

Web Site Software

James Baldo Jr.

SWE 432

Design and Implementation of Software for the Web

**“Where is the wisdom we lost in knowledge?
Where is the knowledge we lost in information?”
– T.S. Eliot**

Motivation – Overview

- Modern web sites are:
 - Distributed (world-wide)
 - Heterogeneous (hardware and software)
 - Highly user interactive
 - Built on new technology
- The software is:
 - Very loosely coupled
 - Written in multiple languages
 - Often generated dynamically

Diverse: In terms of software, communication, and people

Motivation – Overview (2)

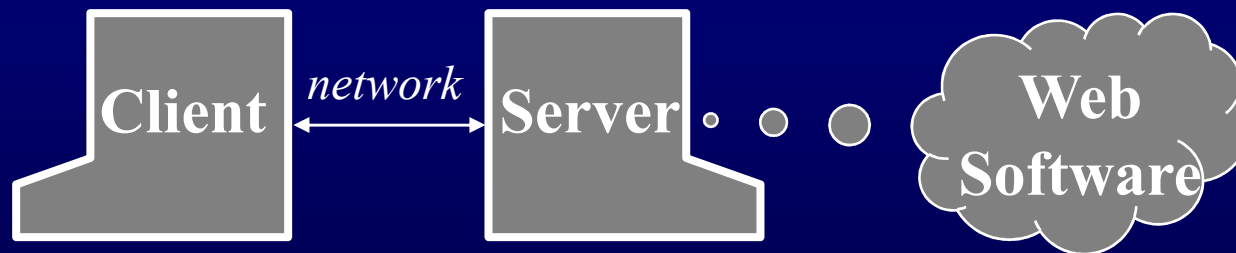
- Web site software has to be better than most shrink-wrap or contract software
- The combination of higher quality requirements and unique technologies make for a very *interesting* situation

(Academics think “interesting” means fun, managers think “interesting” is scary ...)

