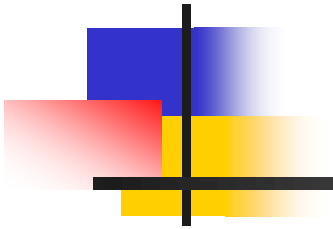
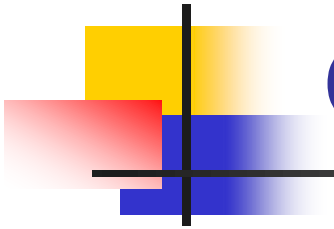


Introduction to Systems Design

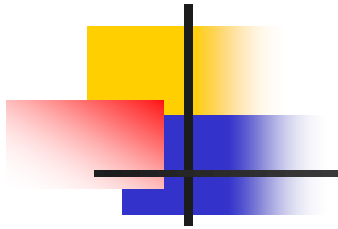


Outline

- What Is Systems Design?
- Design Activities
- System Controls and Security

Learning Objectives

- Describe systems design and contrast it with systems analysis
- List the documents and models used as inputs to or output from systems design
- Explain each major design activity
- Describe security methods and controls



Overview

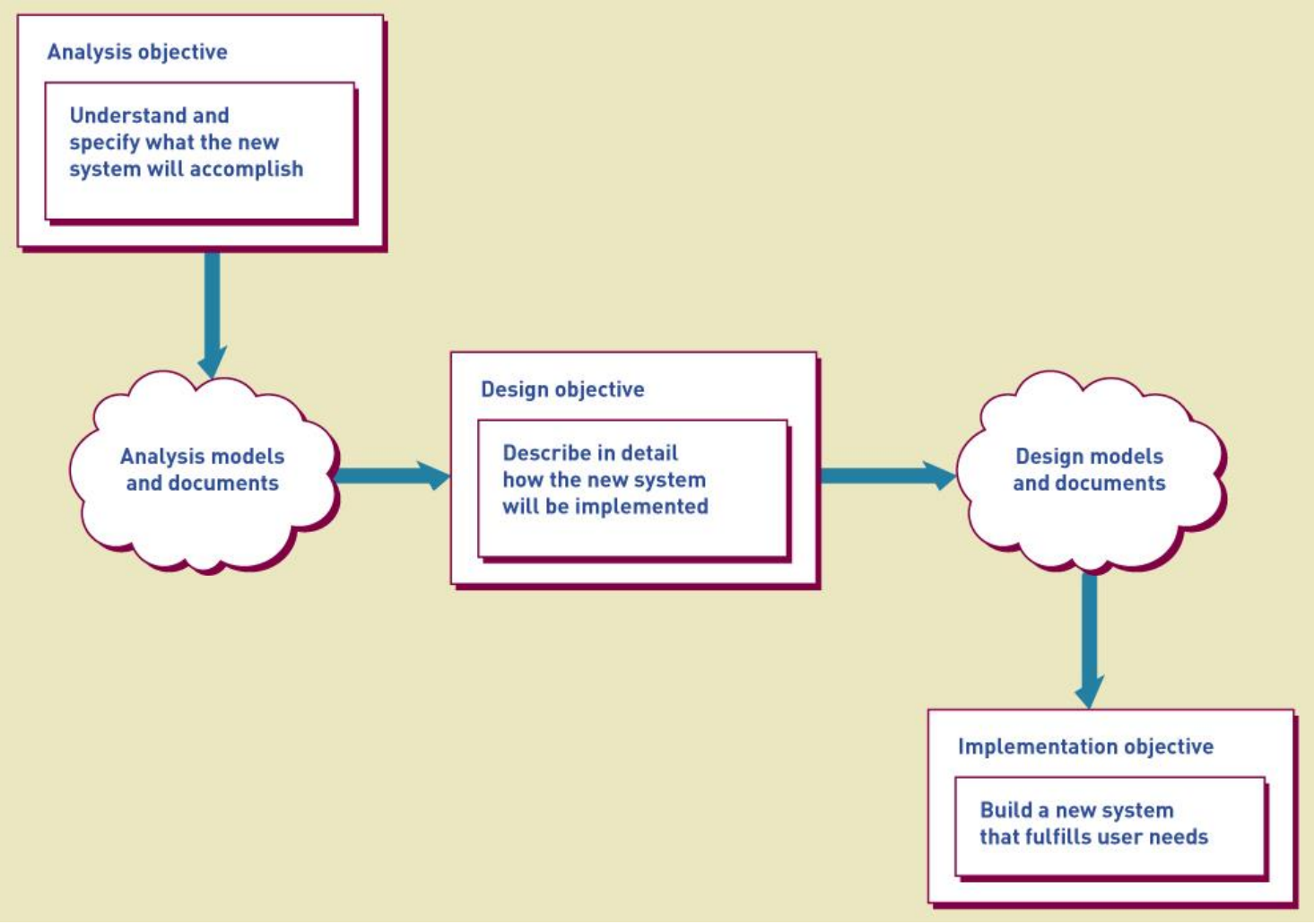
- Analysis says “what” is required and design tells us “how” the system will be configured and constructed

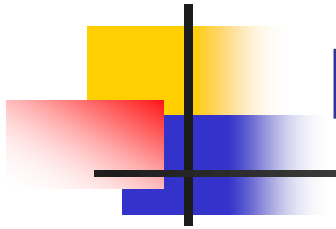
- Design bridges the gap between requirements to actual implementation

What is Systems Design

- Analysis provides the starting point for design
- Design provides the starting point for implementation
- Analysis and design results are documented to coordinate the work
- Objective of design is to define, organize, and structure the components of the final solution to serve as a blue print for construction

