Which is faster, python webpages or php webpages? [closed]

Asked 16 years, 3 months ago Modified 5 years, 11 months ago

Viewed 59k times Part of PHP Collective

▲

42

▼

Closed. This question is opinion-based. It is not currently accepting answers.

Want to improve this question? Update the question so it can be answered with facts and citations by editing this post.

Closed 5 years ago.

Improve this question

Which is faster, python webpages or php webpages?

Does anyone know how the speed of pylons(or any of the other frameworks) compares to a similar website made with php?

I know that serving a python base webpage via cgi is slower than php because of its long start up every time.

I enjoy using pylons and I would still use it if it was slower than php. But if pylons was faster than php, I could maybe, hopefully, eventually convince my employer to allow me to convert the site over to pylons.

PHP php python performance pylons

Share

Improve this question

Follow

edited Dec 23, 2018 at 11:20

user3942918

26.4k ● 13 ● 56 ● 68

asked Sep 16, 2008 at 21:05

Echo says Reinstate Monica

3,917 ● 6 ● 38 ● 40

10 Answers

Sorted by: Highest score (default)

90

✓

It sounds like you don't want to compare the two **languages**, but that you want to compare two **web systems**.

This is tricky, because there are many variables involved.

For example, Python web applications can take advantage of [mod_wsgi](#) to talk to web servers, which is faster than any of the typical ways that PHP talks to web servers (even mod_php ends up being slower if you're using Apache, because Apache can only use the Prefork MPM with mod_php rather than multi-threaded MPM like Worker).

There is also the issue of code compilation. As you know, Python is compiled just-in-time to byte code (.pyc files) when a file is run each time the file changes. Therefore, after the first run of a Python file, the compilation step is skipped and the Python interpreter simply fetches the precompiled .pyc file. Because of this, one could argue that Python has a native advantage over PHP. However, optimizers and caching systems can be installed for PHP websites (my favorite is eAccelerator) to much the same effect.

In general, enough tools exist such that one can pretty much do everything that the other can do. Of course, as others have mentioned, there's more than just speed involved in the business case to switch languages. We have an app written in oCaml at my current employer, which turned out to be a mistake because the original author left the company and nobody else wants to touch it. Similarly, the PHP-web community is much larger than the Python-web community; Website hosting services are more likely to offer PHP support than Python support; etc.

But back to speed. You must recognize that the question of speed here involves many moving parts. Fortunately, many of these parts can be independently optimized, affording you various avenues to seek performance gains.

Share Improve this answer Follow

edited Mar 31, 2013 at 12:54

Kai
423 ● 5 ● 16

answered Sep 17, 2008 at 3:44

Ross
1,425 ● 10 ● 10

18 The benefit of compiled .pyc byte code files in Python is overrated when using something like Apache/mod_wsgi. This is because they only come into play at time module is first loaded. After that it is all cached in memory between requests. Byte code caching systems are a bigger deal for PHP because the application is effectively thrown away after every request and so it has to reload code for every request. – Graham Dumpleton Jun 25, 2009 at 2:26

31

There's no point in attempting to convince your employer to port from PHP to Python, especially not for an existing system, which is what I think you implied in your question.

The reason for this is that you already have a (presumably) working system, with an existing investment of time and effort (and experience). To discard this in favour of a trivial performance gain (not that I'm claiming there would be one) would be foolish, and no manager worth his salt ought to endorse it.

It may also create a problem with maintainability, depending on who else has to work with the system, and their experience with Python.

Share Improve this answer Follow

answered Sep 16, 2008 at 21:24

Rob
48.4k ● 5 ● 75 ● 93

1 Any benchmarks I have seen as well show python underperforming vs newer versions of PHP. Although I have yet to see someone compare performance in a full application. – michael.schuett Dec 17, 2014 at 20:29

▲

8

▼

I would assume that PHP (>5.5) is faster and more reliable for complex web applications because it is optimized for website scripting.

Many of the benchmarks you will find at the net are only made to prove that the favoured language is better. But you can not compare 2 languages with a mathematical task running X-times. For a real benchmark you need two comparable frameworks with hundreds of classes/files an a web application running 100 clients at once.

