

Is functional programming relevant to web development? [closed]

Asked 16 years, 1 month ago Modified 4 years, 1 month ago

Viewed 26k times



92



Closed. This question needs [details or clarity](#). It is not currently accepting answers.

💡 **Want to improve this question?** Add details and clarify the problem by [editing this post](#).

Closed 4 years ago.

The community reviewed whether to reopen this question 6 months ago and left it closed:

Opinion-based Update the question so it can be answered with facts and citations by [editing this post](#).

[Improve this question](#)

I've been seeing so much recently about functional programming and Clojure looks particularly interesting. While I 'understand' the basic description of what it is, I can't figure out how I would use it on a day to day basis as a web developer, if I can at all. A lot of what I have read focuses on the maths side of functional

programming rather than typical programming situations found in regular OO.

Have I got the wrong end of the stick? Is functional programming totally un-related to web development? If not, are there any examples of it being used 'for the web'?

closure

functional-programming

Share

Improve this question

Follow

edited Jul 22, 2016 at 22:03



Alex Miller

70.1k ● 25 ● 124 ● 168

asked Nov 15, 2008 at 2:27



Hates_

68.6k ● 6 ● 33 ● 37

19 Answers

Sorted by:

Highest score (default)



63

Functional programming matches web apps very well. The web app receives a HTTP request and produces a HTML result. This could be considered a function from requests to pages.



Compare with desktop apps, where we typically have a long running process, a stateful UI and dataflow in several directions. This is more suited to OO which is



concerned about objects with state and message passing.

Share Improve this answer

answered Nov 15, 2008 at 7:42

Follow



JacquesB

42.6k ● 13 ● 75 ● 88



32



A few examples off the top of my head:

- Yahoo! Store is powered by Lisp (originally named [Viaweb](#) prior to acquisition)
- [Reddit](#) was fully prototyped in Lisp, although they [switched to Python](#) in 2005
- [Hacker News](#) is written entirely in Arc (a Lisp dialect)

Share Improve this answer

answered Nov 15, 2008 at 2:41

Follow



Greg Case

47.4k ● 3 ● 29 ● 21

7 Yahoo! Store was rewritten in c++ a few years back. – [apg](#) Nov 17, 2008 at 3:01

6 language != programming paradigm. – [phkahler](#) Oct 24, 2010 at 22:58

59 Odd choice of examples... Yahoo! Store was written by Paul Graham, who is a big Lisp advocate. He also wrote Hacker News, which is, effectively, a Reddit clone. Presumably, HN was written somewhat in response to Reddit's switch from Lisp to Python while the Reddit guys were part of YCombinator: an accelerator program that is run by Paul Graham. I'd count this as ONE example, not three. – [Brandon Bloom](#) Oct 25, 2010 at 1:18

@ShermPendley server configuration, network setup, and database access are also major factors in performance. Script execution times of web apps are typically a tiny fraction of overall cost. – [AgmLauncher](#) Nov 29, 2015 at 16:05



21

I don't see why not - so long as you're delivering standards-compliant HTML to browsers, they don't care what you used to produce it, be that a functional language, an imperative language, or trained monkeys.



Share Improve this answer

answered Nov 15, 2008 at 3:00



Follow



[Sherm Pendley](#)

13.6k ● 3 ● 47 ● 57



20

Pure functional programming might not map very well into the web programming environment. But the main impediment is just the lack of infrastructure (frameworks and APIs). It will be a long time (probably never, honestly) before a functional language has as rich a web programming environment as Java, Python, or Ruby.



That said, there are some options.



- [HAppS](#) - a Haskell web application server
- [Haskell Server Pages](#)
- [Links](#) - a functional web programming language
- [mod_caml](#) - an OCaml CGI module for Apache

- [Ocsigen/Eliom](#) - an OCaml web server/programming framework
- [OPA](#) - a commercial web application framework
- [Yesod](#) - a web framework for Haskell

I don't have any experience with any of these. Maybe commenters can weigh in on what's worked well for them.

Share Improve this answer

Follow

edited Jun 7, 2011 at 9:07



[Brian McKenna](#)

46.2k ● 6 ● 63 ● 60

answered Nov 15, 2008 at 15:31



[Chris Conway](#)

56k ● 43 ● 131 ● 155



14

[Twitter rewrote their backend in Scala](#), a JVM language that supports both the Object Oriented and Functional paradigms.



