ISM 50 - Business Information Systems

Lecture 12

Instructor: John Musacchio
UC Santa Cruz
May 4, 2005

Outline For Today

- Class Announcements
- Student Presentations
- Finish Chapter 7
- Start Ch 11, 15

Class announcements

- Database assignment out
  - (Due in 19 days.)
- Access Database tutorial (BE 109)
  - Thursday 2-4
  - Friday 4-6
  - Please go to session you signed up for. If you indicated "indifferent," go to the Friday session.
- Read MySQL and finish Ch 15 Case for Monday.
- Student Presentations Monday
  - Lei, Miaoting (news)
  - Lo, Christopher Jing-Ping (mySQL case)
  - Please – no more stories on MAC OS!

Database Project (due May 23)

- Recommend a PDA for SoE to buy for students.
  - (this is pretend!)
- Design a Survey
  - Give to at least 20 friends/students
  - Collect data into MS Access Database
- Analyze data to recommend
  - Brand, Model, Features
- Report
  - 1 page summarizing final recommendation
  - 3-4 page analysis
  - Also turn in: Access files, and all survey forms

Student Presentations

Moya, Antonio Julian

Shen, David

Standardization

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

**Reference model**

Decide decomposition of system
- where interfaces fall
Defines the boundaries of competition and ultimately industrial organization
- competition on the same side of an interface
- complementary suppliers on different sides
- hierarchical decomposition at the option of suppliers
- (possibly) optional extensions at option of suppliers

**Some issues**

Once a standard is set
- becomes possible source of industry lock-in; overcoming that standard requires a major (~10x?) advance
- may lock out some innovation

In recognition, some standards evolve
- IETF, CCITT (modems), MPEG
- backward compatibility

**Types of standards**

- **de jure**
  - Sanctioned and actively promoted by some organization with jurisdiction, or by government
- **de facto**
  - Dominant solution arising out of the market
  - Voluntary industry standards body
- Industry consortium
- Common or best practice

**Examples**

- **de jure**
  - GSM, ISDN Telephone interface
- **de facto**
  - Hayes command set, Windows API, Pentium instruction set, Ethernet
  - Voluntary industry standards body
  - OMG/CORBA, IAB/IETF, IEEE
  - Industry consortium
  - W3C/XML, SET
  - Best practice
  - Windows GUI
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. OMG, IETF, ATM Forum, 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
- Greater complexity
  - Less physical/performance constraint for either hardware or software

Lock-in

(Particularly open) standards reduce consumer lock-in

- Consumers can mix and match complementary products

Increase supplier lock-in

- Innovation limited by backward compatibility
  - e.g. IP/TCP, x86, Hayes command set

Network effects

Standards can harness network effects to the industry advantage

- Revenue = (market size) x (market share)
- Increases value to customer
- Increases competition
  - Only within confines of the standard
  - But forces customer integration or services of a system integrator

Question

What are examples of standards that serve to tame network effects?

- Internet protocols
- XML
- CORBA
- DVD
- others?
Why standards?

*de jure* are customer driven to reduce confusion and cost
*de facto* standards are sometimes the result of positive feedback in network effects

Customers and suppliers like them because they
- increase value
- reduce lock-in

Governments like them because they
- promote competition in some circumstances
- May believe they can be used to national advantage

Approaches

Consensus
- ISO
Collaborative design
- MPEG
Competitive “bake off”
- IETF
Coordination of vendors
- OMG

Open Standards

- Open standard - a standard that is well documented, unencumbered by intellectual property rights and restrictions, and available to any vendor.
- What are the advantages?
- What are the disadvantages?

Why companies participate

Pool expertise in collaborative design
- e.g. MPEG
Have influence on the standard
Get technology into the standard
- Proprietary, with expectation of royalties
- Non-proprietary
Reduced time to market

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.

