

JavaScript Patterns



O'REILLY® | YAHOO! PRESS *Stoyan Stefanov*

JavaScript Patterns

Stoyan Stefanov

[Download now](#)

[Read Online →](#)

JavaScript Patterns

Stoyan Stefanov

JavaScript Patterns Stoyan Stefanov

What's the best approach for developing an application with JavaScript? This book helps you answer that question with numerous JavaScript coding patterns and best practices. If you're an experienced developer looking to solve problems related to objects, functions, inheritance, and other language-specific categories, the abstractions and code templates in this guide are ideal -- whether you're writing a client-side, server-side, or desktop application with JavaScript.

Written by JavaScript expert Stoyan Stefanov -- Senior Yahoo! Technical and architect of YSlow 2.0, the web page performance optimization tool -- *JavaScript Patterns* includes practical advice for implementing each pattern discussed, along with several hands-on examples. You'll also learn about anti-patterns: common programming approaches that cause more problems than they solve.

Explore useful habits for writing high-quality JavaScript code, such as avoiding globals, using single var declarations, and more

Learn why literal notation patterns are simpler alternatives to constructor functions

Discover different ways to define a function in JavaScript

Create objects that go beyond the basic patterns of using object literals and constructor functions

Learn the options available for code reuse and inheritance in JavaScript

Study sample JavaScript approaches to common design patterns such as Singleton, Factory, Decorator, and more

Examine patterns that apply specifically to the client-side browser environment

JavaScript Patterns Details

Date : Published September 28th 2010 by O'Reilly Media (first published September 1st 2010)

ISBN : 9780596806750

Author : Stoyan Stefanov

Format : Paperback 216 pages

Genre : Computer Science, Programming, Science, Technology, Technical, Software, Nonfiction

 [Download JavaScript Patterns ...pdf](#)

 [Read Online JavaScript Patterns ...pdf](#)

Download and Read Free Online JavaScript Patterns Stoyan Stefanov

From Reader Review JavaScript Patterns for online ebook

David Zukowski says

Goodreads fix your broken UX please.

Adrian Li says

This is a good book and most people who work with Javascript should spend at least a couple hours going through it. But there is a caveat: it is old.

This book was published in 2010, and there are many things that are talked about, which we really don't have to worry about anymore. In this book, there are many IIFE examples that are no longer needed in the world of ES2015 and beyond (i.e. ESNEXT).

In any case, you need to read through this book in a smart manner. Don't force yourself to go through absolutely everything, and definitely skip some sections if it's quite obvious that you won't really need to go through them in detail since a lot of today's best practices avoid some of the mentioned patterns completely.

Nevertheless, this is an important book to read. And if you have a nuanced reading strategy (like I mentioned above), you can probably get through this book within 3 hours or so and learn a decent amount while doing so.

Dave Lorentz says

Do not re invent the wheel! Patterns are a cornerstone of systems design and this book shows how they can be applied. I would like to see data access patterns added, especially now that HTML5 is so wide spread. But that does not take away from the good set of patterns described here.

Andrew says

Very insightful!

Daniel R. says

A grab bag of practices and techniques for JavaScript development for the core language and some browser specific concerns. The Essentials, Literals and Constructors, Functions, Object Creation Patterns, and Code Reuse Patterns chapters mostly rehash material from "JavaScript: The Good Parts". The author provides additional exposition on the concepts but doesn't introduce much new material. The Design Patterns chapter includes some good examples of JavaScript implementations but accurately points out many already exist in

third party libraries. The DOM and Browser Patterns chapter provides a brief introduction to browser specific concepts and concerns but you'll want to consult the referenced books for more detailed discussions. This book is a great primer for JavaScript development due to the breadth of topics it covers but is worth a pass if you've already read other JavaScript books covering these topics.

Rod Hilton says

JavaScript Patterns was the first book that made me feel like I really *GOT* JavaScript. I've come away with a much greater appreciation for the language than I used to have, and I feel much more comfortable with the idiomatic usage of the language.

The book, written by a Senior Yahoo! programmer, is more than just a "here is how to do the XYZ Design Pattern from the GOF book in JS" (though it has quite a bit of that at the end). It's really a book about patterns that have evolved out of JavaScript's usage - these are *NEW* Patterns, that are specific to JavaScript, which the author has observed and annotated. As the GOF book was to Object Oriented languages, JavaScript Patterns is to JavaScript.

The book covers a lot of ground, helping the reader understand not only the what, but the why of various JavaScript conventions. It is very well written and always interesting - if I didn't have to occasionally take breaks to avoid melting my brain, I'd have been virtually unable to put it down.

I'd say that if you're a web programmer and you want to take your JavaScript up to the next level, you should read The Good Parts, High Performance JavaScript, and this book. It was an eye-opener for me, and I highly recommend it.

Johnny Graber says

Only a small part of JavaScript Patterns is about design patterns. Much more pages are spent on best practices and the deployment story. While these parts are very useful, they may not be what you are looking for.

The Singleton pattern is one example I like to pick to illustrate some of the problems with the book. In books about design patterns for other languages like Ruby that doesn't have the concept of a Singleton, the explanation of the pattern is simply: Not supported here. In his book Stoyan Stefanov keeps going and going and uses 5 pages with workarounds. This is great if you must implement a Singleton, but most likely you are better off if you just would stop forcing it into JavaScript.

As long as you read this book as a way to write JavaScript and not as the definitive guide on JavaScript Patterns you can learn a lot.

