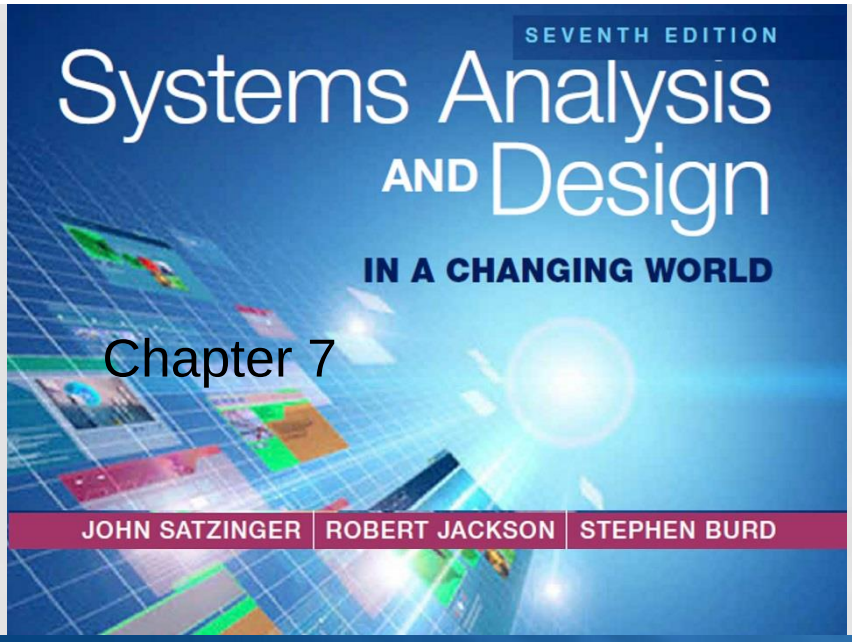


Defining the System Architecture

Chapter 7

Systems Analysis and Design
in a Changing World 7th Ed

Satzinger, Jackson & Burd

Chapter 7 Outline

- Anatomy of a Modern Information System
- Architectural Concepts
- Interoperability
- Architectural Diagrams
- Describing the Environment
- Designing Application Components

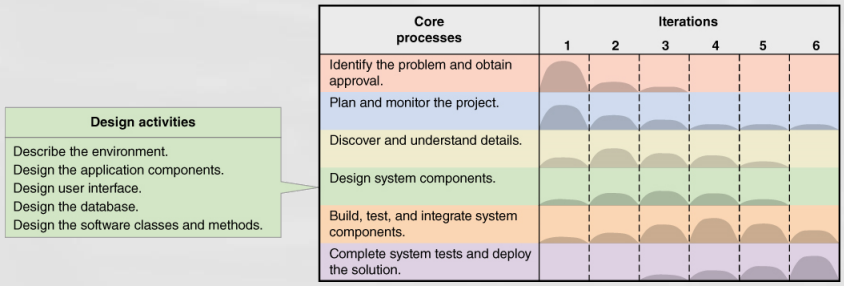
Learning Objectives

- Explain architectural concepts that influence system design, including ubiquitous computing and software, components, protocols, interoperability, and distributed architectures
- Describe and draw location, network, and deployment diagrams
- Describe a system's environment by drawing architectural diagrams and answering key questions
- Design larger application components based on use cases and other analysis models

Overview

- An important part of new system development is choosing appropriate technologies
- Explain and provide a summary of technology and architectural concepts
- describe the details for the activity – *Describe the Environment*
- Describe the details for the activity – *Design the application components*

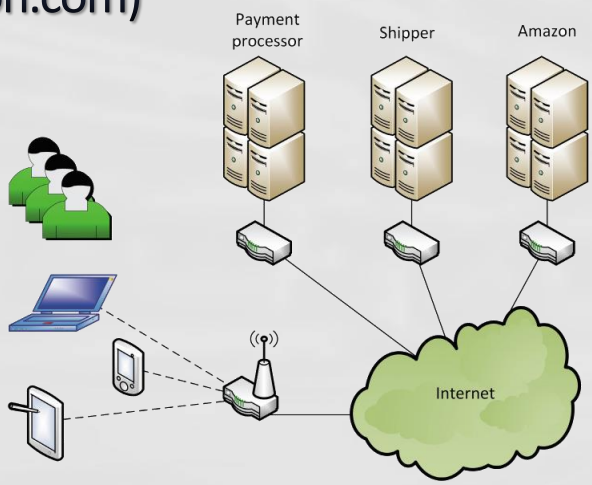
Activities of "Design System Components"



