

# Things you should know about HTTP / 2 protocol

The HTTP version we are using today is HTTP / 1.1, which has been around since 1999, and will soon be replaced by a newer standard, HTTP / 2. This update promises to improve the way browsers and servers communicate with each other, thereby providing faster data transfer speeds while minimizing the power needed to handle access requests.

**Hypertext Transfer Protocol is a method that the browser uses to request and retrieve information from the server where the website is located, and then display the content to us. The HTTP version we are using today is HTTP / 1.1, which has been around since 1999, and will soon be replaced by a newer standard, HTTP / 2. This update promises to improve the way browsers and servers communicate with each other, thereby providing faster data transfer speeds while minimizing the power needed to handle access requests.**

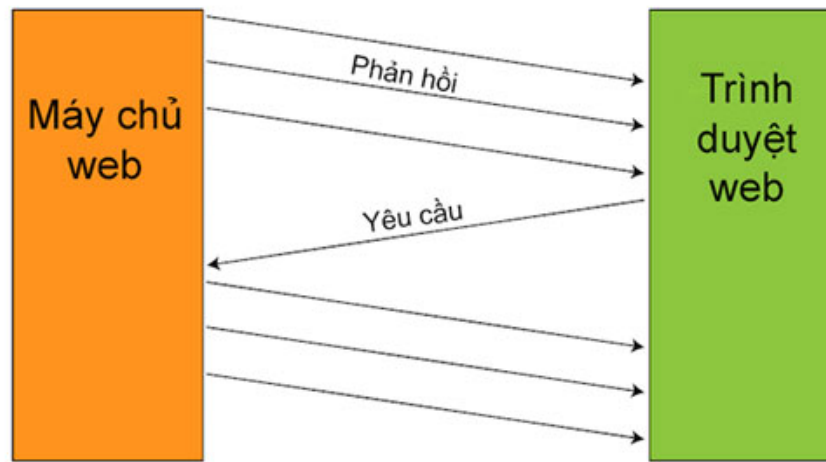
## HTTP / 2 standard has completed draft technical configuration

A brief history of HTTP, this is a protocol used to exchange and convey hypertext - are structured character segments used to link this node to another node of an information system. An example of a hypertext is a piece of text in which links are embedded to help you go from site to website. HTTP appears in many places but it is most widely known in the World Wide Web, which is the web world that we are using every day.

HTTP is jointly developed by the Internet Technical Association (IETF) and World Wide Web Organization (W3C). HTTP / 1.1 was released in June 1999, and is also the popular version of HTTP currently used. By February 2015, HTTP / 2 completed the draft development phase and is awaiting confirmation.

## Why is HTTP / 2 important?

As mentioned above, because HTTP / 1.1 was developed so long ago, it began to reveal the weaknesses of that "aging". Today's websites not only have simple HTML code, but they are also combined with tools for decoration - design (CSS), executable code (JavaScript), images, videos, even Flash content. To transmit that amount of data, the browser has to create a lot of connections to the server, each connection contains a lot of information about who the source is, where it is taken, what content it takes, how it is taken, and many other things half. All of this has created a huge amount of work on our browser ( *or mobile device* ) as well as the server where the website is located.



*The browser will send requests for content (HTTP request) to the server, the server will respond (HTTP response) to the content that the browser needs.*

The above connections, combined with the processing power needed to set up, resolve information requests for each connection, can make our site load much slower. The more things the website contains, the more serious this situation becomes. For our users, it requires increasing speed as our Internet connection is getting better, so only a few seconds of delay when loading web pages can be frustrating. For businesses, slow-loading website means losing money ( *think of online shopping sites, slow pages, no one will buy and people will switch to using rival services* ). It also makes the user experience a lot worse.

People have tried to find ways to speed up the Internet since the dial-up connection was popular and not ADSL. One of the most common techniques is to use the buffer ( *cache, verb is caching* ). With this technique, some content will be stored on the user's device, then the browser just downloads the new content instead of having to reload everything from the beginning. In addition, people also think about reducing the size of photos and video, some others spend a lot of time optimizing the source code to cut down the milliseconds of page loading time.

These things are quite useful, but it is just like a bandage bandage that 'helps to *relieve pain* ' but does not solve the problem. So Google and some other companies decided to refresh HTTP / 1.1 to create SPDY ( *read speedy* ), a new transmission technology to get better results. Overall, communication between server and browser with SPDY is much faster even when encryption tools are applied. At the lowest level, the data transfer rate with SPDY can be about 10% faster, and in some cases it can improve up to 40%.

The impressive result above made Google engineering group decided to develop a new protocol set based on SPDY technology, and this is the starting source of HTTP / 2.

## More information about SPDY

SPDY ( *read: "Speedy"* ) is a network protocol developed primarily by Google to convey web content. SPDY will perform compression and prioritization operations to communicate more effectively, while allowing the browser to send multiple requests to multiple servers in one connection. SPDY has the main goal of reducing the delay when loading web and improving security. Google now integrates SPDY into **Chrome, Firefox, Internet Explorer, Opera** , server companies and many large websites have also incorporated this protocol into their products. Google itself also uses SPDY for its services and speeds them up to **20% to 40%** from before.

