How the Web Works

Chapter 1

Objectives

1. Definition and History
2. Internet Protocols
3. Client-Server Model
4. Where is the Internet?
5. Domain Name System
6. Uniform Resource Locators(URL)
7. Hypertext Transfer Protocol(HTTP)
8. Web Servers

Definitions and history

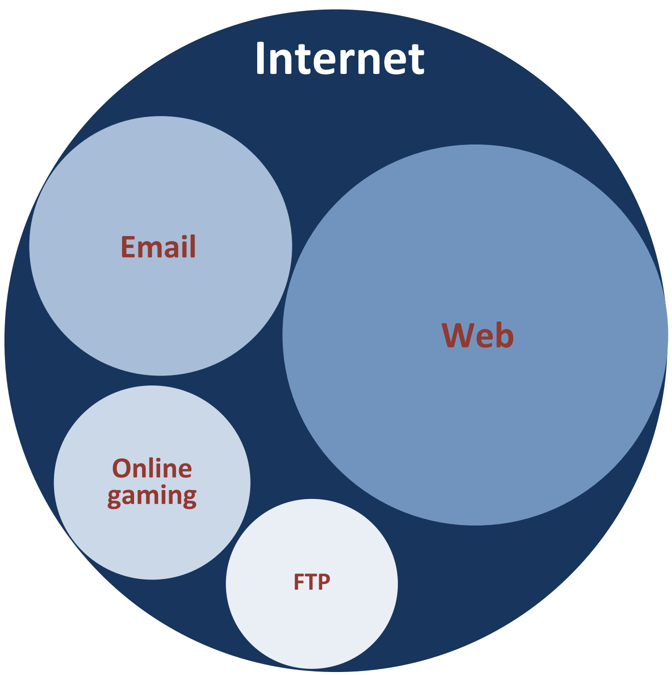
Section 1 of 8

Internet = Web?

The answer is no

The World-Wide Web (WWW or simply the Web) is certainly what most people think of when they see the word “internet.”

But the WWW is only a subset of the Internet.



Communication Definitions

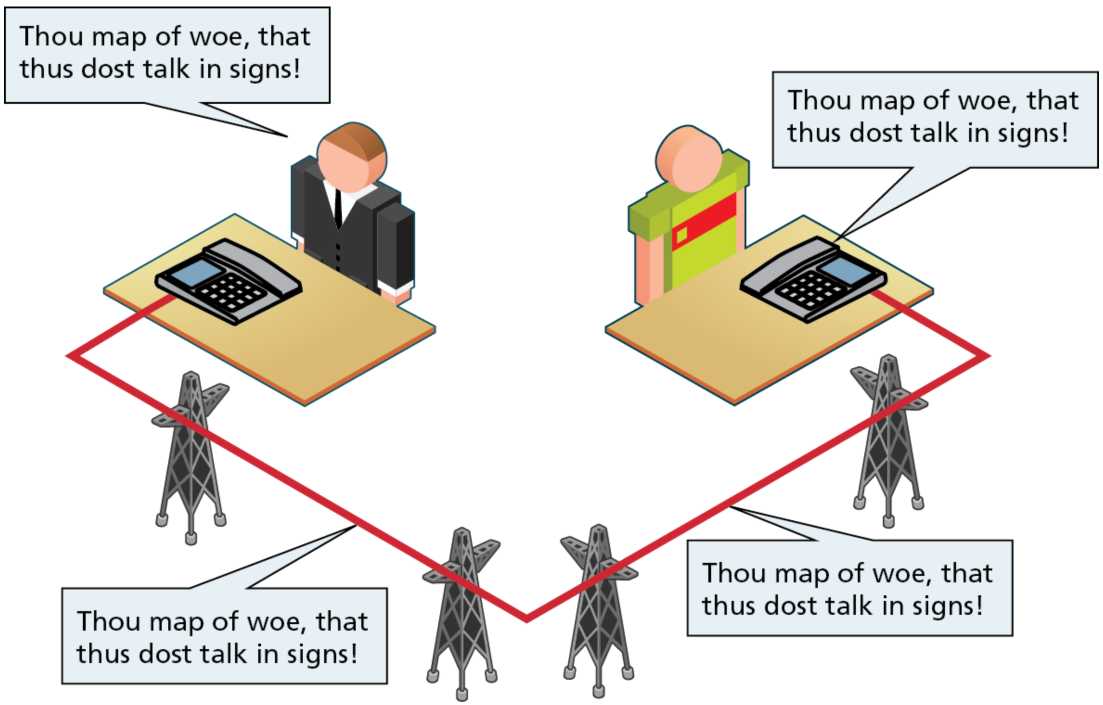
We will begin with the telephone

Telephone networks provide a good starting place to learn about modern digital communications.

In the telephone networks of old, calls were routed through operat ors who physically connected caller and receiver by connecting a w ire to a switchboard to complete the circuit.

Circuit Switching

A **circuit switching** establishes an actual physical connection between two people through a series of physical switches.



Circuit Switching
Its Limitations

Circuit Switching Weaknesses

- You must establish a link and maintain a dedicated circuit for the duration of the call
- Difficult to have multiple conversations simultaneously
- Wastes bandwidth since even the silences are transmitted

ARPANET

The beginnings of the Internet

The research network ARPANET was created. In the 1960s

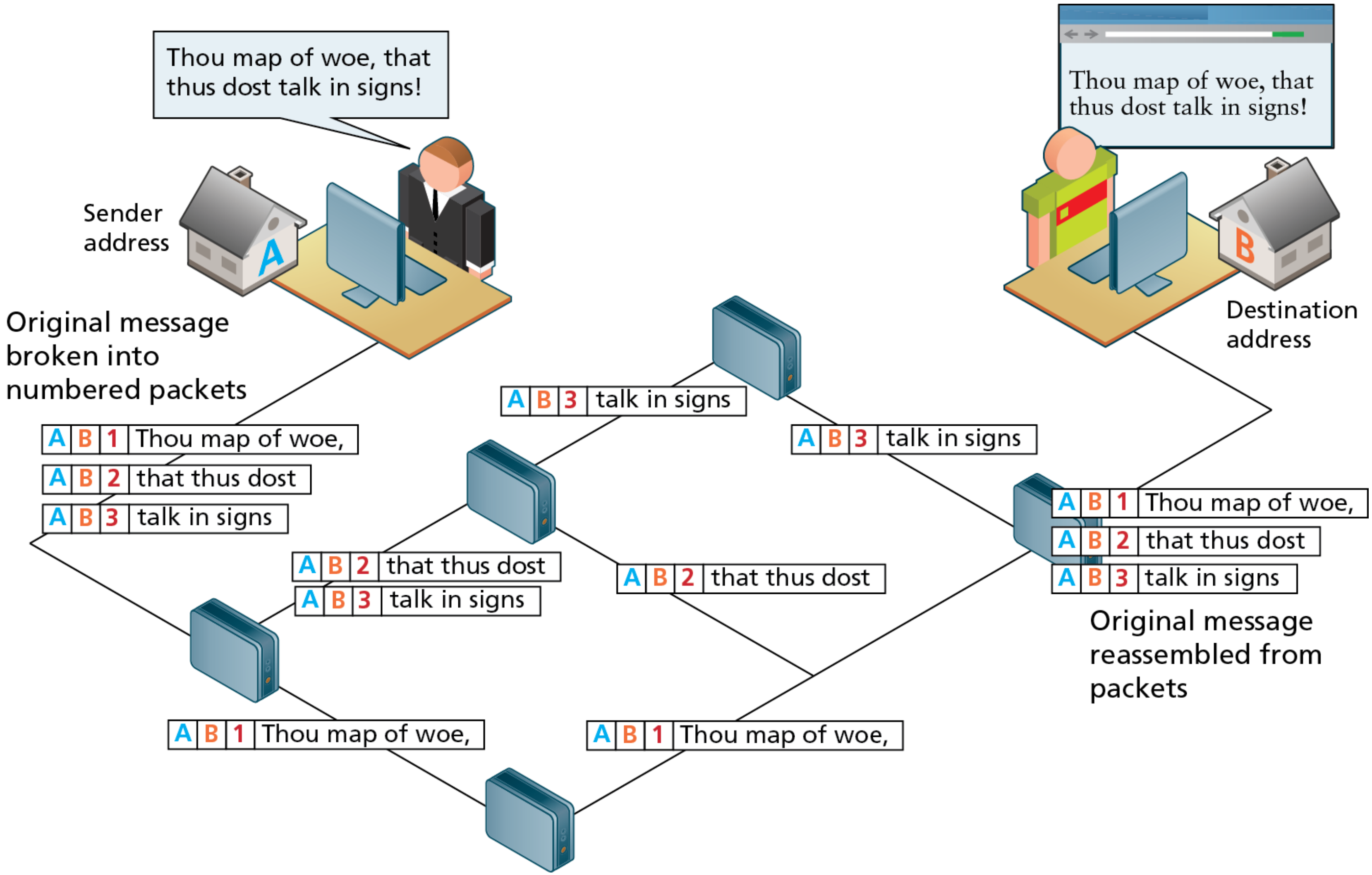
- ARPANET did not use circuit switching
- it used **packet switching**

A packet-switched network does not require a continuous connection.

Instead it splits the messages into smaller chunks called **packets** and ro utes them to the appropriate place based on the destination address.

The **packets** can take different routes to the destination.

Packet Switching



Packet Switching

Isn't this more complicated?

While **packet switching** may seem a more complicated and inefficient approach than **circuit switching**, it is:

- more robust (it is not reliant on a single pathway that may fail) and
- a more efficient use of network resources (since a circuit can communicate multiple connections).

Short History of the Internet

Perhaps not short enough

The early ARPANET network was funded and controlled by the Unite d States government and was used exclusively for academic and scie ntific purposes.

The early network started small with just a handful of connected ca mpuses in 1969 and grew to a few hundred by the early 1980s.

TCP/IP

Rides to the rescue

To promote the growth and unification of the disparate networks a suite of protocols was invented to unify the networks together.

By 1981, new networks built in the US began to adopt the **TCP/IP** (**Transmission Control Protocol / Internet Protocol**) communicatio n model (discussed in the next section), while older networks were transitioned over to it.

Tim Berners-Lee

The invention of the WWW is usually attributed to the British Tim Berners-Lee, who, along with the Belgian Robert Cailliau, publish ed a proposal in 1990 for a hypertext system while both were wor king at CERN in Switzerland.

Core Features of the Web

Shortly after that initial proposal Berners-Lee developed the main features of the web:

1. A URL to uniquely identify a resource on the WWW.

2. The HTTP protocol to describe how requests and responses o perate.

3. A software program (later called web server software) that ca n respond to HTTP requests.

4. HTML to publish documents.

5. A program (later called a browser) to make HTTP requests fro m URLs and that can display the HTML it receives.

W3C

The World Wide Web Consortium

Also in late 1994, Berners-Lee helped found the **World Wide Web Consortium (W3C)**, which would soon become the international st andards organization that would oversee the growth of the web.

This growth was very much facilitated by the decision of CERN to n ot patent the work and ideas done by its employee and instead left the web protocols and code-base royalty free.

Web Apps Compared to Desktop Apps

First the advantages of web apps

Some of the advantages of web applications include:

- Accessible from any internet-enabled computer.
- Usable with different operating systems and browser platforms.
- Easier to roll out program updates since only need to update software on server and not on every desktop in organization.
- Centralized storage on the server means fewer concerns about local st orage (which is important for sensitive information such as health car e data).

Web Apps Compared to Desktop Apps

Now the disadvantages of web apps

Some of the disadvantages of web applications include:

Requirement to have an active internet connection (the internet is not always available everywhere at all times).

Security concerns about sensitive private data being transmitted over the internet.

Concerns over the storage, licensing and use of uploaded data.

Problems with certain websites on certain browsers not looking quite right.

Limited access to the operating system can prevent software and hardware from being installed or accessed (like Adobe Flash on iOS).

What is an "Intranet"?

A short digression

One of the more common terms you might encounter in web develop ment is the term "**intranet**" (with an "**a**"), which refers to an internet n etwork that is local to an organization or business.

Intranet resources are often private, meaning that only employees (or authorized external parties such as customers or suppliers) have access to those resources.

Thus Internet (with an "e") is a broader term that encompasses both p rivate (intranet) and public networked resources.

What is an "Intranet"?

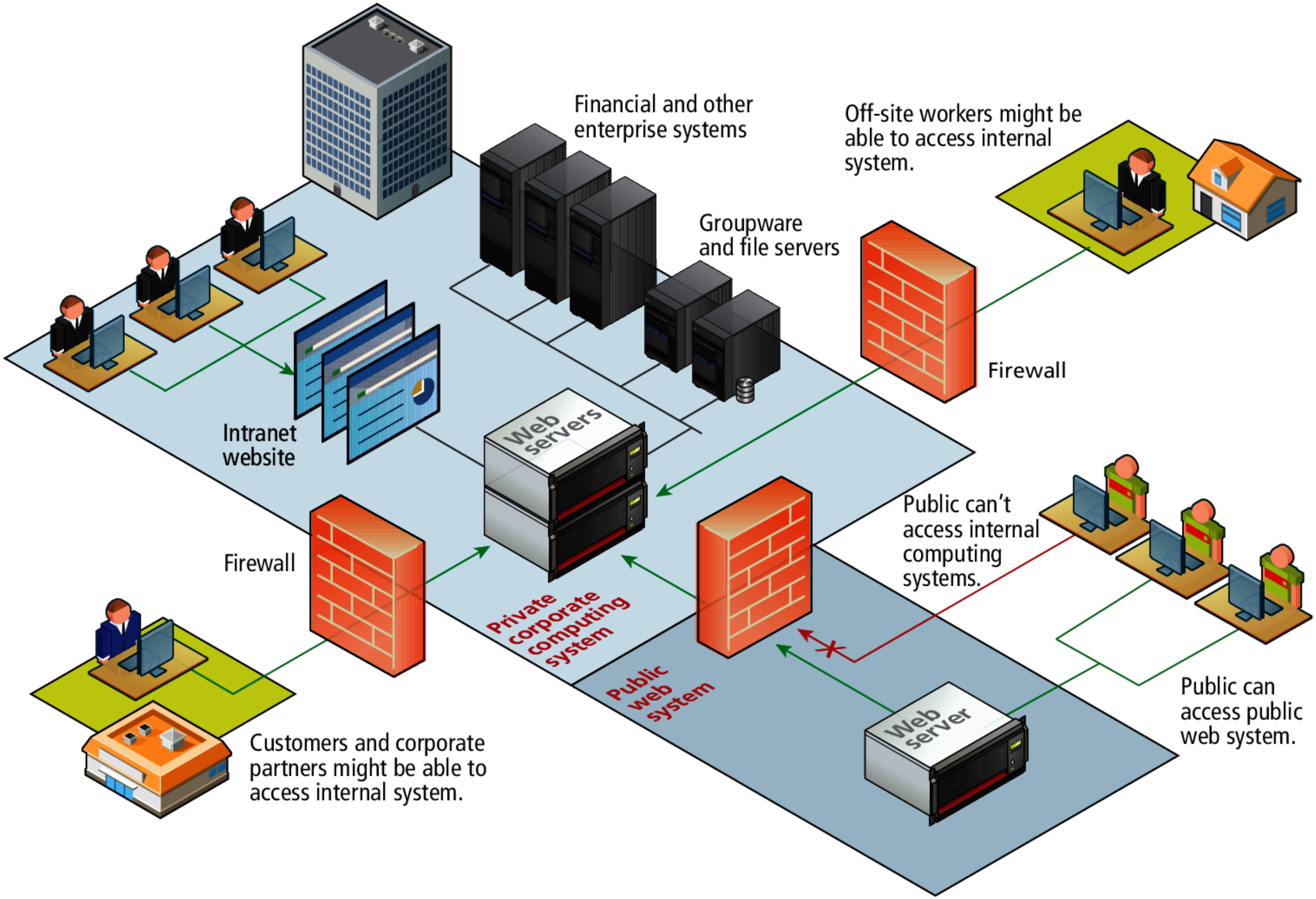
Intranets are typically protected from unauthorized external access via security features such as firewalls or private IP ranges.

Because intranets are private, search engines such as Google have limited or no access to content within a private intranet.

Due to this private nature, it is difficult to accurately gauge, for inst ance, how many web pages exist within intranets, and what techn ologies are more common in them.

- Some especially expansive estimates guess that almost half of all we b resources are hidden in private intranets.

Intranet versus Internet



Intranets and the Job Market

Being aware of intranets is also important when one considers t he job market and market usage of different web technologies.