Chapter 11

by

David G. Messerschmitt
Algorithms and protocols

by
David G. Messerschmitt

Algorithm

Specified sequence of steps that
- accomplish a designated task
- in a finite number of steps

Representation:
- simple algorithm: flowchart
- complicated algorithm: program

Example: one turn at monopoly

Start turn
Throw dice
Move token number of squares indicated on dice

Move to "jail" square

Yes
Land on "go to jail"?

No
Do not move; follow policies for square (like "pay rent")

Finish turn

Algorithm building blocks

Start
Action
Decision
Action
Action
Action
Finish
Sequence
Selection
Loop

Start
Action

Programming languages support these three building blocks

Protocol

- Distributed algorithm ...
- Realized by two or more modules to coordinate their actions or accomplish some shared task
- Module interoperability requires a protocol
  - Prescribed order of method invocations
  - Part of interface documentation

Monopoly players protocol

Player 1
Player 2

One-turn algorithm

This is a protocol interaction diagram
Application and infrastructure

The application defines its own application-level protocols.

Internally, the network uses protocols to implement the services it provides.

Three simple protocols

One-way message: send-receive
Two-way interaction: request-response
Push: publish-subscribe

Client Server

Time

send receive

request response subscribe responses

Example: HTTP

User activates URL

HTTP request

HTTP server

HTTP client (browser)

HTTP response (embedded document)

Browser displays document (if HTML) or invokes "helper application"

HTML documents

Locating things

by David G. Messerschmitt

Three ways of locating things

Name
- "Joe Bloe"
Address
- "1299 Hearst St, Berkeley, CA"
Reference
- "Postmaster of Berkeley CA"
**Name**

- Symbolic (character string) representation
- Easy for people to remember or guess
- Identifies but does not locate directly
  - Distinction important for mobile entities
- Not unique: entities can have more than one name (called aliases)

**Hierarchical names**

Hierarchy makes names easier to remember or guess.

- **Host domain names:**
  - "info.sims.berkeley.edu"
  - designates administrative hierarchy
- **File names:**
  - "c:\My Documents\Docs\Resume.doc"
  - designates folder hierarchy

**Address**

- Route or path to entity
  - is directly specified, or
  - can be inferred
- Independent of who or what is locating entity
- Topological specification

**Example**

- Path from to is (1,2,2,1,1,1)
- Not an address, because it depends on starting point

**Example**

- Address of is (6,5)
- Route from can be inferred

**Reference**

Abstract representation of an entity

- Interaction is with representation
  - infrastructure arranges redirection to actual entity
  - especially appropriate for things that move
Name services

1. name
2. address or reference
3. interaction

Name service

Chapter 15

by
David G. Messerschmitt

Databases

by
David G. Messerschmitt

Treat data as a separate asset
- May be shared by multiple applications
Provide protection and integrity features appropriate to mission-critical data
- Access control
- Integrity constraints
- Persistence
- etc.

Two capabilities

Aggregation: accessing multiple databases
Sharing: two or more applications accessing the same databases

Relational table

<table>
<thead>
<tr>
<th>Table</th>
<th>Employee</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Address</td>
</tr>
<tr>
<td>Record</td>
<td></td>
</tr>
<tr>
<td>Field</td>
<td></td>
</tr>
</tbody>
</table>
Database operations

Each operation results in a new table, so they can be concatenated.

SQL interface

- SQL (Structured Query Language)
- Presents single abstract interface to the application logic
- Standardized, not vendor specific
- Encapsulates various internal details
  - Data partitioning and replication
  - Host mapping
  - File representation
  - etc.

Object-relational database

- A column can store object instances of a given class rather than data of a given simple or compound data type
- Because of the table structure, SQL can be extended to this case
- Standard SQL queries can be extended to methods returning simple data types
- Many other good ideas

Benefits of ORDBMS

- Extension: manage arbitrarily complex data types
- Migration: preserve and extend existing databases
- Preserve SQL interface
  - OR extensions in latest standard
- All the benefits/experience of earlier databases
  - Access control, data integrity, persistence, etc.
- Killer app: Behind Web/CGI
  - Images, video, audio, animation, applets, etc.