Class Announcements

- Assignment 4
  - Due Thursday 5/14

- Business paper draft due in 1 week!
  - Due Tuesday 5/19

- Database Assignment 2 posted
  - Due Tuesday 5/26
Two ways to design a system

System requirements

Requirements

Available components

Decomposition from system requirements

Assembly from available components

Slide adapted from slides for *Understanding Networked Applications* by David G. Messerschmitt. Copyright 2000. See copyright notice.
A component implementation is encapsulated (although often configurable)

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

(Alternative – building your own subsystem)
The Linux OS we are buying “off the shelf” and integrating into our architecture. The Linux OS is a component.
Other Examples of components

Computer
Disk drive
Network
Network router
Operating system
Integrated circuit
Database management system

Why is a component implementation encapsulated?

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

- Components are interoperable when they interact properly to achieve some desired functionality.

- Increasingly component interoperability cannot be dependent on end-user integration:
  - PC and peripherals
  - Enterprise, inter-enterprise, consumer applications
  - Role for standardization
Outsourcing: A subsystem design is contracted to an outside vendor.

Responsibility is delegated.
Suppose we choose to pay another firm to develop the user interface.
This is called **Outsourcing**.
Why would we do this?
System Integration

Suppose we
- Bring together all these subsystems
- and test them...

This is called System Integration
System integration

- Bring together subsystems;
- make them work together;
- to achieve a goal.

- Requires
  - Testing
  - Making modifications to
    - architecture and/or
    - subsystem implementation
Supplier Types

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

- (Some suppliers are 2 or even 3 of above.)
Two ways to sell Software

**Product**
- Customer installed and operated
- Often (but not necessarily) sold or licensed at a fixed price

**Service**
- Functionality provided over a wide-area network
- Often (but not necessarily) sold by subscription

Slide adapted from slides for *Understanding Networked Applications* by David G Messerschmitt. Copyright 2000. See copyright notice.
Recall: Infrastructure and Applications

Infrastructure
- Equipment and/or software used by many applications

Applications
- Provide specific capabilities and features serving individual users.
Four possibilities

Product          Service

Microsoft Office Hotmail

Application

Infrastructure

Personal computer Internet DNS

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

- Two types
  - Bundled
    - An infrastructure provider bundles applications with their infrastructure
      - Example: Comcast, telephony service providers
  - Unbundled
    - A provider of an application service without providing an infrastructure service
      - Examples?
Examples of unbundled ASP model

- Yahoo: Web-based calendar
- gmail: Web-based email
- Schwab: Web-based stock trading
Unbundled ASP model

Advantageous to user

- Proven way to reduce installation, integration, and maintenance costs
- Contractual obligation for availability and quality
- Location independence
Unbundled ASP model (cont’)

Advantages to supplier

- Ongoing revenue stream supporting upgrade and maintenance
- Usage-based revenue better aligned with user’s value proposition
- Opportunity for price discrimination, advertising revenue, etc.

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

Price discrimination?
Usage dependent?
Terms and conditions
  - fixed, leasing, per-use, subscription
  - warrantee, service level agreements
Bundles
  - maintenance, support, releases, provisioning and operations
Who pays?
  - sometimes not the end user
Infrastructure acquisition

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

Trend:
- Outsourced operations
- System integrator
- Service provider

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

--- or ---

**Turnkey Solution**

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

**Integrated Infrastructure**

- Separate infrastructure that can support many applications

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

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

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

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