

# Client-Server

1

## Why distributed systems? Benefits & Challenges

- The Sydney Olympic game system:  
see text page 29-30
- Divide-and-conquer
- Interacting autonomous systems
- Concurrency
- Transactions

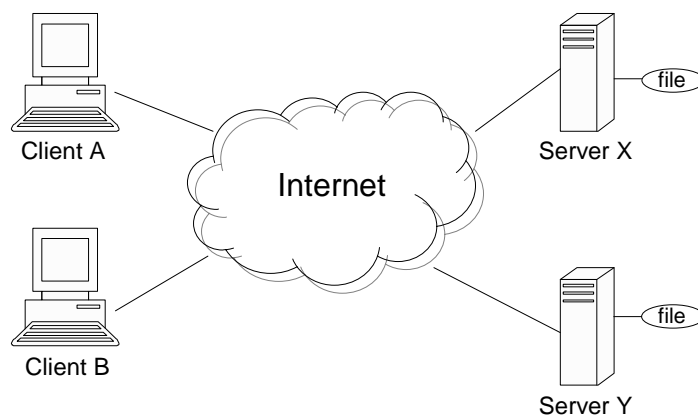
2

## How to get it work

- Open systems, standards, protocols
- Message passing
- Distributed objects technology:
  - CORBA (multi-language technology)
  - RMI (Java-based technology)
  - etc
- etc

3

## Client-Server Model



Not mainframe-terminal, not necessarily big vs. small (rather specialized)

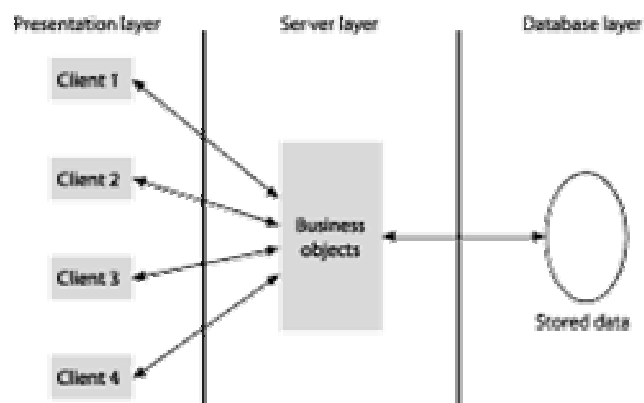
4

## Type of servers

- Database servers
- Web servers
- Mail servers
- Print Servers
- Object servers
- Groupware servers
- etc

5

## A three-tier architecture



What is middleware?

6

## Client-Server vs. P2P

- The new hype
- Uniform role of all nodes
- Direct communication
- Decentralized control
- Scalable

7

## Web client-server

- Web client (browser): an application program that runs on a client computer and requests and displays web pages
  - can also support other features such as email and newsgroups
- Web server: a server-side application program that runs on a host computer and responds to requests for web pages

8

## Commercial Web browsers

Statistics from August 2002<sup>1</sup>:

- Internet Explorer (96%)
- Netscape Navigator (3.4%)
- Other browsers (0.6%)
  - Opera – fastest browser
  - NeoPlanet - customizable, including skins
- How has IE become such a dominant player in the browser market?



<sup>1</sup>Source: [http://www.statmarket.com/cgi-bin/sm.cgi?sm&feature&week\\_stat](http://www.statmarket.com/cgi-bin/sm.cgi?sm&feature&week_stat) retrieved Sept. 15, 2002.

9

## Web client/server communication

- Like any other client/server application, web browsers and servers need a way to:
  1. Locate each other so they can send requests and responses back and forth
    - Uniform Resource Locator (URL)
  2. Communicate with one another
    - Hypertext Transport Protocols (HTTP)

10

# HTTP

- Hypertext Transfer Protocol (HTTP)
  - Lightweight protocol that browsers and servers use to communicate with one another
  - Internet protocol for transferring Web pages
  - runs in the application layer of the TCP/IP model
  - HTTP session begins when a client's browser requests a Web page. Once the server responds by sending the page requested, the HTTP session ends

11

## HTTP is stateless!!!

- stateless because every request that a browser makes opens a new connection that is immediately closed after the document is returned
- even a web page with many objects (graphics, sound, video, etc) requires separate HTTP requests for each object
- each operation is unaware of any other connection
- the server cannot maintain state information about successive requests easily

12

What challenges does “statelessness”  
present to e-commerce?

13

## MIME

- Every document that is returned by a web server is assigned a MIME (Multipurpose Internet Mail Extension) header which describes the contents of the document.
  - “Content-type: text/html” describes an html page
  - This way the browser knows how to display the contents of a web page

14

## What does a HTTP request/response look like?

- Header data + object file
  - Browser request = header only
  - Server response = header + object file (generally)
- Header:
  - Plain text
  - Info about the object (MIME, etc.)
  - Methods allowed
  - Etc.
- Browser send a header to server each time you ask for information
- Server responds with a header and possibly content

15

## HTTP: Hyper Text Transfer Protocol

client request:      GET /index.html HTTP/1.1  
                          User-Agent: Lynx/2.4  
                          Connection: Keep-Alive  
                          Host: www.openaccess.com  
                          Accept: text/html

Server response:    HTTP/1.1 200 OK  
                          Date: Thu, 22 July 1998 18:40:55 GMT  
                          Server: Apache 1.3.5 (Unix) PHP/3.0.6  
                          Last-Modified: Mon, 19 July 1997  
                          16:03:22 GMT  
                          Content-Type: text/html  
                          Content-Length: 12987  
                          ...

16



# HTML

- Hypertext Markup Language

- presentation language
- embedded text that tells the client's browser how to display the page elements
- standard formatting language that all browsers understand
  - theoretically !
  - example tag
    - `<h1> ECMM 6010 </h1>`

**ECMM 6010**

17

## Client-side interactivity

- HTML is a presentation language – *not* a programming language
- controls the appearance of the information on the client's screen but does not support processing or manipulating information
- early web browsers were limited to displaying static pages
  
- But ... many business applications require interactivity

18

## Client side processing

- Examples:

1. HTML
2. JavaScript
3. VBScript
4. Applet
5. Plug-ins

- Other types of client-side processing

1. Cascading style sheets (CSS)
2. Dynamic HTML (DHTML)

19

## HTML

```
<html>  
<TITLE> HTML sample </TITLE>
```

```
<BODY bgcolor="#FFFF80">  
<CENTER>  
<H3>HST Calculator </H3>
```

To compute the amount of HST you need to pay, take the price of the  
item and  
multiply it by 0.15

```
</BODY>
```

```
</html>
```

20

Figure 3.6, Davis and Benamati, pg. 75. Using TCP/IP protocols to download a web page