Anatomy of a Modern System – Computing Devices

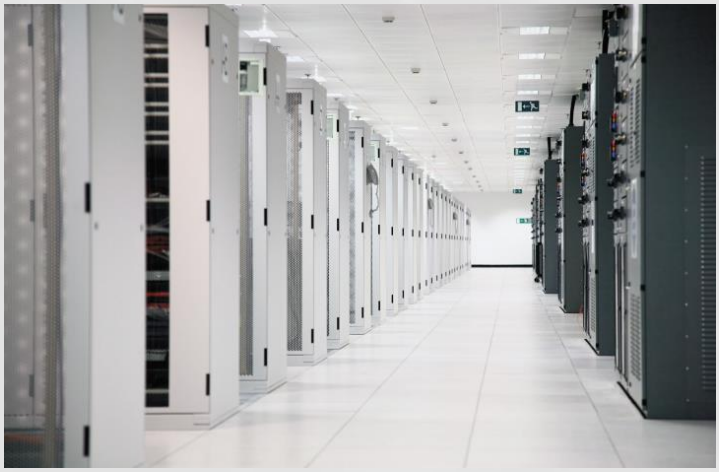
- Server – manages shared resources and enables users and other computers access to these resources
- Personal computing devices or clients
  - Desktops, laptops, tablets, smartphones…

Simplified architecture for application (Amazon.com)



Server Farm –

- Very large databases and very high use

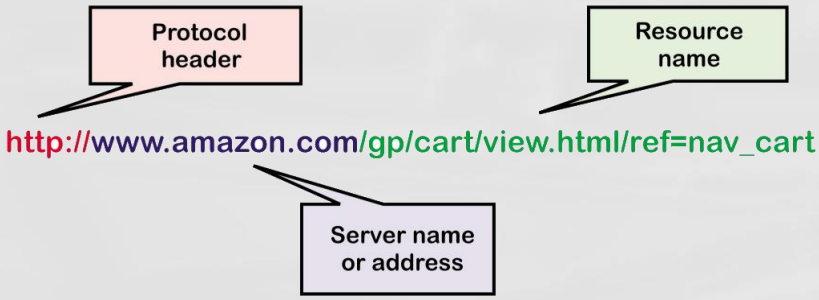


Anatomy - Networks

- Computer network – hardware, software, transmission media
- Internet backbone –
  - High-capacity with high-bandwidth trunk lines and large high-speed computers
  - Owned by governments and telecom companies
- Local area network (LAN) –
  - Small network for a single site
- World Wide Web (WWW)
  - All the interconnected resources accessed through the Internet

Anatomy - Networks

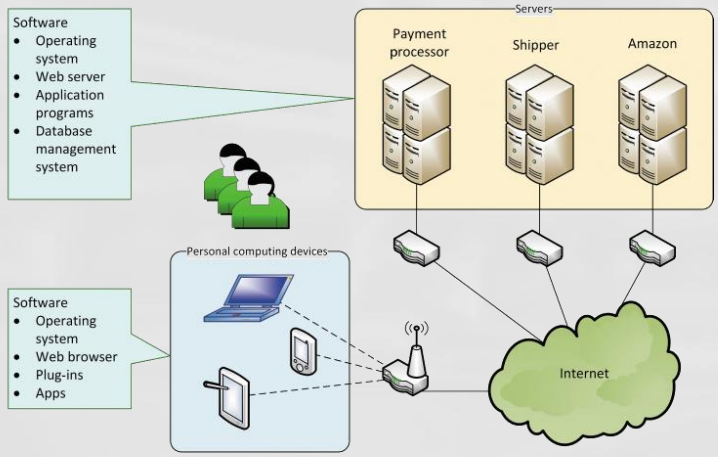
- Uniform Resource Locator (URL)
  - The identifier for the Web to locate a particular resource
- Hyperlink –
  - The URL of a resource embedded within another resource



Anatomy - Software

- Application software – programs that perform work for users
  - Either a custom app or a Web-based application
- App –
  - A custom program usually for a laptop or smartphone
- System Software –
  - Behind the scene software, works as glue to hold everything together