If one focuses just on the public internet, it will appear that, for instance, PHP, MySQL, and WordPress, are absolutely dominant in their market share.

But when one adds in the private world of corporate intranets, other technologies such as ASP.NET, JSP, SharePoint, Oracle, SAP, and IBM WebSphere, are just as important.

Static Web Sites

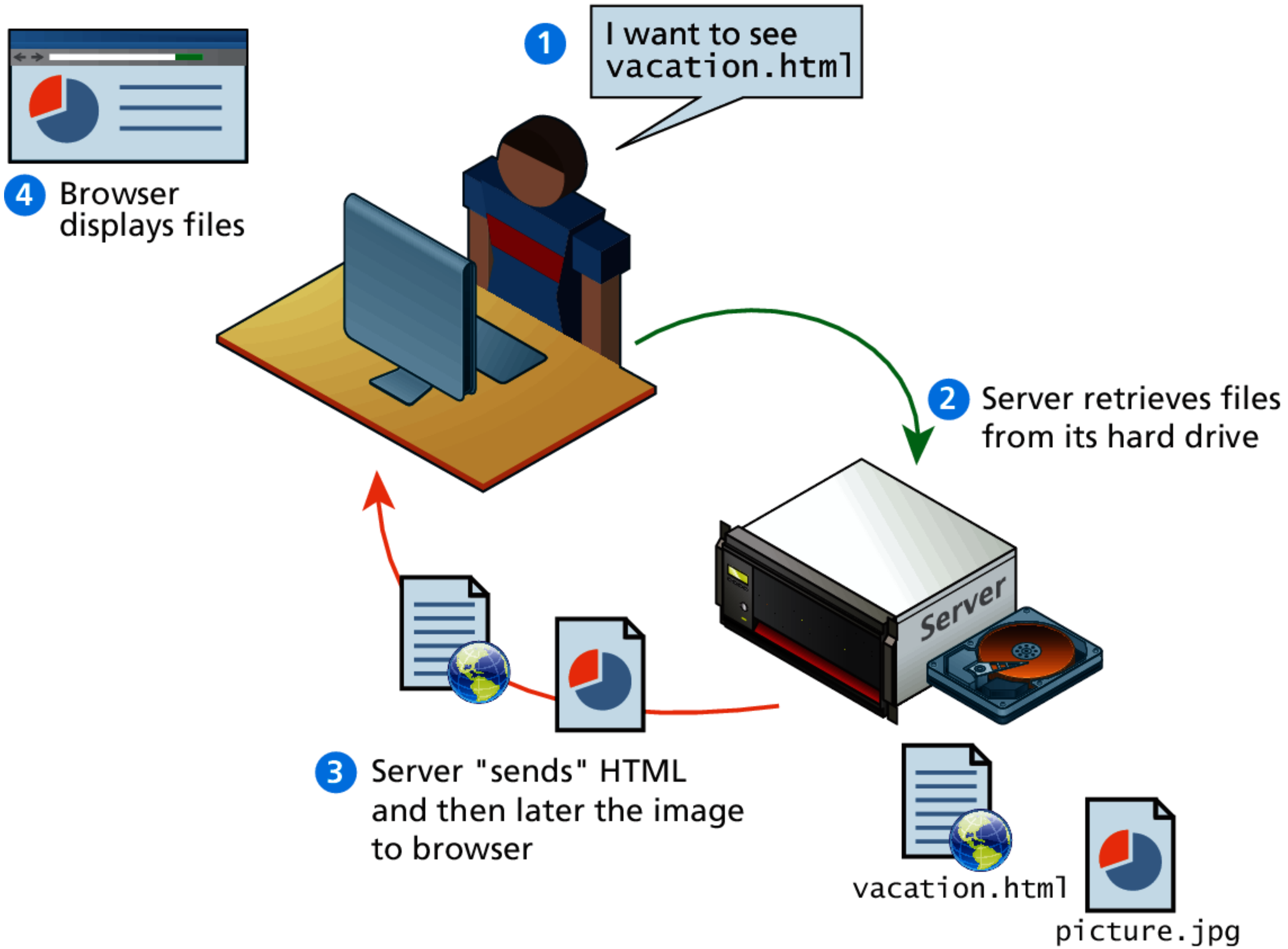
Partying Like It's 1995

In the earliest days of the web, a **webmaster** (the term popular in the 1990s for the person who was responsible for creating and supportin g a web site) would publish web pages, and periodically update them.

In those early days, the skills needed to create a web site were pretty basic: one needed knowledge of the HTML markup language and per haps familiarity with editing and creating images.

This type of web site is commonly referred to as a **static web site**, in t hat it consists only of HTML pages that look identical for all users at al l times.

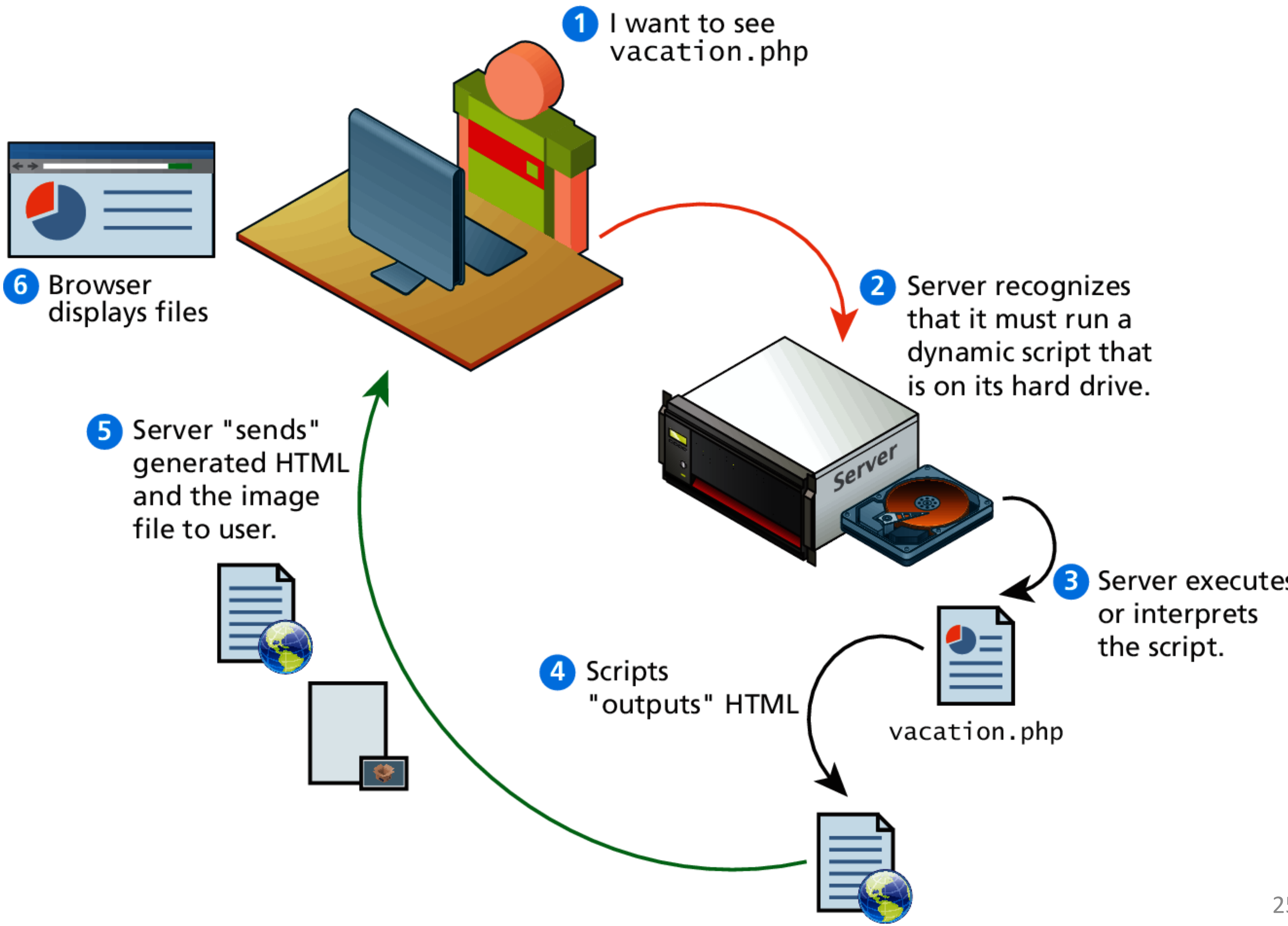
Static Web Sites



Dynamic Web Sites

Within a few years of the invention of the web, sites began to get more complicated as more and more sites began to use programs running on web servers to generate content dynamically.

Dynamic Web Sites



Dynamic Web Sites

These server-based programs would read content from databases, interface with existing enterprise computer systems, communicate with financial institutions, and then output HTML that would be se nt back to the users' browsers.

This type of web site is called here in this book a **dynamic web site** because the page content is being created at run-time by a progra m created by a programmer; this page content can vary for user to user.

Web 2.0 and Beyond

In the mid 2000s, a new buzz-word entered the computer lexicon: **web 2.0**.

This term had two meanings, one for users and one for developers.

Web 2.0

Its meaning for users

For the users, Web 2.0 referred to an interactive experience where users could contribute *and* consume web content, thus creating a m ore user-driven web experience.

Web 2.0

Its meaning for developers

For software developers, Web 2.0 also referred to a change in the paradigm of how dynamic web sites are created.

Programming logic, which previously existed only on the server, be gan to migrate to the browser.

This required learning Javascript, a rather tricky programming lang uage that runs in the browser, as well as mastering the rather diffic ult programming techniques involved in asynchronous communicat ion.

Internet protocols

Section 2 of 8

What’s a Protocol?

The internet exists today because of a suite of interrelated communications protocols.

A **protocol** is a set of rules that partners in communication use when they communicate.

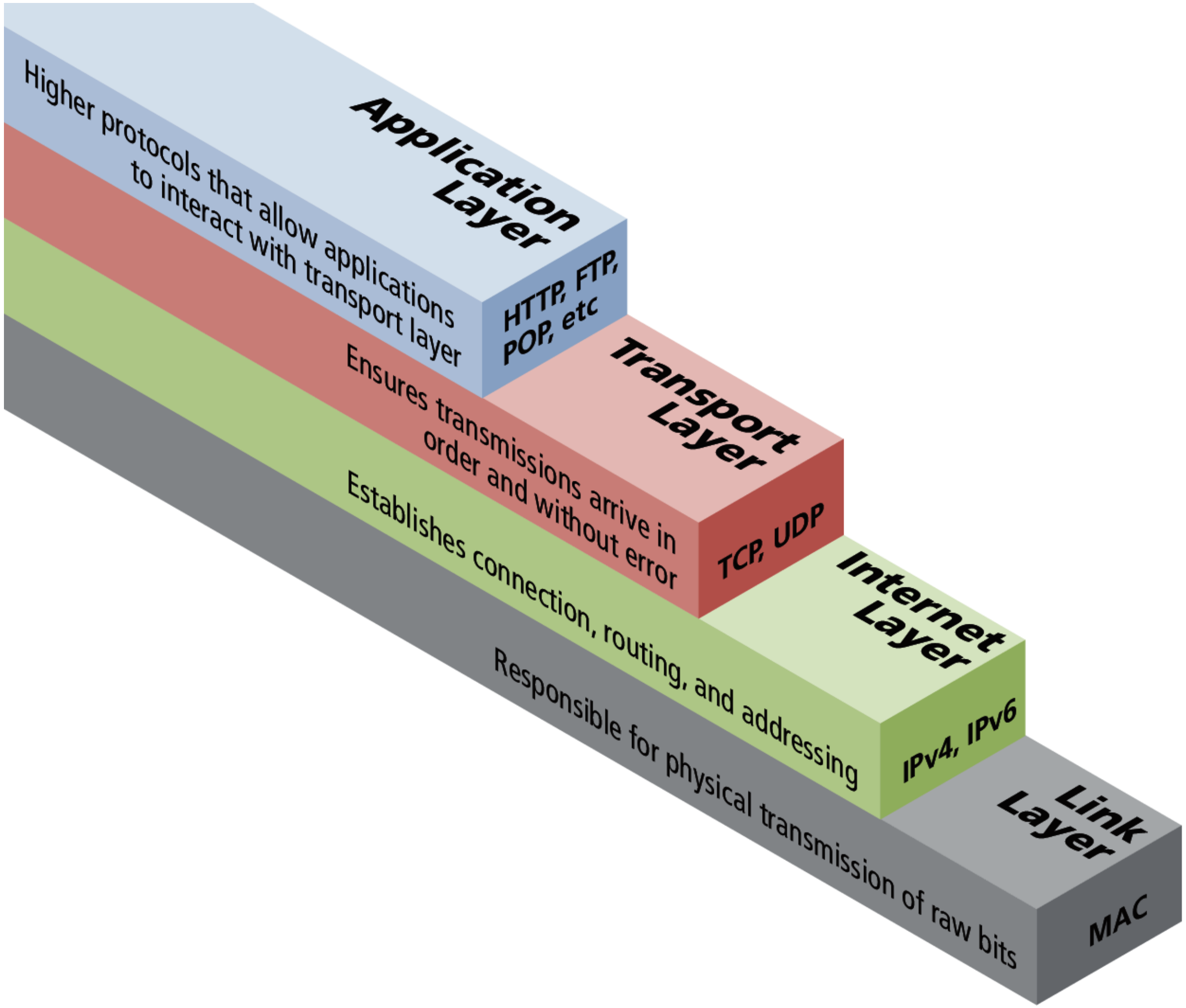
A Layered Architecture

The TCP/IP Internet protocols were originally abstracted as a four -layer stack.

Later abstractions subdivide it further into five or seven layers.

Since we are focused on the top layer anyhow, we will use the ea rliest and simplest **four-layer network model**.

Four Layer Network Model



Link Layer

Save this for your networking course

The **link layer** is the lowest layer, responsible for both the physical transmission across media (wires, wireless) and establishing logical links.

It handles issues like packet creation, transmission, reception and error detection, collisions, line sharing and more.

Internet Layer

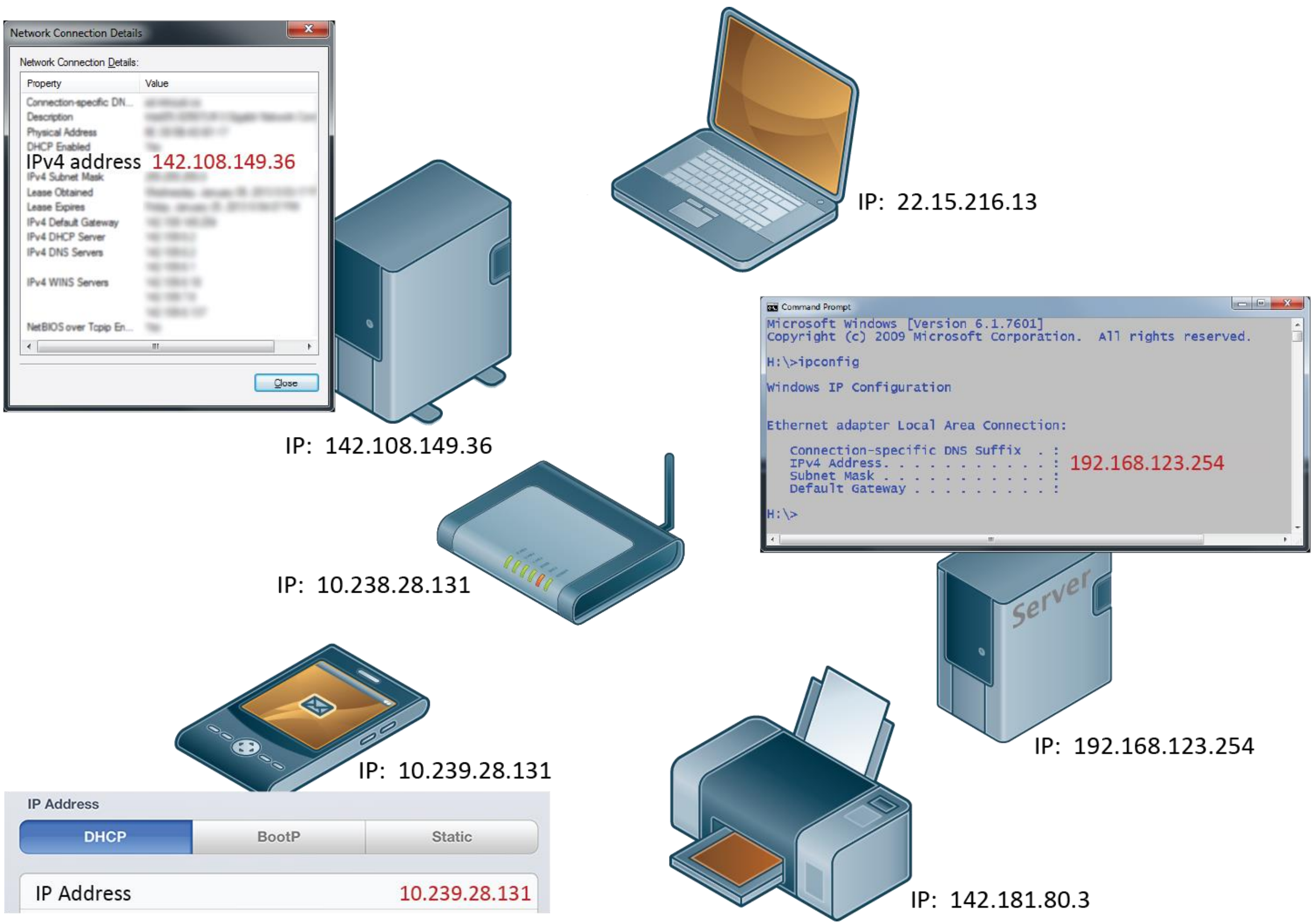
The **internet layer** (sometimes also called the IP Layer) routes packets between communication partners across networks.

