

Web Design with HTML5, CSS3 and JavaScript

Web Design Concepts

Trần Anh Tuấn A

tuantran261083@gmail.com

Edit slide from Telerik Software
Academy

<!DOCTYPE
<HTML>
<HEAD>
<TITLE>RA
<LINK REV
<META NAME

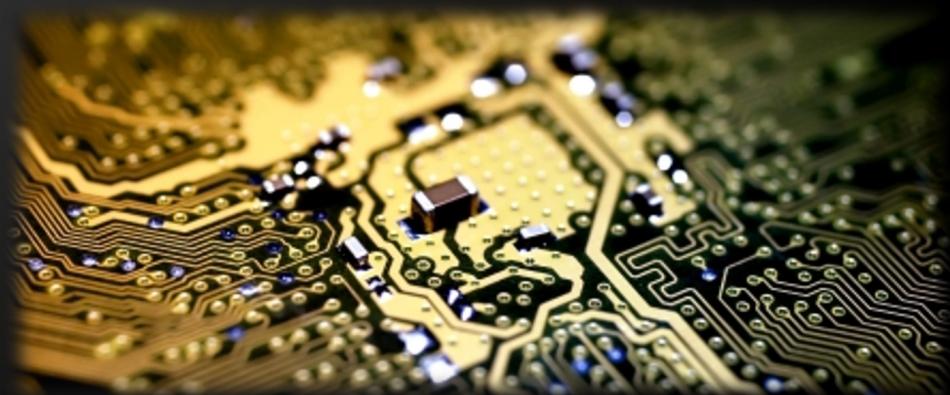


Table of Contents

- ◆ Web Design
- ◆ Web Sites and Web Applications
- ◆ Web 1.0, 2.0, 3.0
- ◆ Web Browsers and Layout Engines
- ◆ Hardware Servers
- ◆ Web Servers
- ◆ Client-Server Architecture
- ◆ 3-Tier / Multi-Tier Architectures



Web Design



What is Web Design?

Web Design

- ◆ Set of the processes for creating of a Web Site
 - Planning – management of the requirements
 - Done by managers
 - Design – designing a UI that matches the requirements
 - Done by graphical designers
 - Implementation – implementing the design and writing the HTML/CSS/JavaScript code
 - Done by Web Front-end developers

Web Sites and Web Applications



Web Site

- ◆ Collection of related web pages containing web resources (web pages, images, videos, CSS files, JS files or other digital assets)
- ◆ Common navigation between web pages
- ◆ A website is hosted on at least one web server
- ◆ Accessible via a network (such as the Internet)
- ◆ All publicly accessible websites collectively constitute the World Wide Web

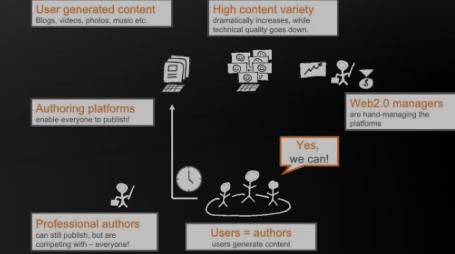
Web Page

- ◆ Document or information resource that is suitable for the World Wide Web
- ◆ Can be accessed through a web browser and displayed on a monitor or mobile device
- ◆ This information is usually in HTML or XHTML format, and may provide navigation to other web pages via hypertext links
- ◆ Web pages frequently refer to other resources such as style sheets (CSS), scripts (JavaScript) and images into their final presentation

Web Application

- ◆ Next level web sites
- ◆ High interactivity
- ◆ High accessibility (Cloud)
- ◆ AJAX, Silverlight, Flash, Flex, etc.
- ◆ Applications are usually broken into logical chunks called "tiers", where every tier is assigned a role
- ◆ Desktop-like application in the web browser
- ◆ Web applications on desktop (Windows 8)





