

## What are Long-Polling, Websockets, Server-Sent Events (SSE) and Comet?

[Ask Question](#)

I have tried reading some articles, but I am not very clear on the concepts yet.

Would someone like to take a shot at explaining to me what these technologies are:

1. Long Polling
2. Server-Sent Events
3. Websockets
4. Comet

One thing that I came across every time was, the server keeps a connection open and pushes data to the client. How is the connection kept open, and how does the client get the pushed data? (How does the client use the data, maybe some code might help?)

Now, which one of them should I use for a real-time app. I have been hearing a lot about websockets (with socket.io [a node.js library]) but why not PHP?

[php](#) [websocket](#) [comet](#) [long-polling](#) [server-sent-events](#)

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

edited Nov 12 '16 at 21:27



DOK

27.8k 7 52 88

asked Jun 18 '12 at 6:28



user1437328

6,130 9 26 38

- 
- 1 Realtime websocket or webrtc? There is a library for websocket in php, you do need to write extra code in order for it to work using ZMQ or just socket programming, nodeJs is built for this so its easily available. The reason websocket is not readily available in php is that you have to run an extra terminal and kept it running so that websocket server is readily available, you will have two servers bottom line. and the structure, php is not an event structure like javascript so there's that, websocket uses a event structure in order to catch and send messages. – [PauAI](#) Apr 11 '17 at 0:31

---

Additionally: Comet and ServerSent Events are PHP's workaround of achieving almost realtime(not really) without creating 2 servers. – [PauAI](#) Apr 11 '17 at 0:49

---

### 3 Answers

---

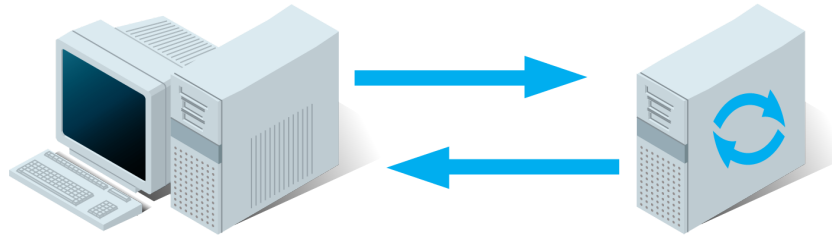
*In the examples below the client is the browser and the server is the webserver hosting the website.*

Before you can understand these technologies, you have to understand *classic* HTTP web traffic first.

#### Regular HTTP:

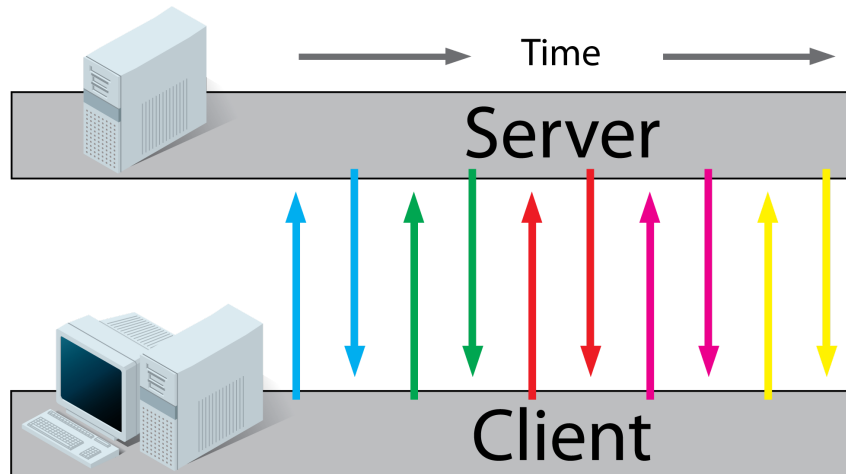
1. A client requests a webpage from a server.
2. The server calculates the response
3. The server sends the response to the client.

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).



## Ajax Polling:

1. A client requests a webpage from a server using regular HTTP (see HTTP above).
2. The client receives the requested webpage and executes the JavaScript on the page which requests a file from the server at regular intervals (e.g. 0.5 seconds).
3. The server calculates each response and sends it back, just like normal HTTP traffic.