Internet Protocol (IP)

The Internet uses the **Internet Protocol** (**IP**) addresses to identify destinations on the Internet.

Every device connected to the Internet has an **IP address**, which i s a numeric code that is meant to uniquely identify it.

IP addresses and the Internet



IP Addresses

Two types

IPv4 addresses are the IP addresses from the original TCP/IP protocol.

In IPv4, 12 numbers are used (implemented as four 8-bit integers), wri tten with a dot between each integer.

Since an unsigned 8-bit integer's maximum value is 255, four integers together can encode approximately 4.2 billion unique IP addresses.

IP Addresses

Two types

To future proof the Internet against the 4.2 billion limit, a new version of the IP protocol was created, **IPv6**.

This newer version uses eight 16-bit integers for 2^{128} unique addresses, over a billion billion times the number in IPv4.

These 16-bit integers are normally written in hexadecimal, due to their longer length.

IPv4

2^{32} addresses

4 - 8 bit components
(32 bits)

`192.168.123.254`

IPv6

2^{128} addresses

8 - 16 bit components
(128 bits)

`3fae:7a10:4545:9:291:e8ff:fe21:37ca`

IP Addresses

Inside of networks is different

Your IP address will generally be assigned to you by your Internet Service Provider (ISP).

In organizations, large and small, purchasing extra IP addresses fr om the ISP is not cost effective.

In a local network, computers can share a single IP address betwe en them.

Transport Layer

The **transport layer** ensures transmissions arrive, in order, and without error.

This is accomplished through a few mechanisms.

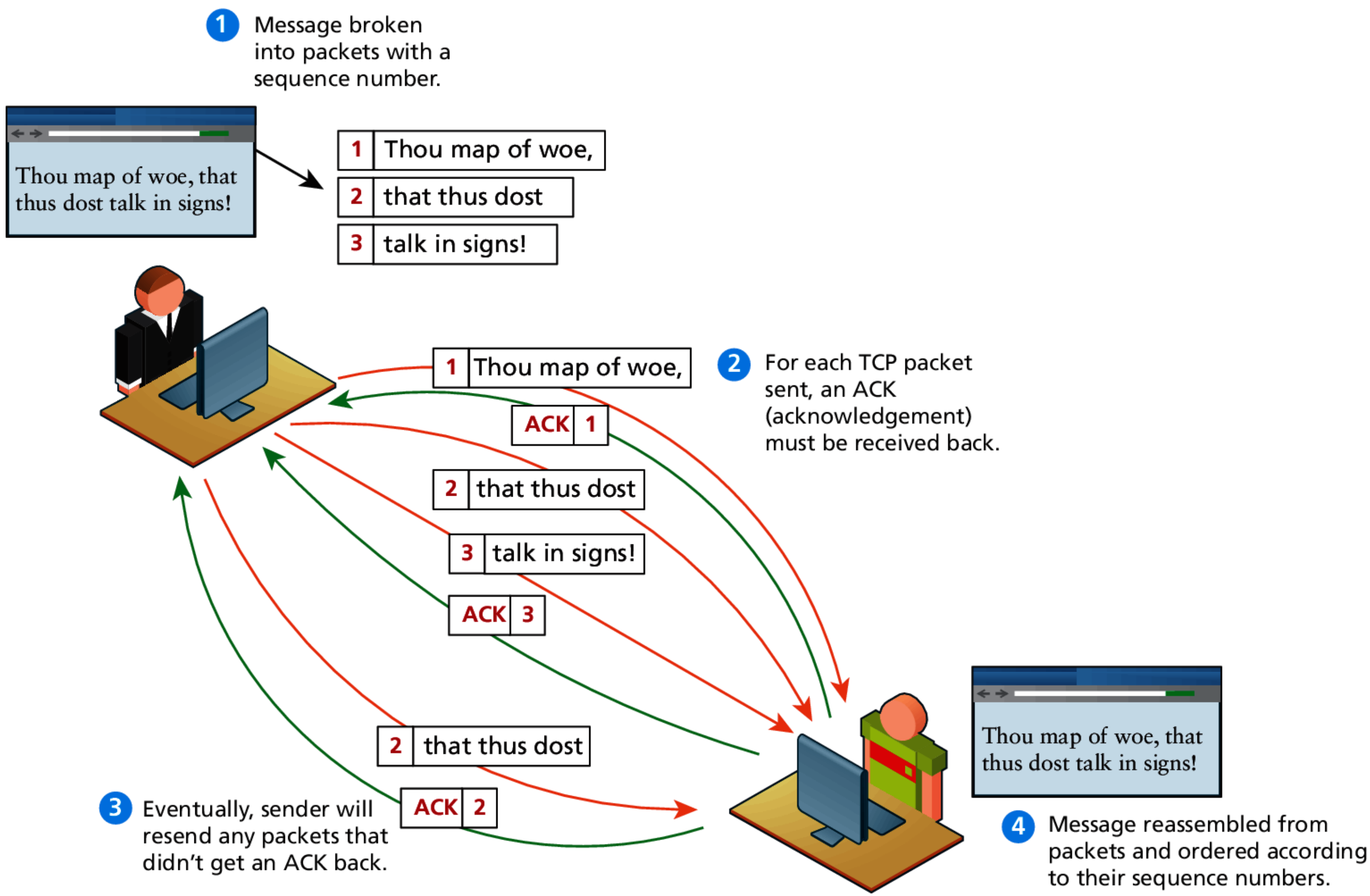
First, the data is broken into packets formatting according to the **Transmission Control Protocol** (**TCP**).

Transport Layer

Secondly, each packet is acknowledged back to the sender so in the event of a lost packet, the transmitter will realize a packet h as been lost since no ACK arrived for that packet.

That packet is retransmitted, and although out of order, is reord ered at the destination.

TCP Packets



1 Message broken into packets with a sequence number.
Thou map of woe, that thus dost talk in signs!
1 Thou map of woe,
2 that thus dost
3 talk in signs!
1 Thou map of woe,
ACK 1
2 that thus dost
3 talk in signs!
ACK 3
2 For each TCP packet sent, an ACK (acknowledgement) must be received back.
2 that thus dost
ACK 2
3 Eventually, sender will resend any packets that didn't get an ACK back.
Thou map of woe, that thus dost talk in signs!
4 Message reassembled from packets and ordered according to their sequence numbers.

Application Layer

With the **application layer**, we are the level of protocols familiar to most web developers.

Application layer protocols implement process-to-process commun ication and are at a higher level of abstraction in comparison to the low-level packet and IP addresses protocols in the layers below it.

Examples: HTPP, SSH, FTP, DNS, POP, SMTP.

Client-Server MODEL

Section 3 of 8

Client-Server Model

The web is sometimes referred to as a client-server model of commu nications.

In the **client-server model**, there are two types of actors: clients and servers.

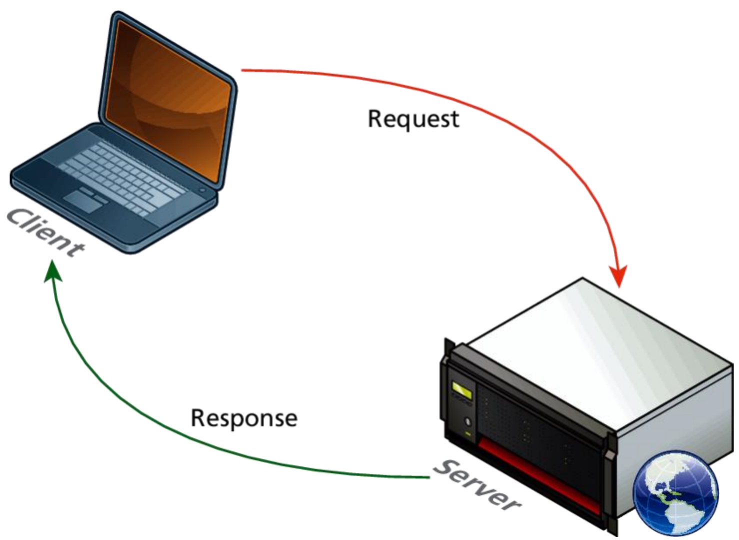
The **server** is a computer agent that is normally active 24 hours a day , 7 days a week (or simply 24/7), listening for queries from any client who make a request.

A **client** is a computer agent that makes requests and receives respo nses from the server, in the form of response codes, images, text file s, and other data.

Request-Response Loop

Within the client-server model, the **request-response loop** is the most basic mechanism on the server for receiving requests and transmitting data in response.

The client initiates a **request** to a server and gets a **response** that could include some resource like an HTML file, an image or some other data.

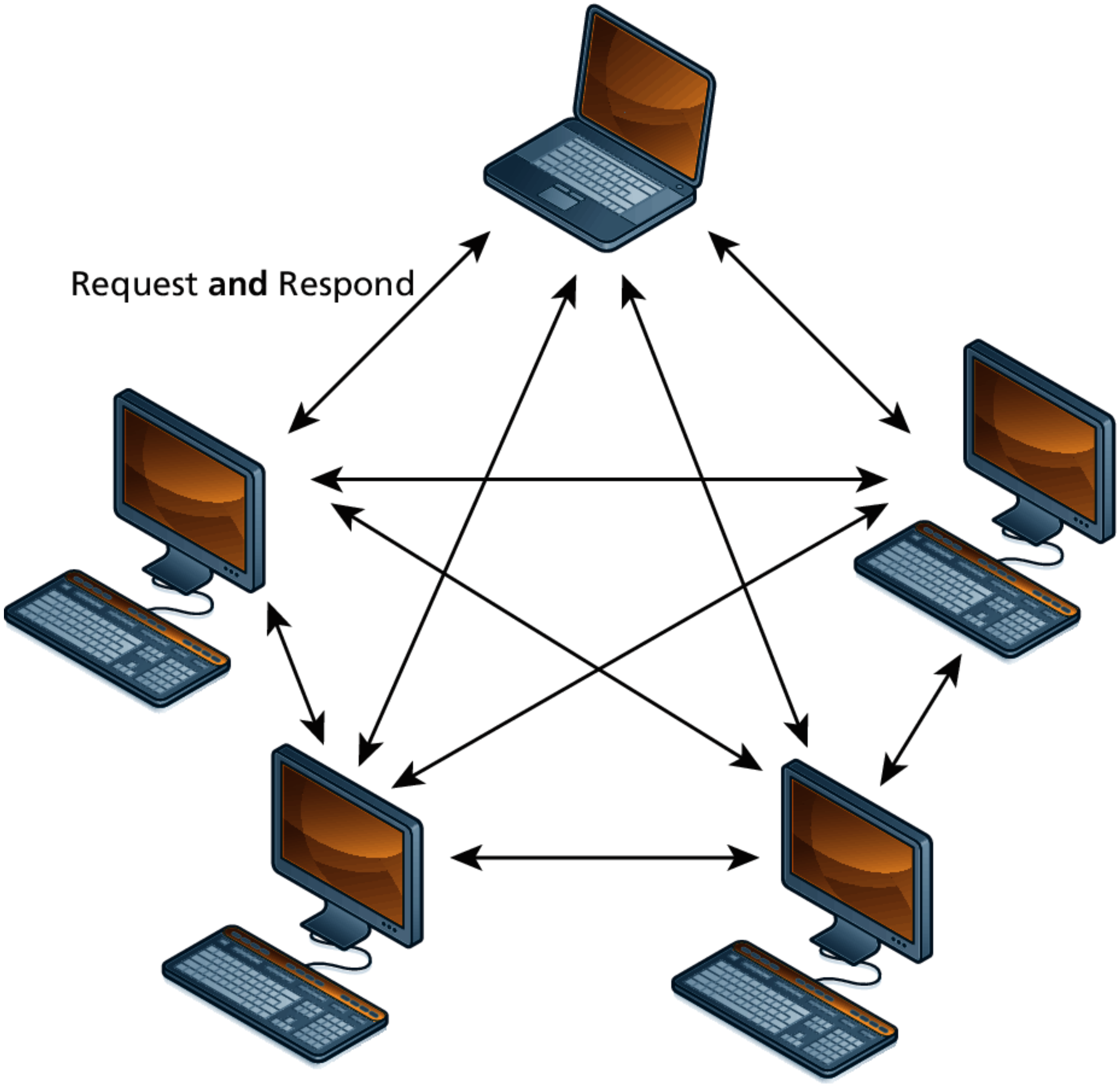


The Peer-to-Peer Alternative

In the **peer-to-peer model** where each computer is functionally identical, each node is able to send and receive directly with on e another.

In such a model each peer acts as both a client and server able t o upload and download information.

Peer-to-Peer Model



Server Types

A server is rarely just a single computer

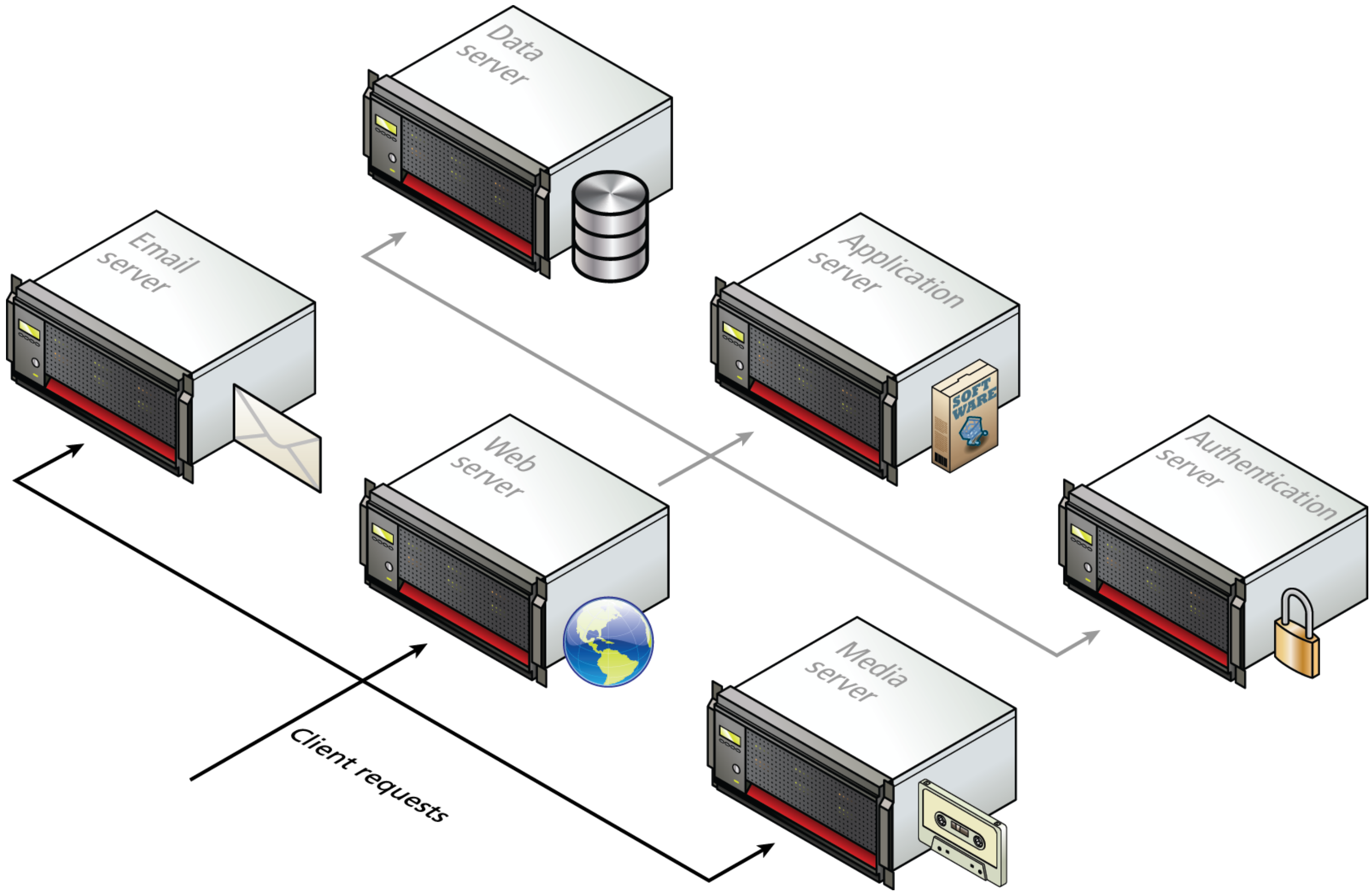
Earlier, the server was shown as a single machine, which is fine from a conceptual standpoint.