Anatomy – Software



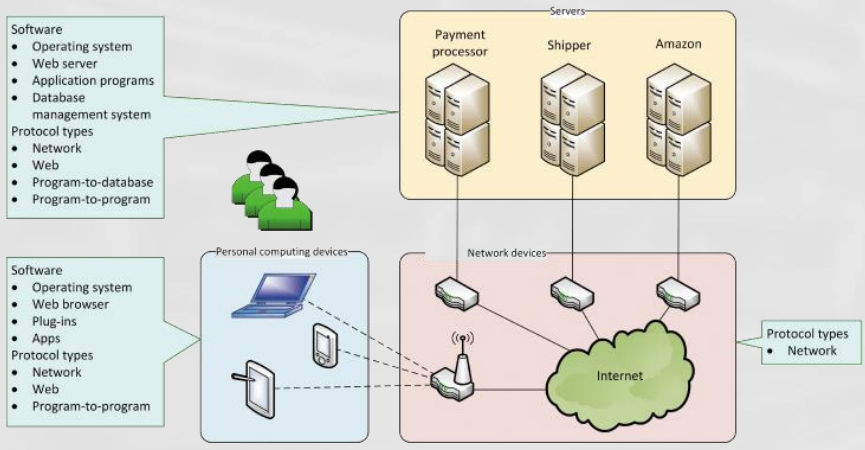
Anatomy – Software

- Web-Based Applications
  - Uses a web browser
  - Accessed through a URL
  - Resides on a Web server
  - Uses standard IP protocols
- Embedded Software
  - Software apps or functions embedded within another app, such as within a browser or O/S
  - Toolbars, Plug-ins, Widgets

Anatomy – Protocols

- Protocol
  - A set of languages and rules to ensure communication and data exchange between hardware and software
- Network protocols
  - Virtual Private Network (VPN)
    - Creates a private network but on the Internet by using secure technologies and encryption

Anatomy – Software and Protocols



Anatomy – Web Protocols

- HTML (hypertext markup language)
  - Protocol for the structure and content of a Web page
- XML (extensible markup language)
  - An extensions of HTML that enables defining semantics of tags
- HTTP (hypertext transfer protocol)
  - Defines format and content for transfer of Web documents
- HTTPS (hypertext transfer protocol secure)
  - Encrypted and secure http transfers

Architectural Concepts

- Technology architecture
  - Computers, network computers and hardware, and system software
- Application architecture
  - The software programs and their configuration

Software as a Service (SaaS)

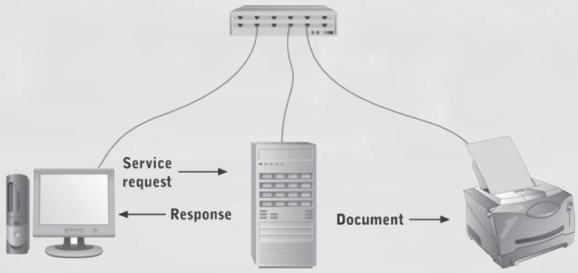
- SaaS
  - No software is installed on the user's device
  - Application services is accessed remotely
  - User data is isolated and stored on common servers

Web Services

- Web service
  - Software function that is executed with Web standards
    - Access via a URL
    - Inputs sent via the URL
    - Executes remotely
    - Data returned within a Web page

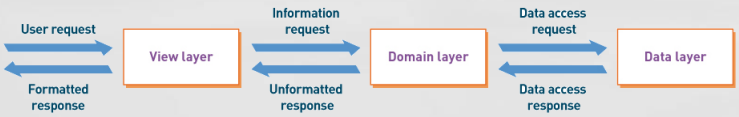
Distributed Architectures

- Client/Server architecture
  - Software design with part of the application on a server and part on the client

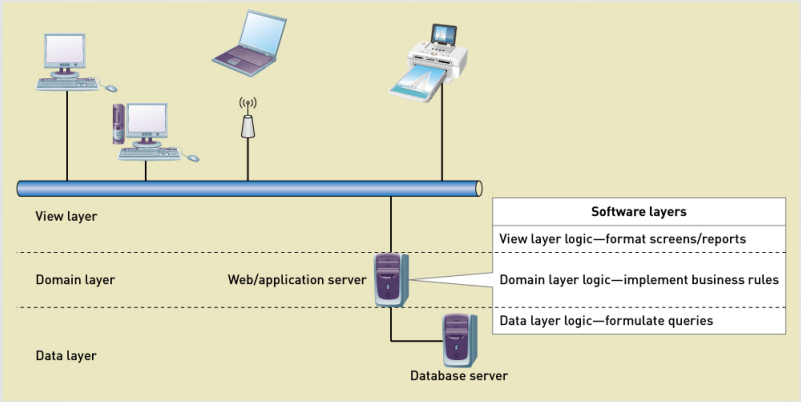


Distributed Architecture

- Three-Layer architecture
  - Client/server architecture with application divided into view layer, logic layer, and data layer
  - View layer – the user interface
  - logic layer – program logic to implement the functions
  - data layer – the functions to access the data



