

Web Service versus regular Http Request

Asked 16 years, 2 months ago Modified 16 years, 2 months ago

Viewed 5k times



2

About 5000 computers will be making a call to a central server, and they will be passing in a GUID to the central server.



The server will then return True/False back to the client.



Is there a big difference in **performance** between a web service and a regular Http request to a Url on the server?



performance

web-services

Share

Improve this question

Follow

asked Oct 9, 2008 at 20:29



Anonymous Cow

You really need to provide a lit more details about the actual web services protocol and message payload you're talking about. REST is HTTP -- so there's no difference. What protocol are you talking about? – [S.Lott](#) Oct 9, 2008 at 22:23

8 Answers

Sorted by:

Highest score (default)





4

Yeah, a SOAP envelope is relatively hefty. I'd suggest a REST-based service that will keep the size of data being marshaled around to a minimum.



Share Improve this answer

answered Oct 9, 2008 at 21:05

Follow



Kon

27.4k ● 12 ● 63 ● 87



+1 agree with this, just walk it through... parsing a XML payload then either building an XML response or sending a cached XML response versus parsing a URL and sending either a 1/true/json body back. – David Feb 17, 2009 at 3:03

what about android i have to use soap since i cant use rest right – shareef May 20, 2013 at 5:24

Why not? stackoverflow.com/questions/6047194/... – Kon May 20, 2013 at 13:57



2

I assume by Web Service you mean SOAP. My experience with various SOAP stacks on different platforms (Java, .NET, Ruby, PHP) is that in this case you're probably looking at an order of magnitude difference in processing such a simple message. There's usually a good deal of overhead with SOAP, which is negligible if you're passing large messages, but overkill for small messages. Using SOAP for this is like an elephant carrying a penny. I would recommend just a simple HTTP handler in this case.



Are all 5000 clients going to be hitting the server at one time? Do you need to guarantee a certain response time?

Share Improve this answer

answered Oct 9, 2008 at 20:58

Follow



[Dave Dunkin](#)

1,097 ● 9 ● 13

The order of magnitude difference may still be acceptable: 20ms vs 100ms. That's why it depends on what kind of load the server will be getting. – [Dave Dunkin](#) Oct 9, 2008 at 21:04

Taking performance aside for a second - the simplicity of a handler from a maintenance PoV is really desireable.

– [stephbu](#) Oct 9, 2008 at 21:14



1



REST web services **are** HTTP.

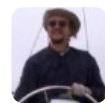
Consequently, I don't understand the question. Perhaps you should provide more information on the protocol, the messages, whether it's RPC-style or document-style, how big the document is, etc.



Share Improve this answer

answered Oct 9, 2008 at 20:47

Follow



[S.Lott](#)

391k ● 82 ● 517 ● 788

the problem is that the 'web-services' buzzword is frequently taken to mean SOAP or other forms of reinventing RPC over HTTP. REST means 'back to the basics' with a few conventions – [Javier](#) Oct 9, 2008 at 21:49

@Javier: the point is not what it frequently means. The point is that the question is bad and unanswerable in that form.



0



I am not 100% sure on a performance benefit in response time, but a WebService request that returns just the true false, rather than a regular http request and then parsing the response I'm guessing would be more efficient overall.



Share Improve this answer

answered Oct 9, 2008 at 20:30



Follow



[Mitchel Sellers](#)

63.1k ● 15 ● 114 ● 174

An HTTP request doesn't mean it will return HTML. – [Neall](#)
Oct 9, 2008 at 22:33



0



I have an app that currently has about 7000 machines calling a .net web service using WCF.

Each machine makes the call once every 15 minutes. The service takes the data and shoves it into SQL server; which is installed on the same box.



Right now it's collecting about 350MB of data a day.



The load is barely registering and we're in the process of rolling it out to 25,000 clients.

I guess my point is that passing a guid to the server with a true / false value coming back is not a whole lot of load to be worried about unless the web server was a POS 5 years ago.

Share Improve this answer

answered Oct 9, 2008 at 20:43

Follow



ChrisLively

88k ● 27 ● 173 ● 247

4 calls per hour x 24 hours = 96 calls per computer x 7000 computers = 672K web service calls per day. Are those calls scattered or every 15 minutes you can hit by 7K calls?

– Anonymous Cow Oct 9, 2008 at 20:56

They are scattered because it's 15 minutes from the time the service starts until it is stopped. Meaning it's completely based on the time the machine starts and therefore random.

– ChrisLively Oct 10, 2008 at 19:29



0



I would think the difference wouldn't much if any difference. The HttpRequest may actually be faster just because its using one less layer in the stack. If you see yourself expanding the services in the future, you might go ahead and use WebService, not because of performance (again the performance difference is probably negligible), but because the WebService is going to be more maintainable as services get more complex.

Share Improve this answer

answered Oct 9, 2008 at 20:45

Follow



Steve g

2,489 ● 17 ● 17



Realistically it won't make much difference. Out of the things that could introduce latency in this request you

0

have:



- Compiled code executing
- Network round-trips
- Database access (presumably that's what you'll be checking against?)



Even if you have a blindingly fast network and database server, the amount of time spent doing the the network round trip and database access (which possibly may involve another network round trip) will render the overhead of executing the compiled code of whatever web service framework you use insignificant.

Share Improve this answer

answered Oct 9, 2008 at 21:25

Follow



[Greg Beech](#)

136k ● 45 ● 209 ● 250



-1

SOAP and REST Web Services imply some overhead, but are certainly the way to go if you think you'll need to scale up to returning some other information other than true/false.



An HTTP response of just 1/0 or true/false will be much smaller, and therefore theoretically faster.



Share Improve this answer

answered Oct 9, 2008 at 21:14

Follow



[Seibar](#)

70.2k ● 38 ● 93 ● 100

a well-designed REST is exactly an HTTP response. – [Javier](#)
Oct 9, 2008 at 21:46

What overhead is there with a REST-based service call?
– [Kon](#) Oct 10, 2008 at 13:21
