SEC201.2 Web-Based Programming

What is a Web Application? - An Overview

Outline

- Web Applications: An Overview
  - Modern Web Applications
  - Historical Perspective to Web Applications
- Web Application Characteristics
  - Web 1.0, Web 2.0, Web 3.0
- What is a Web Application?
  - Web Apps – Architecture (Model)
  - Web Apps – Definition
  - Advantages and Disadvantages
- Evolution of Web Apps
- n-Tier Architecture

Web Applications: An Overview

- Modern Web Applications

  http://www.amazon.com/

  https://www.nytimes.com/

  https://tr.hotels.com/

  https://www.tripadvisor.com/

  http://www.uvelanghe.it/it/

  https://www.coursera.org/

  http://www.vakifbank.com.tr/

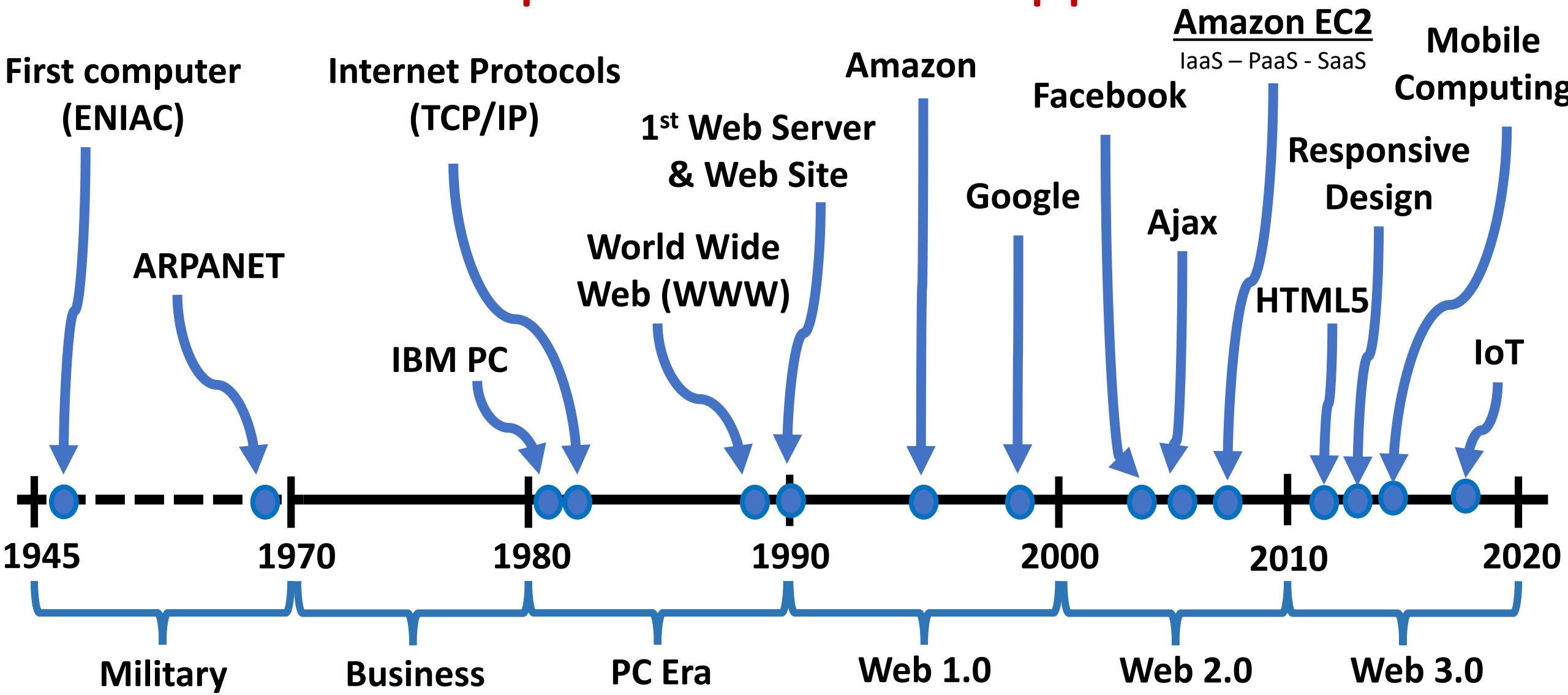
  https://www.bing.com/

  https://www.google.com.tr/

- Historical Perspectives

https://player.vimeo.com/video/2696386

Historical Perspectives to Web Applications



Outline

- Web Applications: An Overview
  - Modern Web Applications
  - Historical Perspective to Web Applications
- **Web Application Characteristics**
  - **Web 1.0, Web 2.0, Web 3.0**
- What is a Web Application?
  - Web Apps – Architecture (Model)
  - Web Apps – Definition
  - Advantages and Disadvantages
- Evolution of Web Apps
- n-Tier Architecture

