Chapter 11: Physical Architecture Layer Design

PowerPoint Presentation for Dennis, Wixom, & Tegarden *Systems Analysis and Design with UML, 4th Edition*

Objectives

- Understand the different physical architecture components.
- Understand server-based, client-based, and client–server physical architectures.
- Be familiar with cloud computing and Green IT.
- Be able to create a network model using a deployment diagram.
- Be familiar with how to create a hardware and software specification.
- Understand how operational, performance, security, cultural, and political requirements affect the design of the physical architecture layer.

Introduction

- Most modern systems span two or more networked computers
- The physical architecture layer design specifies:
  - How the system will be distributed across the computers
  - What hardware and software will be used
- Most systems design is constrained by existing systems and networks
- Physical architecture design is demanding
  - Knowledge of key factors is essential
  - Nonfunctional requirements play a key role

Elements of the Physical Architecture Layer

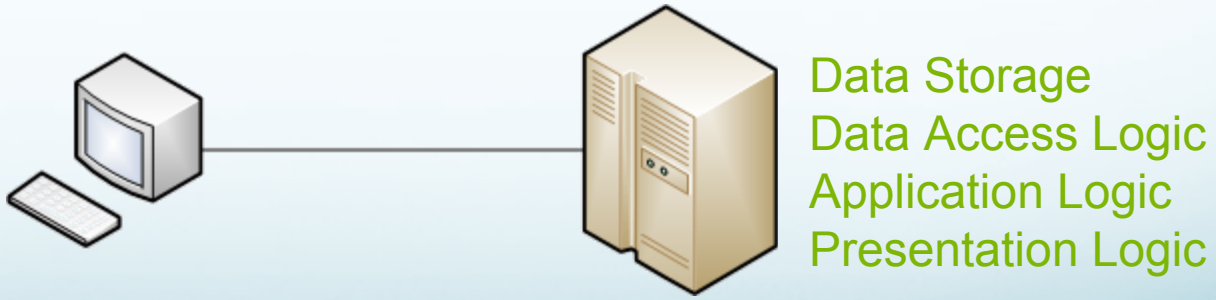
- Purpose is to decide which applications run on what hardware
- Process:
  - Understand the software and hardware options, then
  - Choose from the available alternatives, based on:
    - Cost of acquisition
    - Cost of development
    - Ease of development
    - Interface capabilities
    - Control & security
    - Scalability

Architectural Components

- Software components
  - Data storage
  - Data access logic
  - Application logic
  - Presentation logic
- Hardware components
  - Clients (computers, handhelds, cell phones, etc.)
  - Servers (mainframes, minis, micros, rack mounted)
  - Networks to connect all computers (Dial-up, always-on, medium or high speed, leased lines)

Server-Based Architectures

- The server performs all four application functions
- The client (usually a terminal with display and keyboard) captures keystrokes and sends them to the server for processing



Client-Based Architectures

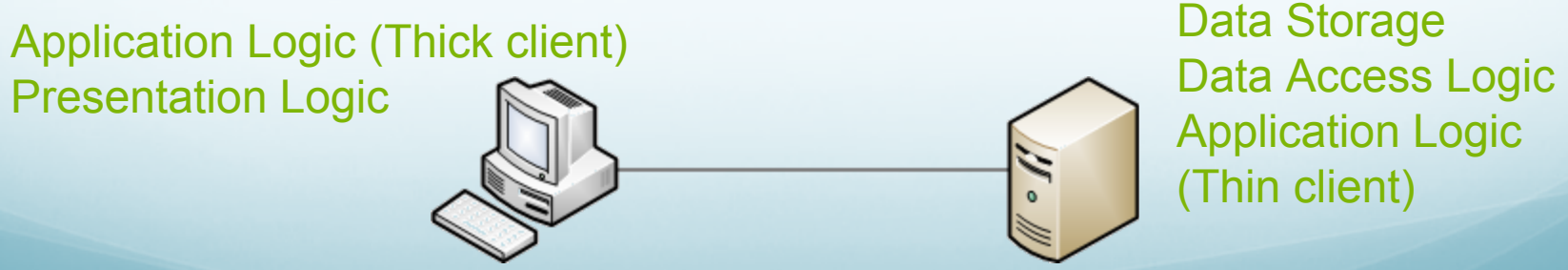
- Clients are personal computers on a network
- Server is a file server on the same network
- Simple to develop, but quickly overloaded
  - All data is downloaded to the client for processing
  - Network traffic may become excessive
  - Client may not have enough computing power