Analysis to Design to Implementation

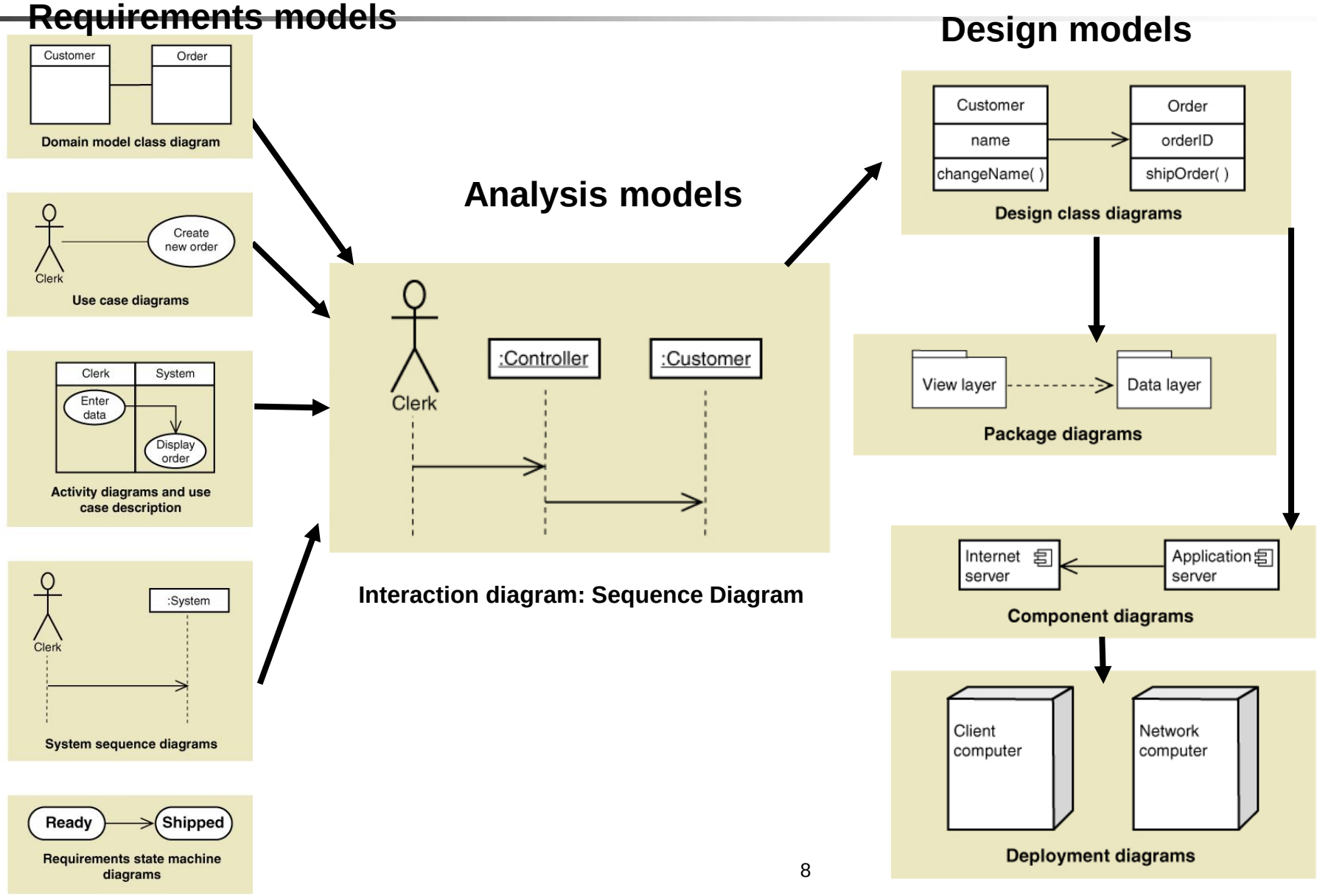




Design Models

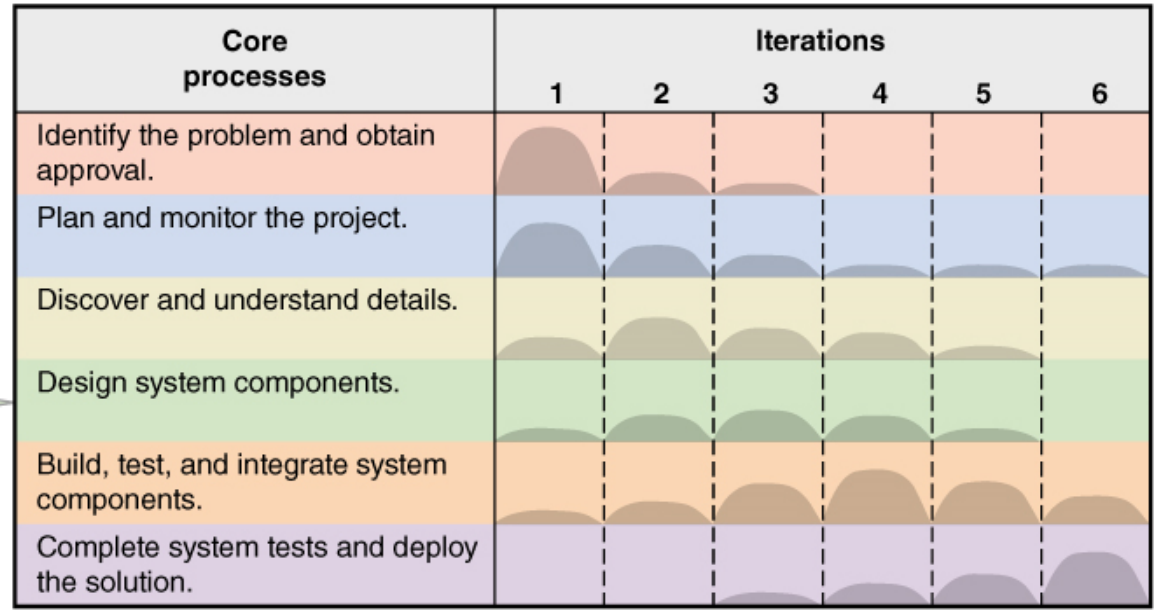
- Design is a model building activity
- The formality of the project will dictate the type, complexity, and depth of models
- Agile/iteration projects typically build fewer models, but models are still created
- Jumping to programming without design often causes less than optimum solutions and may require rework

Analysis: From requirements Models to Design Models



Design Activities and Iterations

Design activities
Describe the environment. Design the application components. Design user interface. Design the database. Design the software classes and methods.



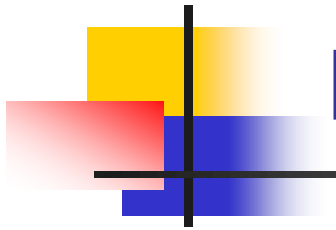
Design Activities

- Design activities correspond to components of the new system

- Activities of Systems Design
  - Describe the environment
  - Design the application components
  - Design the User Interface
  - Design the database
  - Design the software classes and methods

Key Design Questions for each Activity

Design activity	Key question
Describe the environment	How will this system interact with other systems and with the organization's existing technologies?
Design the application components	What are the key parts of the information system and how will they interact when the system is deployed?
Design the user interface	How will users interact with the information system?
Design the database	How will data be captured, structured, and stored for later use by the information system?
Design the software classes and methods	What internal structure for each application component will ensure efficient construction, rapid deployment, and reliable operation?

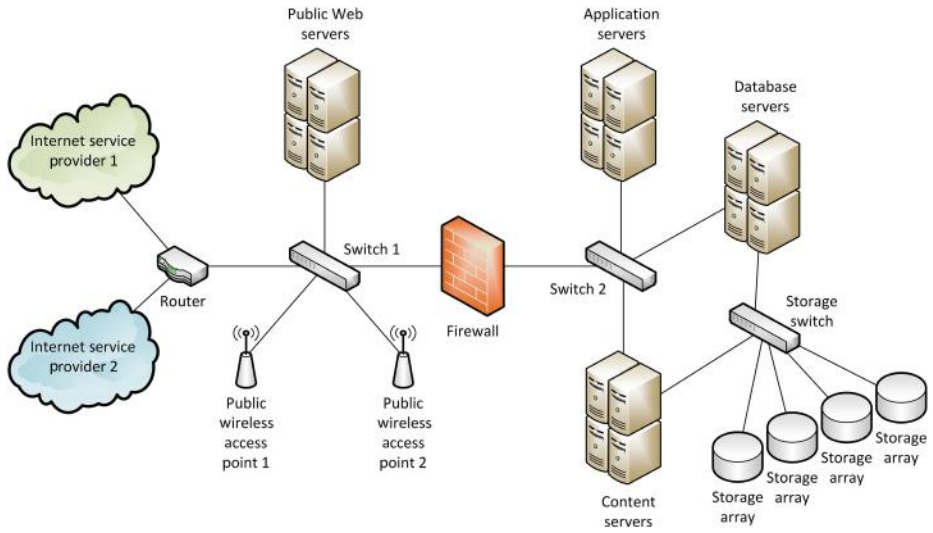


Describe the Environment

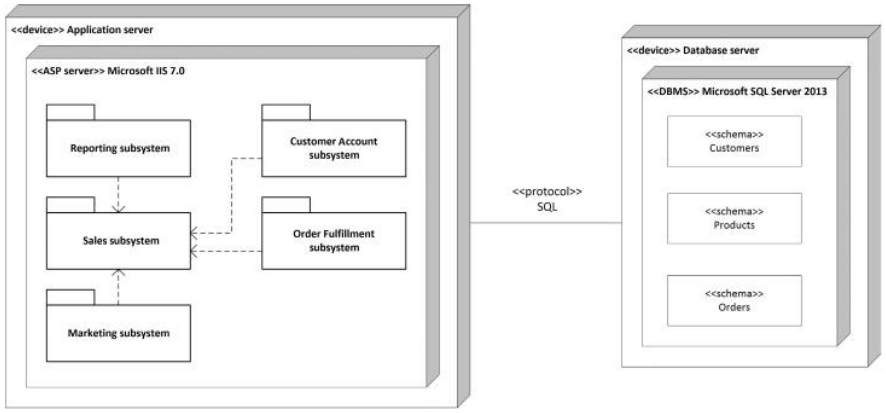
- Two key elements in the environment
  - Communications with External Systems
    - Message formats
    - Web and networks
    - Communication protocols
    - Security methods
    - Error detection and recovery
  - Conforming to an existing Technology Architecture
    - Discover and describe existing architecture