Web 1.0 – Web Application Characteristics

- Static web pages
  - Websites that simply push information out with very little user interactivity
- The first web-based business models
  - First web browser called the World Wide Web, developed by Tim Berners-Lee
  - First graphical or GUI-based web browser → Mosaic
  - First Internet-based company → Netscape
  - Netscape Navigator → Mozilla Firefox
  - Browser Wars → Microsoft Internet Explorer vs. Netscape
    - A competition for dominance between Netscape and Microsoft that took place in the 1990s, eventually won by Microsoft

Web 2.0 – Web Application Characteristics

- Interactivity (Ajax)
  - What Ajax does? It allows for web pages, and thus, web applications, to change content dynamically, without the need to reload an entire page
  - First time, line blurring between web applications and desktop applications, in terms of the user experience

- Social networking, online commerce, wikis, lightweight collaboration

Web 3.0 – Web Application Characteristics

- Ubiquitous/Intelligent web
  - What intelligent web means is machine-facilitated understanding of information on the Worldwide Web
- Recommender systems, semantic web, mobile-friendly, Internet of things (IoT)

Web 2.0 and Web 3.0 Enablers

- JavaScript, XML, JSON (Ajax)
  - Ajax is a term, is no longer an acronym
  - It just means asynchronous delivery of content
- Web services interoperability (REST)
  - The ability to use services from other websites
- Cloud computing
  - The ability to have infrastructure, platforms, software as a service capabilities
- Powerful mobile platforms
- Metadata, linked data, machine processing by intelligent agents
- Web-enabled devices

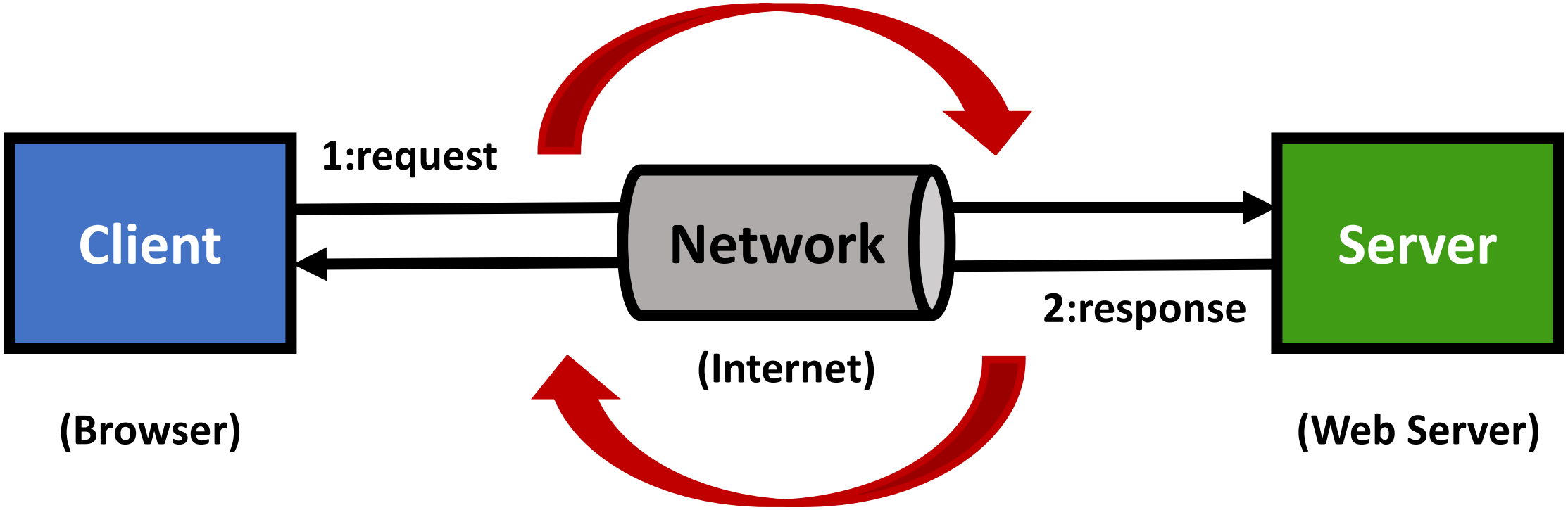
Outline

- Web Applications: An Overview
  - Modern Web Applications
  - Historical Perspective to Web Applications
- Web Application Characteristics
  - Web 1.0, Web 2.0, Web 3.0
- **What is a Web Application?**
  - **Web Apps – Architecture (Model)**
  - **Web Apps – Definition**
  - **Advantages and Disadvantages**
- Evolution of Web Apps
- n-Tier Architecture

Web Apps – Architecture (Model)