Data Access Logic
Application Logic
Presentation Logic

Data Storage

Client-Server Architectures

- Balance processing between client and server
- Predominant architecture in modern systems
- Amount of client processing varies
  - Thin clients do only presentation logic
  - Thick clients do presentation and application
- Highly scalable at incremental cost
- More complex since applications must be written for both client and server



Client-Server Tiers

- Client-server architecture tiers are defined based on how the logic is partitioned:
  - 2-tier: one server responsible for data storage and access; client responsible for application & presentation logic
  - 3-tier: data storage and access logic on one server, application logic on another; client responsible for presentation logic
  - n-tier: application logic split among two servers, data logic on another
    - Common in e-commerce applications
    - Better load balancing
    - More scalable than 2 or 3 tier systems
    - Places higher demands on the network

Selecting a Physical Architecture

- Cost of infrastructure (initial acquisition and future growth)
- Cost of development
- Ease of development
- Interface capabilities
- Control and security
- Scalability (changes in capacity; upgrades)

Architecture Characteristics

	Server-Based	Client-Based	Client-Server
Cost of infrastructure	Very high	Medium	Low
Cost of development	Medium	Low	High
Ease of development	Low	High	Low-Medium
Interface capabilities	Low	High	High
Control and Security	High	Low	Medium
Scalability	Low	Medium	High

Cloud Computing

- Treat IT as a commodity or utility
  - Server is in the "cloud"
  - Client is on the desktop
- The "cloud"
  - A data center, internal or external; or
  - A service provided by a vendor
  - An umbrella technology that includes:
    - Virtualization
    - Service-oriented architectures
    - Grid computing

Green IT

- Anything that reduces the environmental impact of IT
- Topics:
  - E-waste (disposal of toxic materials in old computers)
  - Energy consumption of data centers and desktops
  - The paperless office
- Cloud computing may help to reduce energy consumption and improve the viability of the paperless office

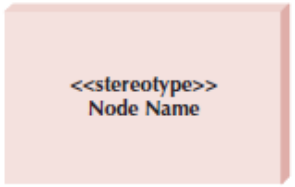
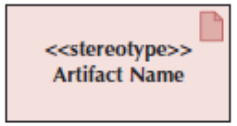
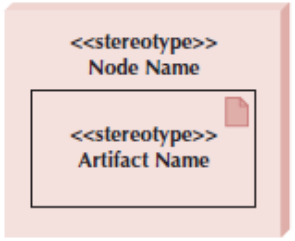
Infrastructure Design

- Although possible, few designs are from scratch
- Most designs utilize systems already in place
  - Change or improve the existing infrastructure
  - Coordination is difficult, but knowledge of elements is essential
    - Deployment diagram
    - Network model