Three Layer Architecture



Interoperability

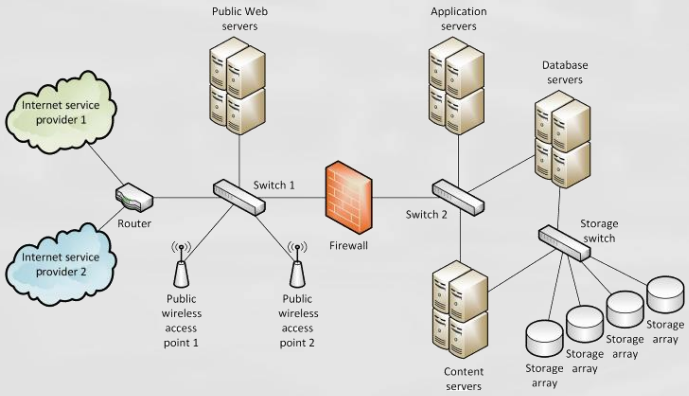
- Interoperability
  - The ability of an application to interact with other software
- Important characteristic in current development projects
  - Understand the environment
  - Reuse software existing components (purchased or in-house)
  - Build components considering interoperability
  - Combine all components into a solution system

Diagrams for System Architectures

- Location Diagrams
  - Identify geographical placement of hardware, software, and users

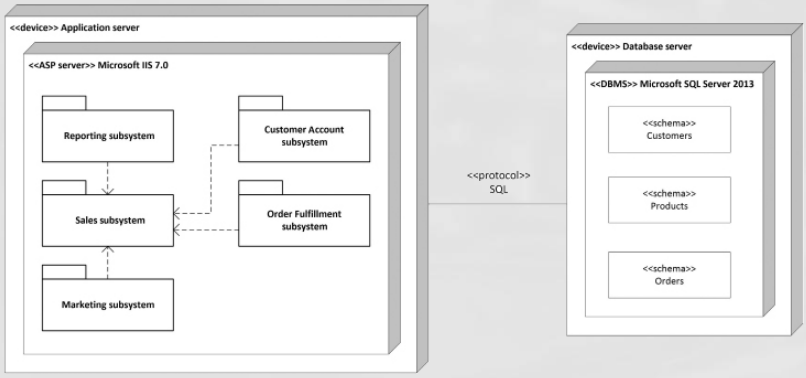
Diagrams for System Architecture

- Network Diagrams
  - How the application software is deployed across the hardware and system software



Diagrams for System Architecture

- Deployment Diagrams
  - How the components of a network are interconnected



Describing the Environment

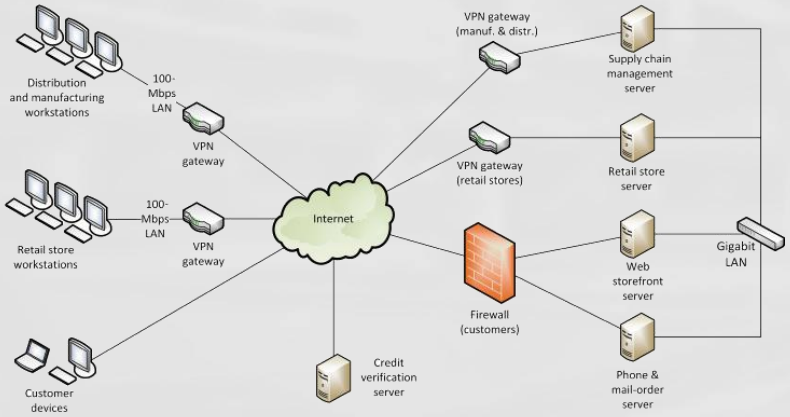
- Key Questions to help describe accurately
  - What are the key features of existing or new environment
    - O/S, system software, networks, tools
  - What are the external systems or DBMSs
    - What kind of interaction
    - What is the data
    - What are the protocols
    - What kind of security