- Client-Server Architecture – the most basic model for describing the relationship between the cooperating programs in a networked software application
- Client-Server Architecture consists of two parts:
  1. Server – "listens" for requests, and provides services and/or resources according to those requests
  2. Client – establishes a connection to the server, and requests services and/or resources from the server

Hypertext Transfer Protocol (HTTP)



Request – Response Cycle

Web Apps – Definition

- Web Application –

A web application is *accessed by users via the Internet, using a browser* as the client, and consists of a collection of client-side and server-side scripts, HTML pages, and other resources that may be spread across multiple *servers, or throughout the World Wide Web (WWW)*

- Examples are
  - Web Mail
  - Online retail stores
  - Online banks
  - Online auctions
  - Wikis
  - Blogs
  - Document Storage

Web Apps and the Web (WWW)

- Web means World Wide Web
- A web application is an application that uses the World Wide Web
- What exactly is World Wide Web?
  - Word Wide Web (WWW) – a system of interlinked documents (web pages) that are accessed via the Internet using HTTP
  - Difference between the Internet and the World Wide Web → ***World Wide Web (web) operates on top of the Internet using HTTP***

- Web pages contain hypermedia, along with hyperlinks to other web pages

Hypermedia can be text, graphics, images, video, and other multimedia content

It's the hyperlinks that give the web its phenomenal structure

The structure of the Web is what makes it useful and gives it value!!!

What are the advantages of Web Apps?

Web Apps: Advantages –

- Most important advantage - ***Ubiquity and convenience of using a web browser as a client***
- Inherent cross-platform compatibility
- Update and maintain web apps without distributing and installing software on potentially thousands of client computers
- Reduction in IT costs

What are the disadvantages of Web Apps?

Web Apps: Disadvantages –

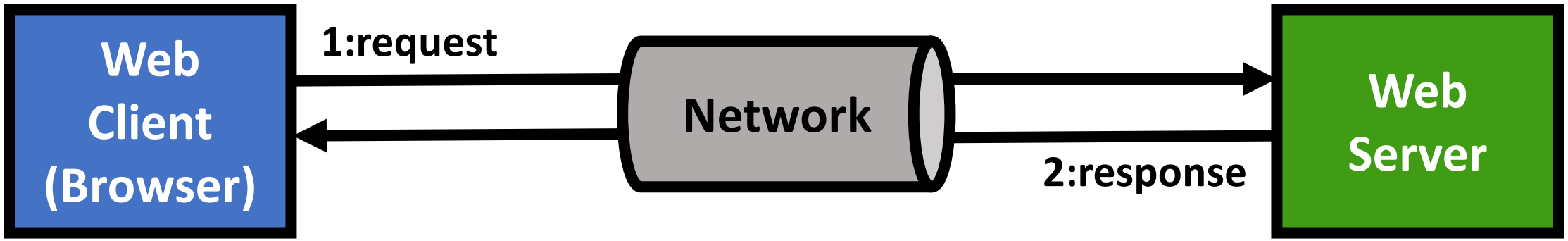
- User experience, as compared to desktop apps –

  historical: not the case any longer!!!
- Privacy and security issues associated with your data
- Programmer's perspective: web apps are difficult to develop and debug – there are a lot of moving parts!

Outline

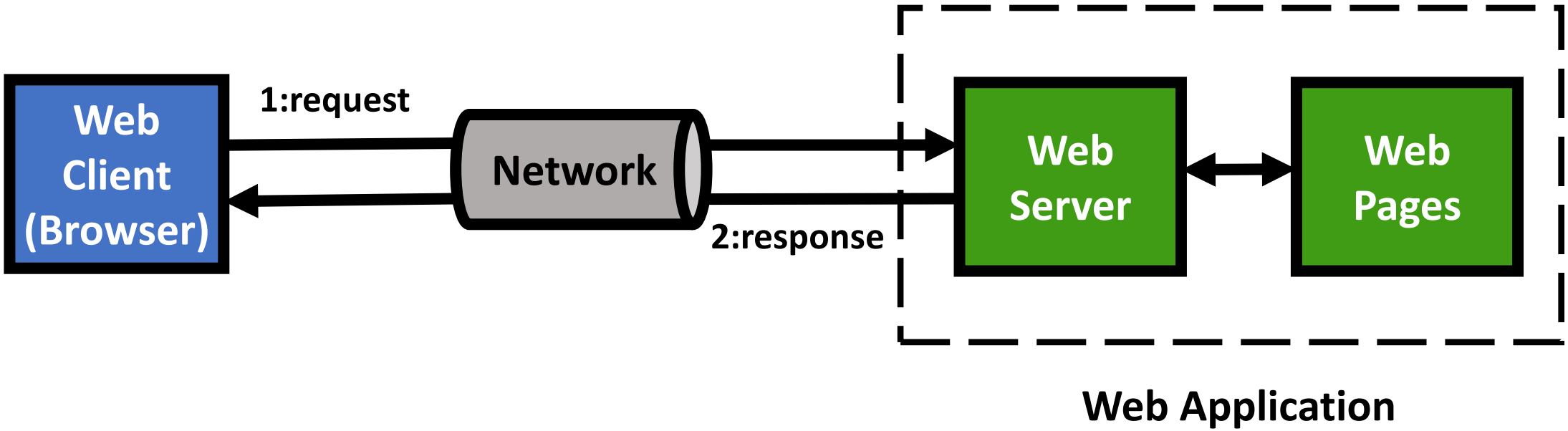
- Web Applications: An Overview
  - Modern Web Applications
  - Historical Perspective to Web Applications
- Web Application Characteristics
  - Web 1.0, Web 2.0, Web 3.0
- What is a Web Application?
  - Web Apps – Architecture (Model)
  - Web Apps – Definition
  - Advantages and Disadvantages
- **Evolution of Web Apps**
- n-Tier Architecture