Clients make requests for resources from a URL; to the client, the se rver *is* a single machine.

However, most real-world web sites are typically not served from a s ingle server machine, but by many servers.

It is common to split the functionality of a web site between several different types of server.

Server Types



Real-World Server Installations

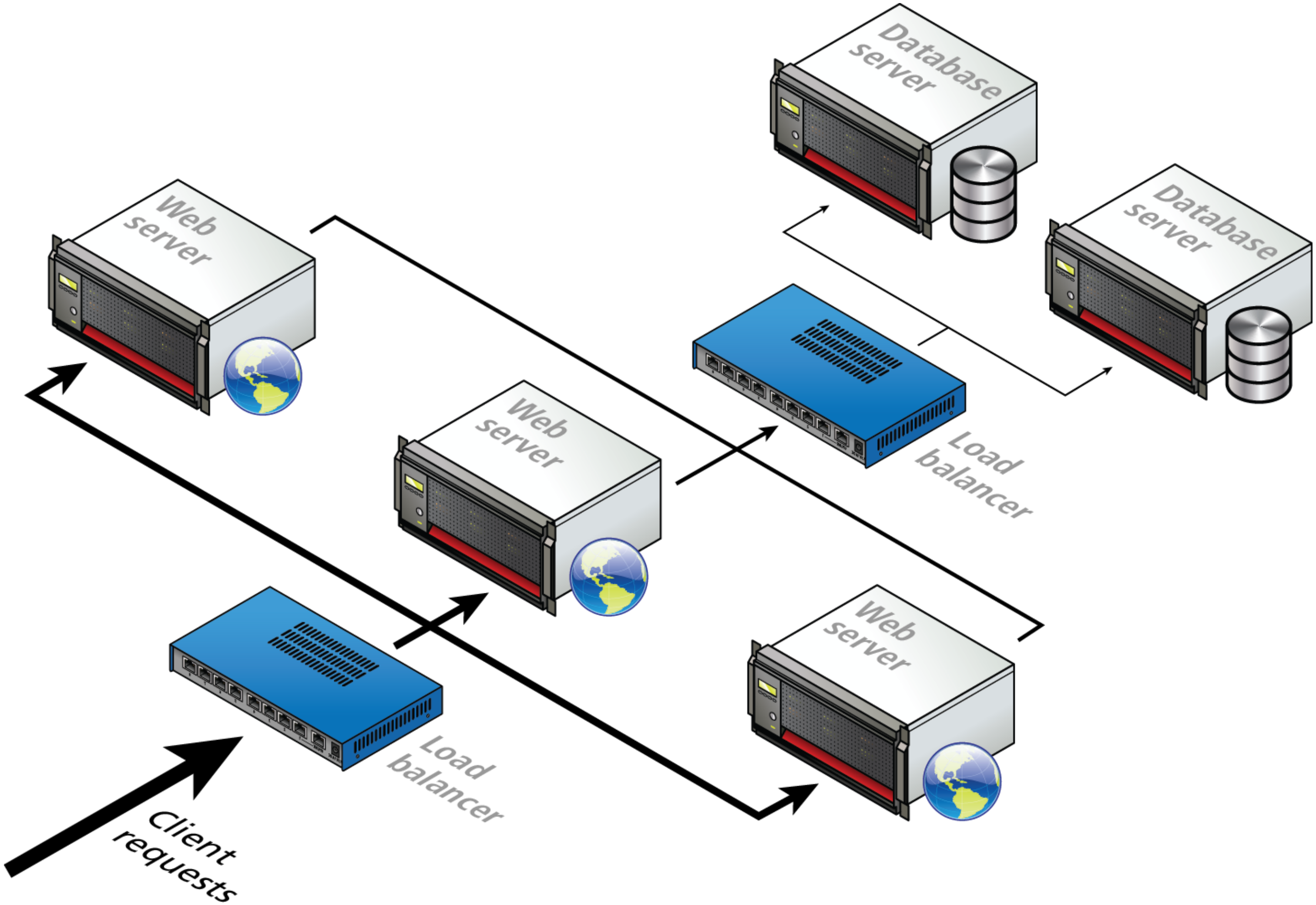
Not only are there different types of servers, there is often replication of each of the different server types.

A busy site can receive thousands or even tens of thousands of reque sts a second; globally popular sites such as Facebook receive millions of requests a second.

Server Farms

A single web server that is also acting as an application or database server will be hard-pressed to handle more than a few hundred requests a second, so the usual strategy for busier sites is to use a **server farm**.

Server Farm



Web server
Web server
Web server
Database server
Database server
Load balancer
Load balancer
Client requests

Server Farms

The goal behind server farms is to distribute incoming requests betwe en clusters of machines so that any given web or data server is not exc essively overloaded.

Special routers called **load balancers** distribute incoming requests to a vailable machines.

Server Farms

Even if a site can handle its load via a single server, it is not uncomm on to still use a server farm because it provides **failover redundancy**.

That is, if the hardware fails in a single server, one of the replicated s ervers in the farm will maintain the site's availability.

Server Racks

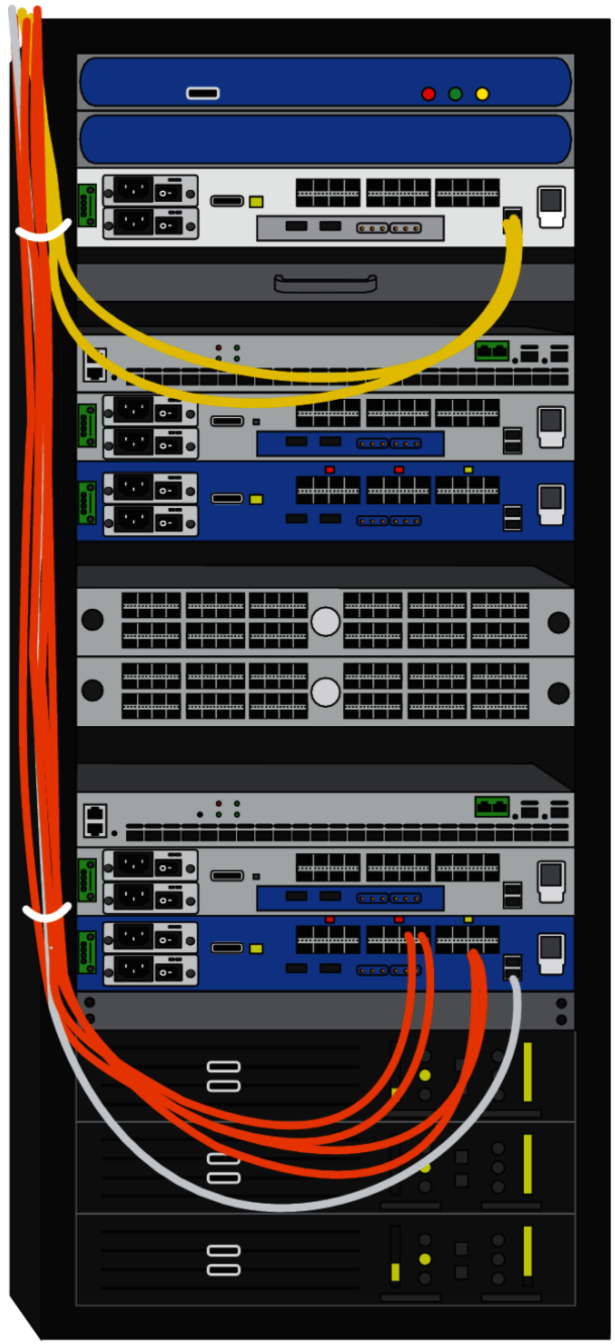
In a server farm, the computers do not look like the ones in your house.

Instead, these computers are more like the plates stacked in your kitchen cabinets.

That is, a farm will have its servers and hard drives stacked on top of each other in **server racks**.

A typical server farm will consist of many server racks, each contai ning many servers.

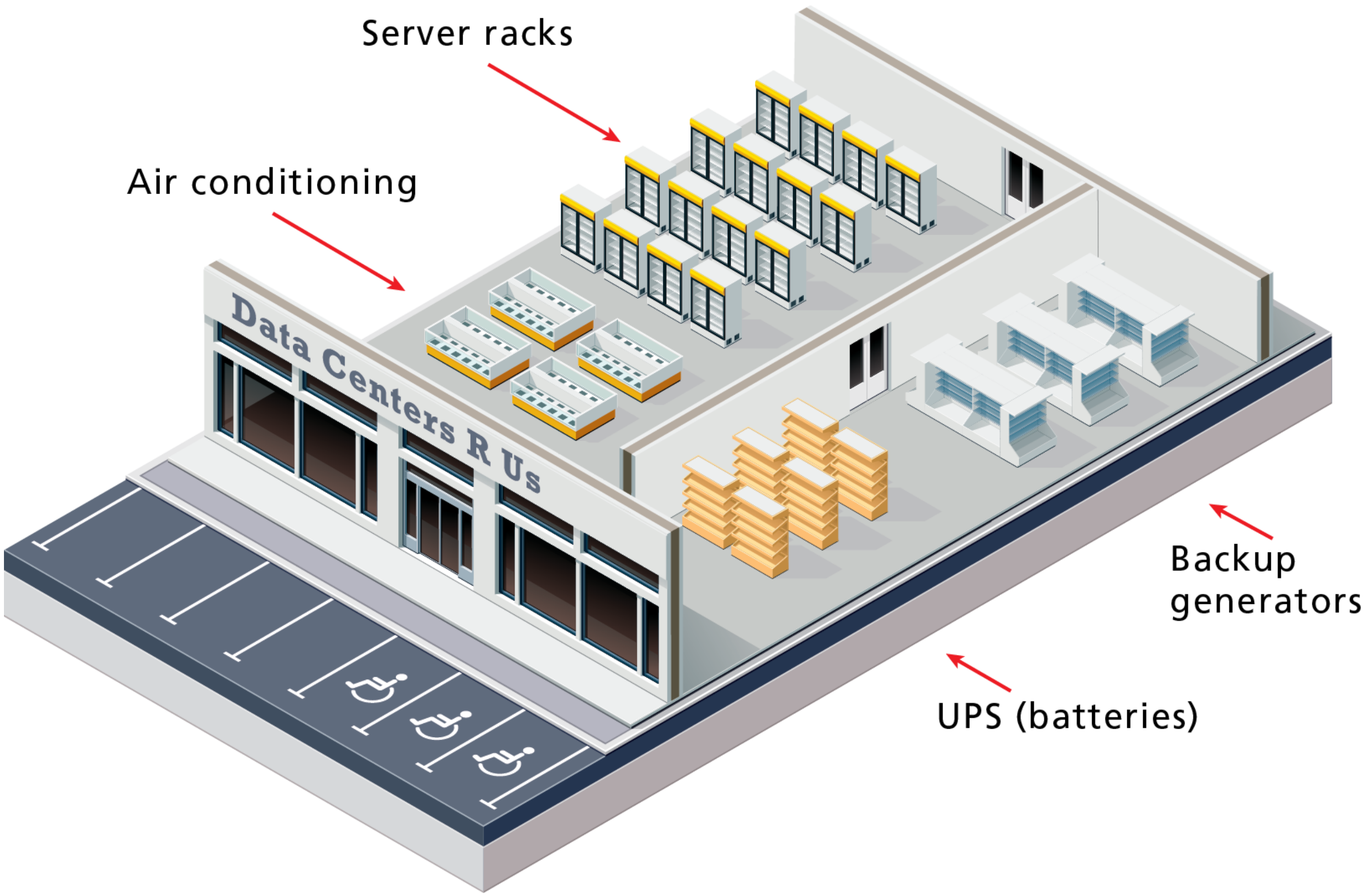
Server Rack


Fiber channel switches
Rack management server
Test server
Keyboard tray and flip-up monitor
Patch panel
Production web server
Production data server
RAID HD arrays
Patch panel
Production web server
Production data server
Batteries and UPS

Data Centers

Server farms are typically housed in special facilities called **data centers**.

Hypothetical Data Center



Data Centers

To prevent the potential for site down times, most large web sites will exist in mirrored data centers in different parts of the country, or even world.

As a consequence, the costs for multiple redundant data centers ar e quite high, and only larger web companies can afford to create an d manage their own.

Most web companies will instead lease space from a third-party dat a center.

Commercial Web Hosting

It is also common for the reverse to be true – that is, a single serve r machine may host multiple sites.

Large commercial web hosting companies such as GoDaddy, Blue H ost, Dreamhost, and others will typically host hundreds or even th ousands of sites on a single machine (or mirrored on several server s).

WHERE is the Internet?

Section 4 of 8

Is the Internet a Cloud?

The Internet is often visually represented as a cloud, which is perhaps an apt way to think about the Internet given the imp ortance of light and magnetic pulses to its operation.

Is the Internet a Cloud?

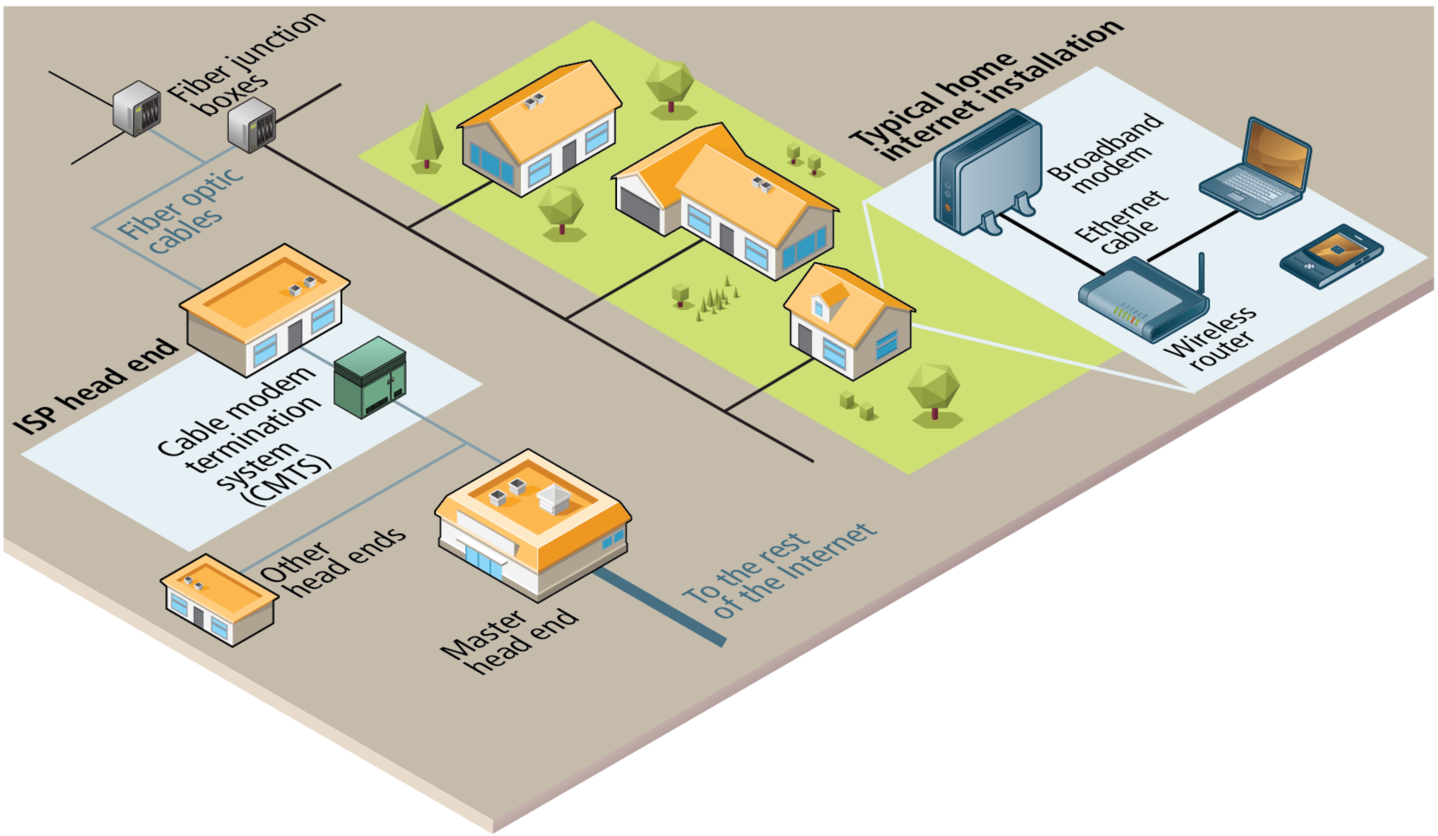
No

It is important to recognize that our global network of networks does not work using magical water vapor, but is implemented via

- millions of kilometers of copper wires and fiber optic cables, as well as via
- hundreds of thousands of server computers
- and probably an equal number of routers, switches, and other networked devices,
- along with many thousands of air conditioning units and specially-constructed server rooms and buildings.