Share Improve this answer Follow

answered Nov 10, 2015 at 9:47

RubbelDeCatc

755 ● 1 ● 8 ● 11

▲

3

▼

PHP and Python are similiar enough to not warrent any kind of switching.

Any performance improvement you might get from switching from one language to another would be vastly outgunned by simply not spending the money on converting the code (you don't code for free right?) and just buy more hardware.

Share Improve this answer

Follow

answered Sep 16, 2008 at 21:16

SCdF

59.3k ● 24 ● 79 ● 114

3

It's about the same. The difference shouldn't be large enough to be the reason to pick one or the other. Don't try to compare them by writing your own tiny benchmarks (`"hello world"`) because you will probably not have results that are representative of a real web site generating a more complex page.

Share Improve this answer

Follow

edited Apr 9, 2014 at 11:15

tshepang

12.4k ● 25 ● 95 ● 139

answered Sep 16, 2008 at 21:13

indentation

9,985 ● 6 ● 23 ● 14

2

If it ain't broke don't fix it.

Just write a quick test, but bear in mind that each language will be faster with certain functions then the other.

Share Improve this answer

Follow

answered Sep 16, 2008 at 21:06

UnkwnTech

90.7k ● 65 ● 191 ● 234

2

You need to be able to make a business case for switching, not just that "it's faster". If a site built on technology B costs 20% more in developer time for maintenance over a set period (say, 3 years), it would likely be cheaper to add another webserver to the system running technology A to bridge the performance gap.

Just saying "we should switch to technology B because technology B is *faster!*" doesn't really work.

Since Python is far less ubiquitous than PHP, I wouldn't be surprised if hosting, developer, and other maintenance costs for it (long term) would have it fit this scenario.

Share Improve this answer Follow

answered Sep 16, 2008 at 21:11
Peter Bailey
106k ● 32 ● 185 ● 206

2

an IS organization would not ponder this unless availability was becoming an issue.

if so the case, look into replication, load balancing and lots of ram.

Share Improve this answer Follow

answered Feb 4, 2009 at 6:28
sara palin

The only right answer is "It depends". There's a lot of variables that can affect the performance, and you can

0

optimize many things in either situation.

Share Improve this answer Follow

answered Sep 16, 2008 at 21:08

Aeon
6,527 ● 5 ● 31 ● 31

-1

I had to come back to web development at my new job, and, if not Pylons/Python, maybe I would have chosen to live in jungle instead :) In my subjective opinion, PHP is for kindergarten, I did it in my 3rd year of uni and, I believe, many self-respecting (or over-estimating) software engineers will not want to be bothered with PHP code.

Why my employers agreed? We (the team) just switched to Python, and they did not have much to say. The website still is and will be PHP, but we are developing other applications, including web, in Python. Advantages of Pylons? You can integrate your python libraries into the web app, and that is, imho, a huge advantage.

As for performance, we are still having troubles.

Share Improve this answer Follow

answered Mar 9, 2010 at 20:13

notageek
39 ● 1 ● 1

---

2 I rather think that classifying an entire language as "kindergarten" is a little easy. Especially since there are actually a good many self-respecting software engineers

currently using PHP to develop web applications.If you ask me, any self-respecting web developer should point to C# as the way to go if you want a "good language", as opposed to PHP or Python, both of which are bad at OOP IMO. Python has its batteries-included philosophy going for it, but ASP.NET has a very decent and coherent framework underlying it. *(edit: I am a PHP web developer, myself, to add a bit of context.)* – toon81 Oct 22, 2011 at 11:18

4 @notageek "You can integrate your python libraries into the web app, and that is, imho, a huge advantage." Can you give me an example when one CANNOT integrate own libraries into a PHP app? – user151851 Dec 12, 2013 at 3:21

You can really go ham with php; I think you're underestimating it. However, python looks nicer and in many ways is actually much simpler and for many tasks much easier than php. – JVE999 Oct 12, 2014 at 1:14

2 This reply wasn't very nice. I know many "self-respecting" software engineers that use PHP. When performance is an issue, C++ and ASM is on the table, not Python. Use the tools to get the job done. – maiko Feb 3, 2016 at 14:39