Models used to describe the environment

Network Diagram



Deployment Diagram



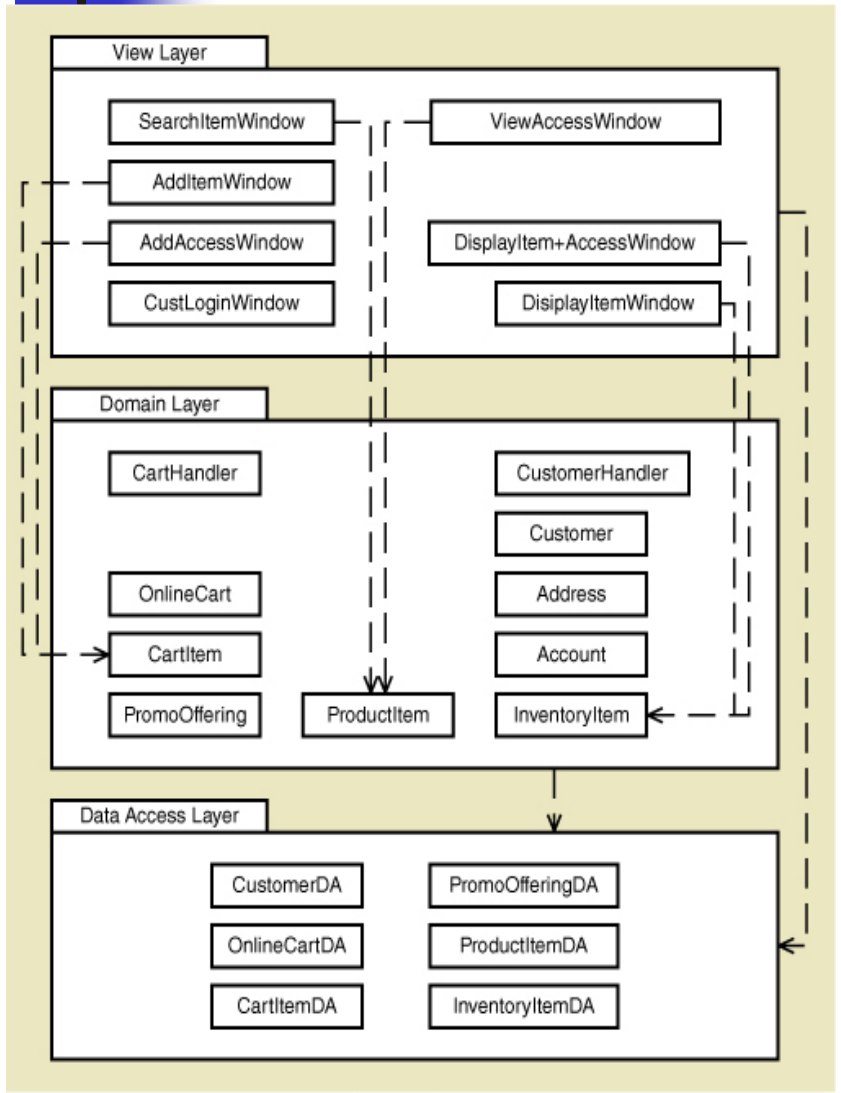
Design the Application Components

- Application component is a well defined unit of software that performs some function(s)
- Issues involve how to package components including
  - Scope and size – what are the functions, boundaries, interfaces?
  - Programming language – what are the accepted languages?
  - Build or buy – is an acceptable version available to purchase?

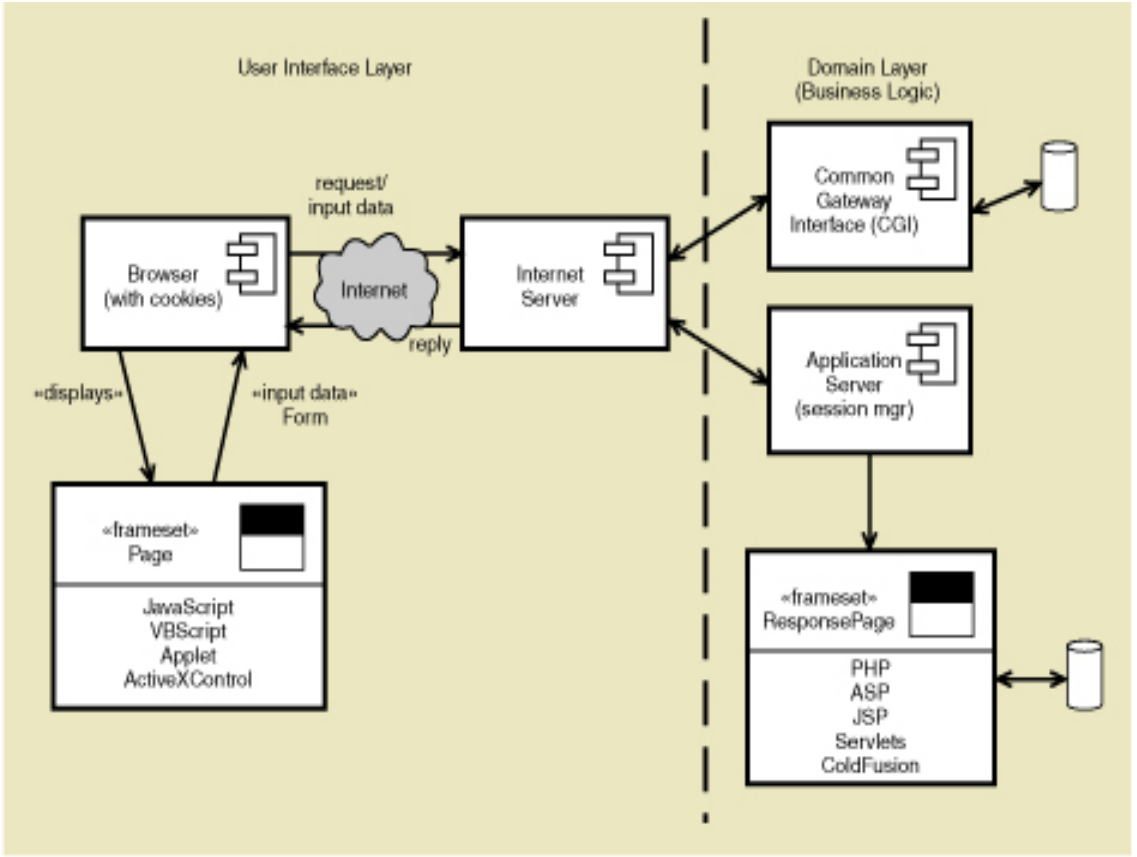
Models developed during Component Design (Architectural Design)

- Package diagrams.
- Component diagrams.
- Deployment diagrams.

