HTTP/2

Hypertext Transfer Protocol (HTTP) is the protocol that governs the connection between your server and the browser of a visitor to your website. It's the first part of most URL's that give you the address of a website. Note on some sites it will be HTTPS, not HTTP (more of later).

Now, for the first time since 1999, we have a **new version of this protocol – HTTP/2**.

In this paper, we'll look at the basics of HTTP/2 as they apply to web design. We will explain some of the key features of the new protocol, look at browser and server compatibility, and detail the things you might wish to think about as we see more adoption of HTTP/2.

First of all, why are we getting a new protocol. Well put simply, HTTP/2 promises faster websites. We produce websites, web-apps and portals so it's quite important to us and we want to ensure our clients can readily understand what it might mean for them.

A brief History

HTTP is an old protocol. It was originally defined in 1991. The last major revision was in 1999 (HTTP/1.1). As you will appreciate, websites in 1999 were very different to websites that we see today. In particular, they are much much bigger. Nobody completely agrees on what the average size is. But many estimates suggest the size of an average homepage is close to 2Mb and has around 100 individual resources required to display it.

HTTP/1.1 – the update done in 1999 does not preform very well when retrieving large numbers of resources. This is the main reason why a further update has been sought. In 2009, Google launched an update of their own called SPDY. Yes – it was meant to sound like SPeeDY - and it set out to provide an enhancement to HTTP. It had a fair degree of take up and it does improve things but it was not universally supported. Possibly because it was from Google and the other companies did not want them to control the new protocol. Microsoft's new Edge browser for instance does not support SPDY. Also, Google themselves are dropping support for it within Chrome.

SPDY is however important in that it has formed the starting point and inspiration for HTTP/2.

For those of a technical bent, it specifically did the following:

*   allows multiplexing - concurrent requests across a single TCP connection
*   allows browsers to prioritize assets so that resources vital to the display of a page could be sent by the server first;
*   compresses and reduce HTTP headers;
*   implements *server push*, whereby a server can push vital resources to the browser before being asked for them.

Current Situation

Right now, all the major browsers offer HTTP/2 support (in their latest versions only). Adoption is also improving; latest figures suggest 12% of websites are now using HTTP/2. Popular sites like YouTube and Facebook have adopted it and Google is very keen to support it across its sites. If you access Gmail on Chrome then you will be using HTTP/2.

We can design and manage your website or web app using the HTTP/2 protocol, as long as your configuration includes the facilities and support required by the standard. In general terms HTTP/2 designed sites will perform faster than 'normal' ones – that's the pay-off for making the change. This is particularly true for sites with HTTPS which tend to run a bit slower than HTTP sites anyway. However, it is not easy to produce a forecast of exactly how much faster a site will be. The best we can say is that if you are using the browser versions that support HTTP/2 that it should be in most situations noticeably faster.

What do you need to make a site ready for HTTP/2

It's worth saying that existing HTTP/1.1 sites will not suddenly become incompatible. HTTP/2 is fully backwards compatible with HTTP/1.1. It's a completely transparent protocol change. What we are really talking about is using the new protocol to enable practices to produce faster websites.

However please note: not all your website visitors will have browser versions that support the protocol anyway, and if they don't they won't get the speed benefits. They will see your website using HTTP/1.1. So importantly, they will still see your website and they won't be particularly disadvantaged given the limitation is more due to their browser rather than your website.

However, to get the main benefits in designing a website there are a number of things that should be done:

1. Whilst it is not a strict requirement within the protocol that the website operates under HTTPS i.e. with SSL protection. Right now, all the main browsers will only use HTTP/2 in preference to HTTP/1.1 **if** the website operates under HTTPS. So, in practical terms, you need an SSL certificate in order to start taking any advantage of HTTP/2. No SSL no HTTP/2…

2. There is no need with HTTP/2 to implement designs that pull images into Sprites or which try to minimise use of lots of small files. In HTTP/2 there is no penalty for having lots of HTTP requests. So managing page resources in HTTP/2 can be done a bit differently.

3. CSS and JavaScript files. The convention with HTTP/1.1 was to try and consolidate these files in order to reduce HTTP requests. Now with HTTP/2 this does not matter and speed-wise, keeping these in a larger number of smaller files that are used as needed, will more likely help speed things up – not slow things down.

There are a few other considerations that are outside the intended scope of this paper that affect how you would design websites under HTTP/2. However, the important thing we would ask readers to appreciate is that we understand the implications of this change. Whilst it is happening now, it is by no means universal and many of your users would likely not see the benefits anyway until they update their browsers.   Where possible we will be designing new websites with HTTP/2 in mind. We can also make changes to older ones as required so they can take more advantage of the new protocol. The single biggest requirement for HTTP/2 is having an SSL certificate applied to the site. Without that, it's currently only going to operate in HTTP/1.1 mode anyway. So if you don't have SSL already, talk to us if you would like it applied to your site (it's not always appropriate).

Lastly, we suggest you beware some of the current web benchmarking tools. Many of them seek to penalise HTTP requests when applying scoring, which in the new world of HTTP/2

does not matter. Indeed your HTTP/2 designed site will likely be designed to use more HTTP requests – not less. As the standard becomes more adopted this will of course change.

Do contact us if you wish to discuss any of the technical aspects of your website, app or portal.