Web 1.0 Architecture



Client-Server Architecture

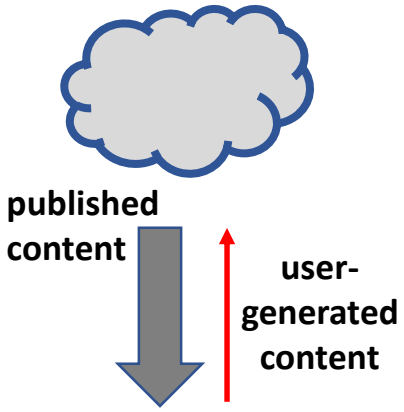
Web 1.0 Architecture



Web 1.0 Context

Web 1.0

100,000 websites
(read-only Web)

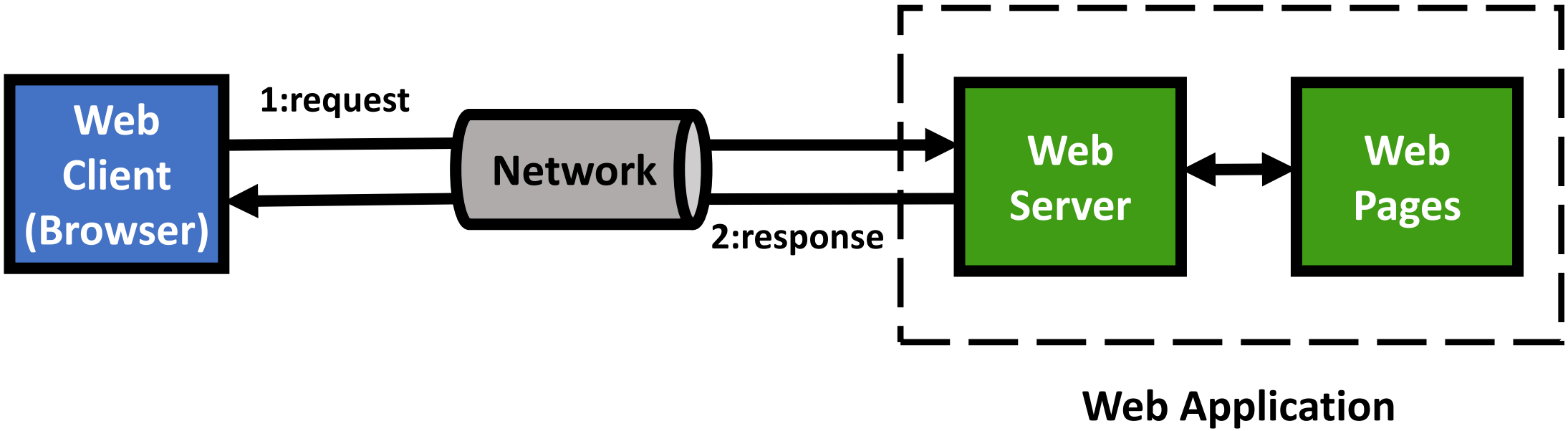


50,000,000 users

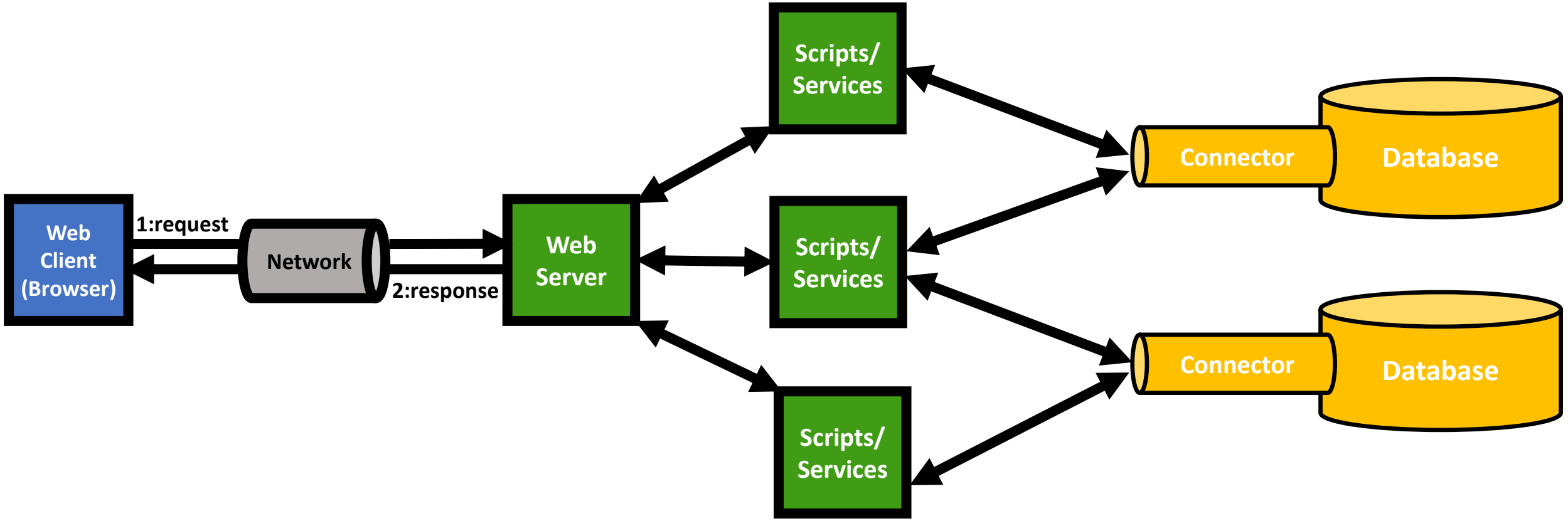
Web 1.0 Application Features

- Richer applications → more complicated server-side scripts → maintenance issues
- “Browser Wars” → more functionality on the client side → compatibility issues
- Developers began creating applications that were more interactive – requires saving state
- New technologies improved performance:
  - Client-side scripts
  - Cookies
  - Faster web servers
  - Web caching
  - CDNs (Content Distribution Network)
  - …