Models developed during Component Design (Architectural Design)



Package diagram



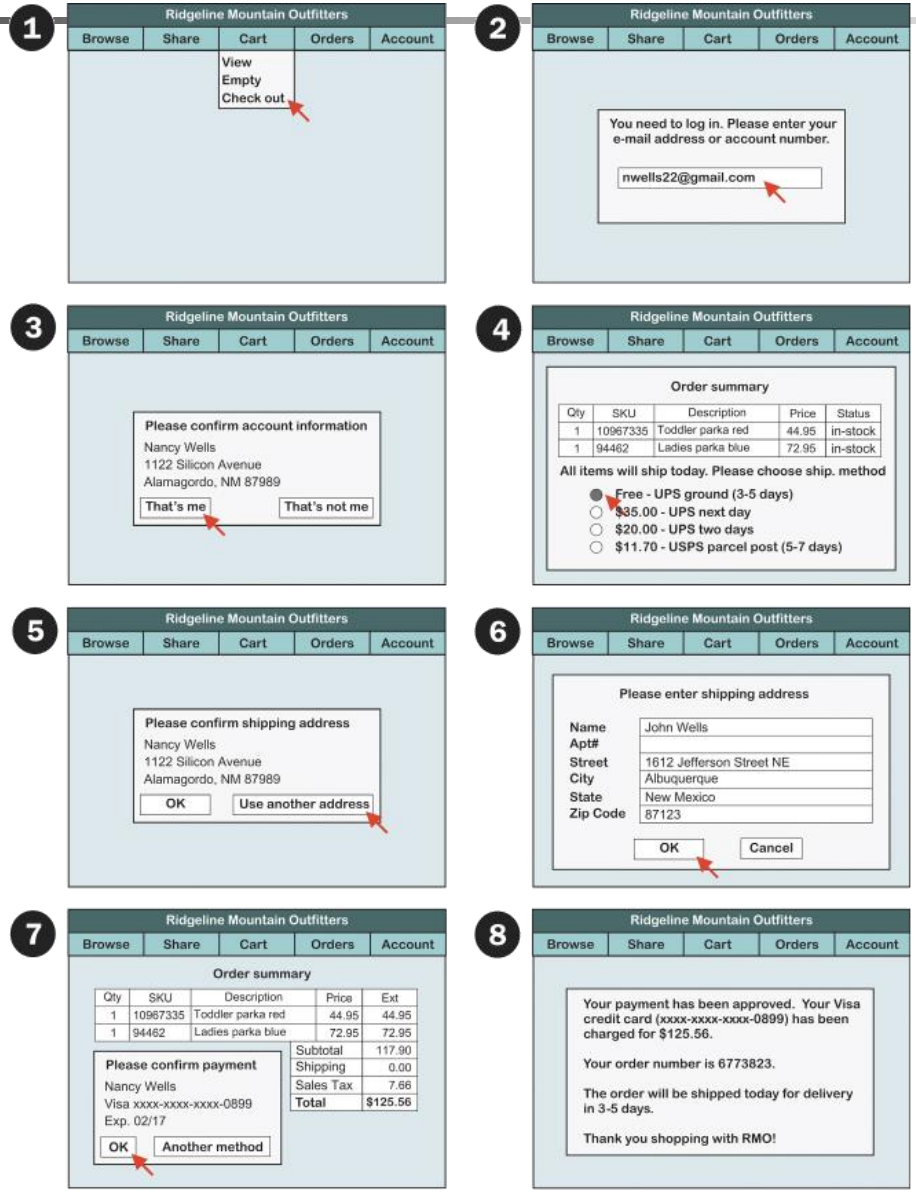
Component diagram

Design the User Interface

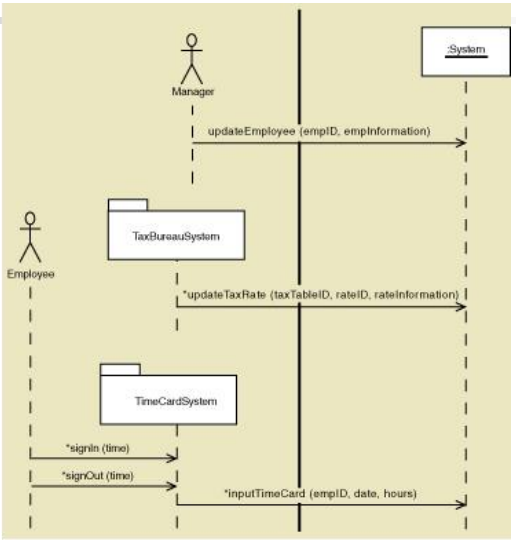
- To the user, the User Interface **is** the system.
- The user interface has large impact of user productivity
- Includes both Analysis and Design tasks
  - Requires heavy user involvement
- Current needs require multiple user interfaces
  - Many different devices and environments

Typical models for user interface design

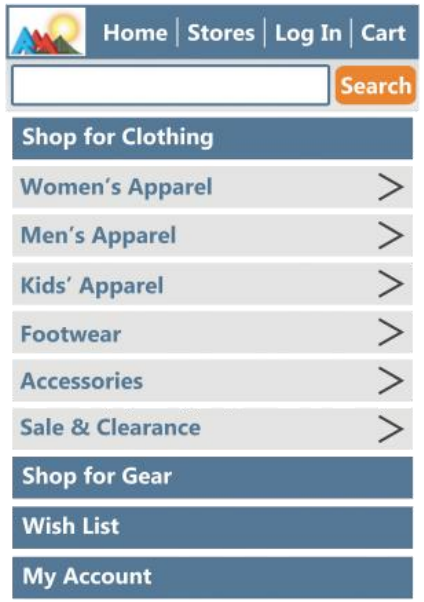
- Story boards
- Screen and report mockups.



Storyboard



System sequence diagram



Small screen menu prototype

RIDGELINE MOUNTAIN OUTFITTERS

Ridgeline Mountain Outfitters—Shopping Cart Order

Customer Name: Fred Westing
Customer Number: 6747222

Order Number: 4673064
Today's Date: May 18, 2015

Shipping Address:

936 N Swivel Street
Hillville, Ohio 59222

Billing Address:

936 N Swivel Street
Hillville, Ohio 59222

Qty	Product ID	Description	Size	Color	Price	Extended Price
1	458238WL	Jordan Men's Jumpman Team J	12	White/ Light Blue	$119.99	$119.99
1	347827OP	Woolrich Men's Backpacker Shirt	XL	Oatmeal Plaid	$41.99	$41.99
2	8759425SH	Nike D.R.I. – Fit Shirt	M	Black	$30.00	$60.00
1	5858642OR	Puma Hiking Shorts	L	Tan	$15.00	$15.00
					Subtotal	$236.98
					Shipping	$8.50
					Tax	$11.25
					Total	$256.73

Shipping Information:

Shipping Method:	Normal 7–10 day
Shipping Company:	UPS
Tracking Number:	To be sent via email
Email Address:	FredW253@aol.com

Payment Information:

American Express ☐ **MasterCard** ☐ **VISA** ☒ **Discover** ☐

Account Number

X X X X – X X X X – X X X X – 5 7 8 4

MO YR

Expiration Date 05 / 17

Thank you for your order. It is a pleasure to serve you.
Check back next week for new weekly specials!!

Text vs. Graphical Reports

Monthly Sales Summary

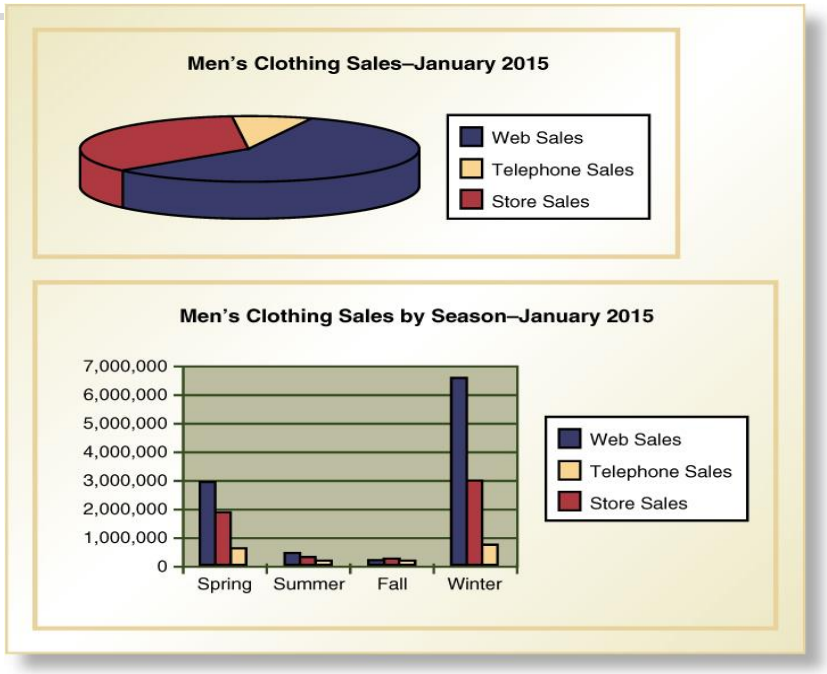
Year **2015** *Month* **January**

Category	Season Code	Web Sales	Telephone Sales	Mail Sales	Total Sales
Footwear	All	$ 289,323	$ 1,347,878	$ 540,883	$ 2,178,084
Men's Clothing	Spring	$ 1,768,454	$ 2,879,243	$ 437,874	$ 4,691,484
	Summer	213,938	387,121	123,590	724,649
	Fall	142,823	129,873	112,234	384,930
	Winter	2,980,489	6,453,896	675,290	10,109,675
	All	1,839,729	4,897,235	349,234	7,086,198
Totals			1,747,368	$ 1,698,222	$ 23,391,023
Women's Clothing	Spring				965,610
	Summer				
	Fall				
	Winter				
	All				
Totals					

