Chapter 17
Web Servers

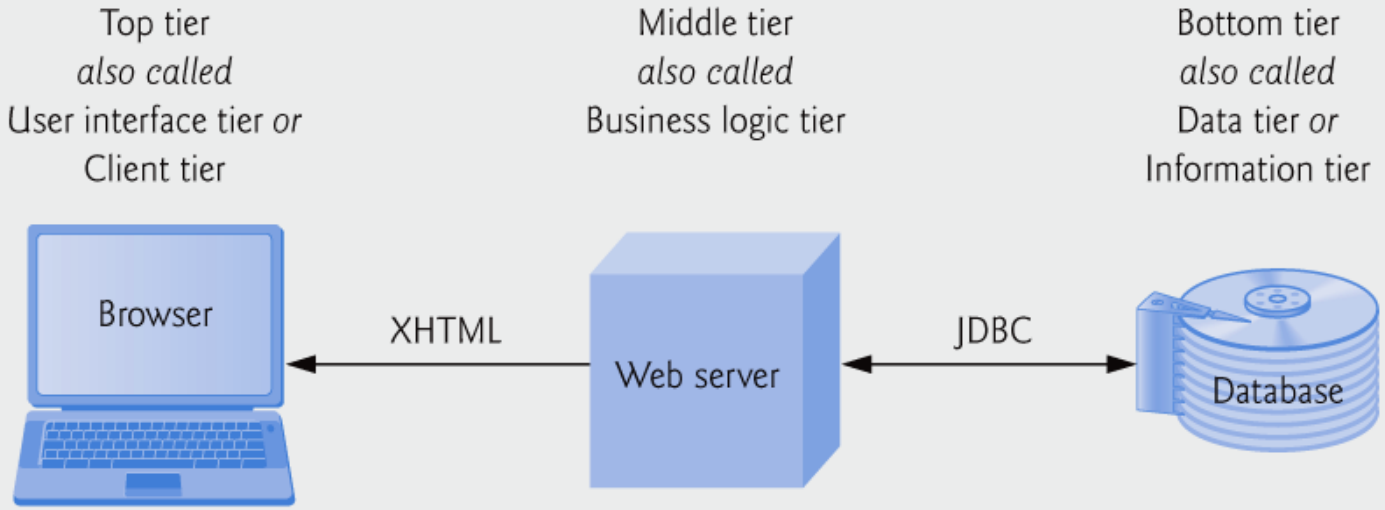
Introduction

- When entering a URL address into a browser
  - Requesting a document from a Web server
- Web servers respond to client requests by providing resources such as HTML documents
- Server and client communicate with HTTP
- HTTP uses URLs to identify resources on the Internet
  - `http://` indicates the HTTP protocol

- Hostname (server name is translated into IP address
  - By a domain name system (DNS) server
    - Maintains a database of hostnames and their corresponding IP addresses
    - The process is called a DNS lookup
- The remainder of URL after hostname specifies
  - Both the resource name and its path (or location) on the Web server
  - For security reasons, the path normally specifies the location of a virtual directory
  - A server translates a virtual directory into a real location, thus hiding the true location of the resource

Multitier Application Architecture

- Web-based applications are multitier applications (*n*-tier application) that divide functionality into separate tiers
- Although tiers can be located on the same computer, the tiers of web-based applications typically reside on separate computers



- The bottom tier maintains the application's data
  - Typically stores data in RDBMS
- The middle tier implements the business, controller, and presentation logics to control interactions between clients and data
  - Business logic enforces business rules and ensures that data is reliable before the server application updates the database, or presents the data to users
  - Business rules dictate how clients can and cannot access application data, and how applications process data

- The top tier is application’s user interface
  - Typically a Web browser
  - In response to user actions, the client tier interacts with the middle tier to make requests and to retrieve data from the information tier
  - The client tier then displays the data retrieved
  - The client tier never directly interacts with the information tier

Client-Side versus Server-Side Scripting

- Client-side scripting does
  - Validate user input, access the browser, process the DOM of a page, and add Ajax functionality
- Client-side scripting does have limitations, such as browser dependency
  - Client-side scripts can be viewed by the client by using the browser's source-viewing capability
    - Sensitive information, such as passwords or other personally identifiable data, should not be stored or validated on the client