Also, the [Lift](#) web framework is written in Scala.



Share Improve this answer

Follow

answered Aug 5, 2009 at 21:56



[James McMahon](#)

49.5k ● 69 ● 210 ● 288



4

For Clojure, there's an interesting startup (TheDeadline) that developed using Clojure and Google App Engine. They have a good ppt on Slideshare and an interview on InfoQ.



For a good discussion on deploying Clojure with GAE:
<http://news.ycombinator.com/item?id=1239788>



As far as I know, Clojure has a few web development libraries. Compojure Ring Conjure

Hope this answers some of your questions =) (I'm just starting out too..)

Best, Ryan

Share Improve this answer

answered Oct 25, 2010 at 3:43

Follow



Ryan Teo

59 ● 2



3



It is not totally unrelated to web development. The app sitting on the server can very well take advantage of functional features like closures, higher-order functions, immutability, referential transparency... for instance, you sure have collections that you need to transform or manipulate in whatever way. Functional programming helps here, and it is for a reason that its idioms are penetrating mainstream languages. Functional features help in conciseness, testability, parallelization, and they can also provide native solutions to problems you would otherwise solve with patterns.

Update: there are web frameworks for functional languages too. Weblocks for Common Lisp, Lift for Scala. These are the ones I've heard of, there might be more... however you don't necessarily have to be purely

functional -- for example Scala is not pure and should work with any Java framework, you'd still be able to use functional programming for the business layer, etc.

Share Improve this answer

edited Nov 15, 2008 at 2:53

Follow

answered Nov 15, 2008 at 2:47



Germán

4,565 ● 3 ● 34 ● 38



3

Check out [Ur/Web](#). It's very fast and its static type system knows about things like HTML and SQL so it can guarantee all sorts of nice things about security.



Share Improve this answer

answered Oct 25, 2010 at 2:32

Follow



Cyrus Omar

109 ● 1 ● 1



3

Erlang seems to be getting a lot of use in some of the infrastructure for scalable web apps. the CouchDB and Riak databases are written mostly in Erlang, as is the RabbitMQ message queueing server. One of the keys to its success is that it handles concurrency through message-passing, with no shared mutable state. It's this way of thinking about problems that is useful, more than any particular functional programming language.



Or, look at MapReduce. It's a very functional way of looking at computation, even if your map and reduce functions happen to be internally stateful -- and for that reason, it's a very nice fit to querying large datasets in a fault-tolerant, distributed way.

By all means, go with whatever seems most practical for you. But always keep functional programming in the back of your mind, because you never know when it might come in handy.

Share Improve this answer

answered Oct 25, 2010 at 2:47

Follow



Peter Scott

551 ● 3 ● 4



2

They are doing some [pretty cool things at Edinburgh University](#) with functional programming for the web.

Share Improve this answer

answered Nov 15, 2008 at 10:57

Follow



Ali Afshar

41.6k ● 12 ● 97 ● 110



2

Language and paradigm doesn't matter for web apps, they're all equally good and bad. If you are looking for a reason to learn a new paradigm, just dive in. My advice would be to analyze the problem you're trying to solve and select the appropriate toolset.





Share Improve this answer

answered Oct 24, 2010 at 22:25



Follow



[Chuck McKnight](#)

109 ● 7



2

Javascript (the language of the FE portion of the web and increasingly the BE as well) isn't functional per se but functions are first order functional



Share Improve this answer

answered Oct 25, 2010 at 3:42

Follow



[Ro Marcus Westin](#)

29.6k ● 4 ● 19 ● 11



2

We have just launched an online spreadsheet where the backend is written entirely in Erlang.

<http://hypernumbers.com>



By any standards this is one of the most complex web-apps you can build with a ginormous GUI with mental complexity.



Share Improve this answer

answered Oct 25, 2010 at 7:44

Follow



[Gordon Guthrie](#)

6,254 ● 2 ● 28 ● 52



2

Here is a web developer experience in building web apps using [Haskell](#). Although functional languages are very type safe and have good concurrency, they always lacked



best of the breed api's since it has been darling of the academia since long and yet to be strongly embraced in the real world. I hope it's not too far away. Erlang had already got into it.