From the Computer to the Local Provider

Our main experience of the hardware component of the Internet is that which we experience in our homes.



In the House

The **broadband modem** (also called a cable modem or DSL modem) is a bridge between the network hardware outside the house (typically controlled by a phone or cable company) and the network hardware i nside the house.

These devices are often supplied by the ISP.

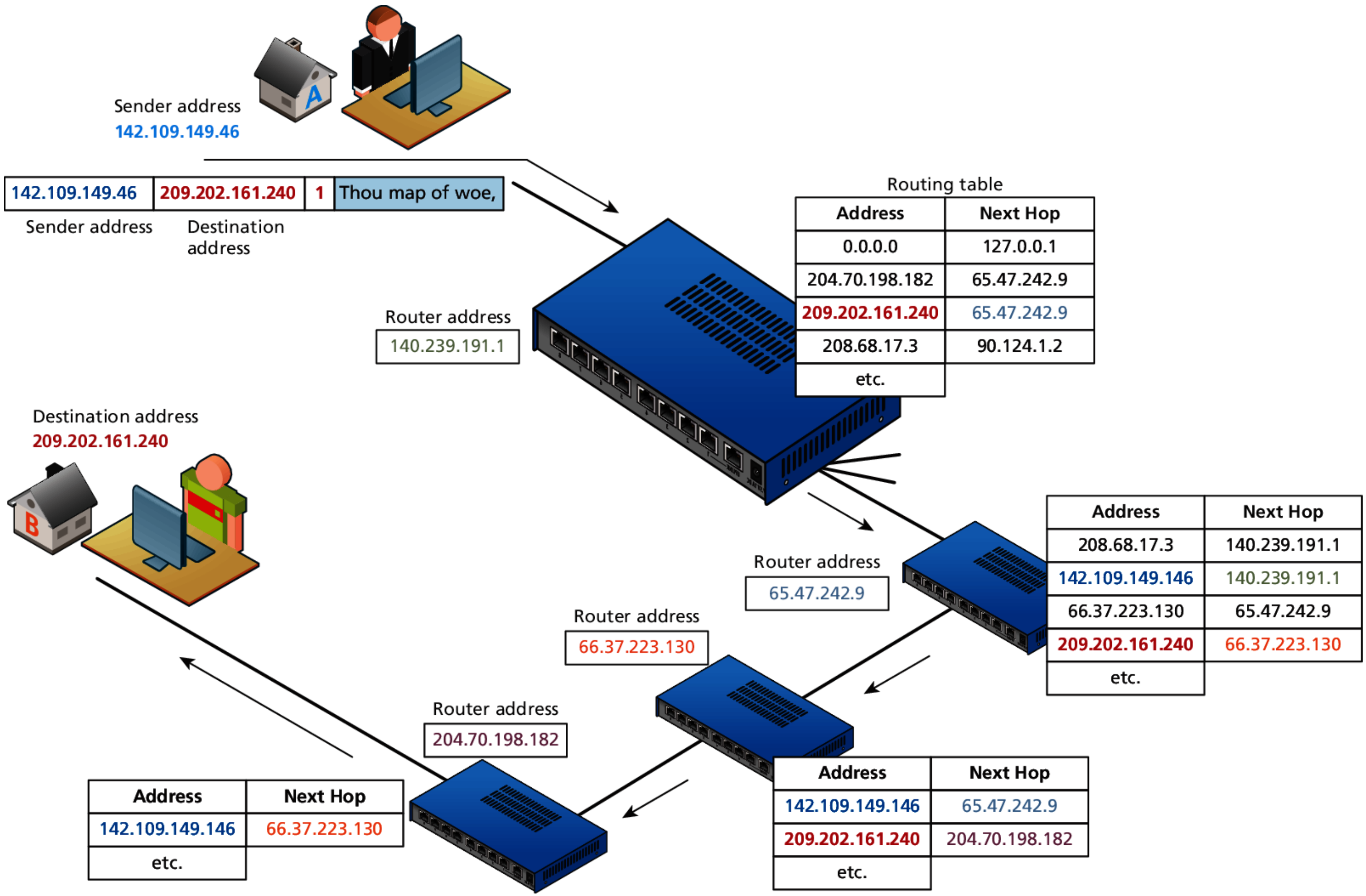
Routers

The wireless router is perhaps the most visible manifestation of t he Internet in one's home, in that it is a device we typically need to purchase and install.

Routers are in fact one of the most important and ubiquitous har dware devices that makes the Internet work.

At its simplest, a **router** is a hardware device that forwards data packets from one network to another network.

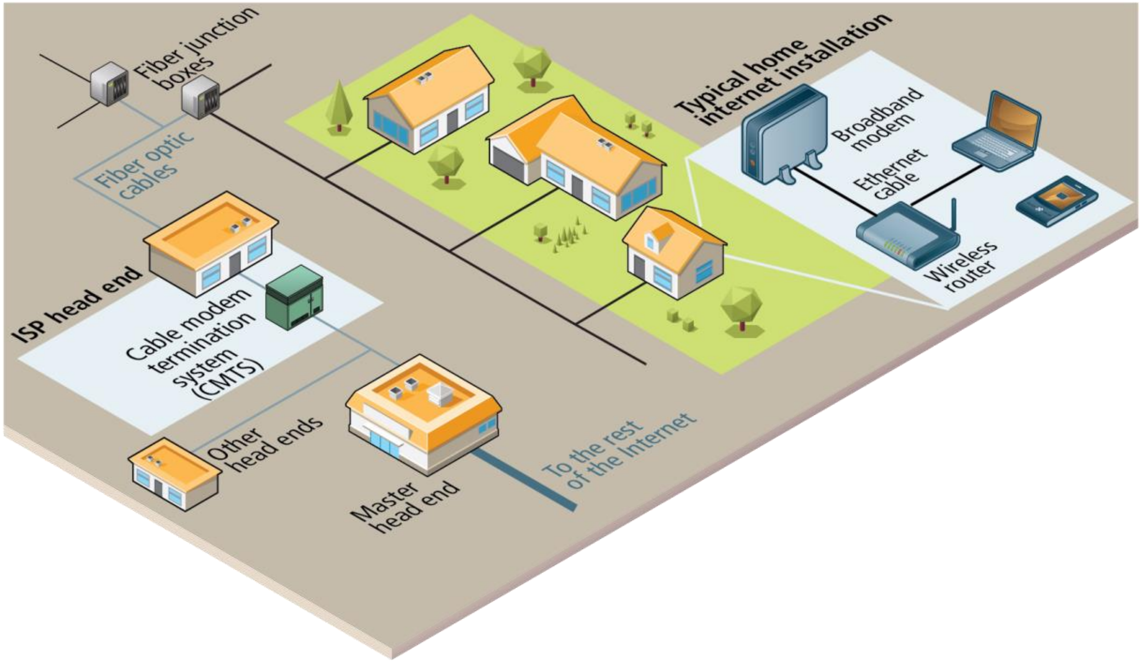
Routers and Routing Tables



Out of the House

Once we leave the confines of our own homes, the hardware of the Internet becomes much murkier.

In the illustration, the various neighborhood broadband cables (whi ch are typically using copper, aluminum, or other metals) are aggre gated and connected to fiber optic cable via fiber connection boxes .



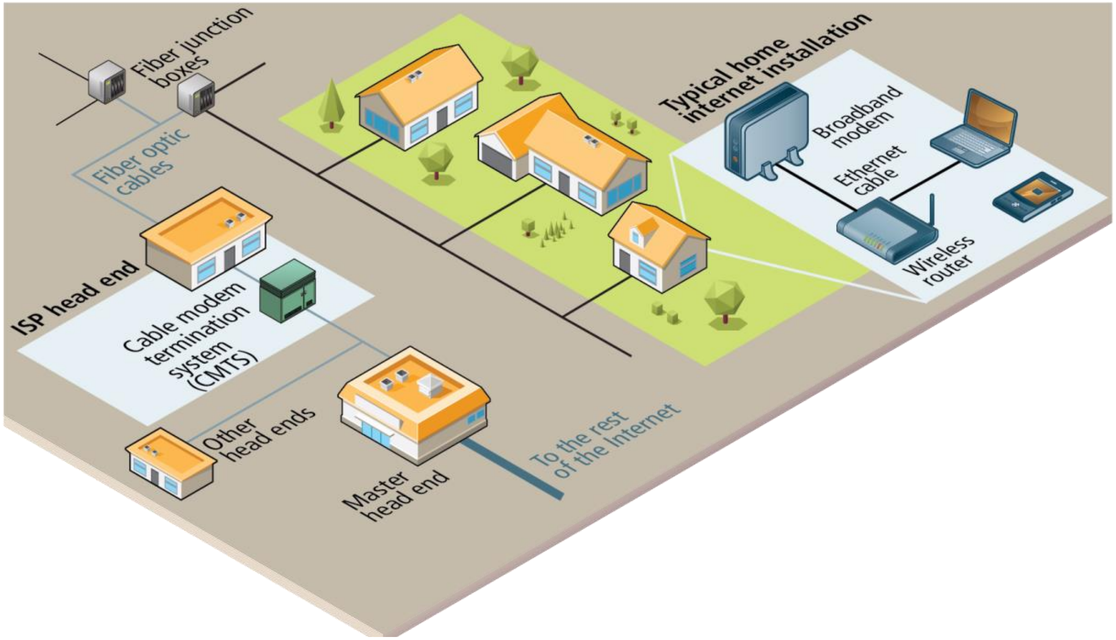
Fiber Optic Cable

Fiber optic cable (or simply optical fiber) is a glass-based wire that transmits light and has significantly greater bandwidth and speed in comparison to metal wires.

In some cities (or large buildings), you may have fiber optic cable going directly into individual buildings; in such a case the fiber junction box will reside in the building.

To the Provider

These fiber optic cables eventually make their way to an ISP's **head-end**, which is a facility that may contain a **cable modem termination system** (CMTS) or a digital subscriber line access multiplexer (DSLAM) in a DSL-based system.



From the Local Provider to the Ocean

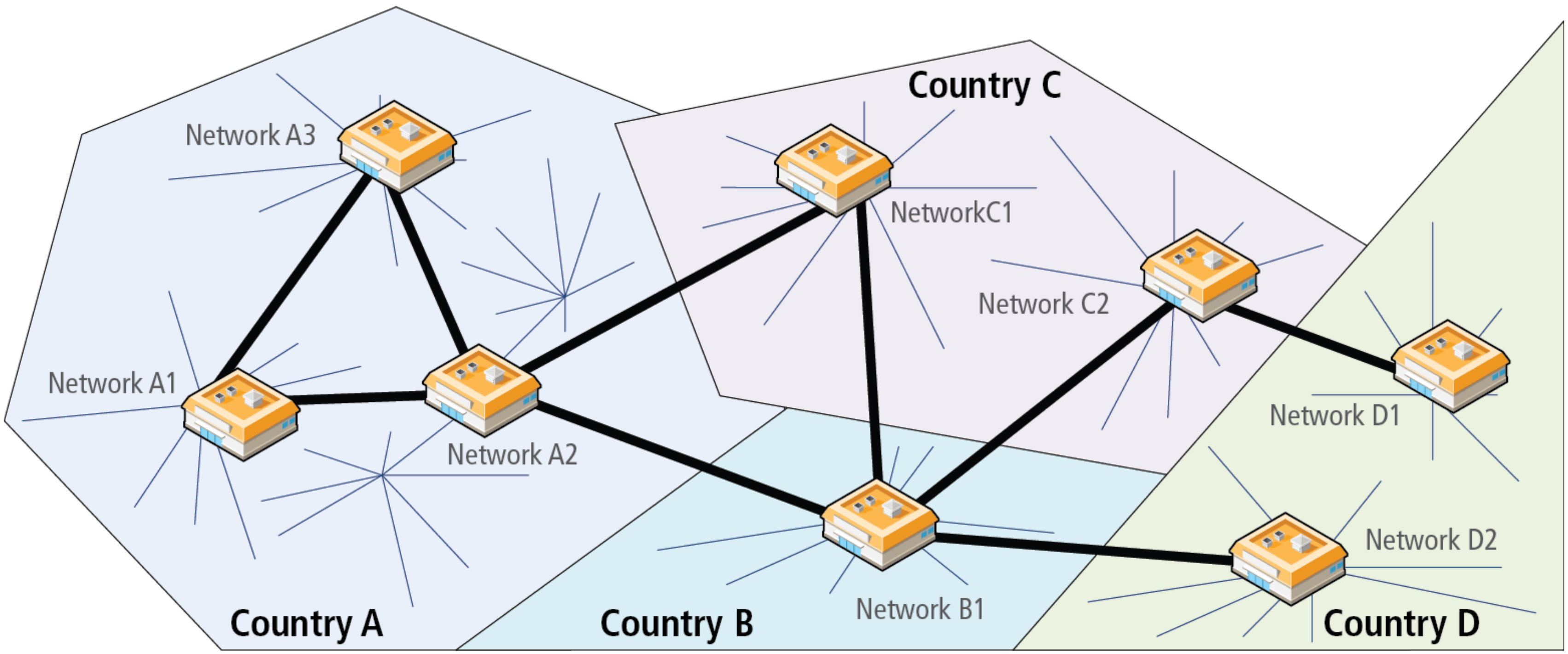
Eventually your ISP has to pass on your requests for Internet packets to other networks.

This intermediate step typically involves one or more regional netwo rk hubs.

Your ISP may have a large national network with optical fiber connec ting most of the main cities in the country.

Some countries have multiple national or regional networks, each wi th their own optical network.