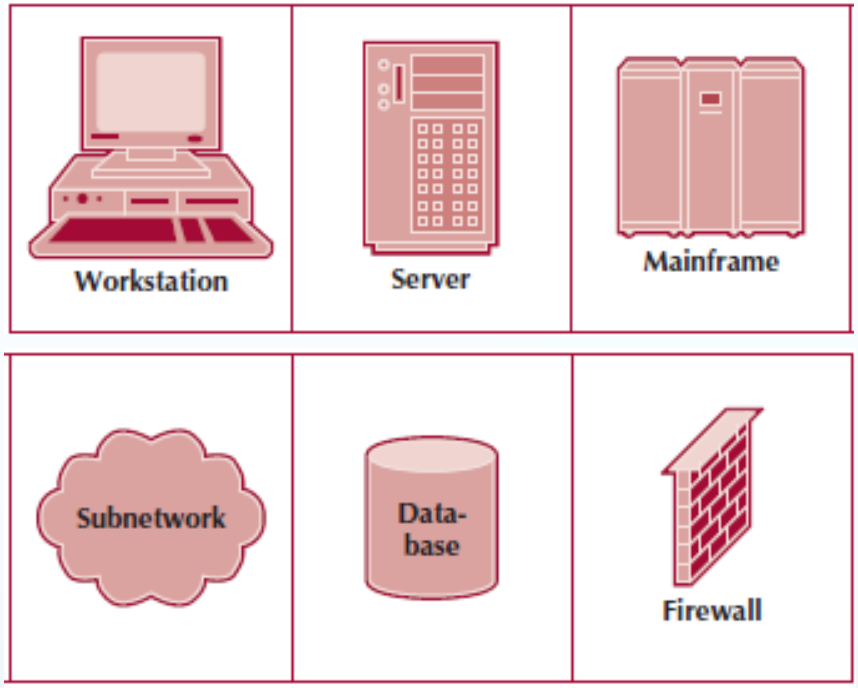
Deployment Diagram

- Represent relationships between hardware components of an information system
- Elements of a deployment diagram
  - Nodes: any piece of hardware (e.g. client computers, servers, networks or network devices)
  - Artifacts: a piece of the information system which will be installed on a node
  - Communication paths: a communication link between the nodes

