Outline

- Announcements
- Components & Industry
- Student Presentation(s)
- Standardization
- MySQL
Assignment 3 out today

Wednesday 11/16:

Reading for next class:

Ch. 15.3.1 - 15.3.3, 15.3.6, 15.4 of Messerchmitt (pp. 426-430, 432-437)
Student Presentations

Today:
Natalja Robinetts (Sun)
Components, Suppliers
Components (Examples?)

Component: A subsystem purchased “as is” from an outside vendor

(Alternative – building your own subsystem)

A component implementation is encapsulated (although often configurable)

Slide adapted from slides for Understanding Networked Applications
By David G Messerschmitt. Copyright 2000. See copyright notice
System Integration, Emergence

System Integration:
take the components, add custom developed subsystems, make them interact → reach higher level goal

Emergence: new capabilities and functions of a system that subsystems and components could not have provided by themselves.
Supplier Types

- Three types of infrastructure/application suppliers:
  - Component Suppliers
  - Custom Subsystem Developers
  - System Integrators

- (Some suppliers are 2 or even 3 of above.)
Four possibilities (examples)

- **Product**
  - Microsoft Office
  - Microsoft Windows

- **Service**
  - Hotmail
  - Internet DNS (Domain Name System)

Slide adapted from slides for *Understanding Networked Applications*
By David G Messerschmitt. Copyright 2000. See copyright notice
Application Service Providers

Two types

- Bundled
  - An infrastructure provider bundles applications with their infrastructure
  - Example: AOL, telephony service providers

- Unbundled
  - A provider of an application service without providing an infrastructure service
  - Examples?
Examples of unbundled ASP model

- Web-based calendar (e.g. Yahoo, Google)
- Web-based email (e.g. Hotmail, Gmail)
- Web-based stock trading (e.g. Charles Schwab)
Application acquisition

Application

- Develop internally
- Buy as product
- Contract development
- Product w/ customization

Software supplier
Outsource developer
Supplier, consultants
The changing industry structure
Stovepipe vs. Integrated Infrastructure

**Stovepipe Architecture**
---or---
**Turnkey Solution**

- Single supplier provides all encompassing solution
- (complete with infrastructure)

**Integrated Infrastructure**

Separate infrastructure that can support many applications

Application and Infrastructure

Application

Infrastructure

Slide adapted from slides for *Understanding Networked Applications* by David G Messerschmitt. Copyright 2000. See copyright notice.
From stovepipe to layering

- Data
- Voice
- Video

Many applications

Integrated Infrastructure (Maybe broken into Additional layers.)

Application-dependent infrastructure

Application-independent
Stovepipe vs. Integrated Infrastructure

- What are some examples of each?
  - Telephone network
  - Broadcast tv
  - Internet
  - Pc

- What are the advantages of each approach?

Slide adapted from slides for *Understanding Networked Applications* by David G Messerschmitt. Copyright 2000. See copyright notice.
Vertical Integration - Diversification

- Two approaches for companies wishing to expand their product offerings

- A company is vertically integrated when it makes rather than buys the subsystems in its products. Example: IBM

- A diversified company produces products across different industry segments. Example: Compaq
Less Vertical Integration - More Diversification?

Why do customers favor less vertical integration?
- Prefer competition amongst component suppliers
- Mix and match components - free choice, but at a price
- Reduced lock in

Disadvantages??
- Customer needs to integrate components from different suppliers.

Slide adapted from slides for Understanding Networked Applications
By David G Messerschmitt. Copyright 2000. See copyright notice
Less Vertical Integration - More Diversification

Why do customers favor diversification (in the application space)?

- Reduce coordination costs by having to deal with fewer suppliers.
- Fewer vendors overall $\Rightarrow$ less chance of toxic interactions
Standardization
Purpose of a standard?