Web 1.0, 2.0, 3.0



Web 1.0

- ◆ Old media model

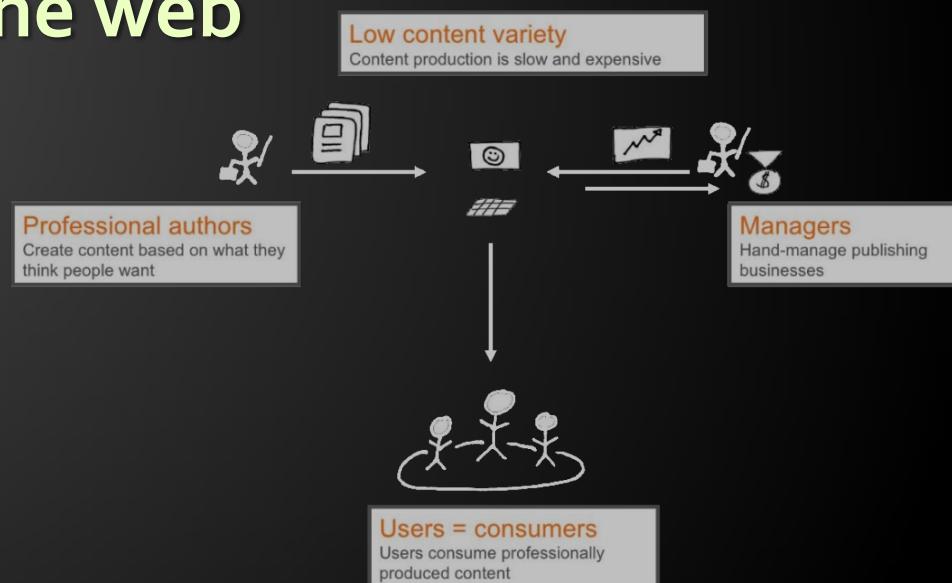
- ◆ It all started with a simple idea

- Just put content in the web

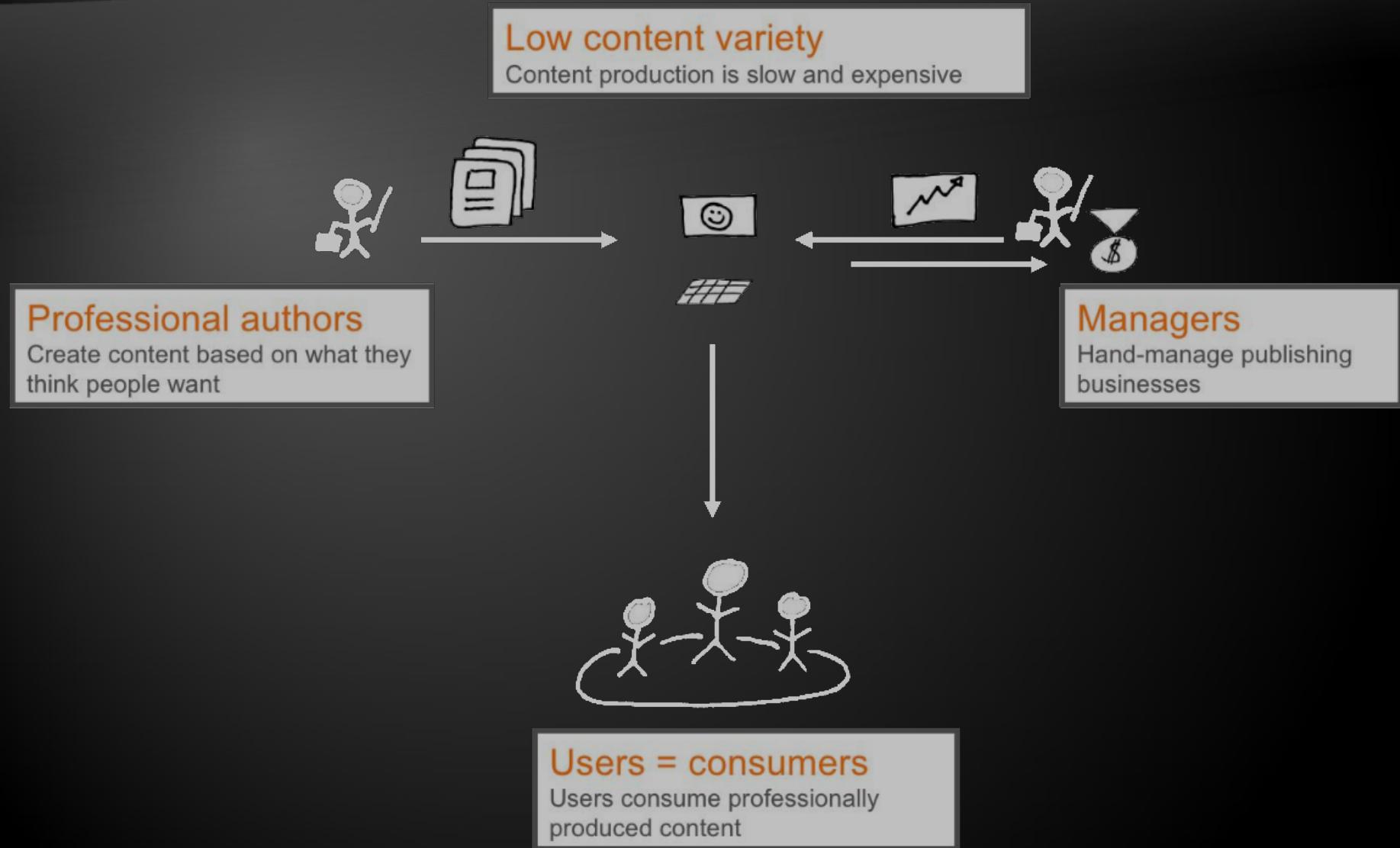
- ◆ Low content variety

- Limited content
 - Limited creativity
 - Limited business

- ◆ 10 000 editors serve 500 000 000 internet users



Web 1.0





- ◆ User generated content
- ◆ New platforms allow users to generate content themselves (YouTube, Wiki, Facebook, Blogs)
- ◆ Everyone can publish!
- ◆ Web 2.0 can be described in 3 parts:
 - Rich Internet application (RIA)
 - Web-oriented architecture (WOA)
 - Feeds, RSS, Web Services, etc.
 - Social Web

Web 2.0

WEB 3.0

User generated content

Blogs, videos, photos, music etc.