Web 2.0 and Web 3.0 Architectures



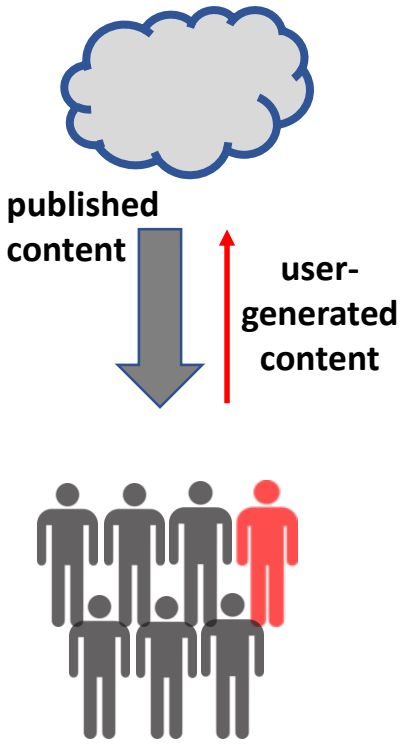
Web 2.0 and Web 3.0 Architectures



Evolution of Web Apps

Web 1.0

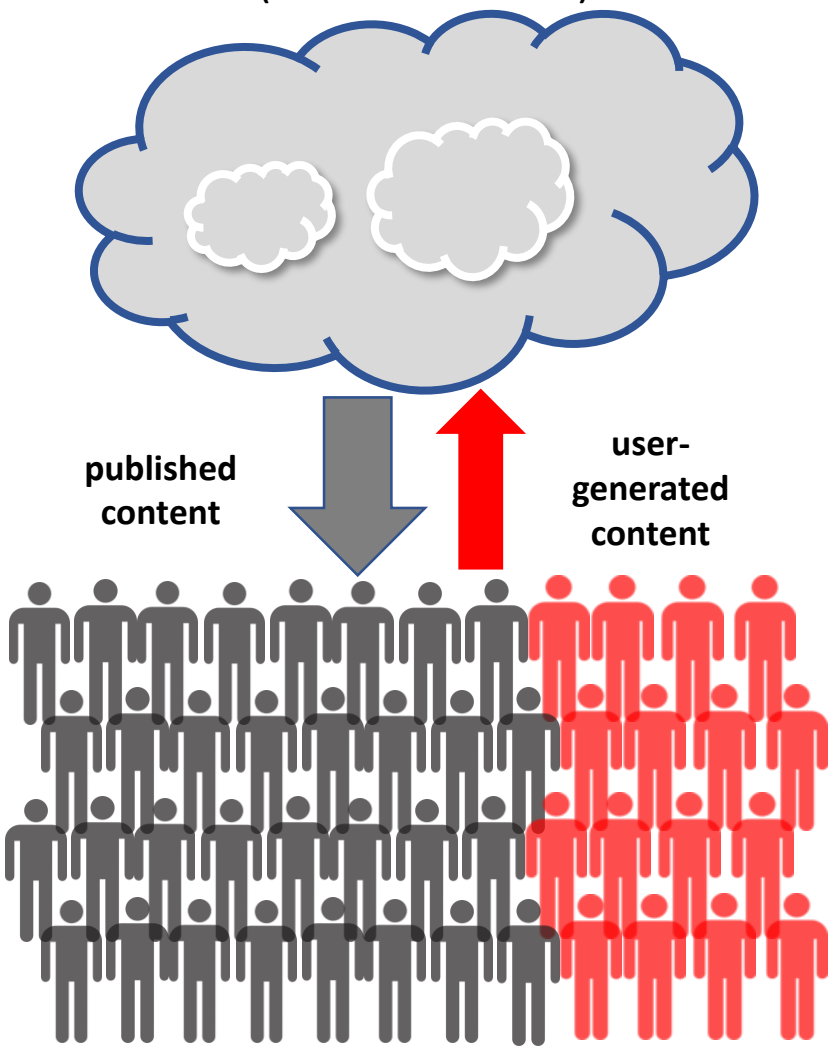
100,000 websites
(read-only Web)



50,000,000 users

Web 2.0

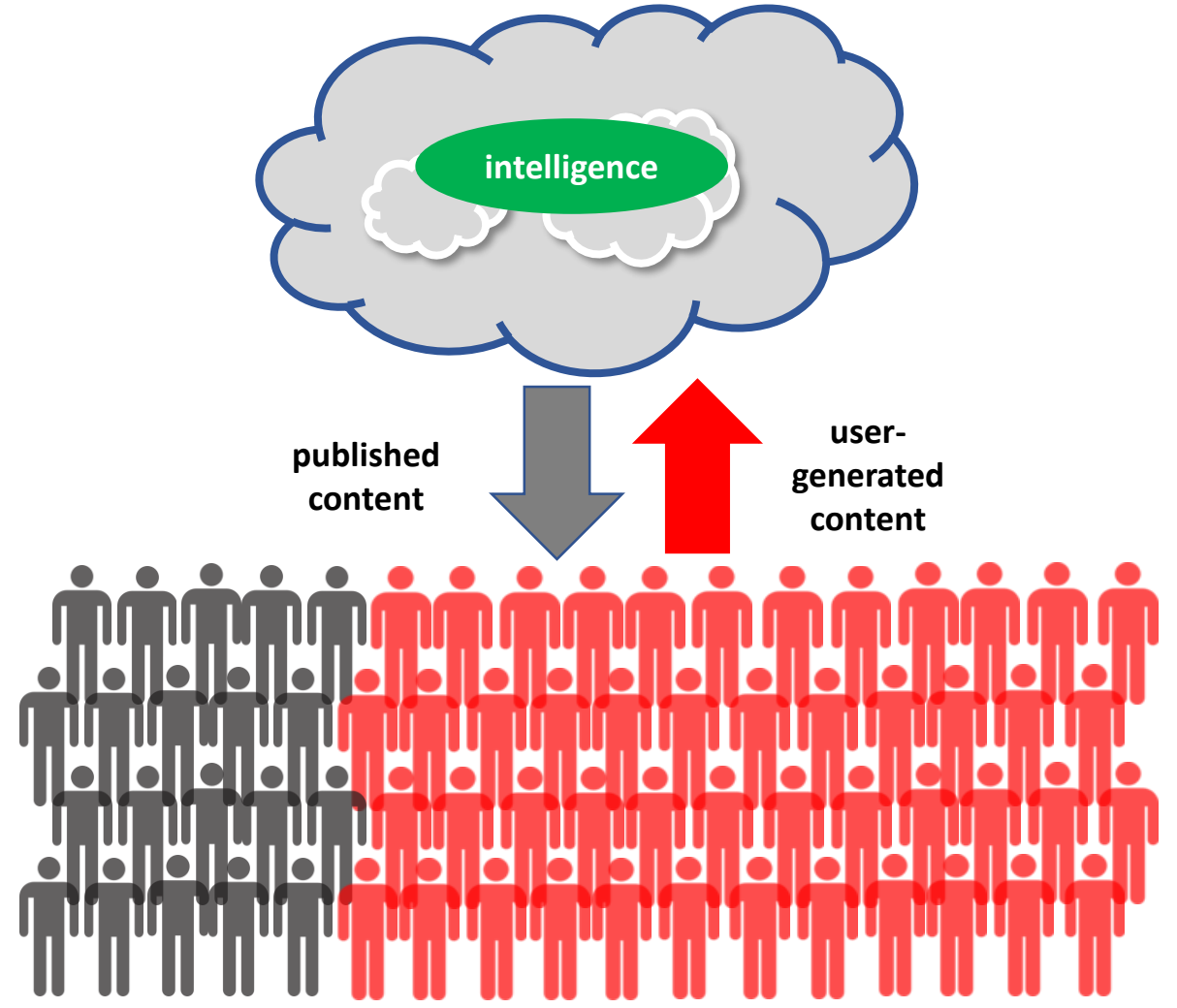
100,000,000 websites
(read-write Web)



