**Networks and the Internet**

Computer network – a connection between two or more computers that allows them to send data back and forth

Local network – typically done through cables or wireless connections between computers at a specific location

The Internet - a network connecting machines all over the planet

The technology to start implementing this vision was developed in the 1980s

A computer uses a network to shoot bits at another computer. For any effective communication to arise out of this bit-shooting, the computers on both ends must know what the bits are supposed to represent. The meaning of any given sequence of bits depends entirely on the kind of thing that it is trying to express and on the encoding mechanism used.

A *network protocol* describes a style of communication over a network. There are protocols for sending email, for fetching email, for sharing files, and even for controlling computers that happen to be infected by malicious software.

The *HyperText Transfer Protocol* (HTTP) is a protocol for retrieving named resources (chunks of information, such as web pages or pictures). It specifies that the side making the request should start with a line like this, naming the resource and the version of the protocol that it is trying to use:

GET /index.html HTTP/1.1

Most protocols are built on top of other protocols. HTTP treats the network as a streamlike device into which you can put bits and have them arrive at the correct destination in the correct order. Providing those guarantees on top of the primitive data-sending that the network gives you is already a rather tricky problem.

The *Transmission Control Protocol* (TCP) is a protocol that addresses this problem. All internet-connected devices “speak” it, and most communication on the internet is built on top of it.

A TCP connection works as follows: one computer must be waiting, or *listening*, for other computers to start talking to it. To be able to listen for different kinds of communication at the same time on a single machine, each listener has a number (called a *port*) associated with it. Most protocols specify which port should be used by default. For example, when we want to send an email using the SMTP protocol, the machine through which we send it is expected to be listening on port 25.

Another computer can then establish a connection by connecting to the target machine using the correct port number. If the target machine can be reached and is listening on that port, the connection is successfully created. The listening computer is called the *server*, and the connecting computer is called the *client*.

Such a connection acts as a two-way pipe through which bits can flow—the machines on both ends can put data into it. Once the bits are successfully transmitted, they can be read out again by the machine on the other side. This is a convenient model. TCP provides an abstraction of the network.

**The Web**

The *World Wide Web* (not to be confused with the internet as a whole) is a set of protocols and formats that allow us to visit web pages in a browser. The word *Web* refers to the fact that such pages can easily link to each other, thus connecting into a huge mesh that users can move through.

To become part of the web, all you need to do is connect a machine to the internet and have it listen on port 80 with the HTTP protocol so that other computers can ask it for documents.

Each document on the web is named by a *uniform resource locator* (URL), which looks something like this:

http://eloquentjavascript.net/13\_browser.html

| | | |

protocol server path

The first part tells us that this URL uses the HTTP protocol (as opposed to, for example, encrypted HTTP, which would be *https://*). Then comes the part that identifies which server we are requesting the document from. Last is a path string that identifies the document (or *resource*) we are interested in.

Machines connected to the internet get an *IP address*, a number that can be used to send messages to that machine, and looks something like 149.210.142.219 or 2001:4860:4860::8888. Since lists of more or less random numbers are hard to remember and awkward to type, you can instead register a *domain name* for an address or set of addresses. I registered *eloquentjavascript.net* to point at the IP address of a machine I control and can thus use that domain name to serve web pages.

If you type this URL into your browser’s address bar, the browser will try to retrieve and display the document at that URL. First, your browser has to find out what address *eloquentjavascript.net* refers to. Then, using the HTTP protocol, it will make a connection to the server at that address and ask for the resource */13\_browser.html*. If all goes well, the server sends back a document, which your browser then displays on your screen.