Share Improve this answer

Follow

edited Oct 25, 2010 at 14:17



[nmichaels](#)

50.9k ● 12 ● 111 ● 137

answered Sep 24, 2010 at 23:29



[A_Var](#)

1,046 ● 1 ● 13 ● 23



1



The reason why functional programming is popular with web programming is that it explicit shared and changing state and allows the programmer to express the purely functional parts as pure functions. Pure functions have the advantage of being very simple to run in parallel - as they have no side-effects.



At least that's my reason.

Share Improve this answer

Follow

answered Oct 24, 2010 at 23:57



[rbanffy](#)

2,501 ● 1 ● 17 ● 28



1

Another short answer: <http://www.mlstate.com> -- a full web development platform, based on FP. The clean semantics of the language permit all sorts of automated safety and security analysis, optimizations, etc.



Caveat: I work there.

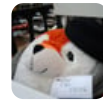


Share Improve this answer

answered Oct 25, 2010 at 6:35



Follow



Yoric

4,073 ● 3 ● 23 ● 28



1



Functional languages may not be directly useful for building great apps but we use functional programming paradigm heavily for building our Apps. Pure functional programming puts the constraint of "no side effects". This ensures that pure functional calls will yield same result in what ever order they are called. This is not ideal for web-development but if functional programming is combined with a state-changing system, a robust web app can be built. Have a look at my paper for more details: [FAST Server](#) Also these [slides](#).

Share Improve this answer

answered Jul 19, 2016 at 10:12

Follow



gopi

46 ● 3



0



Yes, as functional programming can be done in any language, you can use it as a web developer on a day to day basis.

Should you? it depends upon the problem you are solving. Functional programming is a programming paradigm and where you should use it depends upon the problem you are solving.

To make the decision simpler, think whether it is easier to solve some problem by using OOPs concepts, where encapsulation, polymorphism, inheritance like features can make your life easier?

If yes, do not go for functional programming there and simply use OOPs. If your application is going to perform complex computation / calculation / business logic and involving heavily concurrent processing, Functional programming can offer lots of tools and advantages in such cases.

These are just different styles of building the structure and elements of programs, so it's all about using right tool for the right job. rest anything can be done using anything.

Functional programming in web development:

JavaScript supports functional programming and it is very much supportive when we are in context of web development. React framework is heavily influenced by functional programming principles and used in many web applications out there.

Also, you can find many web applications built and running with frameworks developed on functional programming languages listed below:

- WebSharper (F#)
- Snap (Haskell)
- Lift (Scala)

- Ocsigen (OCaml)
- Chicago Boss, Zotonic (erLang)

Hope So my answer will help anyone.

Share Improve this answer

answered Apr 10, 2019 at 13:56

Follow



Amit Agrawal

21 ● 1



-3



You probably won't use it and shouldn't use it but when you say that someone will always find an exception to the rule (Viaweb, etc.). Basically there is no "super language" there are only working lines of code, usually in "Blub." Even Paul Graham says that the main (in fact only) benefit to a Lisp is one's ability to rapidly prototype.

Also "super languages" usually impinge rather than increase code readability, meaning that the one "genius" who wrote it has to maintain it forever since no one else can understand it, especially since he is likely to write it in his own modified dialect. This decreases the possible scope of any project, which means that even if new, innovative things can be done, they are not extensible and so remain at a relatively small scale (like Hacker News in Arc).

That's not to say that someone can't have a genius idea and implement it in a incomprehensible style that then can be re-written in Blub and extended so that lots of people can benefit from it. Actually, that's exactly what

has happened in all of the Lisp success stories, not to mention every famous philosopher that's ever lived. But of course, if you are a "genius" you might also be able to prototype your product some other way.

As for FP on the JVM, there are limited but cool things possible. Although I would personally use it for prototyping only, it is possible you might have a use case (usually something to do with multi-threading) where it provides some improvement.

Share Improve this answer

answered Oct 25, 2010 at 2:11

Follow



Joel

11 ● 1

-
- 1 Actually, Paul Graham says that Lisps big advantage is that it's "the most powerful" language available. He supports his argument by saying that something like a quarter of viaweb (by lines) was Lisp macros. Meta-programming in most languages (particularly those available in the mid-90s) is hard. It's not a prototype if it's pushed to the production server. – [nmichaels](#) Oct 25, 2010 at 14:23
-