Connecting different networks within and between countries



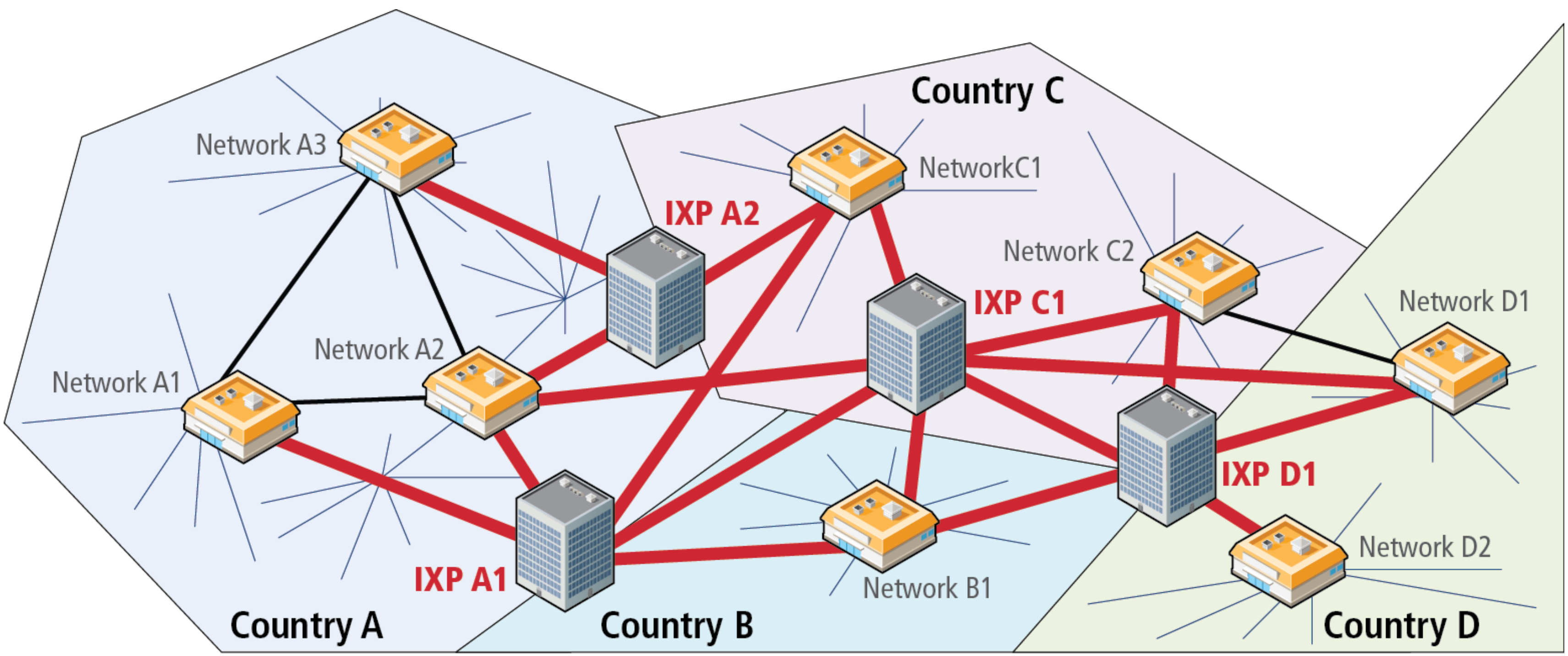
Internet Exchange Points

Connecting different networks

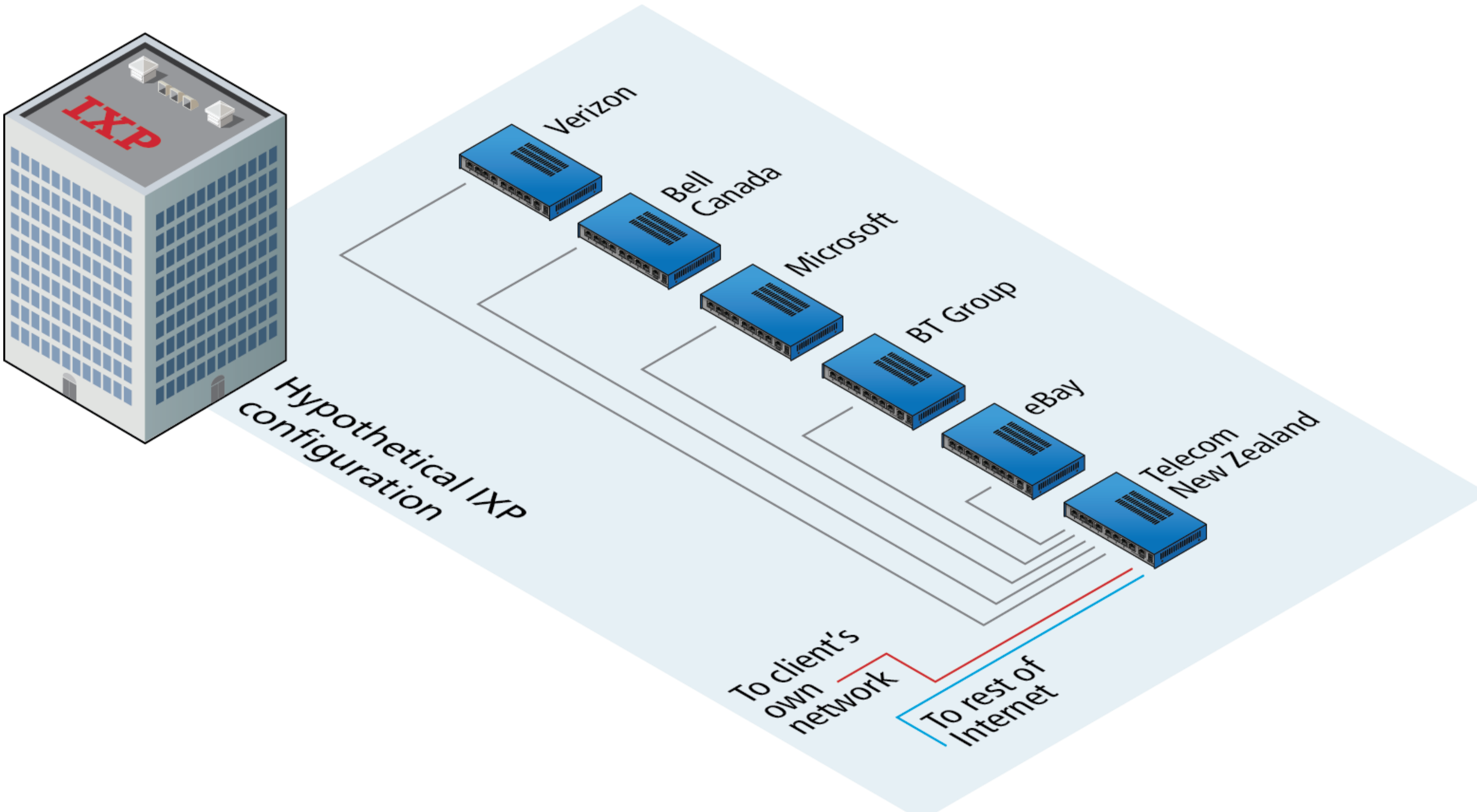
This type of network configuration began to change in the 2000s, as more and more networks began to interconnect with each other usi ng an **Internet Exchange Point** (**IX** or **IXP**).

These IXPs allow different ISPs to **peer** with one another (that is, int erconnect) in a shared facility, thereby improving performance for e ach partner in the peer relationship.

National and regional networks using Internet Exchange Points

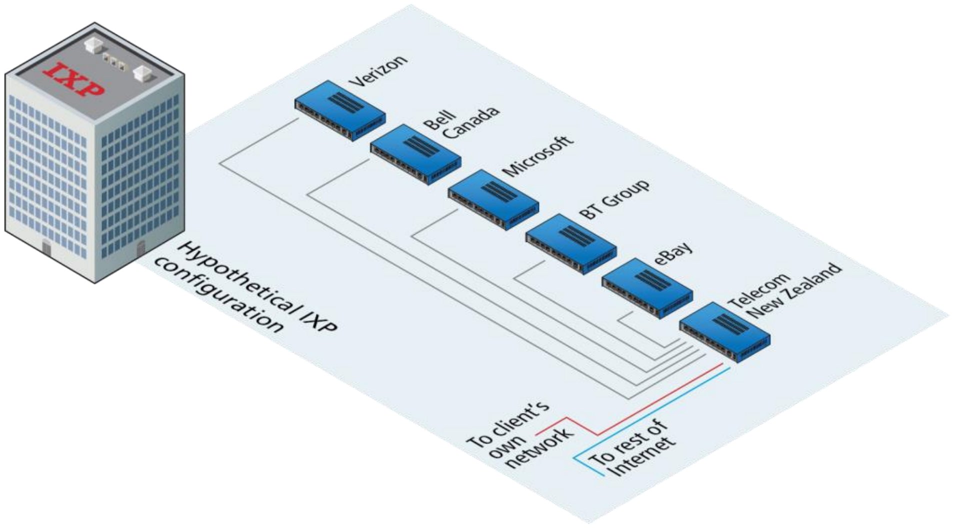


Sample Internet Exchange Point



IXPs
Not just for large networks

Different networks connect not only to other networks within an IXP, but now large web sites such as Microsoft and FaceBook are also con necting to multiple other networks simultaneously as a way of improv ing the performance of their sites.



Real IXPs

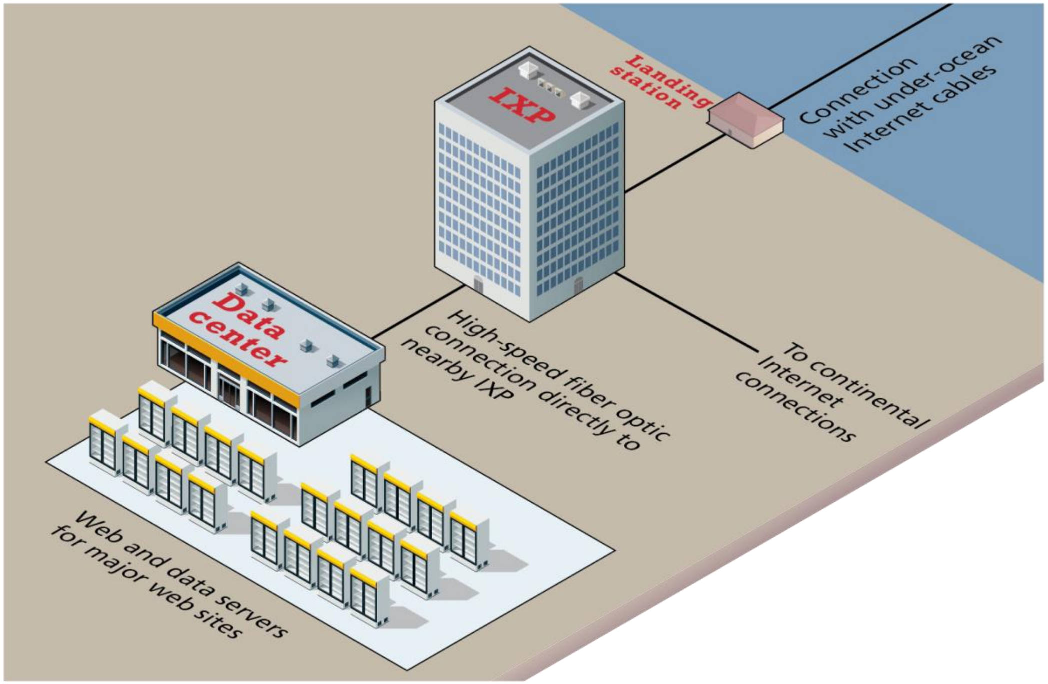
Real IXPs, such as at Palo Alto (PAIX), Amsterdam (AMS-IX), Frankfurt (CE-CIX), London (LINX), allow many hundreds of networks and com panies to interconnect and have throughput of over 1000 gigabits pe r second.

The scale of peering in these IXPs is way beyond that shown in the di agram (which shows peering with only five others); companies withi n these IXPs use large routers from Cisco and Brocade that have hun dreds of ports allowing hundreds of simultaneous peering relationsh ips.

IXPs and Data Centers

In recent years, major web companies have joined the network com panies in making use of IXPs.

As shown in the diagram, this sometimes involves mirroring a site's i nfrastructure (i.e., web and data servers) in a data center located nea r the IXP.

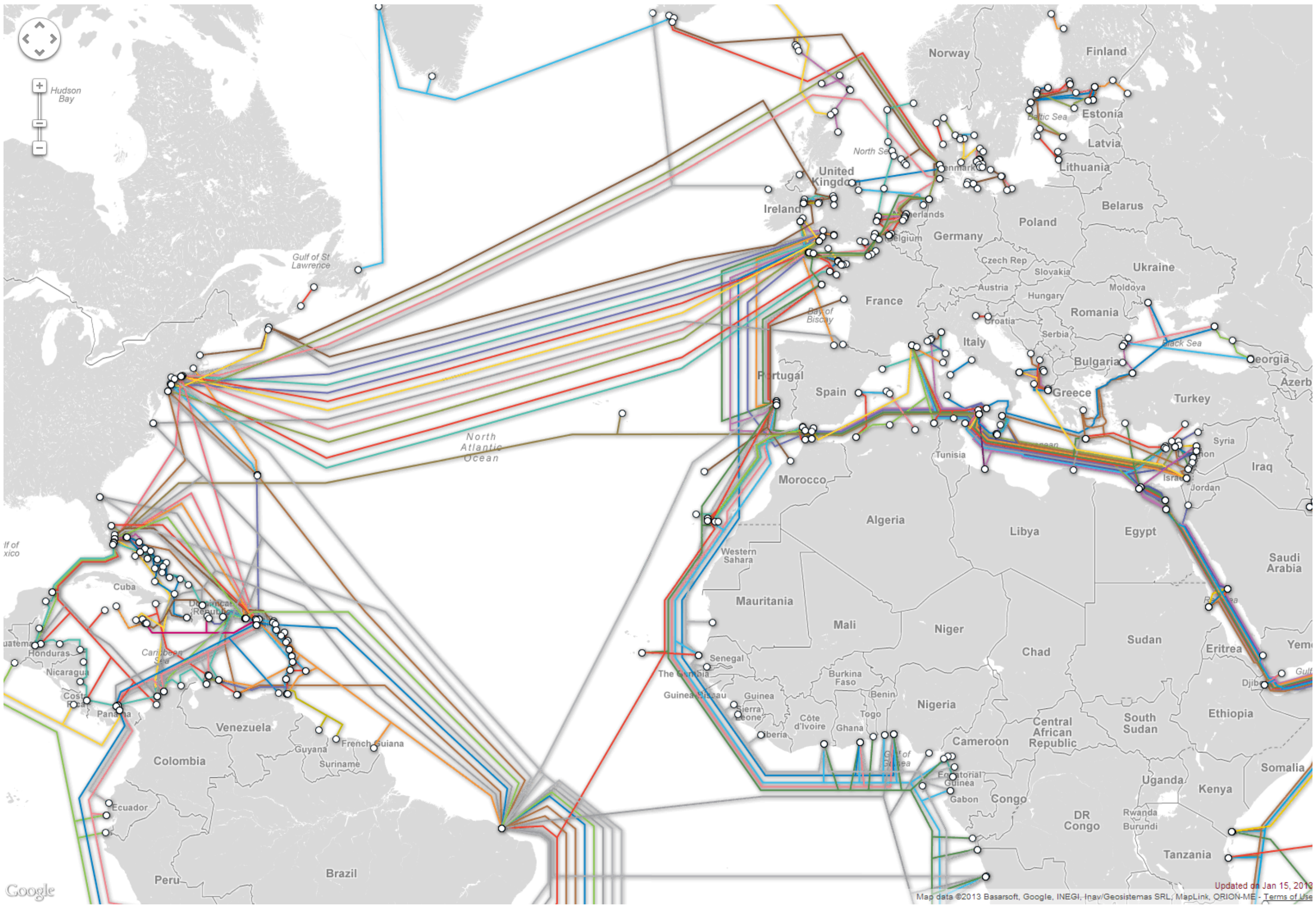


Across the Oceans

Eventually, international Internet communication will need to travel underwater.

The amount of undersea fiber optic cable is quite staggering and is growing yearly.