Monthly Sales Detail

Year **2015** *Month* **January** *Category* **Men's Clothing** *Season* **Winter**

Product ID	Product Description	Web Sales	Telephone Sales	Mail Sales	Total Sales
RMO12987	Winter Parka	$ 1,490,245	$ 3,226,948	$ 337,640	$ 5,054,833
RMO13788	Fur-Lined Gloves	149,022	322,695	33,765	505,482
RMO23788	Wool Sweater	596,097	1,290,775	135,058	2,021,930
RMO12980	Long Underwear	298,050	645,339	68,556	1,003,005
RMO32998	Fleece-Lined Jacket	447,075	1,258,079	100,271	1,805,425
Total		$ 2,980,489	$ 6,743,836	$ 675,290	$ 10,394,615



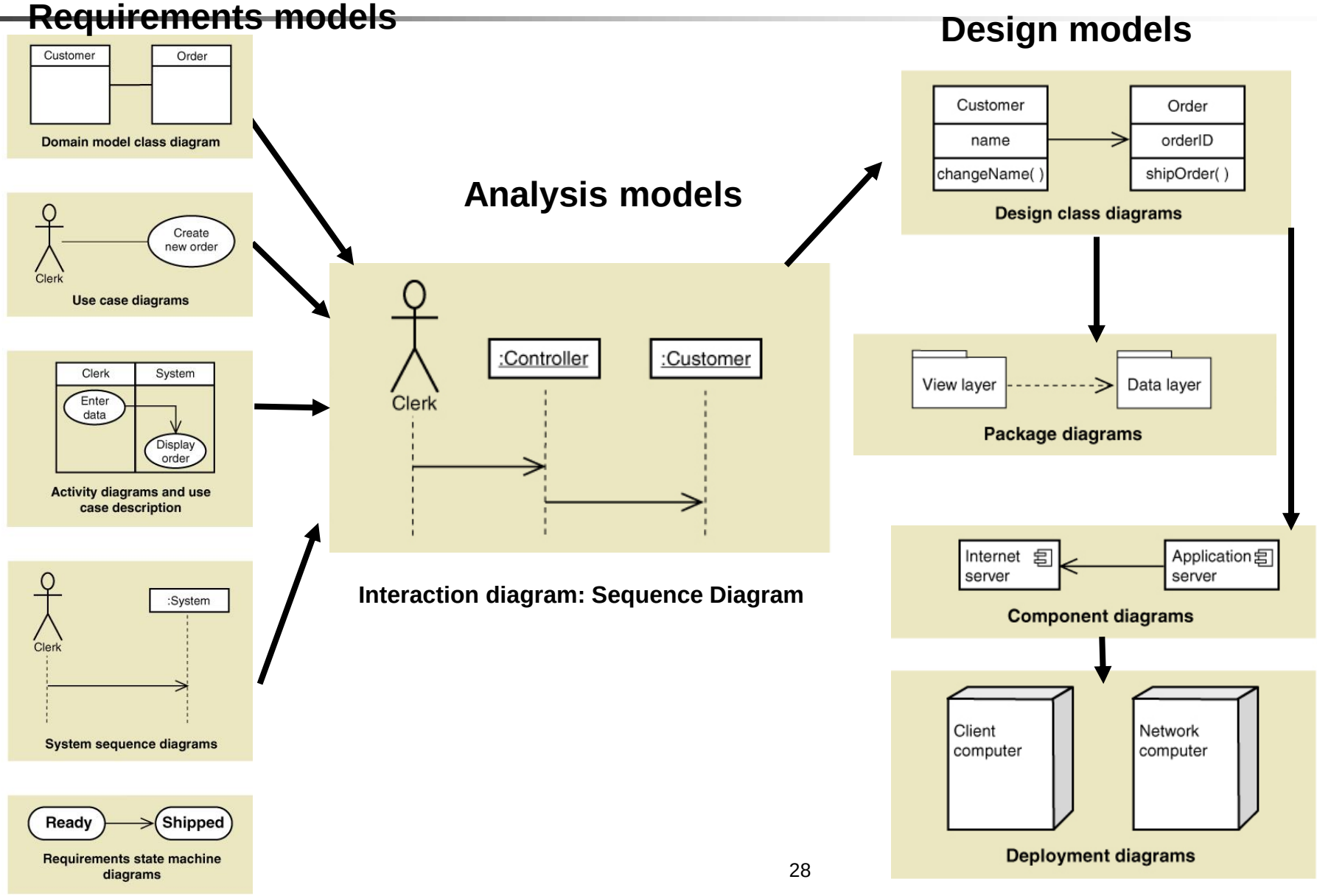
Design the Database

- By definition, an Information System requires data – usually in a database
- Current technology frequently use Relational Database Management Systems (RDBMS)
- Requires converting the data model to a relational database
- Requires addressing of many other technical issues
  - Throughput and response time
  - Security