This talk discusses why and in what ways web software must be better

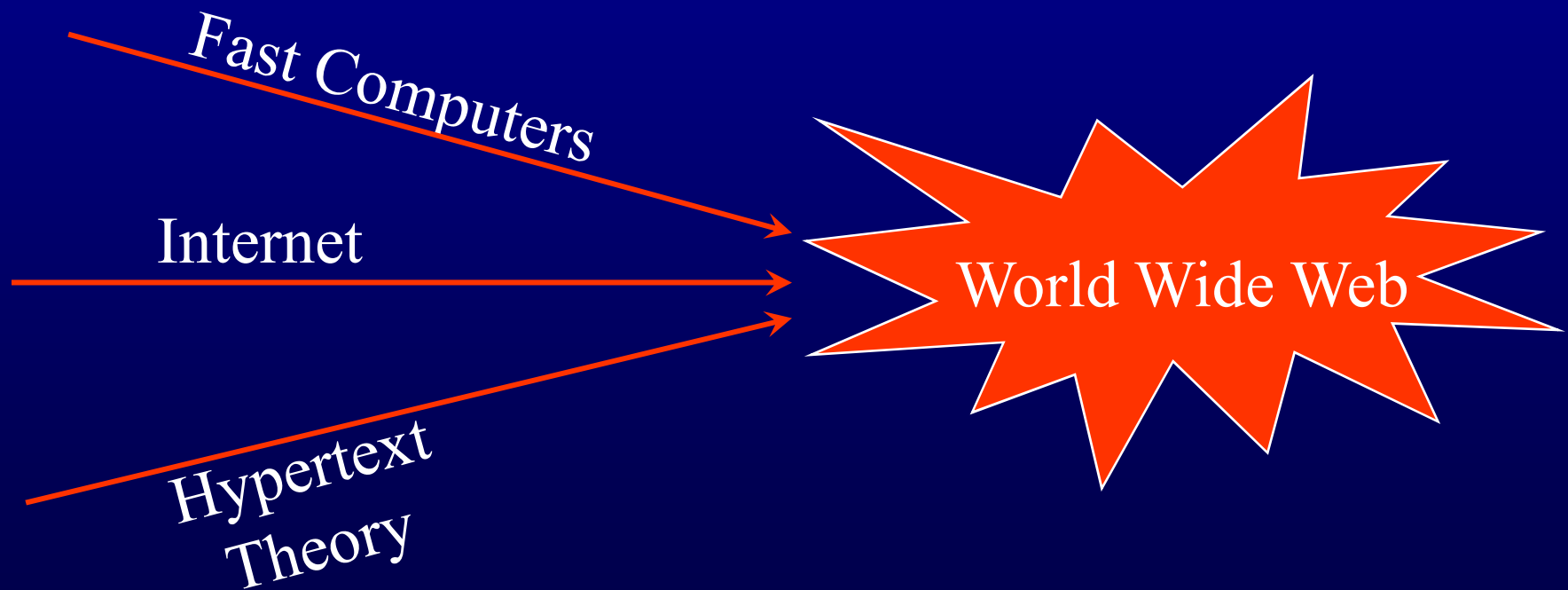
Web Sites and Software

- Web Page : Data that fits in one browser screen.
- Web Site : A number of connected web pages.
- Web Site Software : Software that makes web sites dynamic.



Formation of the WWW

The World Wide Web is a result of the *confluence* of three developments



Historical Highlights

Hypertext, Internet, and the Web

1945: Vannevar Bush proposes **hypertext**

1965: Ted Nelson coins the term “**Hypertext**”

Douglas Engelbart invents **mouse**

(The ACM Douglas Engelbart award is given yearly at the HyperText conference)

1968: Engelbart’s hypertext system “**Augment/NLS**”

Brown university’s **HES**

1969: **Arpanet**

1970: **Unix**

1972: Tomlinson invents **email**

1977: **UUCP**

1978: First **bulletin board system** – Christensen and Sues

Historical Highlights (cont.)

1979: Truscott and Bellovin's **usenet news**

1982: < **100 sites** on "internet" (netnews)

1983: **TCP / IP** defines the internet

1986: **NSFNET**

SGML

1987: **HyperCard** (Apple)

First hypertext **workshop**

1989: **WWW** proposed by Berners-Lee

1990: **HTML** defined

1992: CERN (Switzerland) **releases WWW**

1993: NCSA **Mosaic**

Historical Highlights (cont.)

1994: **WWW** surpasses gopher

1994: **Netscape**

1995: **Lycos** search engine

Java

DEC's **Alta Vista** search engine

15 million pages

1996: **US Telecommunications Bill**

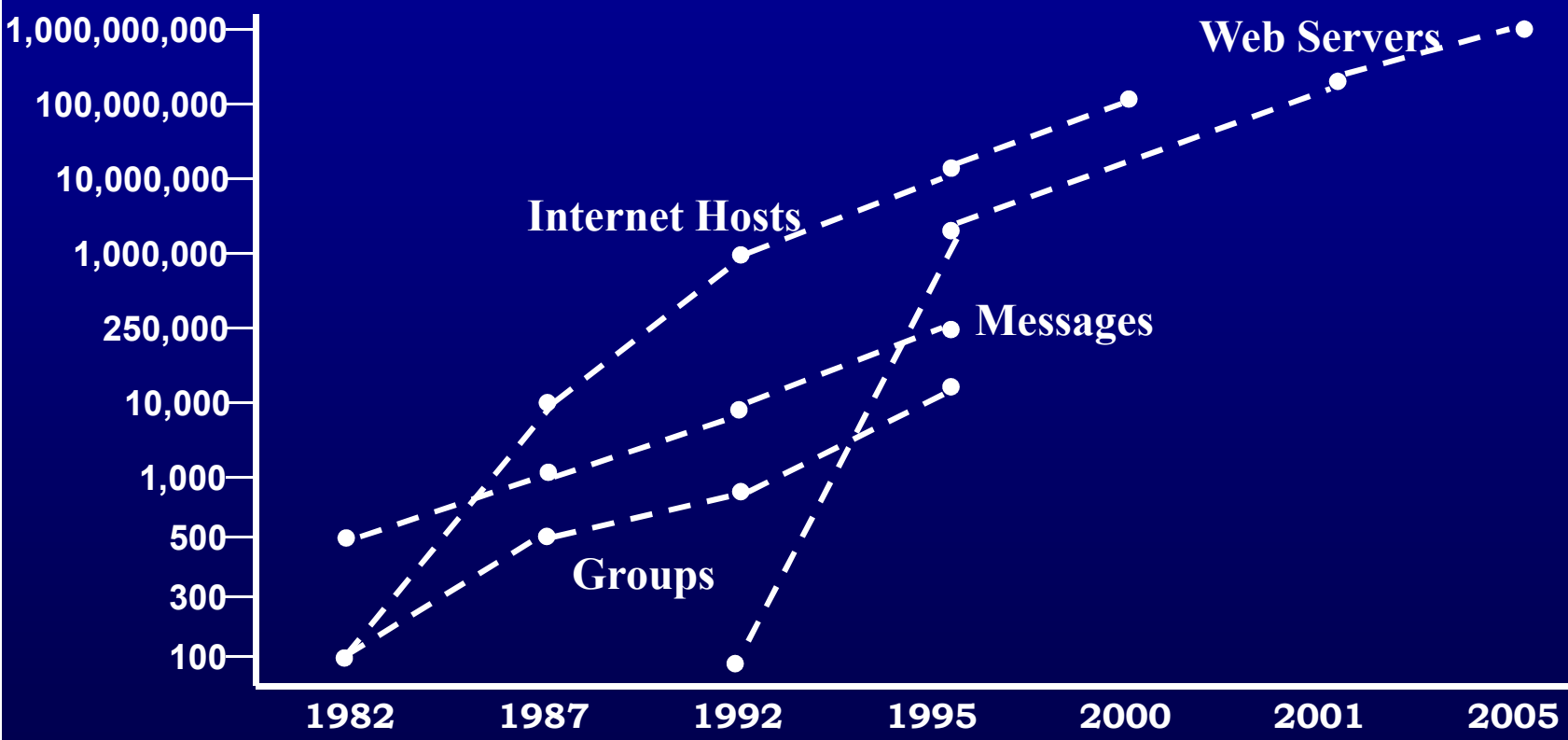
1997: **31,000,000 pages**, 476,000 servers, 14,000 news groups