Describing the Environment

- Key Questions to help describe accurately
  - What devices will be required
    - Protocols for devices
    - Security
    - What APIs
  - What user-interface technology will be used
    - Where and who are users, and what skills
    - What hardware and devices
    - What client O/S will be used
    - Security requirements
    - What APIs are needed

RMO Environment - Existing

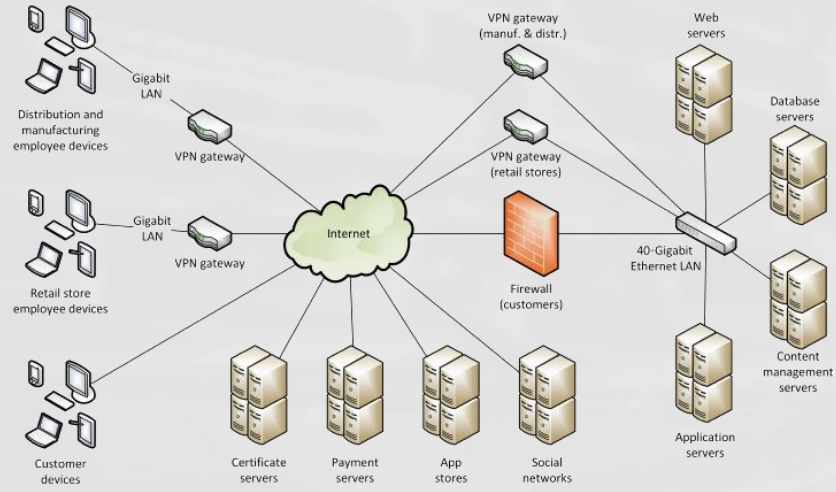
- Current environment prior to new development



RMO Environment - Proposed

- More mobile devices and apps
- Web application software and content
- Social networking applications
- Security issues
- External hosting of portions

RMO Environment - Proposed



Designing Application Components

- Application Component Boundaries
  - Which components perform which functions
  - How to group functions to build components
    - Actors – what functions to particular actors use
    - Shared data – what functions use the same data
    - Events – what functions occur in common business events

RMO CSMS Application Architecture

- Grouping by customer actor – part 1

Use case	User/actor	Domain class(es)	Event(s)	Group
Create phone sale	Phone sales representative	ProductItem, InventoryItem, SaleItem, Sale, SaleTrans	Customer request while shopping by phone	A
Create store sale	Store sales representative	ProductItem, InventoryItem, SaleItem, Sale, SaleTrans	Customer request while shopping in store	B
Create/update customer account	Customer, phone or store sales representative	Customer, Account, Address	Customer request or sale to a new customer	C
Look up order status	Shipping, customer, management, phone or store sales representative	ProductItem, InventoryItem, SaleItem, Sale, SaleTrans, Shipment, ReturnItem	Customer, representative, shipping, or management request	
Track shipment	Shipping, customer, management, phone or store sales representative	Shipment, Shipper, SaleItem	Customer, representative, shipping, or management request	
Create item return	Customer, phone or store sales representative	SaleItem, ReturnItem	Customer requests return	
Search for item	Customer, phone or store sales representative	ProductItem	Customer request while shopping online, by phone, or in store	
View product comments and ratings	Customer, phone or store sales representative	ProductItem, ProductComment	Customer request while shopping online, by phone, or in store	
View accessory combinations	Customer, phone or store sales representative	ProductItem, AcessoryPackage	Customer request while shopping online, by phone, or in store	