- Allow products or services from different suppliers or providers to be interoperable
Scope of a standard

- **Included:**
  - interfaces (physical, electrical, information)
  - architecture (reference model)
  - formats and protocols (FAP)
  - compliance tests (or process)

- **Excluded:**
  - implementation
  - (possibly) extensions
The Standardization Process

- Before something becomes a standard requires:
  - recognition of its need by a standards body/industry/government
  - commitment of monetary and human resources by participants
- Usually, is an ongoing process
  - Refinements/Extensions

Examples:

ISO:  http://standards.iso.org/ittf/PubliclyAvailableStandards/
W3C:  http://www.w3.org/
The Standardization Process

- Each organization participates in Working Groups/Committees of interest
  - Hold periodic meetings for debates/arguments/negotiation
  - When reaching a consensus, publish a RFC (Request for Comments) draft
  - Others can give feedback/Send comments etc.
  - The committee should answer to all comments and incorporate needed changes
  - Time-consuming process

- Results in extensive documentation and sometimes in system prototypes

- Usually standards evolve
  - Backward compatibility (e.g. MPEG)
  - Compatibility with existing standards (e.g. XQuery, XSLT based on XPath)
Some issues

- Slow and cumbersome process

- Once a standard is set
  - becomes possible source of industry lock-in; overcoming that standard requires a major advance
  - may lock out some innovation
Why do companies participate?

- Influence the standard
- Gain expertise and implement prototypes
  - Faster time to market than competitors
- Gain intelligence about competitors
  - That might be part of the standardization body as well
- May benefit financially through patent protection and royalties
  - Maintaining ownership of proprietary technology
- Many companies contribute their expertise to design something bigger
Types of standards

- **de jure**
  - Sanctioned and actively promoted by some standardization body, or by government

- **de facto**
  - Standard practice
  - Dominant solution arising out of the market, OR
  - Recommended by voluntary industry standards body

- Examples?
Examples

de jure

- GSM (global for mobile communication),
- ISDN (Integrated Services Digital Network)
  Telephone interface

de facto

- Windowed GUI
- Java
- Internet protocols

Voluntary industry standards body

- IEEE (Institute of Electrical and Electronic Engineers)
- IETF (Internet Engineering Task Force)
- EPCglobal (RFID standard for UHF)

Industry consortium

- W3C (World Wide Web Consortium)
- SET (Secure Electronic Transactions)

Slide based on slides for Understanding Networked Applications
By David G Messerschmitt. Copyright 2000. See copyright notice
The changing process

- As technology and industry move more quickly, the global consensus standards activity has proven too unwieldy
  - e.g. ISO

- “New age” standards activities are more informal, less consensus driven, a little less political, more strategic, smaller groups
  - e.g. W3C, IETF, WAP

- Programmable/extensible approaches for flexibility
  - e.g. XML, Java
Reasons for change

- From government sanction/ownership to market forces
  - Increasing fragmentation
  - Importance of time to market
Open vs. Proprietary Standards

- **Open standard** - a standard that is well documented, unencumbered by intellectual property rights and restrictions, and available to any vendor
  - e.g. Internet protocols

What are the advantages?

What are the disadvantages?
Standards applied to Business Processes?

- Can you standardize business processes?
- Yes!
  - ISO 9000
    - A set of standardized business processes for Quality Management.
    - Supports TQM (Total Quality Management)
  - RosettaNet
    - A set of standardized business processes, and accompanying standardized data interfaces/formats for conducting e-business.
  - BPEL (Business Process Execution Language)
    - An XML-based language for the formal specification of business processes and business interaction protocols
The role of Venture Capital in Computing

- **Start-Ups**: Open interfaces allow small firms to contribute components without having to develop entire solution

- High risk for VCs

- Diversification
  - Each VC funds multiple start-ups
  - Each start-up funded by multiple VCs

- Is this model successful? For the start-up? For the VC? For the customers? Why?
mySQL Case
mySQL

What does mySQL make?

How Successful is mySQL?
- Visibility: Fortune magazine, more mentions on www
- Reaction from giants
- Revenue growth 2001 700k, 2002 6.2m, 2003 10m
- Good performance reviews
- Recent SAP alliance
- But Market share tiny:
  - $10 million out of $10 billion market!

Why Success?
- Good Technology
- Large DBMS bloated with features most don’t need
- Innovative OSS model
mySQL

How does OSS work?