2000: More than **100,000,000 hosts**

Amount of **back-end programming** exceeds the front-end
hypertext

2004: 3,307,998,701 **pages** (google), 500,000,000 **hosts**,
1,200,000,000 **users**

Internet Size



2000 : Over 2.1 Billion web pages

The World Wide Web

The world wide web was invented by Tim Berners-Lee, an Englishman at CERN, the physics research lab in Switzerland

Tim Berners-Lee's vision of the world wide web was to make all information available to all people at all times.

General Hypertext Terminology

- Node : Atomic amount of text – smallest unit of information known to system (pages)
- Link : Connects one node to another. Depicted by buttons or highlighted region.
 - Link Region (anchor) : Location in a node where link starts.
- Traverse (visit or view) : Moving from one node to another
 - Backtrack : Following links in reverse
- Landmark : Semantically important nodes that are linked to many nodes
- Hypermedia : Incorporates sound, video, pictures, or other non-textual information

General Web Terminology

- Web Page : Data that fits in one browser screen
 - Static Web Page : A web page whose HTML exists as a file on a computer
 - Dynamic Web Page : A web page that is created as needed
- Web Site : A number of web pages that are *semantically* related and *physically* linked
- Web Site Software : Software that makes web sites dynamic.
 1. Dynamic Web Pages
 2. User Interaction

Modern Web Sites

Web sites have continuously evolved in the last decade

We have moved from:

- ftp/email ...
- to gopher ...
- to simple html pages ...
- to web sites ...
- to dynamic html ...
- to web commerce ...

with amazing speed!

Modern Web Sites

- Web sites are now too complicated for individuals to manage.
- They need to be engineered by teams of people with diverse talents:
 - Programming skills
 - Graphics design
 - Usability
 - Information layout and engineering
 - Data communications
 - Data base

We need web site engineering

Important Quality Attributes for Traditional Software

Traditional

1. Efficiency of process (time-to-market)
2. Efficiency of execution (performance)



50. Reliability

51. Safety

52. Maintainability

53. Security



Important Quality Attributes

1. Reliability
 2. Usability
 3. Security
-

4. Availability
5. Scalability
6. Maintainability
7. Performance & Time to market

Customers have little “site loyalty” and will switch quickly, thus time to market is much less important than in other application areas.

(but still important!)