High content variety

dramatically increases, while technical quality goes down.



Authoring platforms

enable everyone to publish!

Web2.0 managers

are hand-managing the platforms



Professional authors

can still publish, but are competing with – everyone!

Users = authors

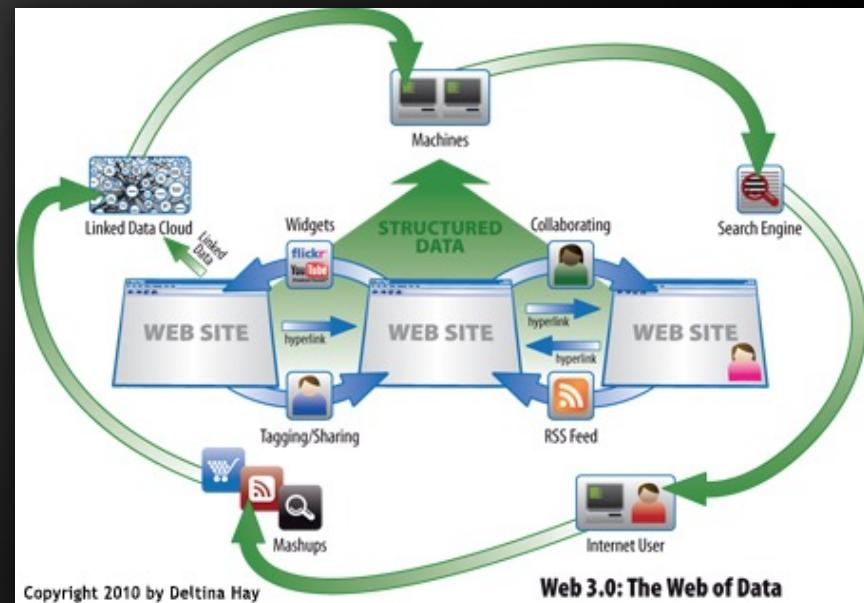
users generate content

Web 3.0



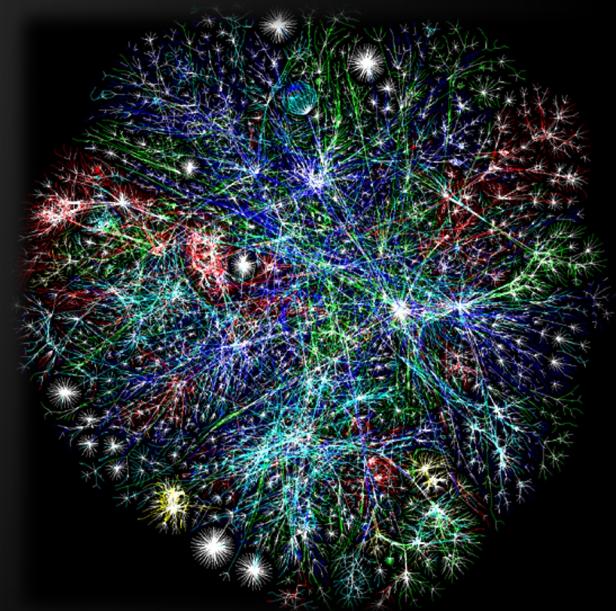
Web 3.0

- ◆ Web 3.0 is where the computer is generating new information, rather than humans.
- ◆ All the new web 3.0 concepts can be divided into 4 parts:
 - Semantic web
 - Artificial intelligence
 - Personalization
 - Mobility



◆ Semantic web

- Changing the web into a language that can be read and categorized by the computers rather than humans
- Makes search engines smarter
- Enables digital collection of all your images, blog post, videos, etc.
- Disadvantage:
it is hard to be implemented



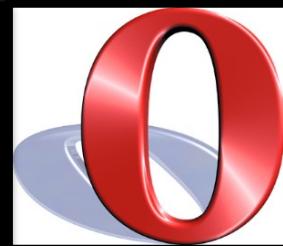
- ◆ Artificial intelligence
 - Extracting meaning from the way people interact with the web
 - Examples: Google suggest, Google translate
- ◆ Personalization
 - Contextualizing the web based on the people using it
 - Different content for different users

◆ Mobility

- Everything
 - Web sites
 - Information
 - Services
- Everywhere
 - You only need your phone or tablet
 - All the time