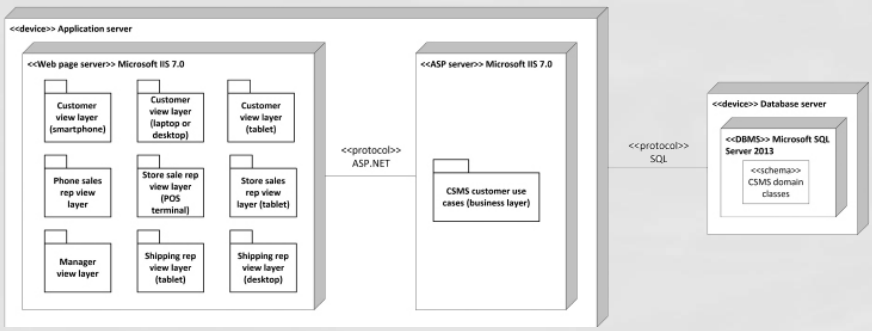
RMO CSMS Application Architecture

- Grouping by customer actor – part 2

Use case	User/actor	Domain class(es)	Event(s)	Group
Fill shopping cart	Customer	ProductItem, InventoryItem, CartItem, OnlineCart	Customer request, usually after sale completed	
Empty shopping cart	Customer	ProductItem, InventoryItem, CartItem, OnlineCart	Customer request while shopping online	
Check out shopping cart	Customer	ProductItem, InventoryItem, CartItem, OnlineCart, SaleItem, Sale, SaleTrans	Customer request while shopping online	
Fill reserve cart	Customer	ProductItem, InventoryItem, CartItem, OnlineCart	Customer request while shopping online	D
Empty reserve cart	Customer	ProductItem, InventoryItem, CartItem, OnlineCart	Customer request while shopping online	
Convert reserve cart	Customer	ProductItem, InventoryItem, CartItem, OnlineCart	Customer request while shopping online	
Rate and comment on product	Customer	Customer, ProductComment, ProductItem	Customer request, usually after sale completed	
Provide suggestion	Customer	Customer, Suggestion	Customer request while shopping online	
Send message	Customer	Customer, Message	Customer request while shopping online	
Browse messages	Customer	Customer, Message	Customer request while shopping online	
Request friend linkup	Customer	Customer, FriendLink	Customer request while shopping online	E
Reply to linkup request	Customer	Customer, FriendLink	Customer request while shopping online	
Send/receive partner credits	Customer	Customer, CustPartnerCredit, PromoPartner	Customer request while shopping online	
View "mountain bucks"	Customer	Customer, Sale	Customer request while shopping online	
Transfer "mountain bucks"	Customer	Customer, Sale	Customer request while shopping online	

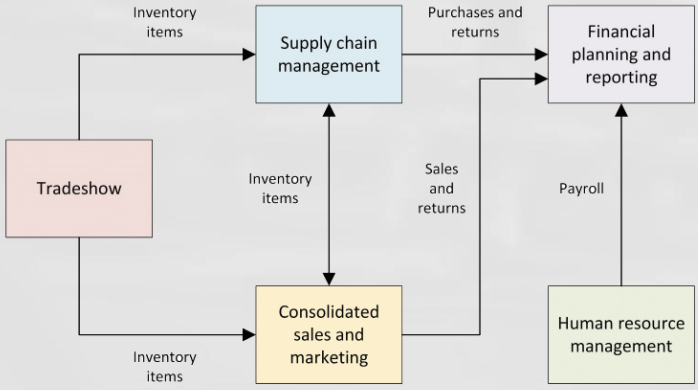
RMO CSMS Deployment Diagram

- Three-layer design with user components grouped by user functions



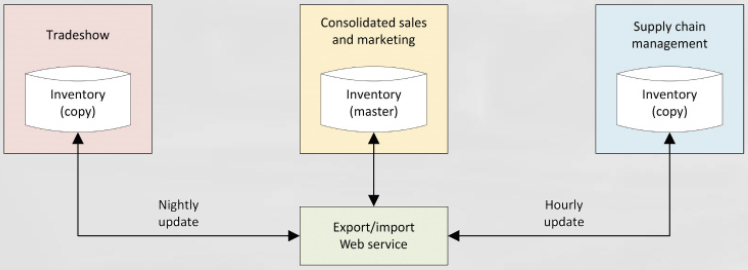
RMO CSMS Component Integration

- Subsystem integration and data flows



RMO CSMS Data Ownership

- Who "owns" the data
  - System of record
    - What system is responsible to maintain the data
    - What system has a copy or can access the data



Summary

- Anatomy of a Modern Information System
  - Consist of computing devices, networks, software, and protocols
  - Deployed as stand-alone software, network based, Web based
- Architectural Concepts
  - SaaS – software as a service
  - Web services
  - Distributed architectures
    - Client/server and three-layer architecture

Summary (cont)

- Interoperability
  - Getting all the components to work together
- Architectural diagrams
  - Location diagrams
  - Network diagrams
  - Deployment diagrams

Summary (cont)

- Describing the Environment
  - External systems
  - Technology architecture
  - Key questions requiring answers
- Designing Application Components
  - Application component boundaries
  - Grouping functions into components
  - System of record – who owns the data