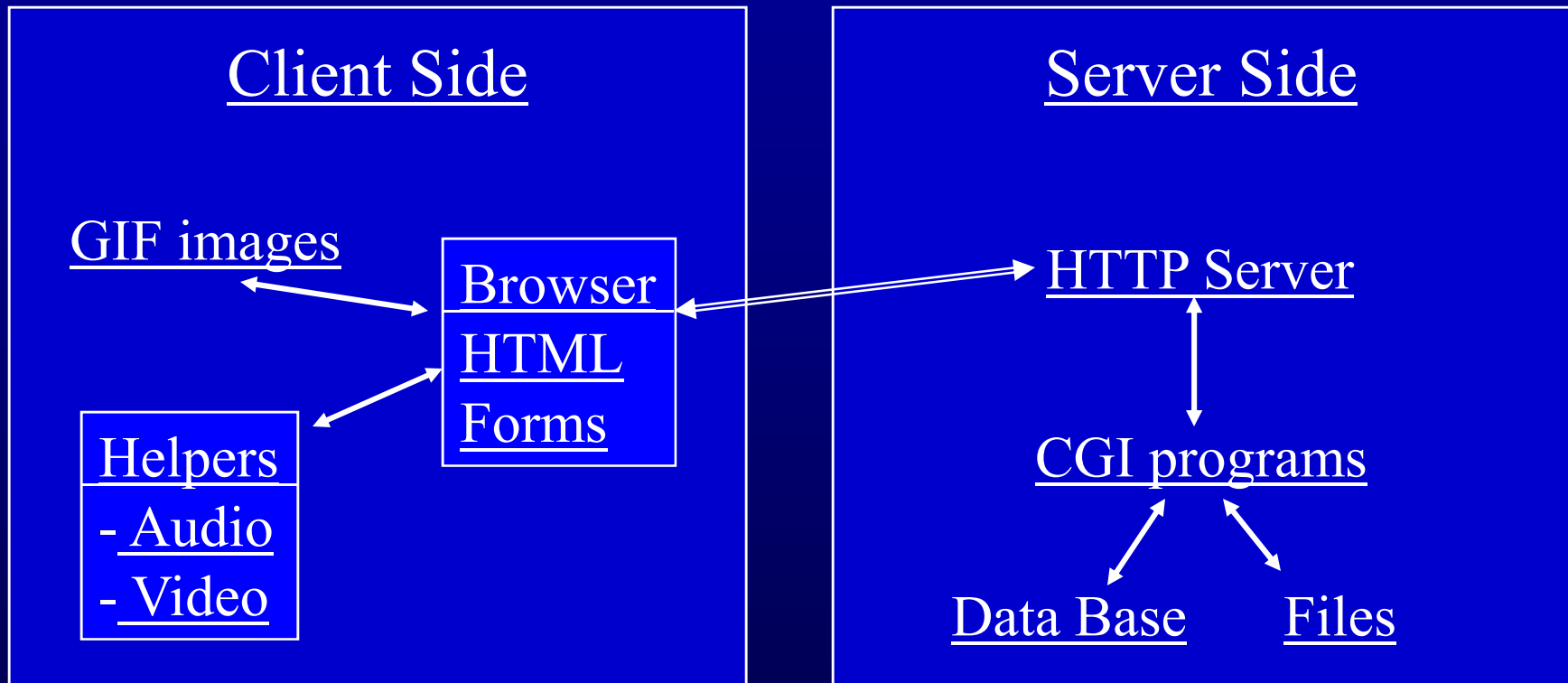
Based on an informal survey of around a dozen software development managers, 2000.

First Generation Web Sites (Mosaic, '93)

- Text with few images
- Weak layout facilities
- Very limited CGI programming – forms
- No background color

One “webmaster” could do it all!

First Generation Web Sites



Web sites were usually created by one “early adopter.”
Most were poorly designed and incomplete.

