System Integration

Mini Case Studies

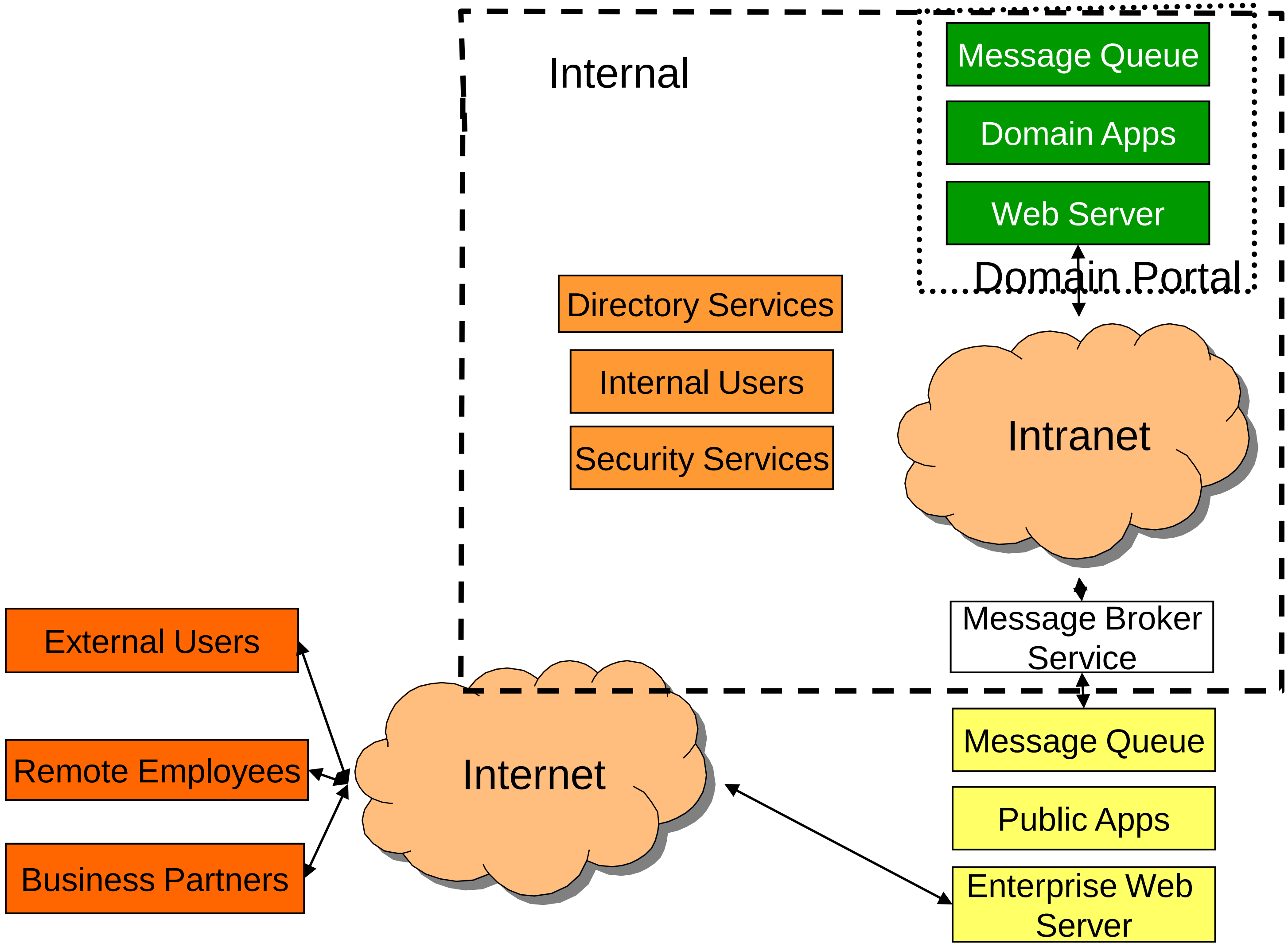
Enterprise Integration

Shawn A. Butler, Ph.D.
Senior Lecturer, Executive Education Program
Institute for Software Research
Carnegie Mellon University

Objectives

- Understand how web-based designs enable development of enterprise architectures
- Understand how XML and SOAP support enterprise integration

Integration Graph



Clients – Fat or Thin?