Web Browsers and Layout Engines



Web Browsers

- ◆ Program designed to enable users to access, retrieve and view documents and other resources on the Internet
- ◆ Main responsibilities:
 - Bring information resources to the user (issuing requests to the web server and handling any results generated by the request)
 - Presenting web content (render HTML, CSS, JS)
 - Capable of executing applications within the same context as the document on view (Flash)

Layout Engines

- ◆ Software component that displays the formatted content on the screen combining:
 - Marked up content (such as HTML, XML, image files, etc.)
 - Formatting information (such as CSS, XSL, etc.)
- ◆ It "paints" on the content area of a window, which is displayed on a monitor or a printer
- ◆ Typically embedded in web browsers, e-mail clients, on-line help systems or other applications that require the displaying (and editing) of web content

Layout Engines and Web Browsers

- ◆ Trident-based

- Internet Explorer, Netscape, Maxthon, etc.



- ◆ Gecko-based

- Firefox, Netscape, SeaMonkey, etc.



- ◆ WebKit-based

- Chrome, Safari, Maxthon, etc.



- ◆ Presto-based

- Opera



User Agent Strings

- ◆ Identify web browsers and their version
- ◆ Can have some additional information like layout engine, user's operating system, etc.
- ◆ Example:

Mozilla/5.0 (Windows NT 6.1; WOW64; rv:7.0.1)

Gecko/20100101 Firefox/7.0.1

- Web browser: Firefox 7.0.1
- Rendering (layout) engine: Gecko/20100101
- Operating system: 64-bit Windows 7
 - WOW64 = Windows-On-Windows 64-bit
 - Windows NT 6.1 = Windows 7

Hardware Servers



Hardware Servers

- ◆ Physical computer (a hardware system) dedicated to running one or more such services
- ◆ Servers are placed in collocation centers
- ◆ The server may be:
 - Database server
 - File server
 - Mail server
 - Print server
 - VPS servers



Web Servers

Apache, IIS, nginx, lighttpd, etc.



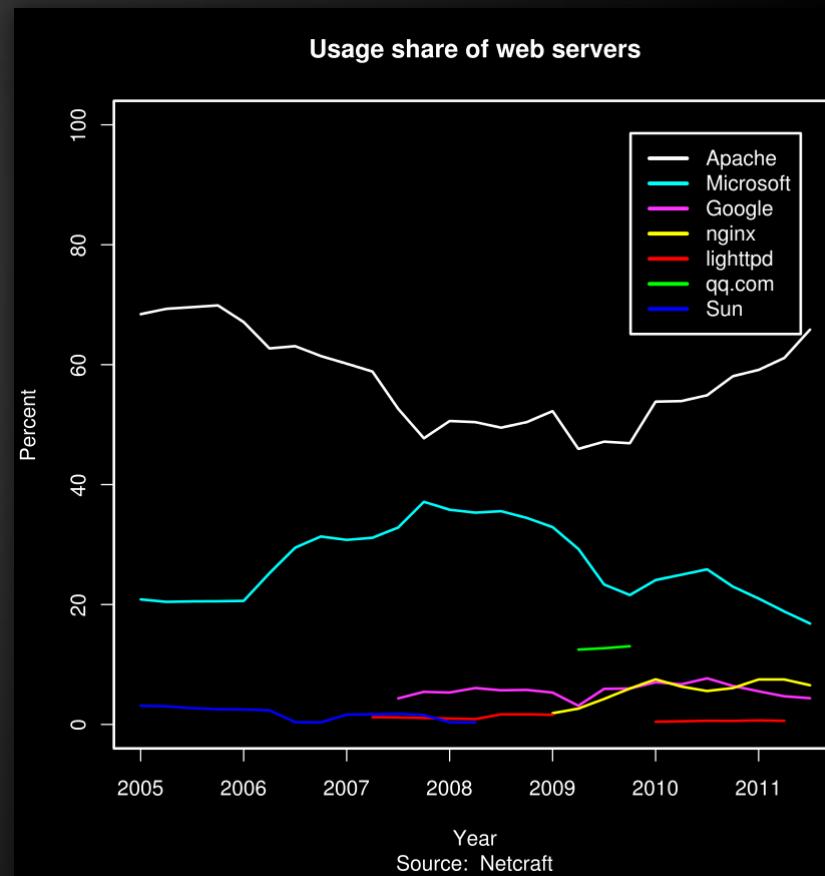
LIGHTTPD
fly light.