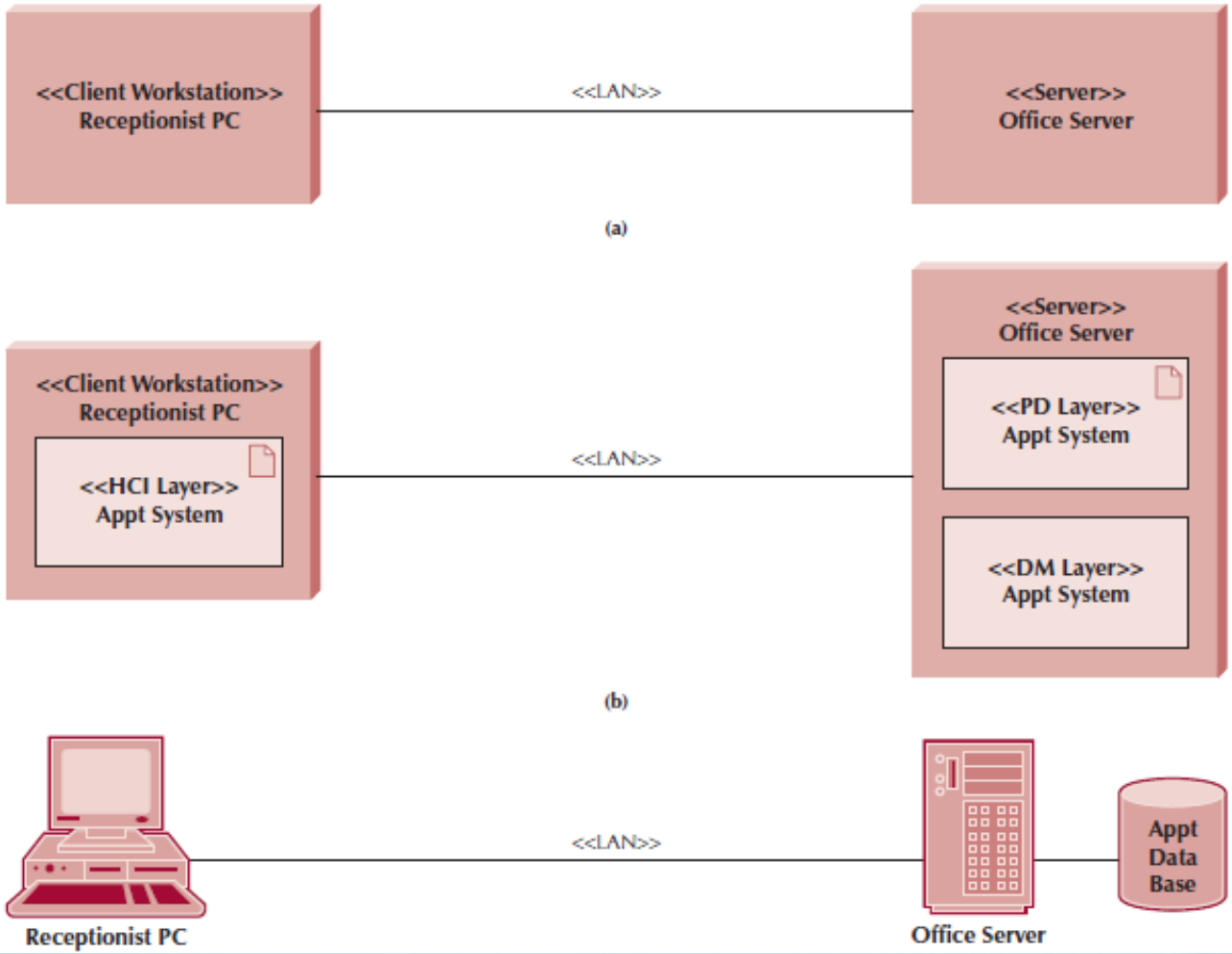
Deployment Diagram Syntax

A node: ■ Is a computational resource, e.g., a client computer, server, separate network, or individual network device. ■ Is labeled by its name. ■ May contain a stereotype to specifically label the type of node being represented, e.g., device, client workstation, application server, mobile device, etc.	
An artifact: ■ Is a specification of a piece of software or database, e.g., a database or a table or view of a database, a software component or layer. ■ Is labeled by its name. ■ May contain a stereotype to specifically label the type of artifact, e.g., source file, database table, executable file, etc.	
A node with a deployed artifact: ■ Portrays an artifact being placed on a physical node.	
A communication path: ■ Represents an association between two nodes. ■ Allows nodes to exchange messages. ■ May contain a stereotype to specifically label the type of communication path being represented, (e.g., Lan, Internet, serial, parallel).	

Extended Node Syntax



Sample Deployment Diagrams



Network Model

- A network diagram that depicts the major components and their geographic locations in the organization
- Purposes of the network model:
  - To convey the complexity of the system
  - To show how the system's software components will fit together
- Can serve as an aid for specifying hardware and software