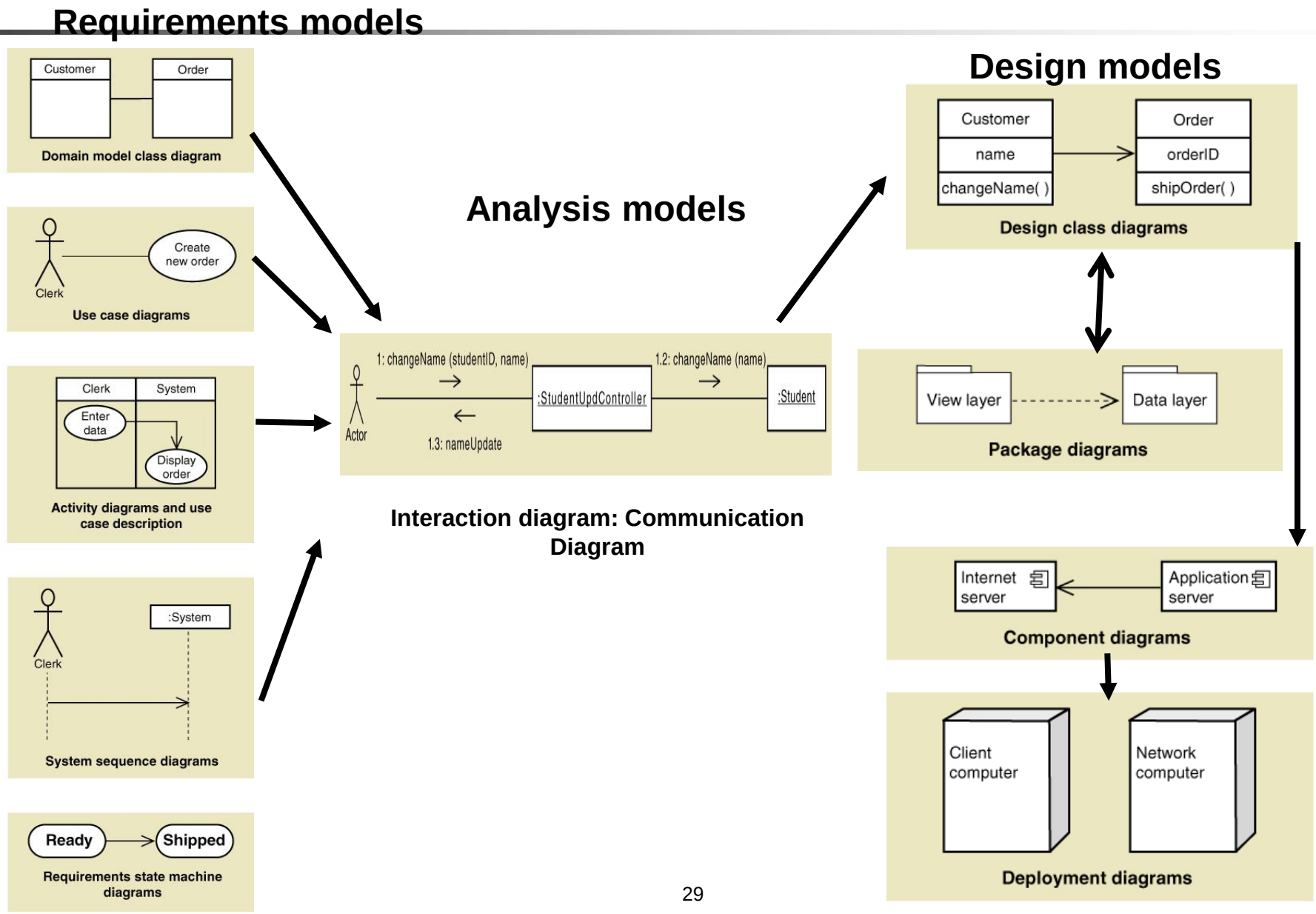
Design Software Classes and Methods

- Detailed Design or Object design
- A model building activity
  - Design Class Diagram
  - Sequence Diagrams / Communication Diagrams
  - State-Machine Diagrams

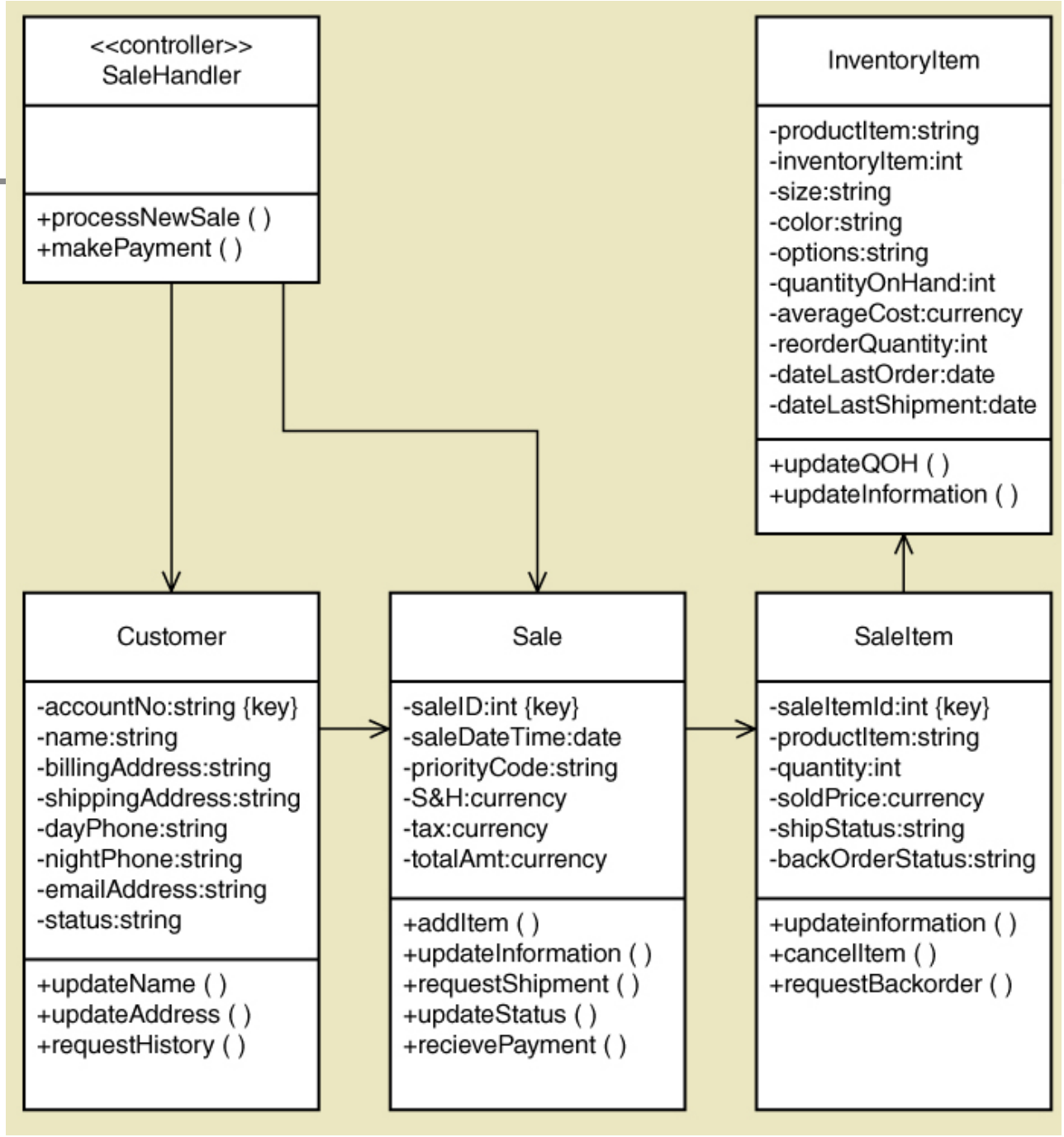
Analysis: From requirements Models to Design Models



Analysis: From requirements Models to Design Models

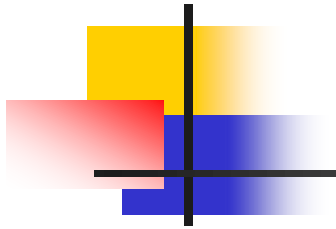


Typical Design Class Diagram with attributes and methods



System Controls and Security

- Integrity Controls
  - Controls that maintain integrity of inputs, outputs and data and programs
- Security Controls
  - Controls that protect the assets from threats, internal and external



- **Compare and contrast integrity controls and security controls. Why isn't there a separate activity to design them?**
- Integrity controls have to do with the maintaining the integrity of the data as it is entered into the system, processed and stored within the system and output from the system. Security controls are more associated with the entire environment from primarily external threats that are malicious. Designing integrity and security controls must be part of every other design activity and hence is not considered a stand-alone design activity.

Designing Integrity Controls

- Integrated into application programs and DBMS
- Objectives of Integrity Controls
  - Ensure that only appropriate and correct business transactions are accepted
  - Ensure that transactions are recorded and processed correctly
  - To protect and safeguard assets such as the database

Input Controls

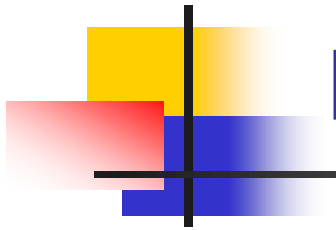
- Prevent invalid or erroneous data from entering the system
- Value Limit Controls
  - Check the range of inputs for reasonableness
- Completeness Controls
  - Ensure all the data has been entered
- Data Validation Controls
  - Ensure that specific data values are correct
- Field Combination Controls
  - Ensure data is correct based on relationships between fields

Output Controls

- To ensure that output arrives at proper destination (for authorized eyes) and is accurate, current, and complete
- Examples
  - Physical access to printers and display devices
  - Discarded data – protect from "dumpster diving"
  - Labels on printed and electronic output to correctly identify source of data

Redundancy, Backup and Recovery

- Protect data and systems from catastrophes
  - Databases
  - Hardware
  - Software applications
  - Networks
- On-site versus off-site copies