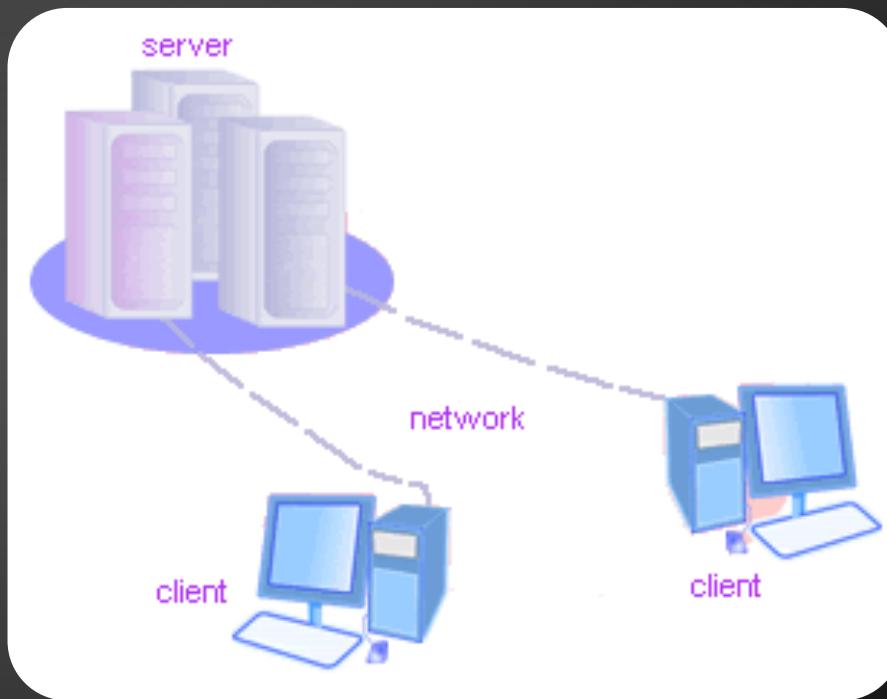
What Do the Web Servers Do?

- ◆ All physical servers have hardware
- ◆ The hardware is controlled by the operating system
- ◆ Web servers are software products that use the operating system to handle web requests
- ◆ These requests are redirected to other software products (ASP.NET, PHP, etc.), depending on the web server settings

Web Servers Market Share 2011

- ◆ Apache
 - ◆ 60.31%
- ◆ IIS (by Microsoft)
 - ◆ 19.34%
- ◆ nginx (by Igor Sysoev)
 - ◆ 7.65%
- ◆ GWS (by Google)
 - ◆ 5.09%
- ◆ lighttpd
 - ◆ 0.60%