Second Generation Web Sites

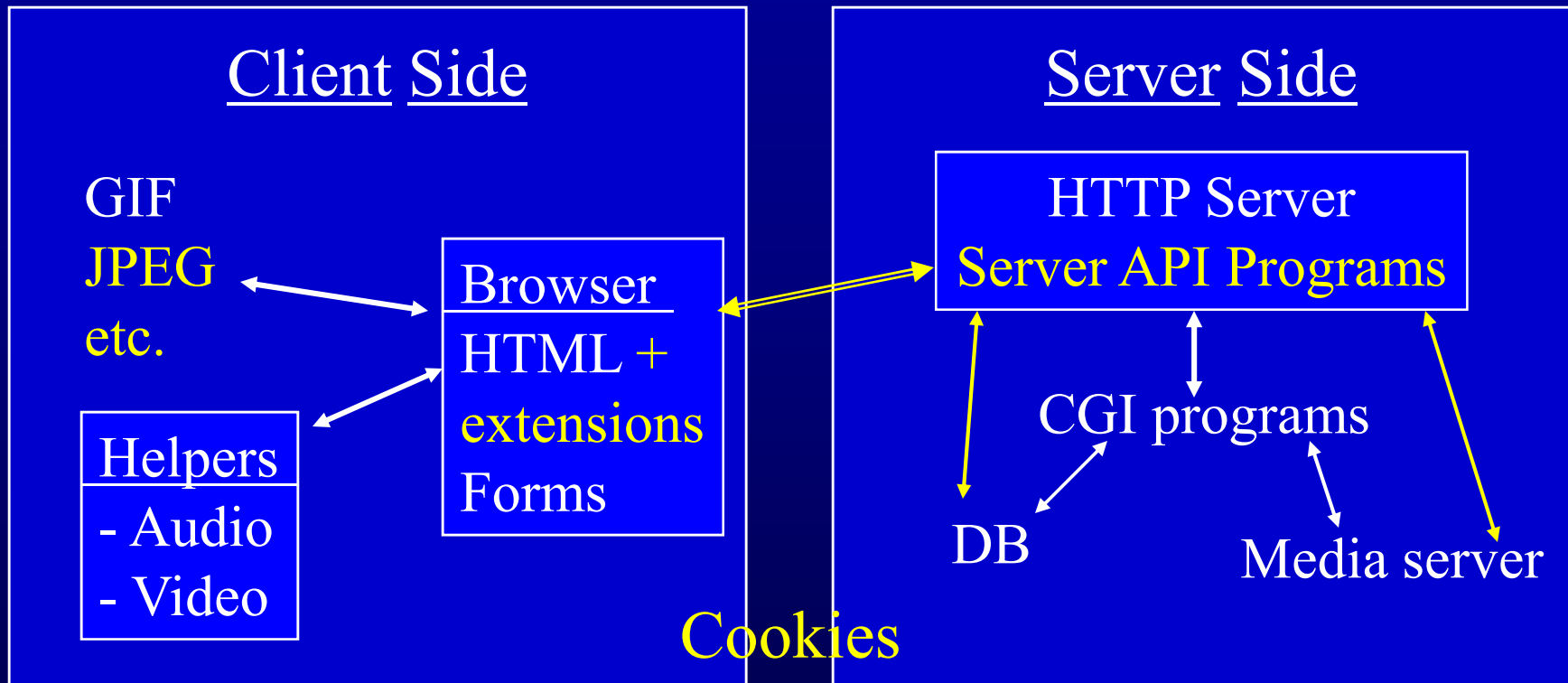
(Netscape, '94)

- Centering and other layout options
- Background colors
- Tables

- Graphics designers started to develop web sites
- Technical & creative jobs diverged
- A lot of emphasis on appearance (over-designed brochures)

- Designers realized: People did not read websites, they visited them for a purpose.

Second Generation Web Sites



More programming, but mostly print-oriented.
Cookies allowed state information to be saved.