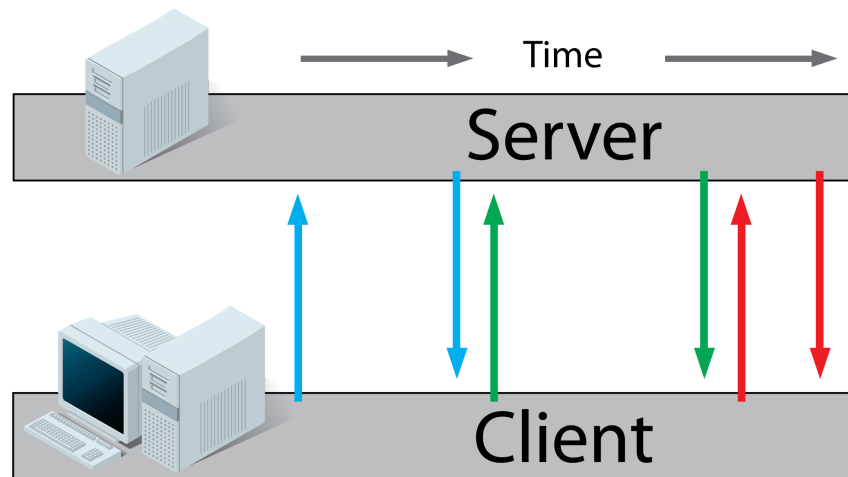


## Ajax Long-Polling:

1. A client requests a webpage from a server using regular HTTP (see HTTP above).

server.

3. The server does not immediately respond with the requested information but waits until there's **new** information available.
4. When there's new information available, the server responds with the new information.
5. The client receives the new information and immediately sends another request to the server, re-starting the process.



## HTML5 Server Sent Events (SSE) / EventSource:

1. A client requests a webpage from a server using regular HTTP (see HTTP above).
2. The client receives the requested webpage and executes the JavaScript on the page which opens a connection to the server.
3. The server sends an event to the client when there's new information available.
  - Real-time traffic from server to client. mostly that's what

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

Home

PUBLIC

 Stack Overflow

Tags

Users

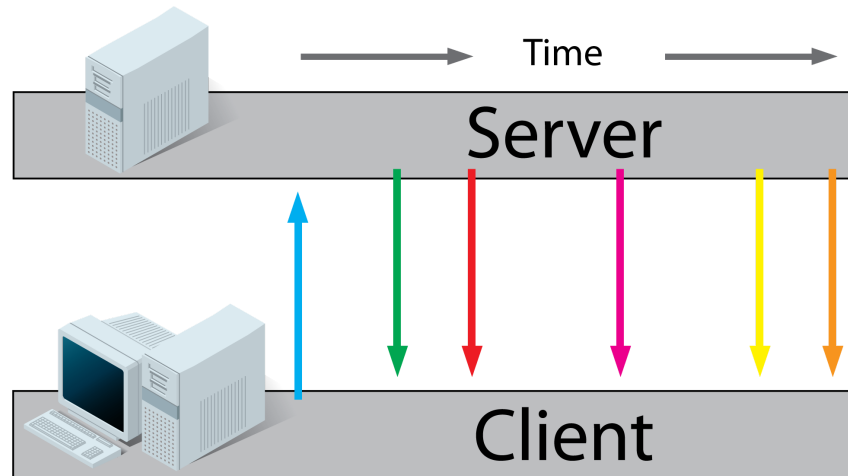
Jobs

**Teams**  
Q&A for work



[Learn More](#)

- Not possible to connect with a server from another domain
- If you want to read more, I found these very useful:  
[\(article\)](#), [\(article\)](#), [\(article\)](#), [\(tutorial\)](#).



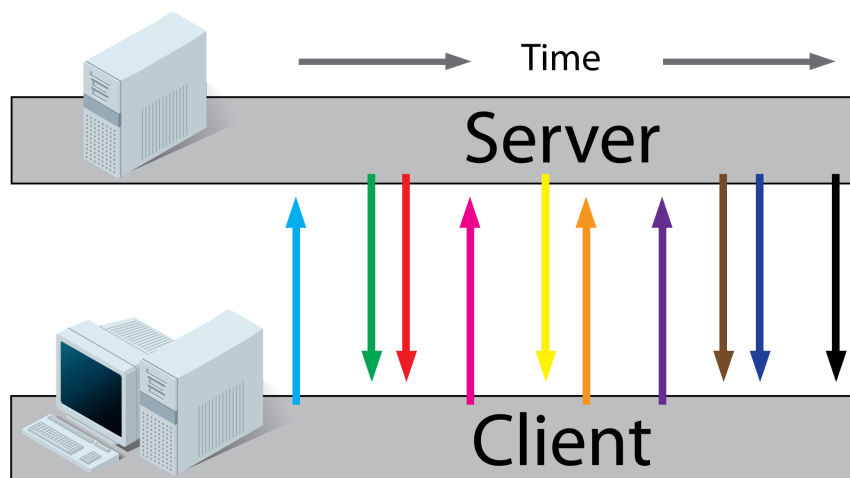
### HTML5 Websockets:

1. A client requests a webpage from a server using regular http (see HTTP above).
2. The client receives the requested webpage and executes the JavaScript on the page which opens a connection with the server.
3. The server and the client can now send each other messages when new data (on either side) is available.
  - Real-time traffic from the server to the client **and** from the client to the server
  - You'll want to use a server that has an event loop
  - With WebSockets it is possible to connect with a server from another domain.
  - It is also possible to use a third party hosted websocket

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

only have to implement the client side, which is very easy!