Client-Server Architecture

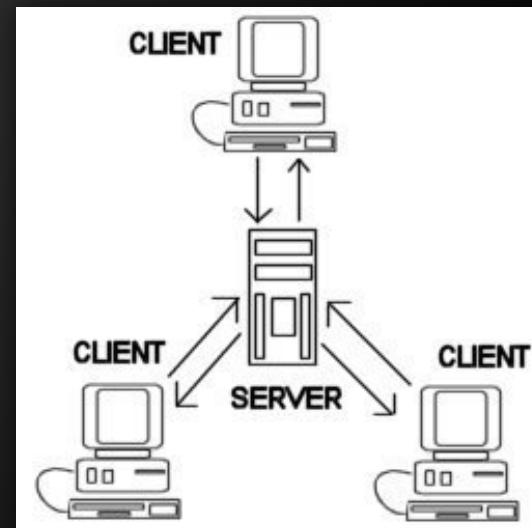
The Classical Client-Server Model

Client-Server Architecture

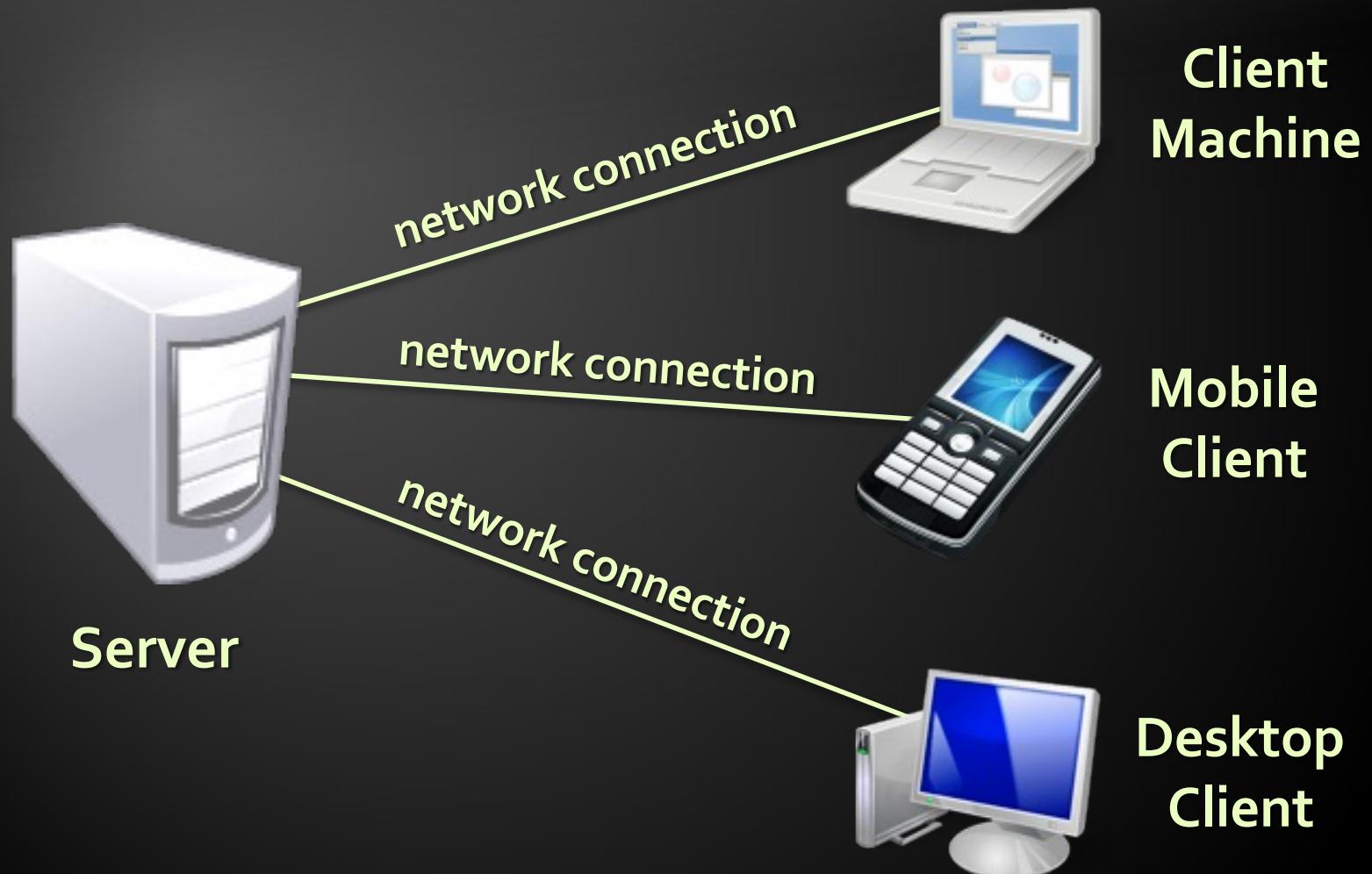
- ◆ The client-server model consists of:
 - Server – a single machine or cluster of machines that provides web applications (or services) to multiple clients
 - Examples:
 - Web server running PHP scripts or ASP.NET pages
 - IIS based Web server
 - WCF based service
 - Services in the cloud

Client-Server Architecture

- ◆ The client-server model consists of:
 - Clients –software applications that provide UI (front-end) to access the services at the server
 - Examples:
 - Web browsers
 - WPF applications
 - HTML5 applications
 - Silverlight applications
 - ASP.NET consuming services