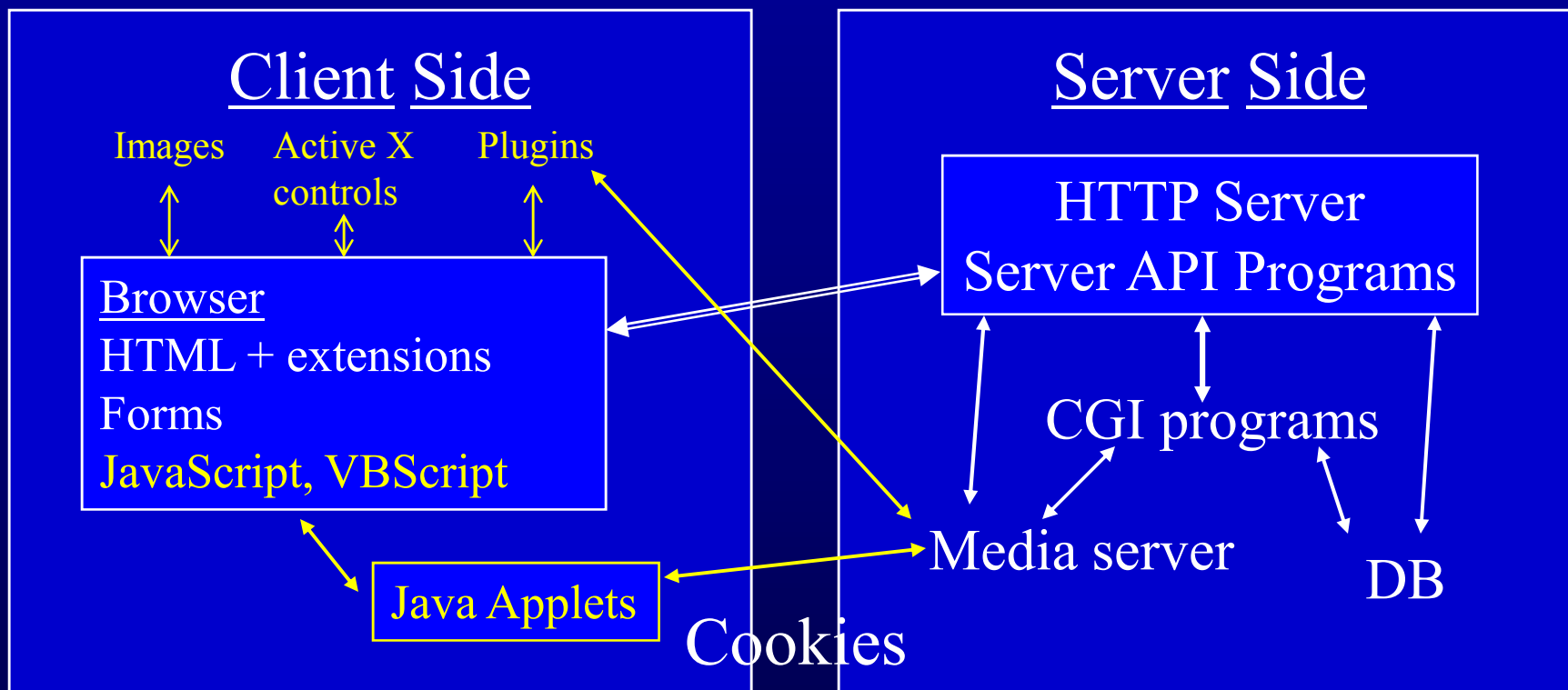
Third Generation Web Sites

(Netscape, IE, 96--97)

- People expected interaction.
- Bandwidth became a serious problem (still is!)
- More layout, proprietary tags, frames, stylesheets, ...

- Java
- JavaScript
- ActiveX
- Applets
- ...

Third Generation Web Sites



Full interactivity and rich software abilities.

Third Generation Web Sites

New interactive applications

- FedEx document tracking
- Airline flight services
- Ticket agencies
- Maps

The birth of E-commerce

Impossible for one person to do everything.