Sample Network Model

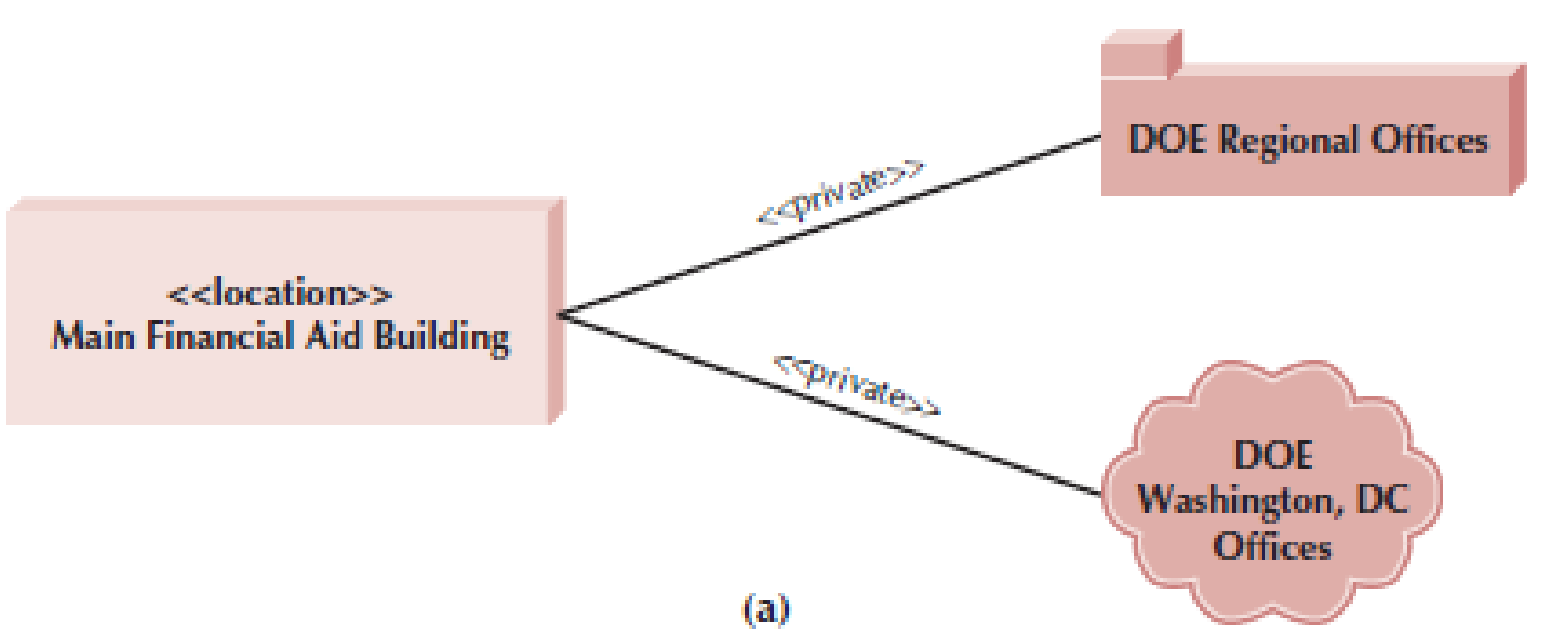
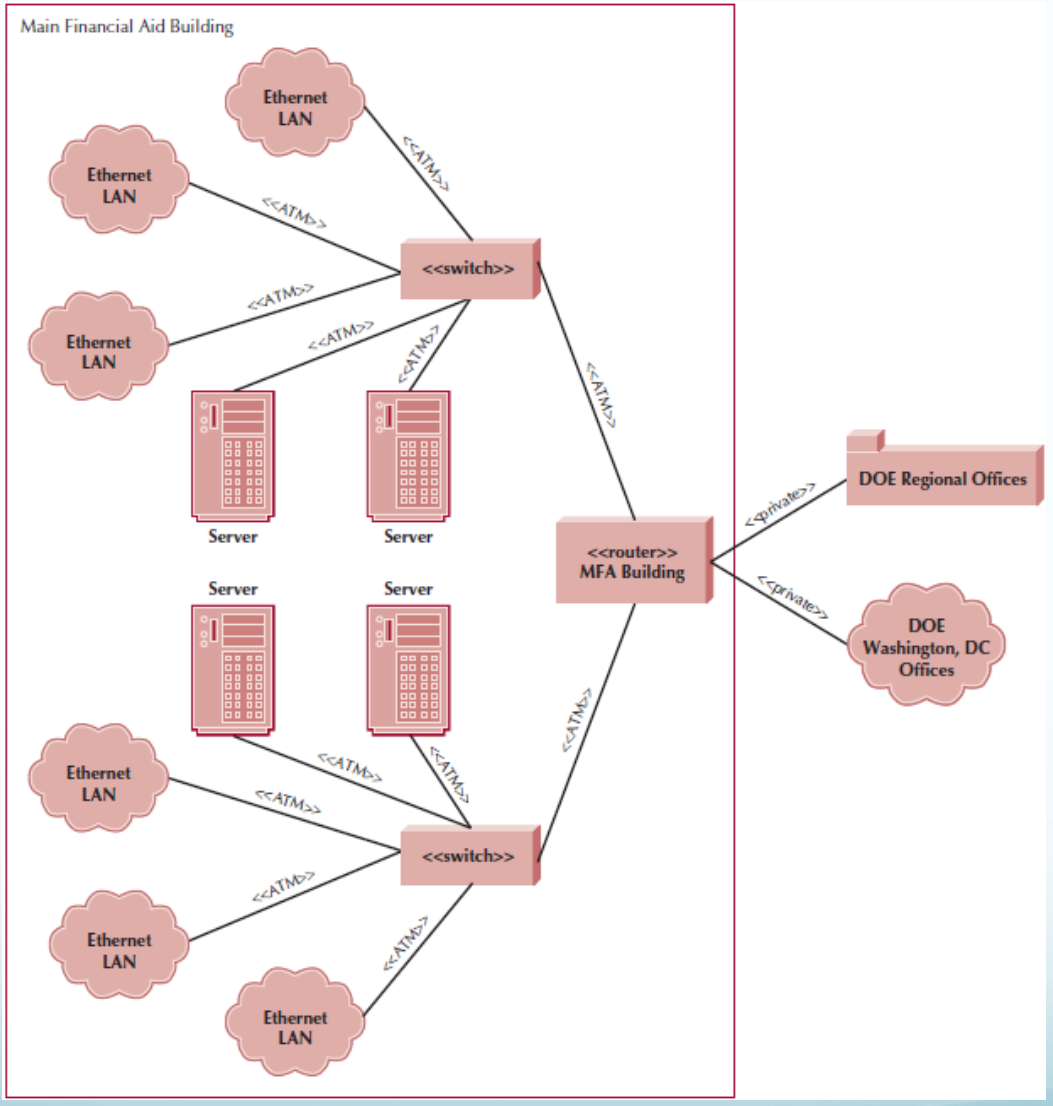


