Class Announcements

- **Midterm 2/9**
  - Study guide posted
  - Midterm Review Sessions
    - **Tuesday**
      - 6-7:10 pm
      - Natural Sciences Annex 101
    - **Wednesday**
      - 11am - 12:10 PM
      - Humanities Lecture Hall 206
  - John M’s office hours this week:
    - Thursday 2–4pm → Tuesday 2-4pm
  - **New assignment to be posted Thursday**
Data and information

by

David G. Messerschmitt
Copyright notice

©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.
Key concept

The key commodity manipulated by information technology is information. To be manipulated by IT, information must be represented by data.

What is information?
From a user (human) perspective...
....recognizable patterns that influence you in some way
(perspective, understanding, behavior...)

In computing, information has a somewhat different connotation:
structure and interpretation added to data
Data

A bit is “0” or “1” — the atom of digital tech

Data is a collection of bits, like

- “010110111010110”

- “Note: the terms data and information are not always used consistently!”
**Representation**

- Take the place of the original
- “Equivalent to,” in sense that original can be reconstructed from its representation
- Often original can only be approximately reconstructed,
  - e.g. audio or video
<table>
<thead>
<tr>
<th>Alphabet</th>
<th>Hex</th>
<th>Binary</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;7&gt;</td>
<td>/x37</td>
<td>00110111</td>
</tr>
<tr>
<td>&lt;8&gt;</td>
<td>/x38</td>
<td>00111000</td>
</tr>
<tr>
<td>&lt;9&gt;</td>
<td>/x39</td>
<td>00111001</td>
</tr>
<tr>
<td>&lt;&gt;</td>
<td>/x3A</td>
<td>00111010</td>
</tr>
<tr>
<td>&lt;;&gt;</td>
<td>/x3B</td>
<td>00111011</td>
</tr>
<tr>
<td>&lt;&lt;&gt;</td>
<td>/x3C</td>
<td>00111100</td>
</tr>
<tr>
<td>&lt;=</td>
<td>/x3D</td>
<td>00111101</td>
</tr>
<tr>
<td>&lt;)&gt;&gt;</td>
<td>/x3E</td>
<td>00111110</td>
</tr>
<tr>
<td>&lt;?&gt;</td>
<td>/x3F</td>
<td>00111111</td>
</tr>
<tr>
<td>&lt;At&gt;</td>
<td>/x40</td>
<td>01000000</td>
</tr>
<tr>
<td>&lt;A&gt;</td>
<td>/x41</td>
<td>01000001</td>
</tr>
<tr>
<td>&lt;B&gt;</td>
<td>/x42</td>
<td>01000010</td>
</tr>
<tr>
<td>&lt;C&gt;</td>
<td>/x43</td>
<td>01000011</td>
</tr>
<tr>
<td>&lt;D&gt;</td>
<td>/x44</td>
<td>01000100</td>
</tr>
</tbody>
</table>

Note that this representation is not unique…

….this one happens to be a standard (ANSI X3.110-1983)
A picture

This picture conveys information

This information is represented in this computer, but how?
Representation of picture: image

Expanding a small portion of the picture, we see that it is represented by square pixels....

....300 tall by 200 wide....

....with a range of 256 intensities per pixel

An approximation!

300 • 200 • 8 bits = 480,000 bits (but it can be compressed)
A color picture can be represented by three monochrome images…

At the expense of three times as many bits
Terminology

Data Representation

Communicate data to another user or organization

Data processing

Slide adapted from slides for *Understanding Networked Applications* by David G Messerschmitt. Copyright 2000. See copyright notice.
Representation needs to be standardized

If the representation is not standardized, the information is garbled!

Communicate data to another user or organization

Slide adapted from slides for Understanding Networked Applications
By David G Messerschmitt. Copyright 2000. See copyright notice
Regeneration

- Make a precise copy of the data (copy bit by bit)
- If you know the representation, equivalent to making a precise copy of the information
- Each such copy is called a generation
- process is called regeneration
Replication of information

Anything that can be regenerated can be replicated any number of times

This is a blessing and a curse
Analog information cannot be regenerated

Analog information can be copied, but not regenerated
Discrete information can be regenerated

Regeneration can preserve data (but not its original physical form)

“A” + noise = “A”

0 + noise  0
1 + noise  1
Replication of information requires knowledge of representation.

Information → Replication of information also presumes knowledge of its representation → Data

Data ← Replication preserves the integrity of the data, but that is not sufficient ← Data

Example: Every .doc file is a representation

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

Digitally represented information can
- be preserved over time or distance
- by occasional regeneration
  - digital library
  - digital telephony
Implications (con’t)

- Replication of information requires knowledge of the structure and interpretation
  - Standardization or some other means
- Extreme supply economies of scale
- You can give away or sell and still retain
- Unauthorized replication or piracy relatively easy
Architecture

by

David G. Messerschmitt
Copyright notice

©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
- Organization
- Functionality
- Responsibility
- Interaction
- 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
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

- Application
- Transport
- Network
- Link
- Physical

Messages ➔ Packets ➔ Frames ➔ Bits ➔ Signals
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>
</body></html>
Client Server Example - Layers Revealed

Client

Application:

Server

Application

Internet

Infrastructure

Packet
Packet

 Packet
Packet

Infrastructure

<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;}//-->

<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;}//-->
3-Tier Client Server Architecture example

Client

Application Server

Clicks, keystrokes

What is Bob’s balance?

Balance $0.50

Shared data

$0.50