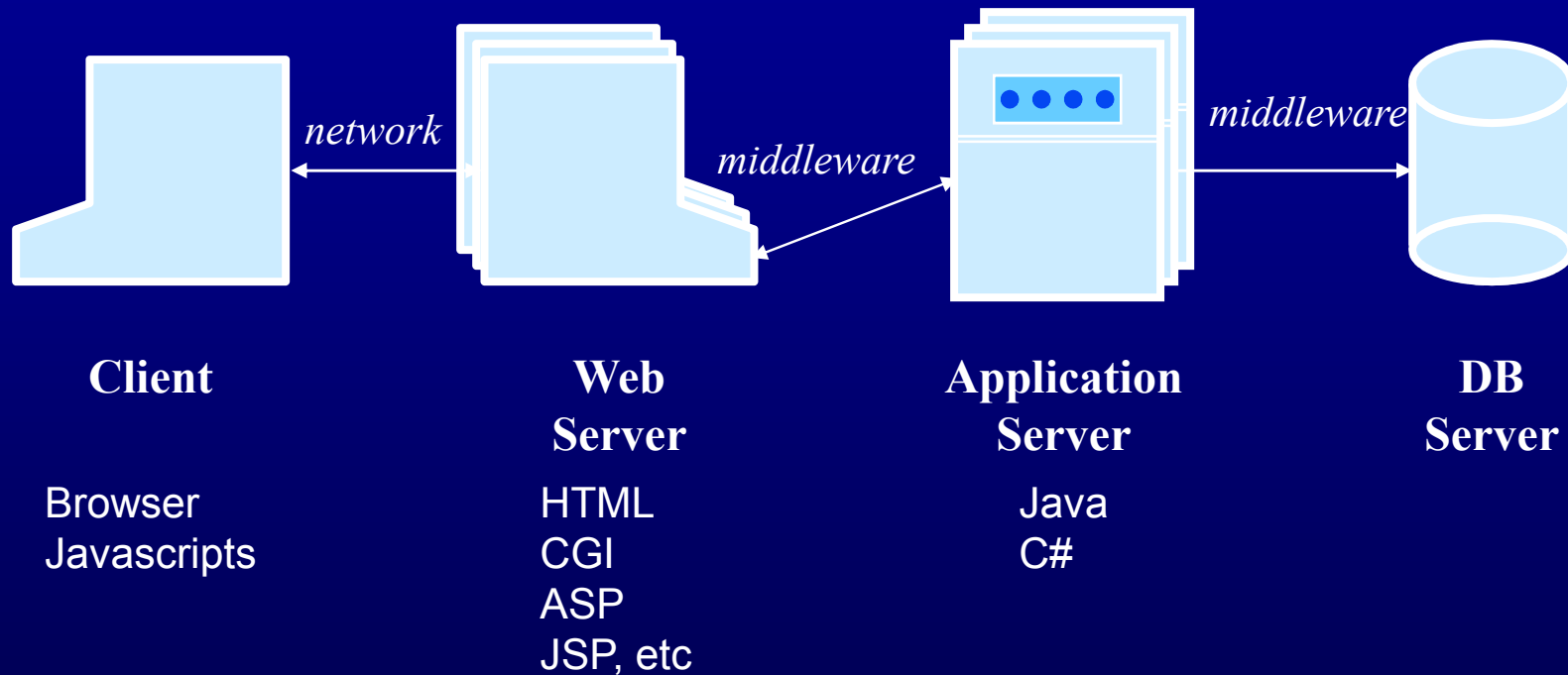
A Web Manager must manage programmers, graphics designers, DB administrators, security experts, usability engineers, business experts,

Third Generation Web Sites

Uses of web sites:

- Marketing
- Promotion
- Commerce
- Entertainment
- Technical support
- Investor relations
- Employee recruitment
- Personal satisfaction and expression
- Distributing research results
- ...

Fourth Generation Web Sites (software-centered)



Client-server ... 3-tier ... N-tier ...

Problems Can Occur Anywhere

- 1995: Web sites were 100% interface
- 1998: Web sites were about 90% interface
- 2001: Web applications are less than 50% interface
- 2005: Web applications about 25% interface

There is still a shortage of knowledgeable, skilled web programmers and software engineers

A Time of Transition

- The “.com-.gone” crash of 2000-2002 has been terrible for many of my friends and students who lost jobs and took pay cuts
- But I see a positive sign – these claims are validated:
 - The companies that “figured it out” are still alive, doing well, growing, making money, and hiring
- Henry Ford said “everybody is selling something”
- I’m not selling tools, languages, technologies or even research – I’m selling engineering

Summary

Changes in Web Technologies

1997

- Static web pages
- “Soft brochures”
- Webmasters
- HTML, CGI, JavaScript

1998-1999

- Dynamic HTML
- Programs (poorly written)
- Confused webmasters
- ASP, CSS, ...

2000-2004

- Functional websites
- ECommerce +
- WebManager + programmers, DB, network, UI, graphics designers, ...
- Java (JSP, Servlets, beans), {HT,U,X}ML, Component-based
- .NET

Summary

Concerns of Software

Traditional

1. Efficiency of process
(time to market)
2. Efficiency of execution



50. Reliability

51. Safety

52. Maintainability

53. Security



Web Software

1. Reliability
2. Usability
3. Scalability
4. Security
5. Availability
6. Maintainability



Subject of 432

- 432 splits web software into two aspects:
 1. The interface aspect
 2. The software aspect
- The **interface** runs on the **client** side
 - Implementation of HTML is easy
 - Achieving usability is a key, and very difficult
- The **software** runs on the **server** side
 - A mixture of lots of technologies
 - The key to other quality criteria (reliability, etc.)

Summary

The concept of Web Software Engineering is just beginning ...