Diagram With Added Detail



Hardware & Software Specifications

- Hardware & software needed for the new application is recorded in a specifications document
- Software requirements:
  - Operating system
  - Special purpose software (e.g., DBMS)
  - Include training needed, maintenance, warranties and licensing agreements
- Hardware requirements
  - Use low level network diagram as a starting point
  - Include type & quantity of servers, peripherals, storage & backup devices
  - Describe minimum requirements
  - Use an alternative matrix to evaluate vendor proposals

Nonfunctional Requirements

- Operational
  - Technical environment
  - System integration
  - Portability
  - Maintainability
- Performance
  - Speed
  - Capacity
  - Availability & reliability
- Security
  - System value
  - Access control
  - Encryption & authentication
  - Virus control
- Cultural & political influence
  - Centralized vs. local control
  - Language differences (keyboard requirements)
- Legal implications
  - Laws & government regulations
  - Global presence requires scrutiny of local laws

Operational Requirements

Type of Requirement	Definition	Examples
Technical Environment Requirements	Special hardware, software, and network requirements imposed by business requirements	• The system will work over the Web environment with Internet Explorer. • All office locations will have an always-on network connection to enable real-time database updates. • A version of the system will be provided for customers connecting over the Internet via a tablet or smartphone.
System Integration Requirements	The extent to which the system will operate with other systems	• The system must be able to import and export Excel spreadsheets. • The system will read and write to the main inventory database in the inventory system.
Portability Requirements	The extent to which the system will need to operate in other environments	• The system must be able to work with different operating systems (e.g., Linux, Mac OS, and Windows). • The system might need to operate with handheld devices such as a Android and Apple iOS devices.
Maintainability Requirements	Expected business changes to which the system should be able to adapt	• The system will be able to support more than one manufacturing plant with six months' advance notice. • New versions of the system will be released every six months.

Performance Requirements

Type of Requirement	Definition	Examples
Speed Requirements	The time within which the system must perform its functions	• Response time must be less than 7 seconds for any transaction over the network. • The inventory database must be updated in real time. • Orders will be transmitted to the factory floor every 30 minutes.
Capacity Requirements	The total and peak number of users and the volume of data expected	• There will be a maximum of 100–200 simultaneous users at peak use times. • A typical transaction will require the transmission of 10K of data.
Availability and Reliability Requirements	The extent to which the system will be available to the users and the permissible failure rate due to errors	• The system will store data on approximately 5,000 customers for a total of about 2 MB of data. • Scheduled maintenance shall not exceed one 6-hour period each month. • The system shall have 99% uptime performance.

Security Requirements

Type of Requirement	Definition	Examples
System Value Estimates	Estimated business value of the system and its data	• The system is not mission critical but a system outage is estimated to cost $50,000 per hour in lost revenue. • A complete loss of all system data is estimated to cost $20 million.
Access Control Requirements	Limitations on who can access what data	• Only department managers will be able to change inventory items within their own department. • Telephone operators will be able to read and create items in the customer file but cannot change or delete items.
Encryption and Authentication Requirements	Defines what data will be encrypted Where and whether authentication will be needed for user access	• Data will be encrypted from the user's computer to the website to provide secure ordering. • Users logging in from outside the office will be required to authenticate.
Virus Control Requirements	Requirements to control the spread of viruses	• All uploaded files will be checked for viruses before being saved in the system.

Cultural & Political Requirements

Type of Requirement	Definition	Examples
Customization Requirements	Specification of what aspects of the system can be changed by local users	• Country managers will be able to define new fields in the product database to capture country-specific information. • Country managers will be able to change the format of the telephone number field in the customer database.
Legal Requirements	The laws and regulations that impose requirements on the system	• Personal information about customers cannot be transferred out of European Union countries into the United States. • It is against U.S. federal law to divulge information on who rented what videotape, so access to a customer's rental history is permitted only to regional managers.

Summary

- Elements of the Physical Architecture Layer
- Cloud Computing
- Green IT
- Infrastructure Design
- Hardware & Software Specifications
- Nonfunctional Requirements