

Waterfall Process

TCP Connection

The TCP connection time involved the timestamps `connectStart`, `secureConnectionStart` and `connectEnd`.

connectStart - `connectStart` is the moment in time immediately before the user agent starts establishing a connection to a server to retrieve a resource. There is a difference between `fetchStart` and `connectStart`. Not all resources may need to connect to be retrieved. The metric `fetchStart` is the beginning of the larger process of fetching data, for example if the resource is already cached there is no connection required, or if a connection that has already been set using `keepalive` / `persistent` connection.

secureConnectionStart - `secureConnectionStart` is the moment in time immediately before the handshake process takes place to secure the connection. The main purpose of this metric is to create a more precise measurement of how long the secure handshake process takes.

If we subtract `secureConnectionStart` from `connectEnd` we will have the SSL time.

connectEnd - `connectEnd` is the time immediately after the browser finishes establishing the connection to the server to retrieve the current document. When you are measuring a network connection, it is helpful to separate the connection itself from the request / response.

When `connectEnd` is combined with the `connectStart` event the entire TCP connection time can be measured. When `connectEnd` is combined with the `secureConnectionStart` event the entire SSL handshake connection time can be measured.

HTTP Connection

The HTTP request / response timeline includes `requestStart`, `responseStart` and `responseEnd`.

requestStart - `requestStart` is the moment the browser requests the current document from the server, relevant application caches, or from local resources. The `requestStart` is the starting point of a HTTP request / response transaction. When

requestStart is reached, the HTTP request for a resource actually takes place. Establishing a start point for the request / response can allow us to examine the HTTP process with precision.

responseStart - responseStart is the time immediately after the browser receives the first byte of the response from the server, or from relevant application caches or from local resources. When a resource is retrieved via the network (rather than the application cache) responseStart represents part of the HTTP request / response timeline.

It is the end time for time to first byte TTFB. Time to first byte is reported by most performance tools and can help a webmaster understand where problems exist with their slow pages. It is the end time of request duration.

Combined with requestStart, responseStart is used to measure the duration of a request.

responseEnd - responseEnd is the time immediately after the user agent receives the last byte of the current document or immediately before the transport connection is closed, whichever comes first. The responseEnd time is measured regardless of the source (server, relevant application caches or from local resources). As such, several web performance metrics "end" with responseEnd. For webmasters it is basically the time where you now can begin determining page speed issues that deal directly with the browser (rendering issues) and the end of the network / retrieval page speed issues (redirects/DNS/file size/etc.).

Document Object Model (DOM)

The DOM is a W3C (World Wide Web Consortium) standard. The DOM defines a standard for accessing documents, the W3C Document Object Model (DOM) is a platform and language-neutral interface that allows programs and scripts to dynamically access and update the content, tree structure, and style of a HTML, XHTML, or XML document.

domLoading - domLoading is the time immediately before the user agent sets the current document readiness to 'loading'. It means the browser has the document and is about to begin processing it.

Other sub-resources such as CSS or images for the page are not a factor yet because the initial document is has just begun parsing, so the browser does not know if they exist. The html itself is currently downloaded (the browser has the html), but it is still being processed.

Since this is the first moment of the actual browser process, it can be used as the start time of measuring how pages are being rendered by a browser.

domInteractive - domInteractive is the time immediately before the user agent sets the current document readiness to 'interactive'. The browser has finished parsing all of the HTML and DOM construction from that HTML complete. It is the beginning of the browser dealing with page sub-resources like CSS and images.

domContentLoaded - domContentLoaded is the point when both the DOM is ready and there are no CSS stylesheets or image related issues that are blocking JavaScript execution. This event typically marks when both the DOM and CSSOM are ready and that the render tree can now be built.

If there is no parser blocking JavaScript then DOMContentLoaded will fire immediately after domInteractive.

This event is useful in optimizing the critical rendering path by measuring the affect of parser blocking JavaScript. If you want DOM to get parsed as fast as possible after the user had requested the page, you should turn your JavaScript asynchronous and optimize loading of stylesheets which slow down page load due to being loaded in parallel with the main HTML document.

domComplete - domComplete is the time immediately before the user agent sets the current document readiness to 'complete'. It simply means the HTML page and all of its sub-resources (CSS, images, JavaScript) have been parsed, loaded, and are ready.

onLoad event - the content is loaded