1,000,000,000 users

Web 3.0

1,000,000,000 websites
(read-write Web)



2,500,000,000 users

Outline

- Web Applications: An Overview
  - Modern Web Applications
  - Historical Perspective to Web Applications
- Web Application Characteristics
  - Web 1.0, Web 2.0, Web 3.0
- What is a Web Application?
  - Web Apps – Architecture (Model)
  - Web Apps – Definition
  - Advantages and Disadvantages
- Evolution of Web Apps
- **n-Tier Architecture**

Design Patterns: An Overview

Complexity of Modern Web Apps

- Modern web apps involve a significant amount of complexity
- This makes developing, maintaining and extending a complex web application extremely difficult
- Using a foundation of solid design principles can simplify development and maintenance

Abstraction

- Software engineers use abstraction to manage complexity
- Abstraction involves representing the essential features of a software design or component, without including the background details
- Design Patterns provide useful abstractions for building software systems

What is a Design Pattern?

Definition:

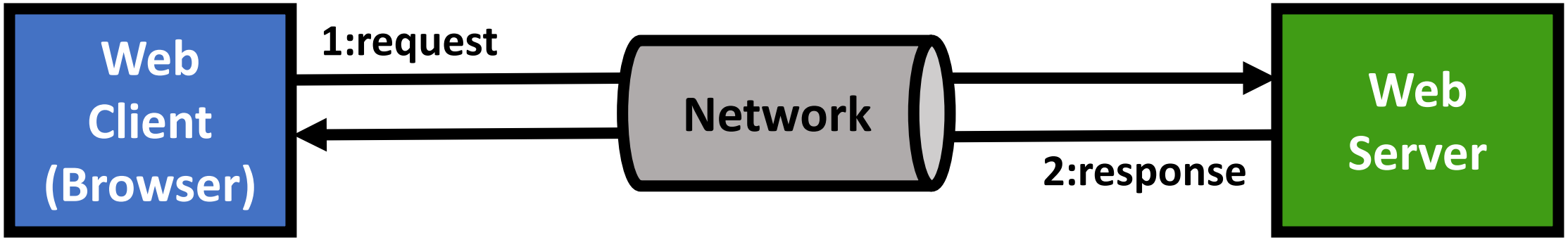
A ***design pattern*** is a reusable solution to a design problem that involves a set of components that interact to solve a general problem within a particular context

Design Patterns

- Abstract templates that can be applied over and over again in many different contexts
  - It's important to understand that design patterns are not actual code
  - They're design ideas that just commonly occur
- Well known design patterns are often used, alone or in combination, to simplify a complex design
- Design patterns provide a way to communication the parts of a design

Example: Client-Server Model

The client-server model is an ***architectural design pattern***



Features of the client-server model:

- A reusable template that can be applied over and over again
- Interacting components
- When I use the term “client-server”, you immediately have an idea of the design concept involved

n-Tier Architecture

A client-server architecture in which application functionality is further partitioned into separate tiers related to:

1. Presentation
2. Application processing
3. Data management

- What is the importance of tiers?
  - If they're designed properly, they support an important design principle known as **separation of concerns**

Separation of Concerns

- Each tier address a separate “concern”, encapsulated within a well-defined interface
- This allows each tier to be developed, modified or replaced, without affecting other tiers
- Encapsulation greatly simplifies development and maintenance of software systems

3-Tier Architecture: In General

- Presentation Tier – The user interface
- Data Tier – Persistent storage of data associated with the application
- Application (logic) Tier – Retrieves, modifies and/or deletes data in the data tier, and sends the results to the presentation tier. Also responsible for processing the data itself

3-Tier Web Application Architecture

Web Apps are often deployed as a 3-Tier Architecture:

- Presentation Tier – User's web browser
- Data Tier – A relational database
- Application (logic) Tier – The web server and logic associated with generating dynamic web content