Note that SPDY is not designed to replace HTTP; instead, it refines how the HTTP protocol sends and receives requests to download content between the browser and the server. When sent via SPDY, a request ( *HTTP request* ) will be processed, stamped, simplified and compressed to match, thereby increasing the transmission speed. Below is a connection diagram using Google's SPDY proxy .

## What is the protocol?

HTTP in general or HTTP / 2 in particular is a protocol. You can think of the term ' *protocol* ' as a set of rules that costs how data is transferred from one computer to another, even the server. Each protocol will vary slightly, but together, they will create data packets consisting of three parts:



1. Header: Contains information about the starting address of the package, destination address and some additional information such as data type, data volume, .
2. Payload: Contains the main data to be transmitted
3. Footer: Contains some information used to detect and correct errors during transmission.

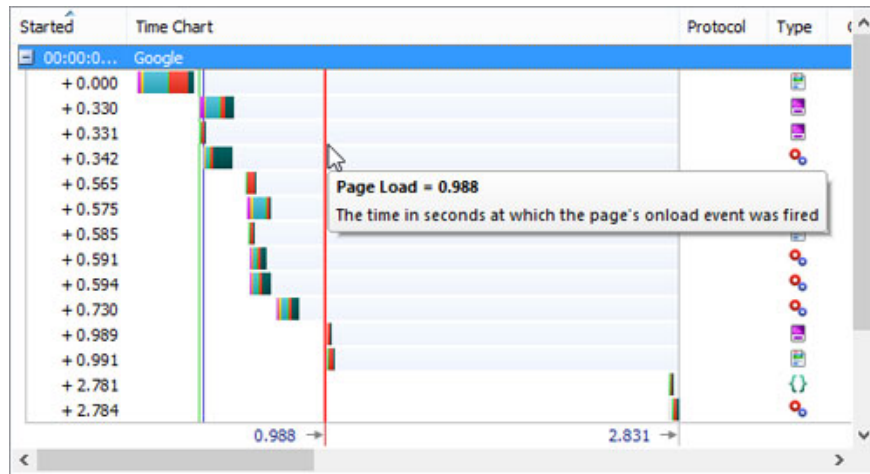
You can imagine the protocol the same way you send it. The header is the outer envelope containing the name and address of the sender and the recipient, and you can note that you send something inside. Payload is the actual letter you need to send, and Footer is like a tape you use to seal the envelope. If the tape is torn, you know that the contents may have been tampered with and something has happened.

Some protocols also support a technology called ' *encapsulation* '. It allows people to add other protocols embedded within the payload of the packet. This is how HTTP in particular and other network protocols transmit data.

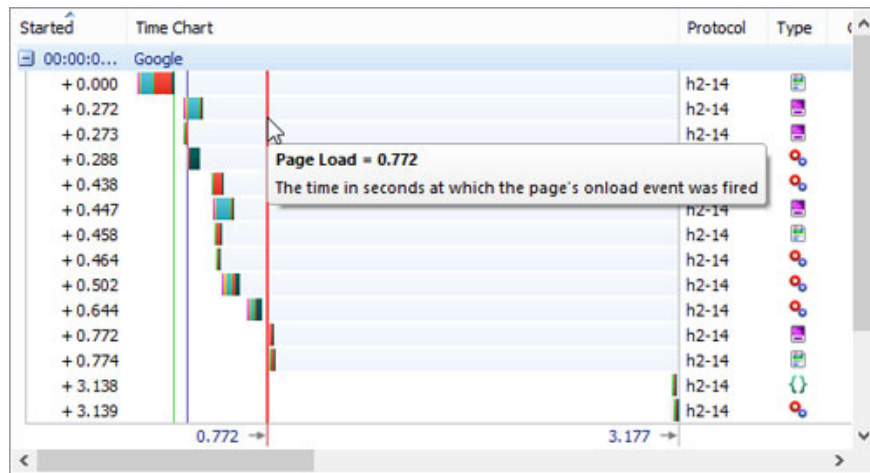
## Why is HTTP / 2 better?

In short, HTTP / 2 helps websites upload faster, saving time for everyone. Simple as that.

In the example below is the data posted by HTTPWatch. It shows that the website's transmission time when using HTTP / 2 increased by 20%, and it is worth mentioning that the server used for testing has not been optimized for this new protocol. If everything is optimized over time, it is normal to improve 30% of load time.



