**Quick word about HTTP/3**

HTTP/2 recap

We’ve seen that HTTP/2 has its advantages. With it, it’s now possible to push more than one resource in a single TCP connection – multiplexing. It thus allows us to minimize the number of request-response cycles and handshakes.

HTTP/2 has now reached around a 50% world adoption rate.

But (ahhh don’t you hate that word) HTTP/2 is not perfect. It uses TCP, and the reliability that TCP provides comes at a cost. There is still overhead from all the round-trips required by handshakes, delivery feedback and ordering guarantees.

This has made TCP a bottleneck of the modern protocol stack.

And this is where UDP steps into the picture.

What is UDP

**U**ser **D**atagram **P**rotocol (UDP) is one part of the Internet Protocol Suite, and dates back to 1980. Yep, it's been around for a long time.

It is a connectionless protocol. This means there are no handshakes and there are no assurances of ordering or delivery. This means that any possible steps for ensuring delivery, data integrity, and other things are left to the application layer (i.e. its left to the browser).

Because UDP has been around for a very long time, it makes it possible to achieve improvements without requiring firmware updates on a wide array of devices connected to the internet, or significant changes in the operating systems.

And this is where HTTP/3 steps into the picture.

What is HTTP/3

HTTP/3 does not use TCP. It uses QUIC.

QUIC (Quick UDP Internet Connections) was created by Google in 2012. It redefines handshakes, reliability features, and security features.

When using HTTP/2, you can use a single TCP connection for multiple streams multiplexed together to transfer data, but when one of those streams suffers a packet loss, the whole connection (and all its streams) are held hostage, so to say, until TCP does its thing (i.e. re-transmits the lost packet).

QUIC is not constrained by this.

With QUIC, building on the on connectionless UDP protocol, the concept of connection does not carry the limitations of TCP and failures of one stream do not have to influence other streams.

**What about support?**

Today, we have HTTP/3 support from major browsers like Google Chrome, Firefox, and Brave. On the infrastructure front, web servers like Litespeed and Nginx both have working implementations of HTTP/3, while network providers like Cloudflare have already deployed full support for HTTP/3.

CONCLUSION

* HTTP/3 uses QUIC, which is built on UDP, while HTTP/2 uses TCP
* As I write this lecture, HTTP/3 is already being used by prominent internet companies such as Google and Facebook and many others
* If you’re using Chrome and connecting to a Google service, you’re probably already using HTTP/3
* HTTP/3 support is optional. When enabled, HTTP/3 can fall back to HTTP/2 or HTTPS if not available
* To use HTTP/3 on your site, you require a web server that supports it
* Don’t worry too much about what it is (it’s not relevant to know about the CRP). All you need to know is that HTTP/3 is the third and most recent version of the HTTP used to exchange information on web