Undersea fiber optic lines (courtesy TeleGeography)



DOMAIN NAME SYSTEM (DNS)

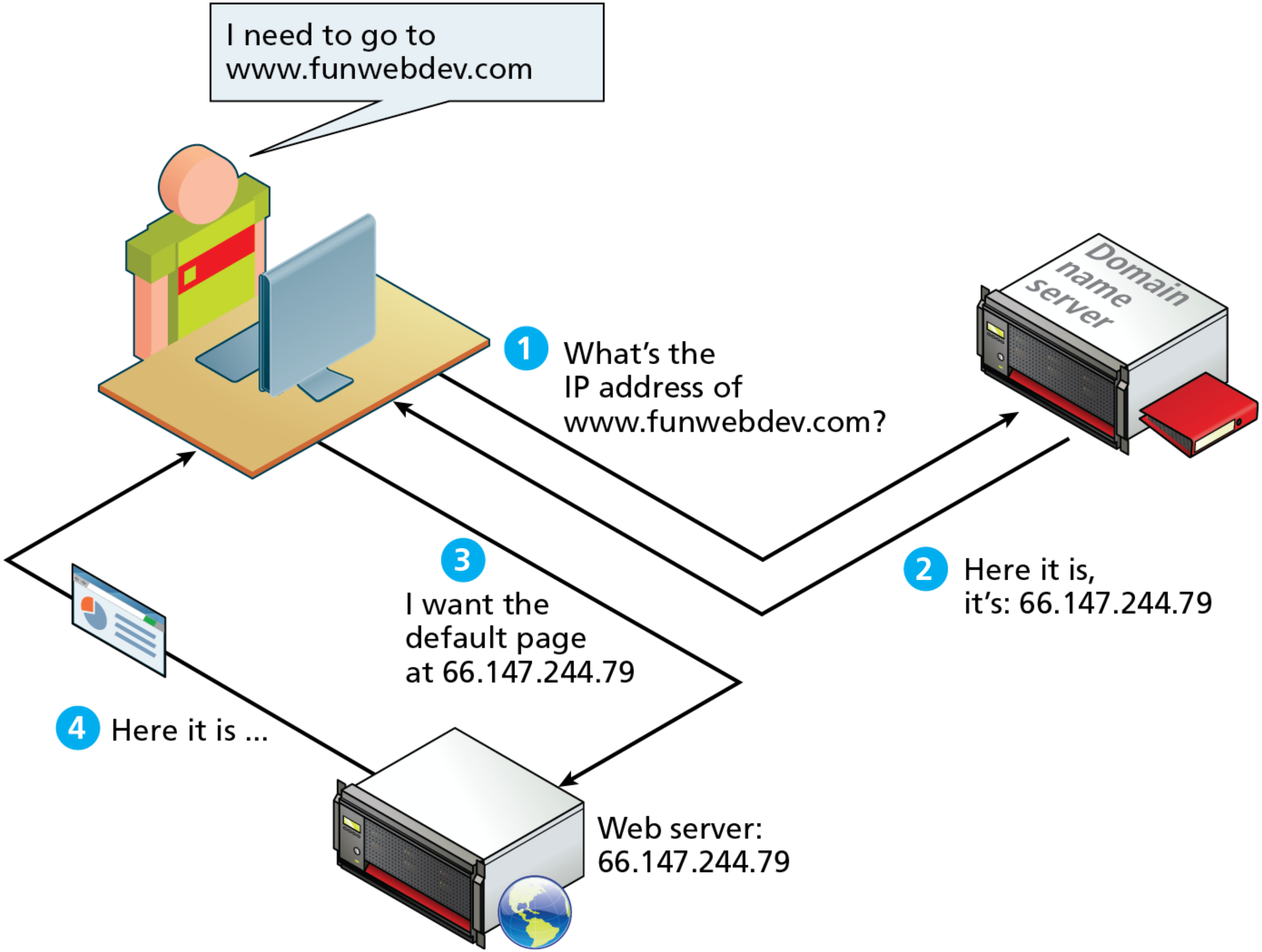
Section 5 of 8

Domain Name System

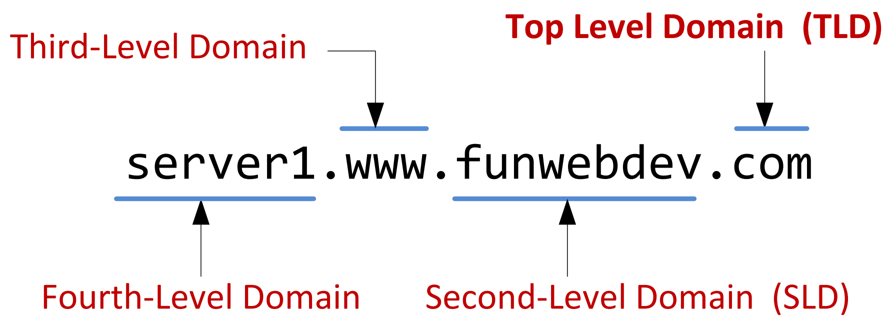
Why do we need it?

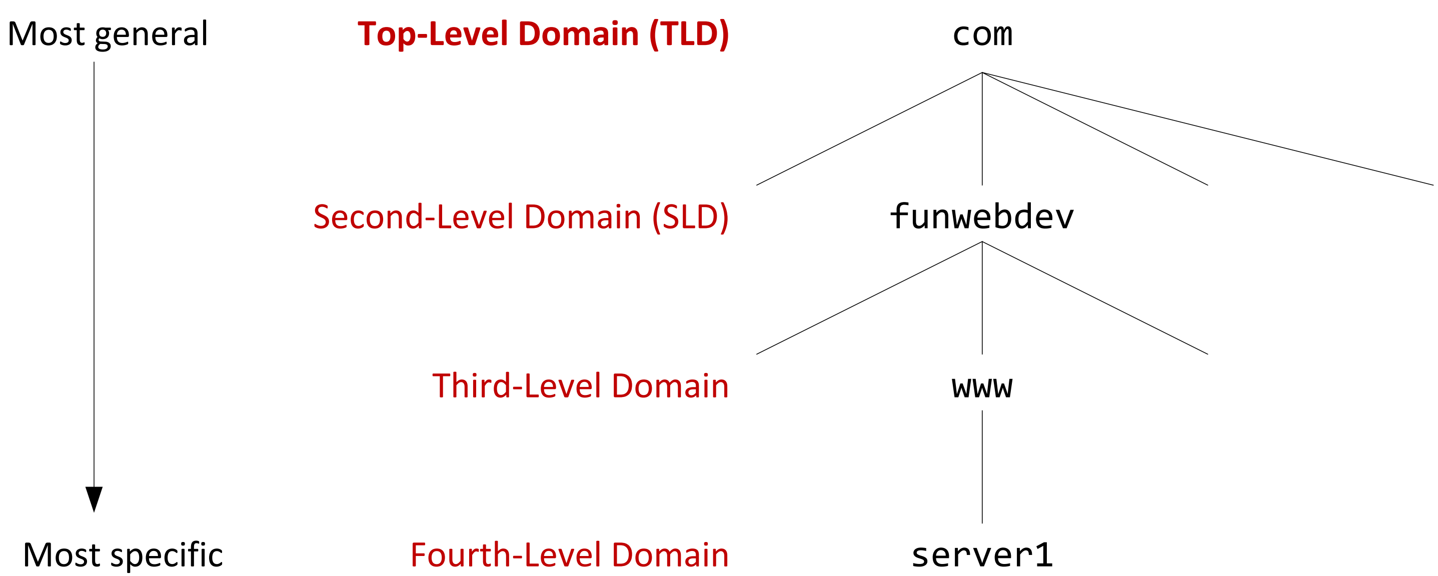
As elegant as IP addresses may be, human beings do not enjoy having to recall long strings of numbers. Instead of IP addresses, we use the **Domain Name System** (**DNS**)

DNS Overview



Domain Levels





Types of TLDs

Generic top-level domains (gTLD)

Country code top-level domain (ccTLD)

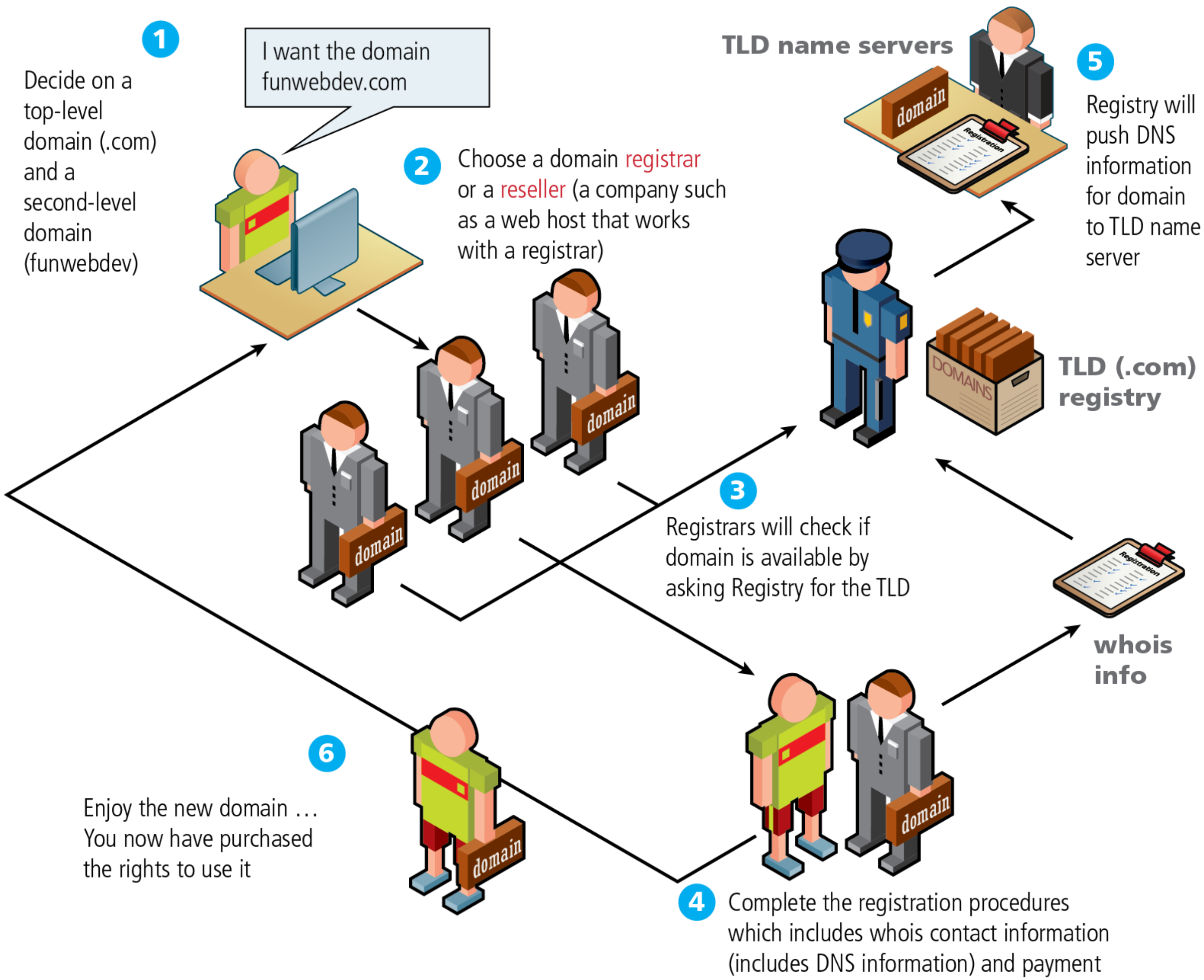
Name Registration

How are domain names assigned?

Special organizations or companies called **domain name registrars** manage the registration of domain names.

These domain name registrars are given permission to do so by the appropriate generic top-level domain (gTLD) registry and/or a coun try code top-level domain (ccTLD) registry.

Domain name registration process



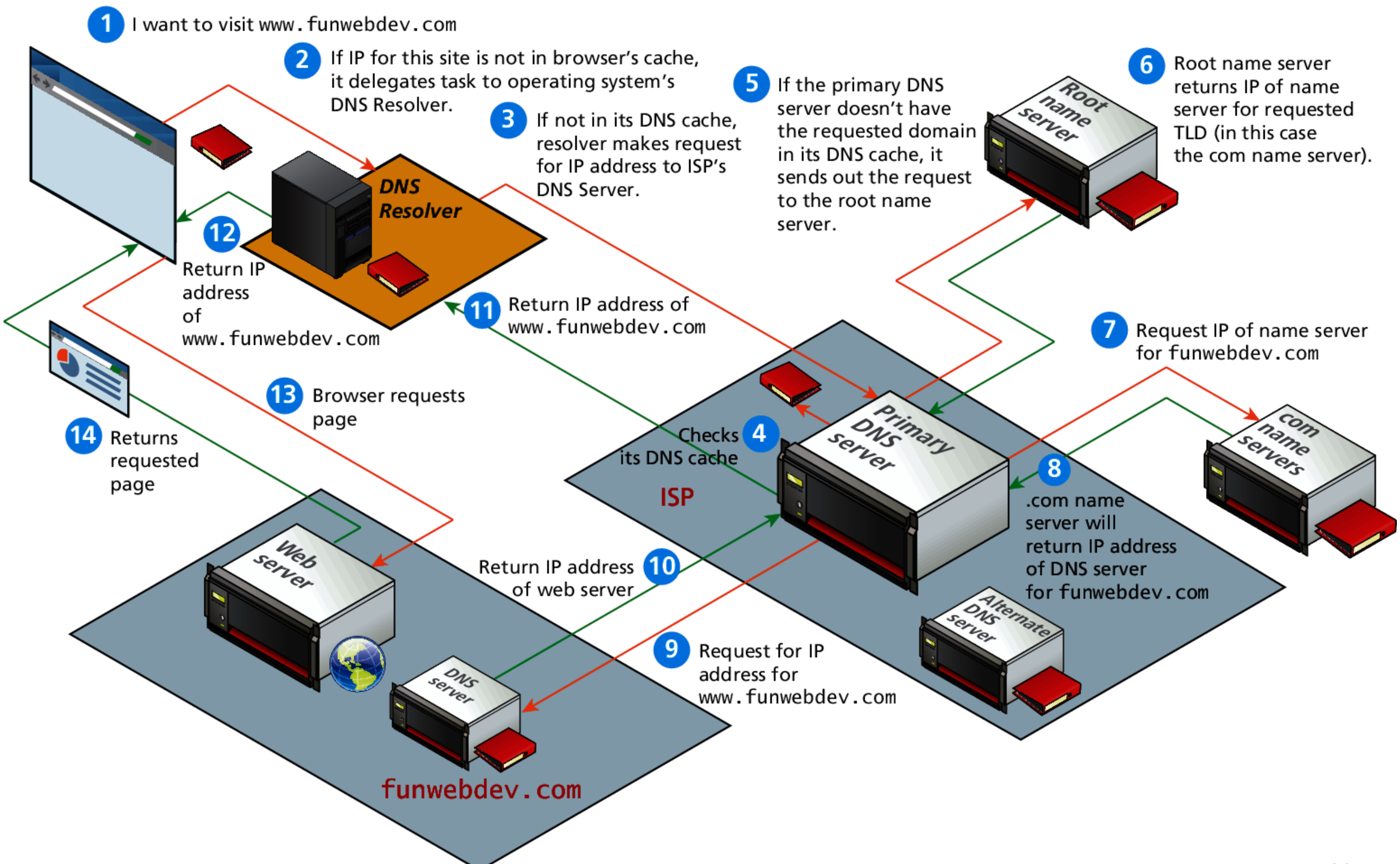
DNS Address Resolution

While domain names are certainly an easier way for users to reference a web site, eventually, your browser needs to know the IP address of the web site in order to request any resources from it.

The Domain Name System provides a mechanism for software to discover this numeric IP address.

This process is referred to here as **address resolution**.

Domain name address resolution process



Uniform Resource Locators (URL)

Section 6 of 8

URL Components

In order to allow clients to request particular resources from the server, a naming mechanism is required so that the client knows how to ask the server for the file.

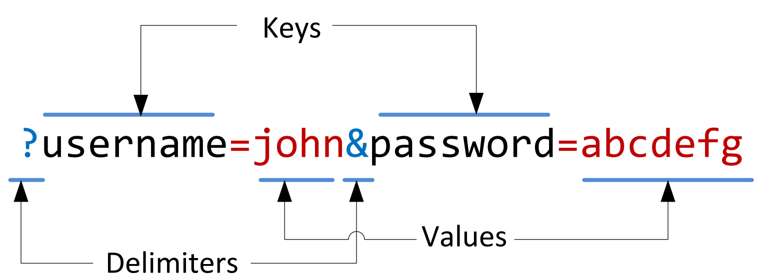
For the web that naming mechanism is the **Uniform Resource Locator** (**URL**).

$\underbrace{\texttt{http}}_{\text{Protocol}}\texttt{://}\underbrace{\texttt{www.funwebdev.com}}_{\text{Domain}}\texttt{/}\underbrace{\texttt{index.php}}_{\textit{Path}}\texttt{?}\underbrace{\texttt{page=17}}_{\textit{Query String}}\texttt{\#}\underbrace{\texttt{article}}_{\textit{Fragment}}$

Query String

Query strings will be covered in depth when we learn more about HTML forms and server-side programming.

They are the way of passing information such as user form input fr om the client to the server. In URL's they are encoded as key-value pairs delimited by “&” symbols and preceded by the “?” symbol.



