakley</td>
<td>Resort</td>
<td>190</td>
</tr>
<tr>
<td>2002</td>
<td>Oakland</td>
<td>Bed&amp;Breakfast</td>
<td>340</td>
</tr>
<tr>
<td>2002</td>
<td>Oakland</td>
<td>Resort</td>
<td>230</td>
</tr>
<tr>
<td>2002</td>
<td>Berkeley</td>
<td>Camping</td>
<td>120000</td>
</tr>
<tr>
<td>2002</td>
<td>Berkeley</td>
<td>Bed&amp;Breakfast</td>
<td>3450</td>
</tr>
<tr>
<td>2002</td>
<td>Berkeley</td>
<td>Resort</td>
<td>390800</td>
</tr>
<tr>
<td>2002</td>
<td>Albany</td>
<td>Camping</td>
<td>8790</td>
</tr>
<tr>
<td>2002</td>
<td>Albany</td>
<td>Bed&amp;Breakfast</td>
<td>3240</td>
</tr>
<tr>
<td>2003</td>
<td>Oakley</td>
<td>Bed&amp;Breakfast</td>
<td>55</td>
</tr>
<tr>
<td>2003</td>
<td>Oakley</td>
<td>Resort</td>
<td>320</td>
</tr>
<tr>
<td>2003</td>
<td>Oakland</td>
<td>Bed&amp;Breakfast</td>
<td>280</td>
</tr>
<tr>
<td>2003</td>
<td>Oakland</td>
<td>Resort</td>
<td>210</td>
</tr>
<tr>
<td>2003</td>
<td>Berkeley</td>
<td>Camping</td>
<td>115800</td>
</tr>
<tr>
<td>2003</td>
<td>Berkeley</td>
<td>Bed&amp;Breakfast</td>
<td>4560</td>
</tr>
<tr>
<td>2003</td>
<td>Berkeley</td>
<td>Resort</td>
<td>419000</td>
</tr>
<tr>
<td>2003</td>
<td>Albany</td>
<td>Camping</td>
<td>7650</td>
</tr>
<tr>
<td>2003</td>
<td>Albany</td>
<td>Bed&amp;Breakfast</td>
<td>6750</td>
</tr>
</tbody>
</table>
Storage Middleware example: DBMS

- **Database Management System (DBMS)**
  - Manage Multiple databases
  - Allow multiple applications to access common databases
  - Implement standard data “lookup” (query) functions.
Client - Server Computing
Client Server Example

Client

“I want to see
www.google.com”

Server

<html><head><meta http-equiv="content-type" content="text/html;
charset=UTF-8"><title>Google</
title><style><!--
body,td,a,p,.h{font-family:arial,sans-serif;}
.h{font-size: 20px;}
.q{color:#0000cc;}
//--> ...
</style></head>
<body>

Google

</body></html>
Client Server Example - Layers Revealed

Client

Application:

Infrastructure

Packet

Internet

Server

Application

<html><head><meta http-equiv="content-type" content="text/html; charset=UTF-8"><title>Google</title><style><!--
body,td,a,p,.h{font-family:arial,sans-serif;}
.h{font-size: 20px;}
.q{color:#0000cc;}
//-->
...</style></head><body><a href="http://www.google.com">Google</a><p>Packet</p>
</body></html>
### 3-Tier Client Server Architecture example

**Client**

- **Clicks, keystrokes**
- **Balance $0.50**

**Application Server**

- **What is Bob’s balance?**
- **$0.50**

**Shared data**
3-Tier Client Server Architecture example

Client

Application Server

Web Server

Common Gateway Interchange

Application Logic

Shared data
What is Bob’s Balance?
3-Tier Client Server Architecture example

In some implementations, Application Logic and Web Server can be put on different machines.

- Client
- Application Server
  - Web Server
  - Java Servlet
  - Database Management System (DBMS)
  - Database

What is Bob’s Balance?

Shared data
Relational Database

<table>
<thead>
<tr>
<th>Customer</th>
<th>Balance</th>
<th>Customer Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alice</td>
<td>$527</td>
<td>Silver</td>
</tr>
<tr>
<td>Bob</td>
<td>$0.50</td>
<td>Bronze</td>
</tr>
<tr>
<td>Charles</td>
<td>$1000000</td>
<td>Gold</td>
</tr>
</tbody>
</table>
DBMS Responsibilities

- Hide Changes in the Database hardware from the Application

- Standard operations on the data, including searches, such a search is called a query.

- Separate Database Management from Applications, so that many applications can access the same data.

- Security, Integrity, Backup, fault tolerance, etc..
3-Tier Client Server Architecture in General

Client

- Accept instructions from user
- Make requests of server
- Display responses of server

Application Server

- Takes inputs from client
- Decides what to be done next
- Decides what shared data to access and manipulates it
- Processes shared data

Shared data

- Support multiple applications with common data
- Protect critical data
- Decouple data administration and application administration