

The Modern Web

Rafael L. Cepeda

December 4, 2017

1 HTTP

HTTP is the famous protocol that governs the web. It runs atop of TCP, providing a reliable service. HTTP is stateless; it does not remember anything about prior requests. Instead, explicit state management must be implemented alongside the protocol: cookies, URL data passing, or client-side storage.

1.1 Quirks

HTTP stands upon the URL syntax, defined in RFC 3986, to establish connections. HTTP formulates two message variants: requests and responses. From the client and server respectively. Various types of congestion control may hinder web server performance: TCP slow start (though optimizations exist like Nagle's algorithm.) HTTP defines a set of methods to differentiate request types: GET, POST, PUT, DELETE, and more. These are the main ones, however, corresponding to the retrieval, creation, modification, and deletion of resources.

1.2 Infrastructure

HTTP has a nifty infrastructure: tunnels blindly forward data, gateways link two different protocols, and proxies glue it all together. Proxy uses vary:

- Caching
- Request or response alteration
- Interception
- Authentication
- Redirection

Proxies are distinguished, in that they must implement both client and server communication.

1.3 Extension

HTTP servers can execute binary programs via CGI. Process startup times and resource demands gave birth to FastCGI, a binary protocol. The execution of binary programs gives the server more flexibility:

- Dynamic responses
- Database lookups
- Scheduling tasks
- File system interaction
- And more

1.4 Niche

Although most languages have an HTTP package, web servers are specifically designed to be the backbone of any serious infrastructure. On the other hand, application servers handle application logic. The combination of both web servers and application servers decouple the infrastructure and provide a robust architecture.

URL syntax, defined in RFC 3986, provides a sufficient stateless notation. Not only for HTTP, URL syntax finds its way into other services like FTP, Micro-kernel servers, MongoDB, and more. The lesser known fact about URLs is that each segment may contain its own parameters; this is different than query strings:

`http://www.ex.com/foo/bar;param=var/baz.html`¹

1.5 HTTP 2—H2

Since '97, HTTP 1.1 serves the web its portion of hypertext. The many connections required to load hypertext documents demanded improvements. HTTP 2 multiplexes connections; it can send multiple responses, in the same connection, at the same time. The real-time nature of the web, and the pitfalls of Ajax polling, H2 includes server push, allowing servers to directly communicate with clients. Lastly, H2 allows header compression.

2 IPv6

The most significant advantage of IPv6 is the substantial increase in IP address quantity. This mitigates expansions achieved through sub-netting. IPv6 also leverages that every NIC has a unique hardware address, each IP address begins with the NIC's physical address. IPv6 has jumbo frames, frames that can be as large as four gigabytes; this decreases header data, routing, and congestion control overhead.

3 Single Page Applications

Sophisticated web applications like Facebook, YouTube, and other real-time demanding applications take the web to its limits. The web has steered toward reducing server load and increasing responsiveness. Offloading computation onto client devices significantly reduces network traffic, server resources, and provides sufficient response. The disadvantages are maintaining consistency and compatibility across browsers and operating systems; accessibility remains a hot topic.

3.1 Developer's Perspective

Single page applications are more difficult to manage:

¹<https://stackoverflow.com/questions/6444492/can-any-path-segments-of-a-uri-have-a-query-component>

- State
- Architecture
- Security
- Compatibility
- Accessibility

In addition, DOM manipulation and logic can easily become intertwined. Hence, front-end frameworks like Aurelia, AngularJS, react, and VueJS boast solutions. JavaScript is becoming more demanding; ensuring correctness is increasingly difficult in proportion to application size and complexity. Microsoft released TypeScript, a to-JavaScript trans-compiled language, to add static analysis. The world remains optimistic yet skeptical of single page applications, but all look forward to the future of the web.