The Client-Server Model

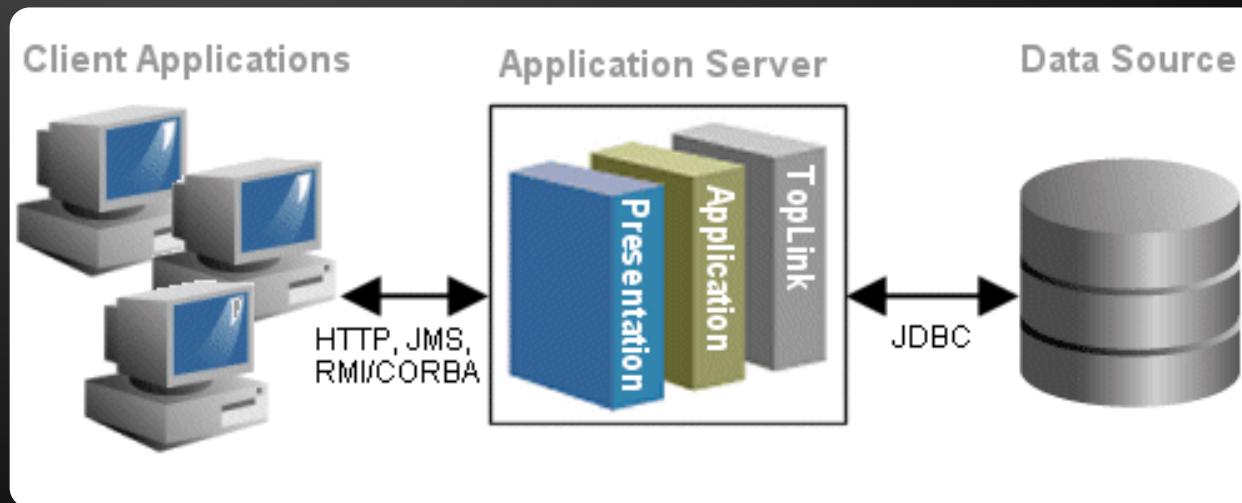


Client-Server Model – Examples

- ◆ Web server (Apache, IIS) – Web browser
- ◆ FTP server (ftpd) – FTP client (FileZilla)
- ◆ EMail server (qmail) – email client (Outlook)
- ◆ SQL Server – SQL Server Management Studio
- ◆ BitTorrent Tracker – Torrent client (μ Torrent)
- ◆ DNS server (bind) – DNS client (resolver)
- ◆ DHCP server (wireless router firmware) – DHCP client (mobile phone /Android DHCP client/)
- ◆ SMB server (Windows) – SMB client (Windows)

3-Tier / Multi-Tier Architectures

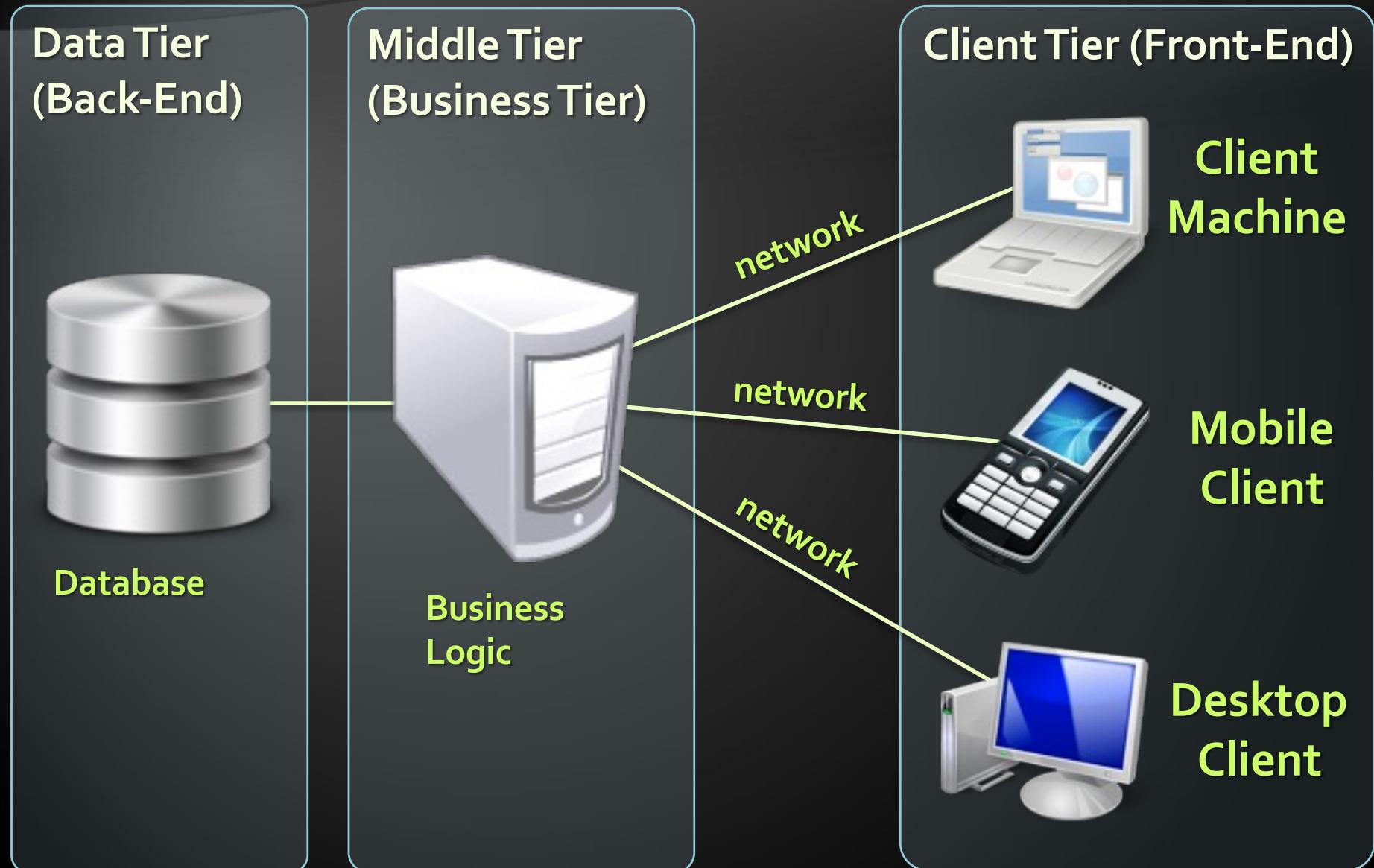
Classical Layered Structure of Software Systems



The 3-Tier Architecture

- ◆ The 3-tier architecture consists of the following tiers (layers):
 - Front-end (client layer)
 - Client software – provides the UI of the system
 - Middle tier (business layer)
 - Server software – provides the core system logic
 - Implements the business processes / services
 - Back-end (data layer)
 - Manages the data of the system (database / cloud)

The 3-Tier Architecture Model



Typical Layers of the Middle Tier

- ◆ The middle tier usually has parts related to the front-end, business logic and back-end:

Presentation Logic

Implements the UI of the application (HTML5, Silverlight, WPF, ...)