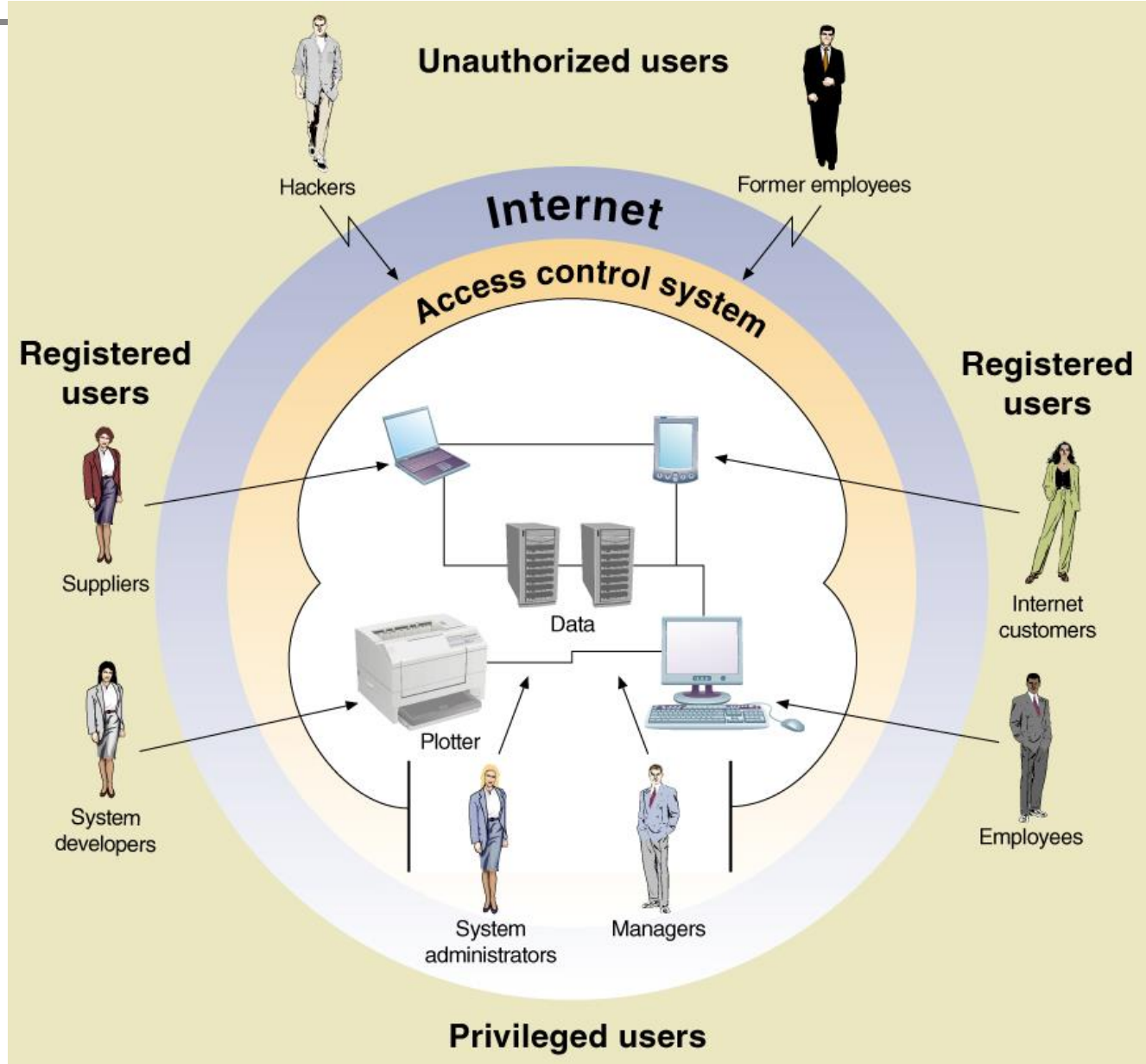
Designing Security Controls

- Protect all assets against external threats
- Main objectives
  - Protect and maintain a stable, functioning operating environment 24/7 (equipment, operating systems, DBMSs)
  - Protect information and transactions during transmission across networks and Internet

Designing Security Controls

- Access Controls – Limit a person's ability to access servers, files, data, applications
  - Authentication – to identify users
    - Multifactor Authentication
  - Access control list – list of valid users
  - Authorization – authenticated user's list of permission level for each resource
- Registered Users – those with authorization
- Unauthorized Users – anyone not registered
- Privileged Users – those that maintain lists and systems

Types of users



Summary

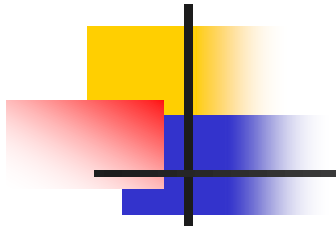
- The concept of Systems Design
  - Analysis is fact finding and modeling
  - Design is modeling to specify how system will be implemented
  - Design is bridge between analysis an implementation

Summary (continued)

- Activities of Systems Design
  - Describe the environment
  - Design the application components
  - Design the User Interface
  - Design the database
  - Design the software classes and methods

Summary (continued)

- System Controls and Security
  - Integrity Controls
    - Input controls
    - Output controls
    - Backup and recovery
  - Security Controls
    - Access controls
    - Data encryption
    - Digital signatures and certificates
    - Secure transactions



- Activities of Systems Design
  - Describe the environment
  - Design the application components
  - Design the User Interface
  - Design the database
  - Design the software classes and methods

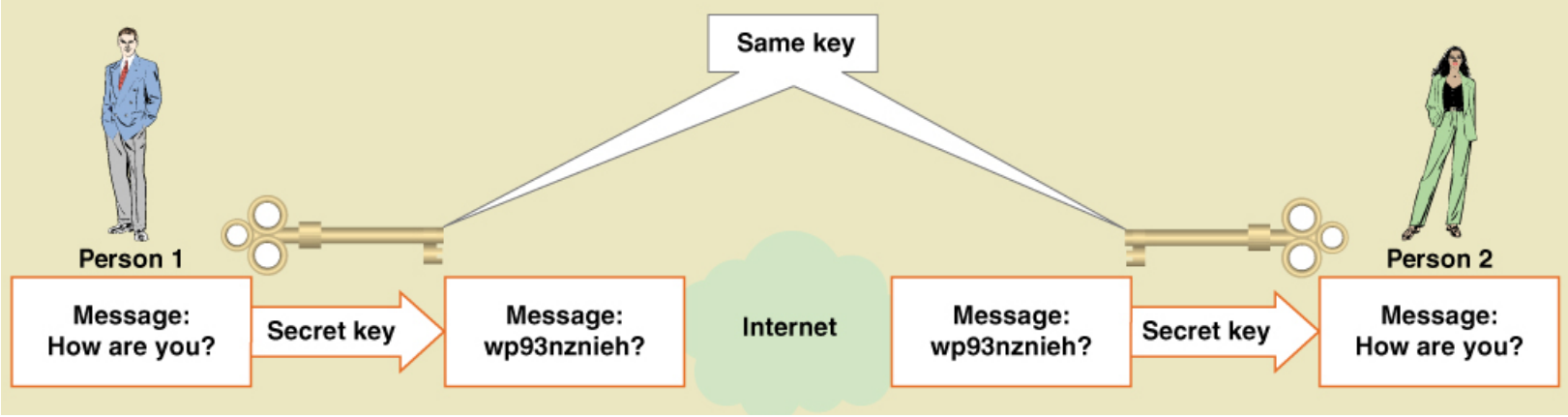
Remaining slides are for self reading and not included in the tests

Data Encryption

- Method to secure data – stored or in transmission
- Encryption – alter data so it is unrecognizable
- Decryption – converted encrypted data back to readable format
- Encryption Algorithm – mathematical transformation of the data
- Encryption Key – a long data string that allows the same algorithm to produce unique encryptions