- Provide adequate interactive performance
- Ensure the integrity of system updates
- Minimize risk of sensitive information exposure
- Minimize network activity
- Enable disconnected user activity

Client Functionality: Forms

- Allows user to enter information
- Generally is not session based
- Information is sent to web server for processing
- Must be some process that knows how to process the information
- Very simple implementation

Client Functionality: Active Content

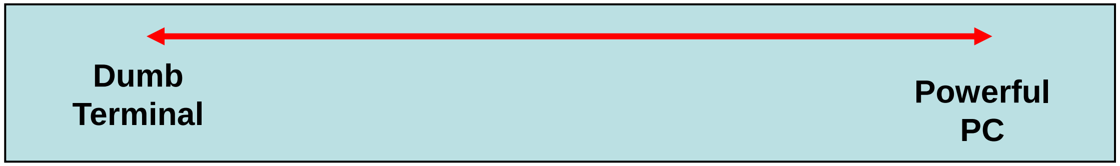
- Java Script
  - Java embedded in HTML documents
  - Interpreted by the browser
  - Provides:
    - Dialog boxes
    - Data entry validation
    - Computations
    - Actions on events
    - Dynamic content changes
    - Update browser properties
    - Cookies
- Applets

Client Functionality: Plug-ins

- A program that is executed when an applicable data type is found
- Advantages
  - Interactive performance is very good
  - Application can function without security constraints
  - Enhanced graphics techniques
  - Plug-ins can be executed off line
- Disadvantages
  - Plug-ins can contain viruses
  - Different versions are required for different platforms
  - Spoofed plug-ins

Client Functionality: Stand-Alone Apps

- Network access not always available
- The nature of the work doesn’t require continuous access to services
- The application can take advantage of the power of the PC
- Application graphics and inputs are large and stable so there isn’t any need for downloads
- Reliance on network introduces added risks to work



Client Functionality: Static Web Pages

- Simplest form of display
- Contains graphics and text
- Embedded links
- HTML based – defines content elements
- May use standard formats encoded in Cascading Style Sheets (CSS)

Server Functionality

- CGI
  - Initial scripting language that provided functionality to client
  - Serious security flaws
  - Compatibility Issues
- Servlets – Java component that executes inside a servlet container
  - Provides interfaces to JDBC and JNDI
  - Not loaded on each request
  - Portable
  - Provide useful services which simplifies programming
- Java Server Pages – Special form of servlet
  - JSP template separates layout from content computations
  - JSP specification defines transformation of JSP code to/XML format

Session Management

- Sessions:
  - Enable completion of an objective or task
  - Ensure security
  - Enhance user experience
- Session Mechanisms
  - URL variables
  - Cookies
    - Temporary or Persistent
  - Secure Socket Layer

Enterprise Client Devices

- Specialized devices are becoming an essential part of the enterprise solution
  - Cell phones/PDA’s
  - Mobile media devices
- Protection of these devices is a significant challenge
- User interfaces change among different devices
- Wireless connections are expanding

Service Oriented Architectures (SOA)

- Characterized by dynamic discovery of services and information
- No single definition exists
- Loose coupling of components
  - Small set of interfaces
  - Extensible schemas
- Stateless – is the norm
- Web Services use SOAP
  - Provides the envelope for delivery
  - Provides RPC capability

XML

- Nonproprietary
- Platform Independent
- HTTP compatible
- International
- Extensible
- Self-defining
- Common Tools
- Transformation

XML Extended Technology

- Name Spaces
  - Unique naming of elements
- XML Schema
  - DTD defines the structure of an XML document
  - Required and prohibited elements
  - ID attributes
  - Incorporation by reference
  - Shared segments
- XML Stylesheet Language

XML and Architectures

- Provides an easy medium of exchange
- Accommodates change
- Resolves potential ambiguities
- Facilitates message transformation
- Provides some security capabilities
- Provides domain specifications for interchange

Summary

- Web-based architectures, XML, and SOAP have revolutionized integration
- Loose coupling and dynamic services have made integration much more extendable
- Information sharing is the largest gain from capitalizing on these latest technologies