Two Types of License:

- **GPL (General Public License (GNU))**
  - Free
  - No Support
  - Any software that uses MySQL as a module must itself be made GPL-compliant

- **Commercial License**
  - Support
  - Could be distributed with non-open source software
  - Not Free:
    - MySQL: Classic $250, Pro $495 (for ~ 50 users)
    - Compare to:
      - MSFT $3150 single proc for 50 users
      - IBM $33000 single proc for 50 users
      - Oracle $40000 single proc for 50 users
Aside: DB’s in different software stacks

- Operating System
  - MS Windows or other OS
- Middleware
  - Oracle or MySQL, IBM, etc.
- Application
  - SAP or Oracle, Axtapa, etc.

**ERP Software Stack**

- Proprietary Business Logic
- Apache Web Server

**Web Application Software Stack**

- MySQL or other DB
- Linux or other OS

**Banking Software Stack**

- Proprietary Banking App.
- Oracle or other DB
- IBM z/OS or other OS

- Which companies are competitors?
- Which are complimenters?
- Which are both!?
mySQL

- Which segments of market is mySQL strong in?
  - Large Companies or Small Companies?
  - Web applications or Critical Enterprise data?

- Why would a major enterprise want to pay so much more for an Oracle or IBM DB?

- How should MySQL proceed? What are the advantages/disadvantages?
### My SQL: market

<table>
<thead>
<tr>
<th>Enterprise wide data (90%)</th>
<th>Small (20%)</th>
<th>Medium (30%)</th>
<th>Large (50%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft</td>
<td></td>
<td></td>
<td>Oracle, IBM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Reliability, Scalability, Support, Longevity</td>
</tr>
<tr>
<td>Web Sites (10%)</td>
<td>My SQL Cost</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

How should mySQL grow in order to meet it’s stated goal of getting to $100 million in revenue?

---

Figure Adapted from “Teaching Note for MySQL Open Source Database,” 6/1/04, Stanford GSB.
- Lack of Brand identity in this segment
- MySQL lacks the organization to offer support
- Large enterprises have high switching costs

Figure Adapted from “Teaching Note for MySQL Open Source Database,” 6/1/04, Stanford GSB.
### My SQL: Growth Strategy

<table>
<thead>
<tr>
<th>Enterprise wide data 90%</th>
<th>Small 20%</th>
<th>Medium 30%</th>
<th>Large 50%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft</td>
<td></td>
<td></td>
<td>Oracle IBM</td>
</tr>
<tr>
<td>Web Sites 10%</td>
<td>My SQL</td>
<td>Cost</td>
<td>Reliability</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Stay Put?</td>
<td>Scalability</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Support</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Longevity</td>
</tr>
</tbody>
</table>

- Not a big enough market to reach stated $100 million goal.

Figure Adapted from “Teaching Note for MySQL Open Source Database,” 6/1/04, Stanford GSB.
### My SQL: Growth Strategy

<table>
<thead>
<tr>
<th></th>
<th>Small 20%</th>
<th>Medium 30%</th>
<th>Large 50%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enterprise</td>
<td>Microsoft</td>
<td>Oracle</td>
<td>IBM</td>
</tr>
<tr>
<td>wide data</td>
<td>90%</td>
<td>Reliability</td>
<td>Support</td>
</tr>
<tr>
<td>Web Sites</td>
<td>My SQL</td>
<td>Scalability</td>
<td>Longevity</td>
</tr>
<tr>
<td>10%</td>
<td>Cost</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Many of these customers already using MySQL with websites
- Less emphasis on global organization
- Leverage SAP alliance
- Up against Microsoft.

Figure Adapted from “Teaching Note for MySQL Open Source Database,” 6/1/04, Stanford GSB.
My SQL: Growth Strategy

- + builds on existing brand and strengths
- - Market not so big

<table>
<thead>
<tr>
<th></th>
<th>Small 20%</th>
<th>Medium 30%</th>
<th>Large 50%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enterprise wide data 90%</td>
<td>Microsoft</td>
<td>Oracle IBM</td>
<td>Reliability, Scalability, Support, Longevity</td>
</tr>
<tr>
<td>Web Sites 10%</td>
<td>My SQL Cost</td>
<td>Maybe?</td>
<td></td>
</tr>
</tbody>
</table>