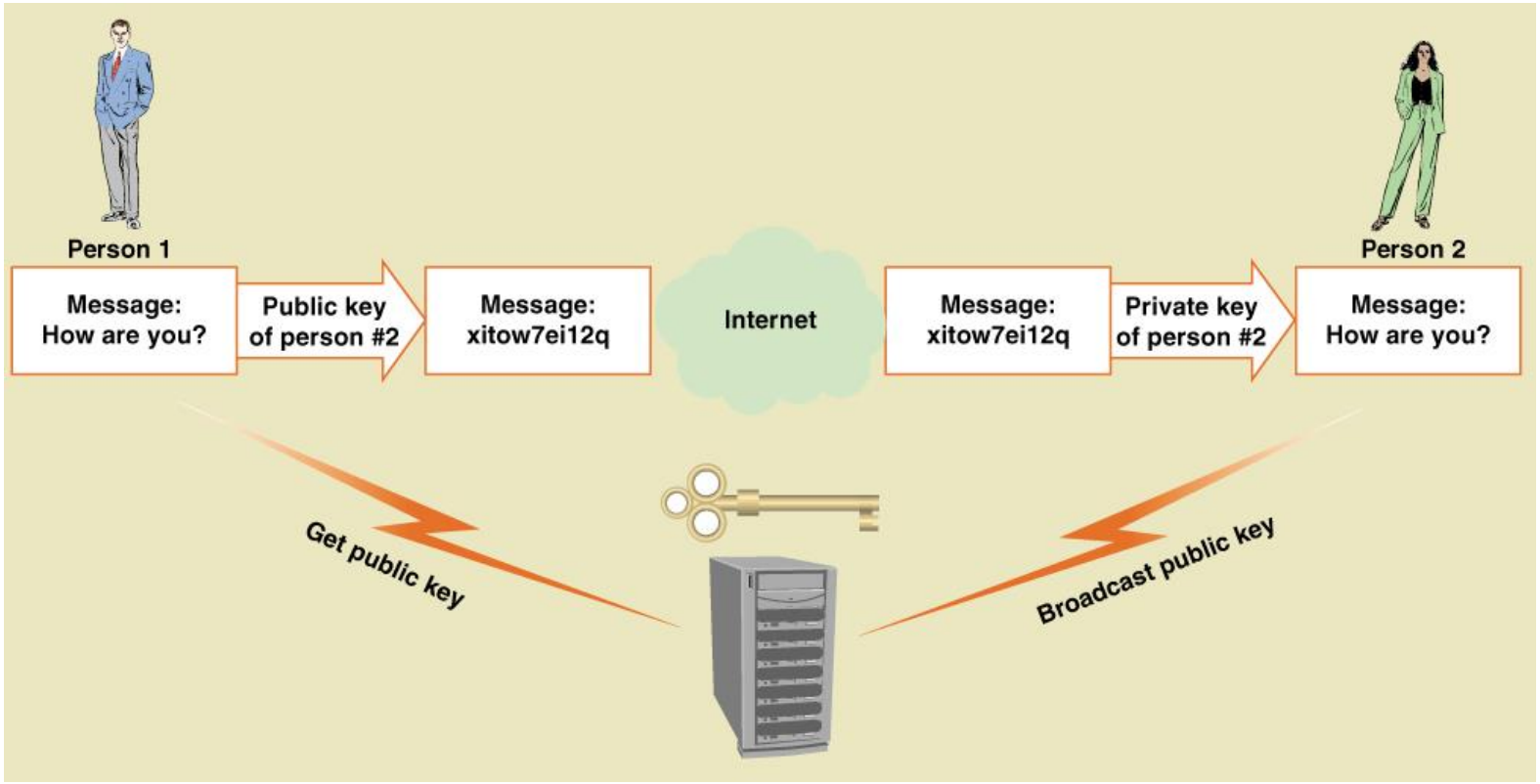
Symmetric Key Encryption

- Encryption method that uses the same key to encrypt and decrypt



Asymmetric Key Encryption

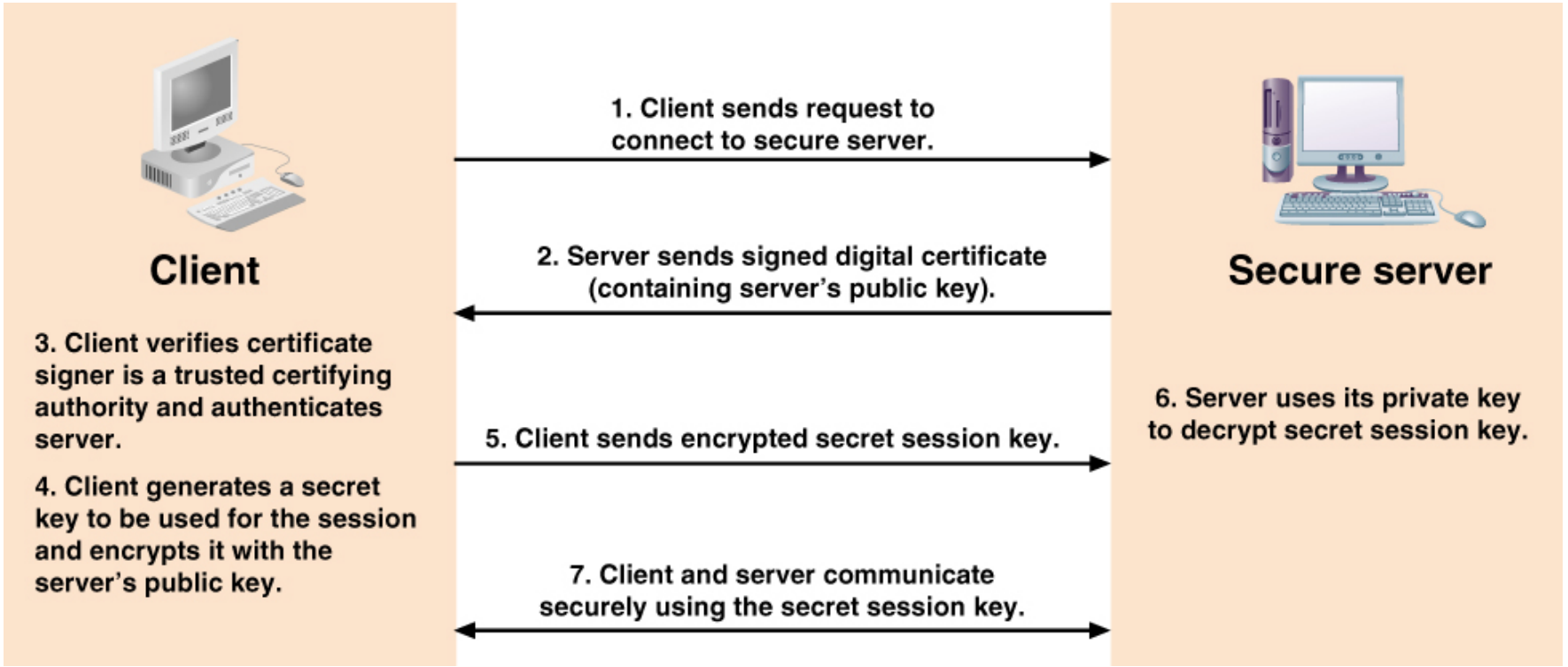
- Encryption method that uses different keys to encrypt and decrypt
  - AKA Public Key Encryption



Digital Signatures and Certificates

- Digital Signature – technique where a document is encrypted using a private key
  - Note – implements previous slide, but in reverse
    - Document is encrypted with private key, but then can only be decrypted with correct public key
- Digital Certificate – An organizations name and public that is encrypted and certified by an authorized third party
- Certifying Authority – the authorized third party
  - Widely known and accepted – built into Web browsers

How a Digital Certificate is used



Secure Transactions

- Secure Sockets Layer (SSL) – standard set of protocols for authentication and authorization
- Transport Layer Security (TLS) – an Internet standard equivalent to SSL
- IP Security (IPSec) – Internet security protocol at a low-level transmission
- Hypertext Transfer Protocol Secure (HTTPS) – Internet standard to transmit Web pages