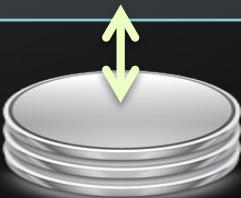
Business Logic

Implements the core processes / services of the application

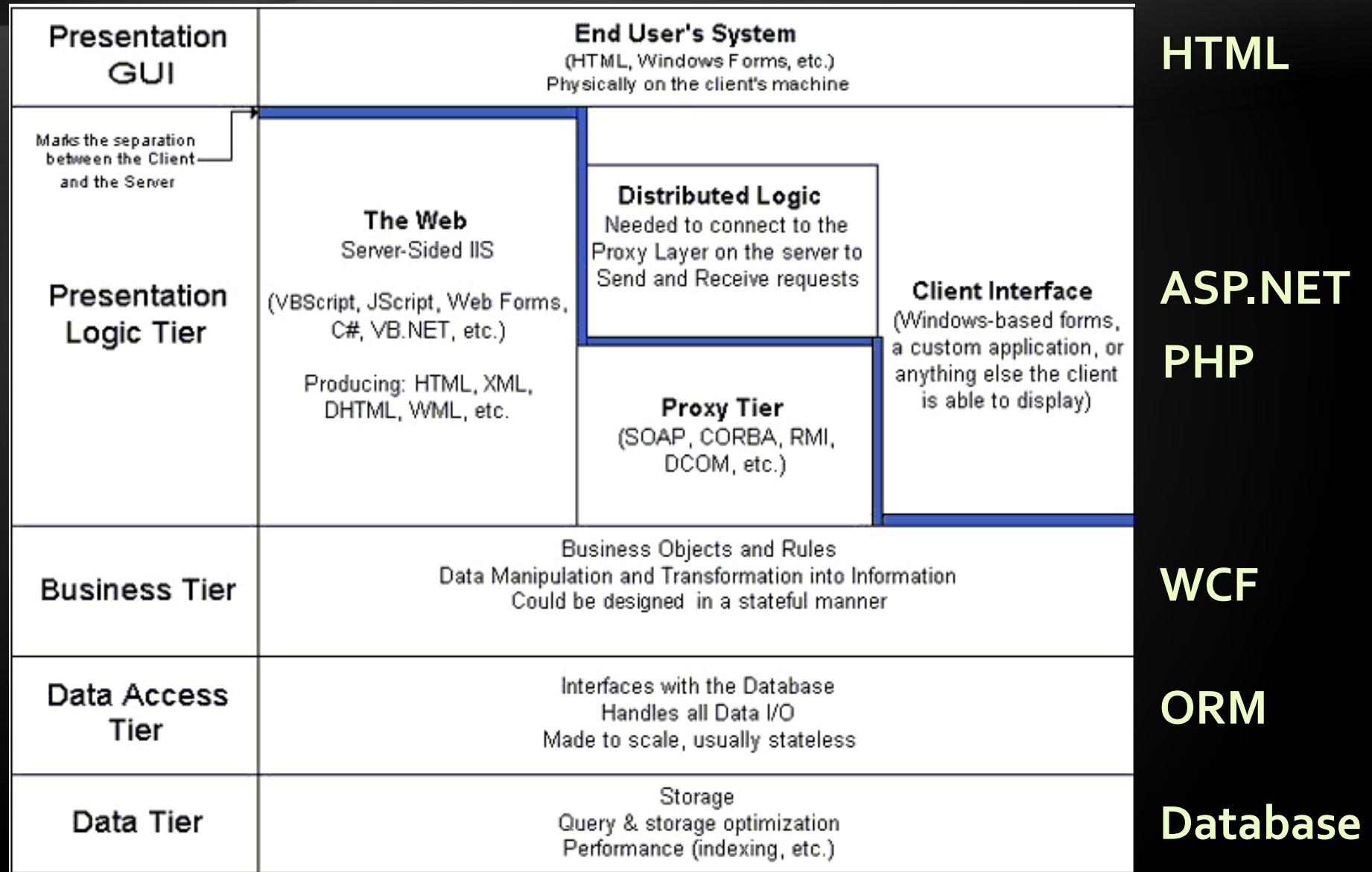


Data Access Logic

Implements the data access functionality (usually ORM framework)



Multi-Tier Architecture



Web Design Concepts

Questions?