Class Announcements
- Database Assignment 2 posted
- Due 5/27
- Business paper draft due 5/20

Infrastructure acquisition

Infrastructure
- Build and operate
- Build but do not operate
- Do not build but operate
- Neither

Trend
- Outsourced operations
- System integrator
- Service provider

Application acquisition

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

Trend
- Software supplier
- Outsource developer
- Supplier, consultants

Stovepipe vs. Integrated Infrastructure

Stovepipe architecture
- Turnkey Solution
- Single supplier provides all encompassing solution
- (complete with infrastructure)

Integrated Infrastructure
- Separate infrastructure that can support many applications

From stovepipe to layering

Many applications

Integrated Infrastructure
(Maybe broken into Additional layers.)

Application-dependent infrastructure

Application-independent
**Stovepipe vs. Integrated Infrastructure**
- What are some examples of each?
- What are the advantages of each approach?

**Vertical Integration vs. Diversification**
- A company is *vertically integrated* when it makes rather than buys the subsystems in its products.
- A *diversified* company produces products across different industry segments.

**Vertical Integration vs. Diversification**
- Why do customers favor less vertical integration?
  - Prefer competition amongst component suppliers
  - Mix and match components
  - Reduced lock in
- Disadvantages??
  - Customer needs to integrate components from different suppliers.

**Vertical Integration vs. Diversification**
- Why do customers favor diversification?
  - Reduce coordination costs by having to deal with fewer suppliers.

**General Trend**
- Less Vertical Integration
- More Diversification
- Of course there are exceptions...

**Today’s supplier structure**
- Applications
- Frameworks and components
- Middleware
- Infrastructure (network, OS) software
- Equipment (network, computers)
- Semiconductors, components
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

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?
Examples

- *de jure*
  - GSM, ISDN Telephone interface

- *de facto*
  - Microsoft Windows API (Application Programming Interface)
  - Intel Pentium instruction set
  - Voluntary industry standards body
  - IEEE (Institute of Electrical and Electronic Engineers)
  - IETF (Internet Engineering Task Force)
  - Industry consortium
  - Blu-ray
  - Best practice
  - Windowed 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. IETF
- 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

Aside: Network Effects

- The value of owning some products goes up if lots of other people have it too.
  - Examples?
- This phenomenon is called “network effects”
- How do standards influence network effects?