- Server-side scripting languages have a wider range of programmatic capabilities
  - For example, server-side scripts often can access the server's file directory structure, whereas client-side scripts cannot access the client's directories
- Properly configured server-side scripts are not visible to the client
  - Only HTML and any client-side scripts are visible to the client

Accessing Web Servers

- To request documents from servers, users must know the hostnames on which the web server software resides
- Users request documents from local or remote servers
- Local web servers can be accessed through your computer's name or through the name `localhost`
  - Translates to the IP address `127.0.0.1` (also known as the loopback address)

Web Server Operation

- When a server starts, it tell its OS it is ready to accept communications through a specific port (80)
  - Apache running under UNIX
- Web servers have two separate directories
  - document root & server root
- Document root is the root directory of servable documents

- E.g., Suppose the site name is www.bloomers.com and the document root is named **topdocs**, and it is stored in the **/admin/web** directory
- /admin/web/topdocs is the document directory address
- If a request URL is:

  http://www.bloomers.com/bulbs/tulips.html

- The server will search for the file with the given path:

  /admin/web/topdocs/bulbs/tulips.html

- The server can have virtual document trees
  - Sometimes a different disk, possibly on a different machine, is used after the original disk is filled

- Server root is the root directory for all of the code that implements the server
  - Usually has four files
    - One is the code for the server itself
    - Three others are subdirectories
      - conf - for configuration information
      - logs - to record activities/errors
      - cgi-bin - for executable scripts
- Provide many services:
  - Virtual hosts: multiple sites on the same system
  - Proxy servers: serve documents from the document roots of other sites
  - Besides HTTP, support for FTP, email, etc
  - Support for database access

Apache under UNIX

- Apache is available for other platforms, but mostly UNIX
- The configuration file is named httpd.conf
- 150 directives control the operation of the server
  - Comments begin with a #
  - Blank lines are ignored
  - Non-blank lines that do not begin with # must begin with a directive name, which may take parameters, separated by white space
- When Apache begins, it reads the configuration files and sets its parameters according to what it reads

- Use the following UNIX commands to force Apache to reset

  cd /usr/local/etc/httpd/logs

  kill -HUP `cat httpd.pid`

  - Works because Apache writes its process id (pid) into httpd.pid when it starts
- Directives
  - ServerName: returned by the hostname command

    ServerName www.bloomers.com
  - ServerRoot: set the server root address
    - Default: /usr/local/etc/httpd

    ServerRoot /usr/local/httpd

- ServerAdmin: email address of the site admin

  ServerAdmin webguy@www.bloomers.com
- DocumentRoot: set the document root address
  - Default: /usr/local/etc/httpd/htdocs

  DocumentRoot /local/webdocs
- Alias: specify a virtual document tree
  - Two parameters: virtual path and the actual path

  Alias /bushes /usr/local/plants/bushes
  - Now, http://www.bloomers.com/bushes/roses.html will be mapped to /usr/local/plants/bushes/roses.html
- ScriptAlias: create a secure place for scripts
    - Creates a virtual directory

  ScriptAlias /cgi-bin/ /usr/local/etc/httpd/cgi-bin/

- Redirect: redirect requests to another system
  - e.g., To move the bushes directory to www.bloomers2.com

    Redirect /bushes http://www.bloomers2.com/local/web/bushes
- DirectoryIndex: URL-specified directories
  - When a request includes a URL that ends with a slash, Apache searches for a document to return, called the welcome page (default: index.html)
  - If there is no welcome page, Apache may try to build a directory listing for the home directory (unless automatic directory listings are turned off)
  - To avoid this, provide more than one welcome page names

    DirectoryIndex index.html contents.html

- UserDir: default is public_html

  UserDir public_html

  - Now, if user ‘bob’ stores stuff.html in his public_html directory, the URL will work like http://site-name/~bob/stuff.html
  - To make a subdirectory of public_html available, include it in the parameter

  UserDir public_html/special_stuff

  - To disallow additions and deletions:

  UserDir disabled

- Logs
  - Access logs record (time, date, HTTP command, URL)
  - Error logs have the form: [date/time] error message