*Page load time by HTTP / 1.1*



*Page load time with HTTP / 2*

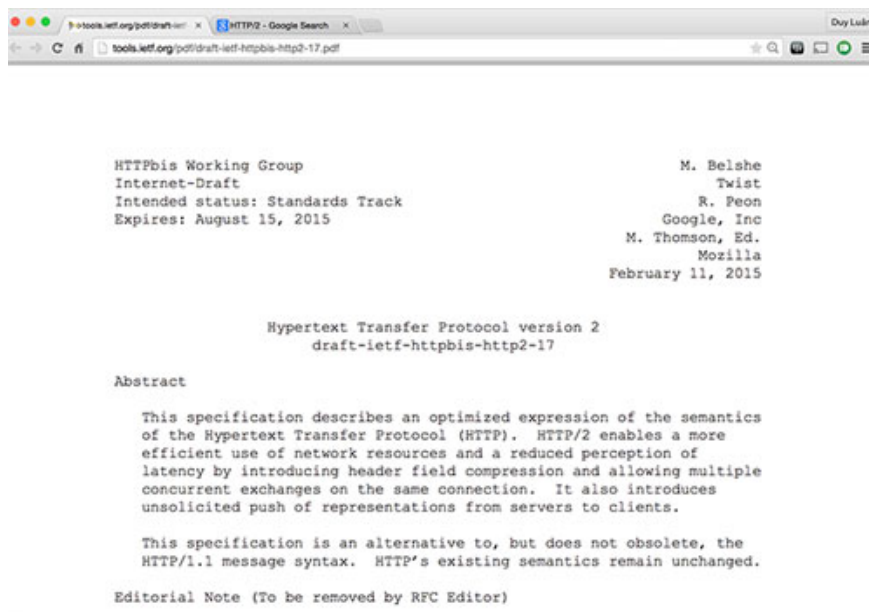
HTTP / 2 improves the main speed by creating a persistent connection between the server and the browser, instead of establishing a new connection each time the browser needs something. This method significantly reduces the amount of information transmitted. In addition, HTTP / 2 also transmits data in binary form instead of the text format as before. This means that your browser does not have to waste time translating text into a format that it can understand.

Some other techniques that HTTP / 2 owns to speed up include multiplexing ( *sending and receiving multiple information at the same time* ), prioritization ( *hitting priority order, important data will be sent first* ), compression ( *compressed to shrink the packet size* ) and push server ( *the server will guess the next data request you need to send the information in advance to save time* ).

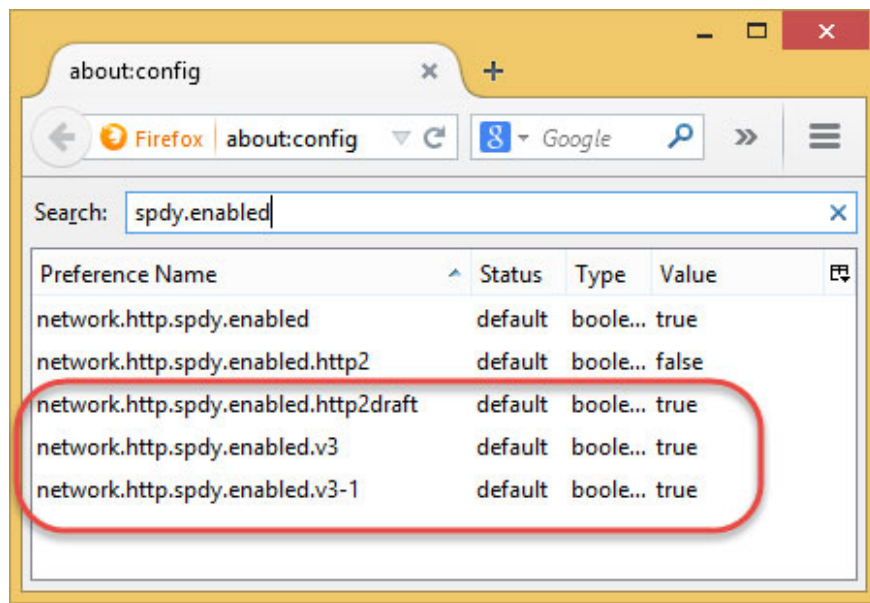
**So when can we see the benefits of HTTP / 2?**

There is currently no specific start time for deploying using HTTP / 2, and many people may be using it without knowing it. The draft configuration of HTTP / 2 published on February 11 will expire in the next 6 months ( *specifically August 15* ). Previously, it needed to be validated and transformed into a document ( *called 'RFC'* ) for developers around the world to research and develop. If not, a new draft configuration will be issued with the necessary changes.

Note that RFC is " *Request For Comments* ", which means ' *request comments* ', but it is actually the name that the IETF organization uses to call a completed document. is a mandatory requirement that websites or servers must follow, instead it suggests how related issues should be designed, of course, for a protocol to work, all people will have to follow certain general rules.



Currently HTTP / 2 technology has been integrated into many servers and browsers, even if it is only a draft. For example, Microsoft has included HTTP / 2 in Internet Explorer in the **Windows 10 Technical Preview** , Chrome has been supporting it for some time ( *although it is disabled by default* ), and Firefox is not far from the protocol either. this new



If we talk about server side, IIS ( a Windows / Microsoft web server solution ) will support HTTP / 2 in the upcoming Windows 10. Other well-known solutions such as Apache or Nginx will soon have this capability ( *although it is now possible to install extensions to use SPDY* ). This means that sooner or later, we will all switch to using HTTP / 2. And most likely you will not even notice that change unless you have the habit of calculating the loading time of websites. Every day, the page load time will be shortened. Also, since you only see 'http' or 'https' in the address bar, life will continue as ever, but it will be a little faster.

You finished reading the article "**Things you should know about HTTP / 2 protocol**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.