- If you want to read more, I found these very useful: ([article](#)), ([article](#)) ([tutorial](#)).



### Comet:

Comet is a collection of techniques prior to HTML5 which use streaming and long-polling to achieve real time applications. Read more on [wikipedia](#) or [this](#) article.

Now, which one of them should I use for a realtime app (that I need to code). I have been hearing a lot about websockets (with socket.io [a node.js library]) but why not PHP ?

You can use PHP with WebSockets, check out [Ratchet](#).

edited Nov 11 '17 at 11:32



38.9k 14 73 122

- 
- 14 This is awesome! I am reading up on SSE and found this article, it's very nice - like I've now compared stuff, can you also include SSE here so we can also cross-check it's difference with Websocket? – [index](#) Nov 7 '12 at 7:34
- 
- 1 @Tieme Oh was that it? I thought SSE meant Server-Sent Events. Anyway, thanks, I see it now. – [index](#) Nov 16 '12 at 3:28
- 
- 1 Q: in php let's say you were using websocket would every client connected to my server using ws: would have one thread allocated to him/her and its size would be ~2mb as is the case with normal requests? how would that differ in nodejs? How many concurrent clients can nodejs handle and when it breaks what happens? – [Muhammad Umer](#) Jul 13 '13 at 7:13
- 
- 5 You can accomplish the same with both solutions but the mechanism is different. Long-polling uses 'regular' http data, SSE uses a different underlying protocol and needs a different server setup compared to long-polling. – [Tieme](#) Oct 23 '13 at 7:53
- 
- 2 Well you could use apache if you want. But a lot of people use Node.js because it has an event loop. But for Apache, see [stackoverflow.com/questions/12203443/...](https://stackoverflow.com/questions/12203443/) – [Tieme](#) Nov 15 '13 at 9:45
- 

---

Tieme put a lot of effort into his excellent answer, but I think the core of the OPs question is how these technologies relate to PHP rather than how each technology works.

PHP is the most used language in web development besides the obvious client side html, css, and javascript. Yet PHP has 2

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

1) PHP started as a very basic CGI. PHP has progressed very far since it's early stage, but it happened in small steps. PHP already had many millions of users by the time it became the embed-able and flexible C library that it is today, most of whom were dependent on it's earlier model of execution, so it hasn't yet made a solid attempt to escape the cgi model internally. Even the commandline interface invokes the PHP library (libphp5.so on linux, php5ts.dll on windows, etc) as if it still a cgi processing a GET/POST request. It still executes code as if it just has to build a "page" and then end it's life cycle. As a result, it has very little support for multi-thread or event driven programming (within PHP userspace), making it currently unpractical for real time, multi-user applications.

Note that PHP does have extensions to provide event loops (such as libevent) and threads (such as pthreads) in PHP userspace, but very, very, few of the applications use these.

2) PHP still has significant issues with garbage collection. Although these issues have been consistently improving (likely it's greatest step to end the life cycle as described above), even the best attempts at creating long running PHP applications require being restarted on a regular basis. This also make it unpractical for real time applications.

PHP 7 will be a great step to fix these issues as well, and seems very promising as a platform for real-time applications.

edited May 7 '15 at 2:15

answered Oct 14 '14 at 7:11



JSON

1,215 13 22

---

1 One small correction: PHP was always written in C. as can be seen

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).



@IMSoP - Thanks for the correction, I've been using PHP for over a decade and have always been under the impression that it's roots were in Perl. The PHP [history](#) page clearly supports that it was originally C as well. I'll edit my answer once I find a moment. – JSON Dec 15 '14 at 2:45

Please edit your answer regarding roots of php. – Netverse Feb 4 '15 at 6:05

1 @JSON [twitter.com/rasmus/status/226405807305138176](https://twitter.com/rasmus/status/226405807305138176) – eis Nov 1 '15 at 20:11

1 though also [svn.php.net/viewvc/phpdoc/en/trunk/appendices/...](http://svn.php.net/viewvc/phpdoc/en/trunk/appendices/...) && [web.archive.org/web/20090426061624/http://us3.php.net/...](http://web.archive.org/web/20090426061624/http://us3.php.net/...) – eis Nov 1 '15 at 20:19 ✎

I have tried to make note about these and have collected and written examples from a **java perspective**.

[HTTP for Java Developers](#)

[Reverse Ajax - Old style](#)

[Async Handling on server side](#)

[Reverse Ajax - New style](#)

[Server Sent Events](#)

Putting it here for any java developer who is looking into the same subject.

edited Dec 7 '17 at 9:08

answered Apr 21 '17 at 14:47

 .John

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

**protected** by [Shankar Damodaran](#) Jul 9 '14 at 8:11

Thank you for your interest in this question. Because it has attracted low-quality or spam answers that had to be removed, posting an answer now requires 10 [reputation](#) on this site (the [association bonus does not count](#)).

Would you like to answer one of these [unanswered questions](#) instead?

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).