Hypertext transfer protocol (http)

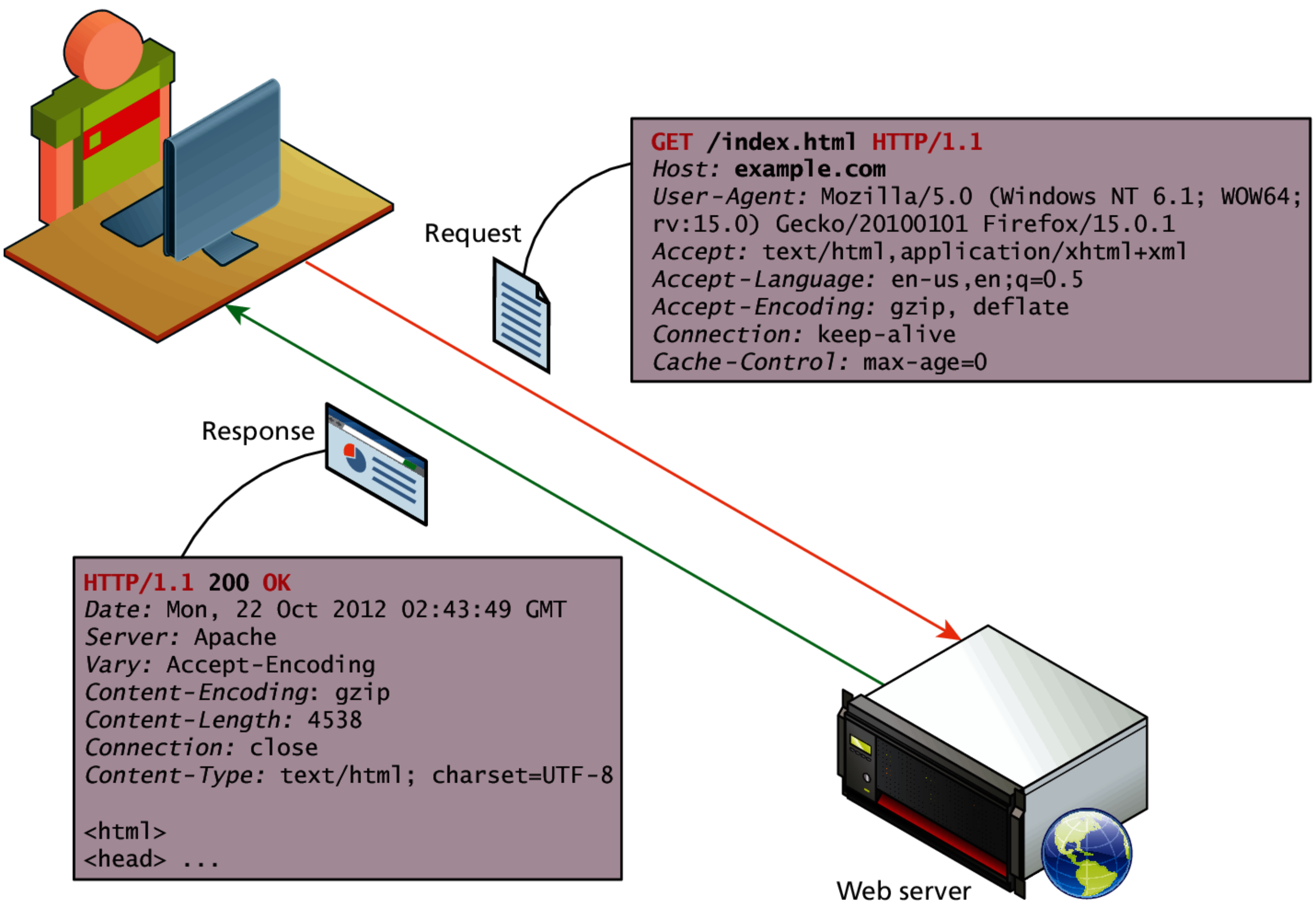
Section 7 of 8

HTTP

The HTTP protocol establishes a TCP connection on port 80 (by default).

The server waits for the request, and then responds with a response code, headers and an optional message (which ca n include files).

HTTP



GET /index.html HTTP/1.1
Host: example.com
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:15.0) Gecko/20100101 Firefox/15.0.1
Accept: text/html,application/xhtml+xml
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Cache-Control: max-age=0
Request
Response
HTTP/1.1 200 OK
Date: Mon, 22 Oct 2012 02:43:49 GMT
Server: Apache
Vary: Accept-Encoding
Content-Encoding: gzip
Content-Length: 4538
Connection: close
Content-Type: text/html; charset=UTF-8
<html>
<head> ...
Web server

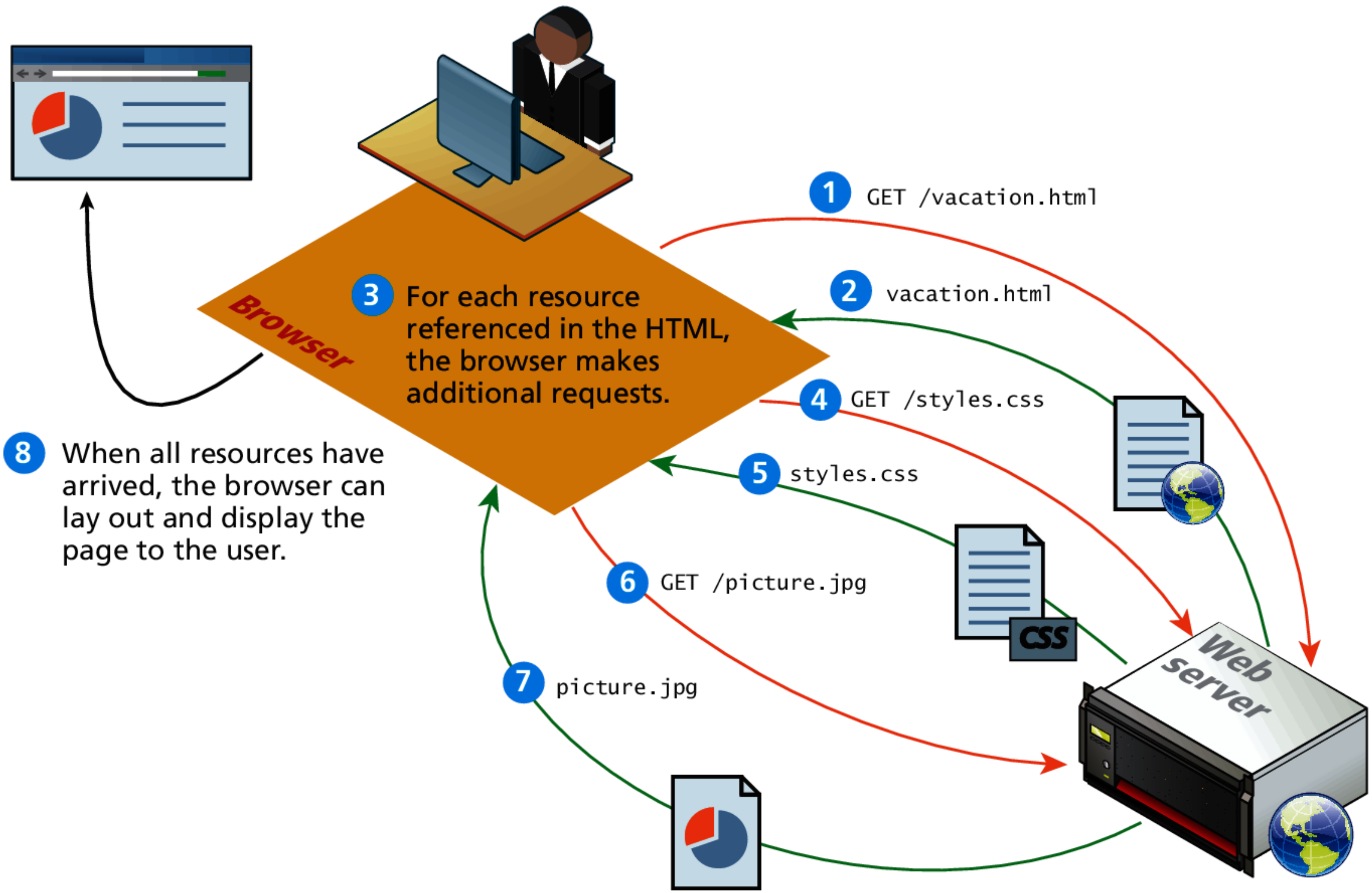
Web Requests

While we as web users might be tempted to think of an entire page being returned in a single HTTP response, this is not in fact what ha ppens.

In reality the experience of seeing a single web page is facilitated by the client's browser which requests the initial HTML page, then par ses the returned HTML to find all the resources referenced from wit hin it, like images, style sheets and scripts.

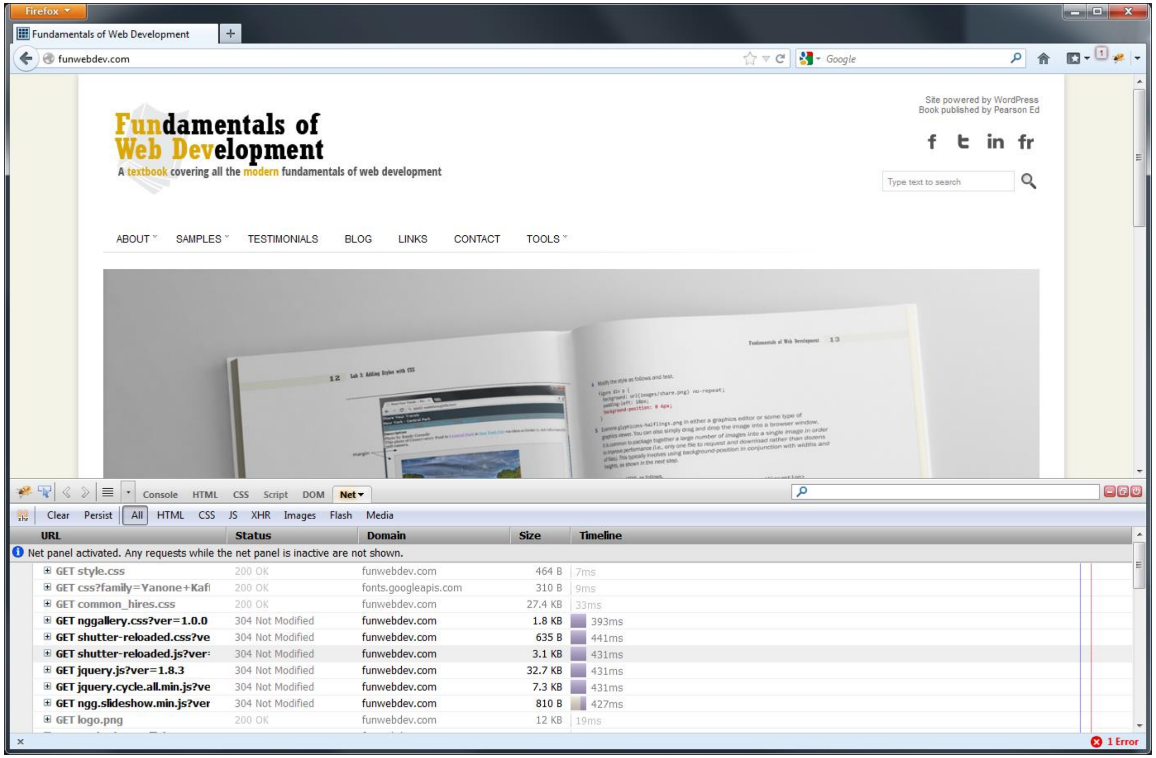
Only when all the files have been retrieved is the page fully loaded for the user

Browser parsing HTML and making subsequent requests



Browser Tools for HTTP

Modern browsers provide the developer with tools that can help us understand the HTTP traffic for a given page.



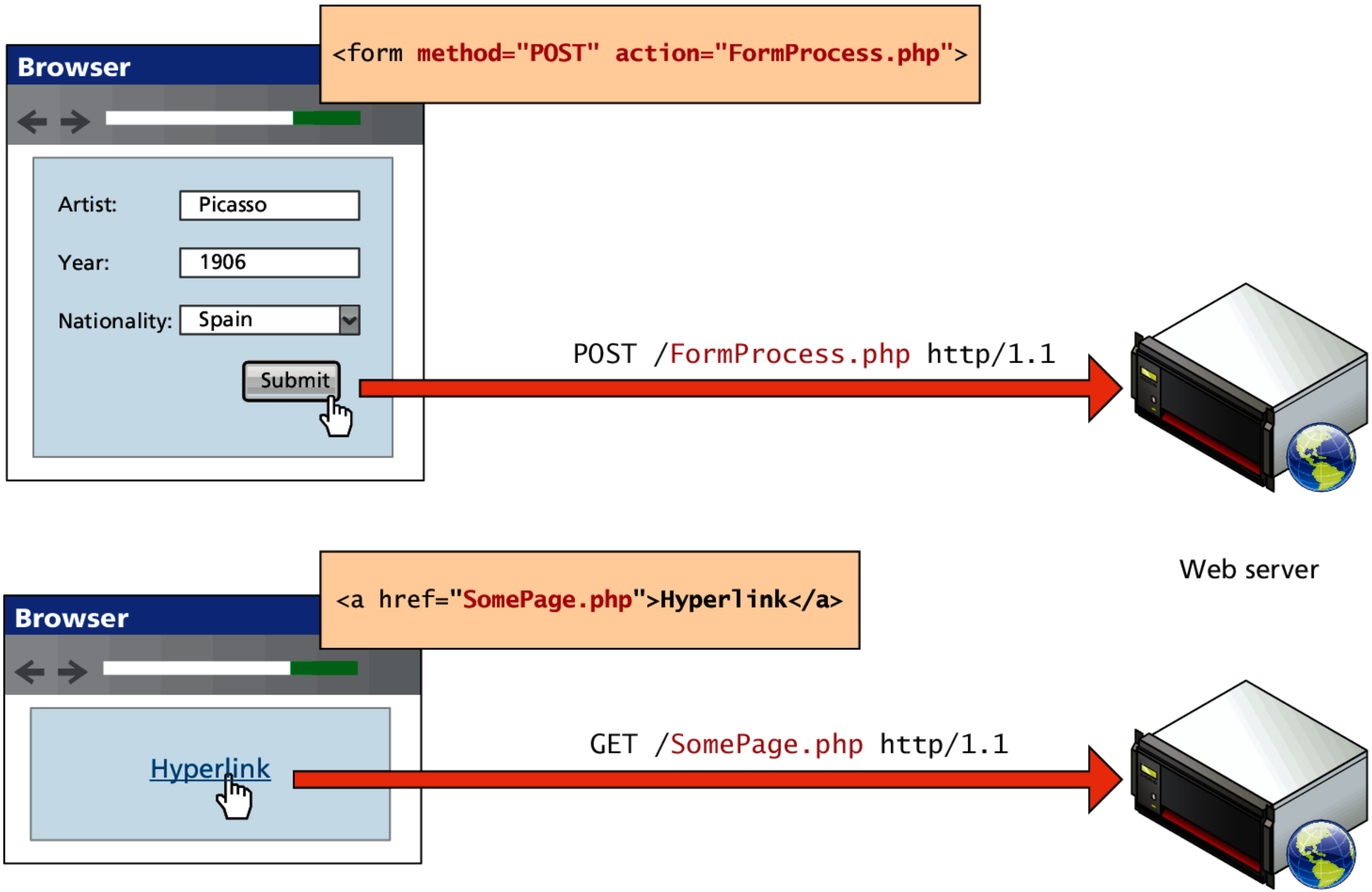
HTTP Request Methods

The HTTP protocol defines several different types of requests, each with a different intent and characteristics.

The most common requests are the GET and POST request, along w ith the HEAD request.

Other requests, such as PUT, DELETE, CONNECT, TRACE and OPTION S are seldom used, and are not covered here.

GET versus POST requests



Web Servers

Section 8 of 8

Web Servers

A **web server** is, at a fundamental level, nothing more than a compu
ter that responds to HTTP requests.

Web Stack

Regardless of the physical characteristics of the server, one must choose an application stack to run a website.

This stack will include an operating system, web server software, a database and a scripting language to process dynamic requests.

LAMP Software Stack

Throughout this textbook we will rely on the **LAMP software stack**, which refers to the Linux operating system, Apache web server, MySQL database, and PHP scripting language

WISA software stack

Many corporations, for instance, make use of the Microsoft **WISA software stack**, which refers to Windows operating system, IIS web server, SQL Server database, and the ASP.NET server-side development technologies.

What You’ve Learned

1. Definition and History
2. Internet Protocols
3. Client-Server Model
4. Where is the Internet?
5. Domain Name System
6. Uniform Resource Locators(URL)
7. Hypertext Transfer Protocol(HTTP)
8. Web Servers