6-Tier Web Application Architecture

The Presentation Tier is often subdivided into two tiers:

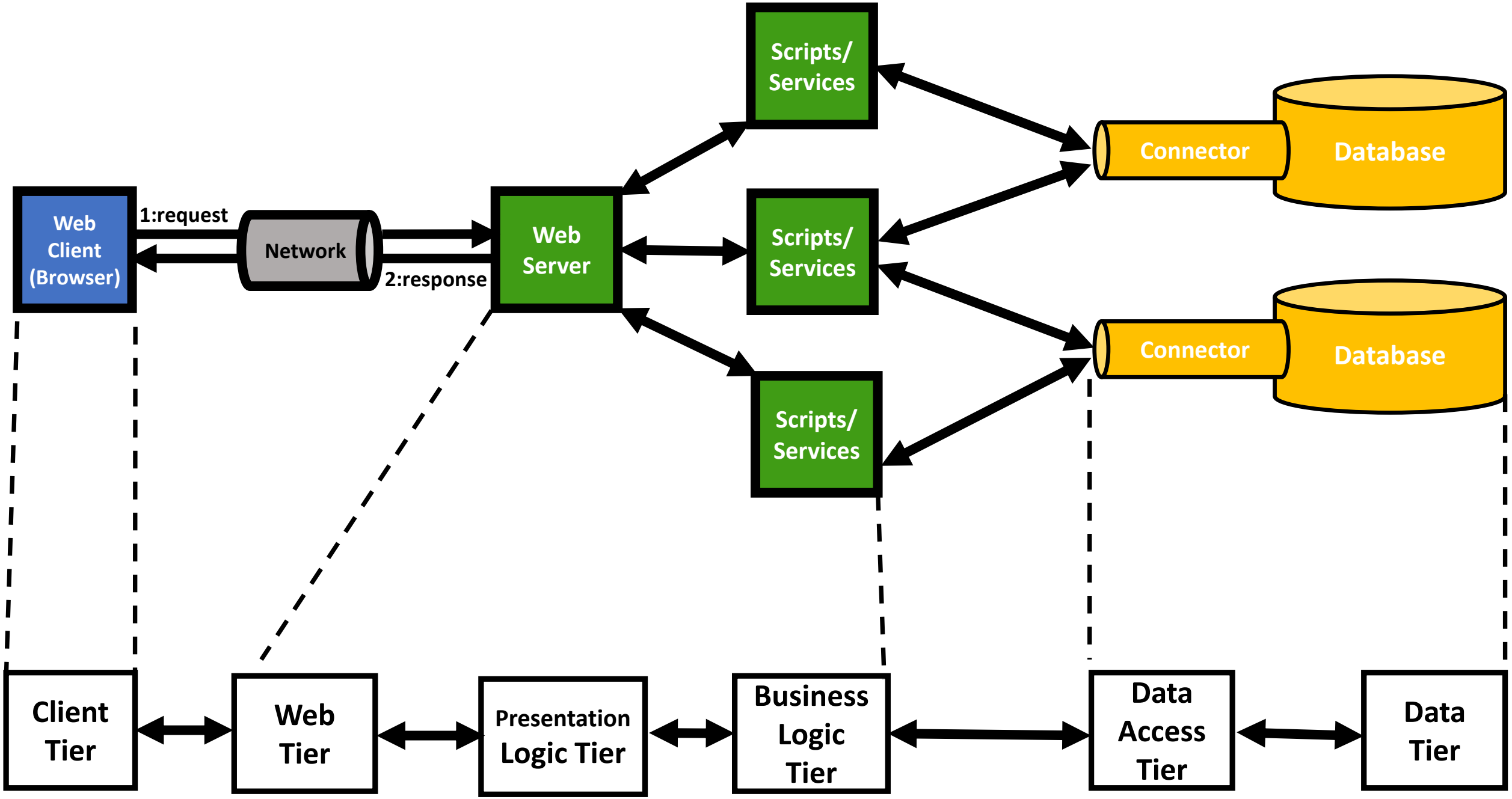
- ***Client Tier*** – client-side user interface components
- ***Presentation Logic Tier*** – server-side scripts for generating webpages

The Data Tier is often subdivided into two tiers:

- ***Data Tier*** – the data used by the application, a persistent data store of some type
- ***Data Access Tier*** – responsible for accessing data from the data tier, and passing it to the business logic tier

The Application Tier is often subdivided into two tiers:

- ***Business Logic Tier*** – models the business objects associated with the application
- ***Web Tier*** – the web server


Web Client (Browser)
1:request
Network
2:response
Web Server
Scripts/ Services
Scripts/ Services
Scripts/ Services
Connector
Database
Connector
Database
Client Tier
Web Tier
Presentation Logic Tier
Business Logic Tier
Data Access Tier
Data Tier