David says

This book covers a lot of different patterns, and is a useful tour. Even experienced programmers will likely learn patterns they haven't seen before. The author's commentary is thoughtful and helps the reader evaluate pros and cons, though I think the author is not sufficiently critical of weaker techniques. Many of them are

fine for small programs but don't scale well for larger ones (i.e., are not very maintainable or extensible or automatically checkable).

The bigger problem is that, like JavaScript itself, there's no remotely complete, coherent set of patterns. Many of the techniques presented don't work well together, and many problems have three or four half-solutions, each with important drawbacks. So while the book covers a wide breadth with useful analysis, the result is very unsatisfying.

Harvey says

Some of the author's suggestions about how to handle loading of JS are getting quite dated, but the discussion of design patterns is still worth buying this book.

Rob says

If you're a JavaScript developer, you would be wise to have this on your bookshelf--nestled nicely between *JavaScript: The Good Parts* and *High Performance JavaScript*. The three make a nice little troika.

And read them in that order: *The Good Parts*, *Patterns*, and then *High Performance*.

Here's why:

What Stefanov gives us with this book is effectively an overview [†] of best practices for JavaScript development, going over the benefits and gotchas of certain important language features, and translating those into design and implementation patterns. Many of these patterns are language-agnostic--and you're likely to recognize them from "The Gang of Four"--but Stefanov puts them in their JavaScript party dresses and takes them out to the ball. Wisely, Stefanov also presents these patterns in an environment/host-independent fashion, so the lessons you learn about encapsulation or inheritance or performance should be equally valid regardless of whether you're coding for the browser [‡] or NodeJS or some image exporting automation for Adobe Illustrator. Stefanov is also a lucid and concise author, clearly illustrating his points about these design patterns; the text is accessible--easy to follow and digest--and he is careful to clearly define words and terms that might be ambiguous or commonly misunderstood (e.g., "function expression" vs. "function declaration" vs. "function literal").

JavaScript patterns makes a great transition guide for intermediate developers--the men and women who have stopped confusing jQuery-the-library with JavaScript-the-language--the folks who are ready to re-evaluate their approach software development with JavaScript. This is for the folks that made it through Crockford's *The Good Parts* knowing that they learned something but also feeling less-than-certain about how to apply that something. This is the follow-on; *JavaScript Patterns* is the application of those lessons. And then after you've written your clean, maintainable, scalable applications--then you make the jump to Zakas' *High Performance JavaScript* to tune things just a little bit further.

So you're probably wondering then: if you recommend it so highly, why only four stars?

The four stars comes mostly from two niggling points:

(1) Relative to *The Good Parts* and *High Performance, JavaScript Patterns* was not published in the order that I recommend reading them. As a consequence, since I'd read the others (and quite a few others above and beyond those two), there is quite a bit of information in there that I'd seen before. This is not a Bad Thing; sometimes it pays to see information presented again--to help it sink in or else to gain another perspective on it. And in some cases Stefanov offers an as-good-or-better explanation on certain topics/techniques as others writing in the field (e.g., his examples for memoization and currying rival Crockford's, and his explanation of the pub/sub pattern (and custom event design) is more concise than the one Zakas presents in *Professional JavaScript for Web Developers*). Sometimes (and I've written this before) you were just hoping for... just a little bit more.

(2) And this is super nit-picky but... The book could have taken another quick editorial pass for spelling and grammar. The one that stuck out at me was right in the intro to Chapter 6: "But it's important to keep the end goal in mind--we want to reuse cod;." Indeed.

† : An in-depth overview, but an overview nonetheless.

‡ : Stefanov is careful to "keep the browser out of it" and dedicates only one chapter (Chapter 8: DOM and Browser Patterns) to the subject; though everyone's favorite host environment does creep in a couple of times, in a couple of examples.

Diego Higuchi says

Este livro me proporcionou maiores conhecimentos de padrões de projeto (Singleton, Factory, Iterator, Decorator, Strategy, Façade, Proxy, Mediator, Observer), performance javascript entre outros, como pude aprender também o que são anti-padrões. O livro cita exemplos primeiramente como "não fazer" e como "fazer" (anti-padrão e padrão), bastante objetivo. Recomendo à todos desenvolvedores.

Greenicicle says

So far I'm a little disappointed that practices like "write sensible comments" and "indent properly" are really listed as patterns, especially when the book claims to be targeted to professional devs. But I've still got a bunch of pages to go through.

UPDATE: Finished now. The title does not lie: the book is full of patterns. There is a focus on organizational patterns (object creation, inheritance). Of these, there is a large variety of variants; mostly presented in an evolutionary chain that leads to increasingly better solutions. This is interesting for an understanding why a pattern is recommended, it's not the most compact way of representation on the other hand.

There is a lack of macroscopic, task-oriented patterns. This is not a book learning about common patterns in general, applied to JavaScript.

I found reading JavaScript patterns beneficial; coming from C# as my primary language it filled some of the gaps on how objects and functions can be utilized to create a richer programming platform. For that purpose

however, I found "The Good Parts" more compact.

Paul Pacurar says

basics or not so basics, but anyone should know these patterns

Maciej Cieslar says

Great book!

Marcus says

Let me start by presenting my 'credentials' in the field - I have two decades of programming experience in different object-oriented languages, mainly in .NET platform and SQL. Most of the time I deal with client-server applications in Windows environment. Over last couple of years I started increasingly dabbling with HTML5 and JavaScript, a challenge which I admit freely is not very enjoyable to me mostly due to JavaScript's chaotic nature (in my opinion at least).

When I reached for this particular volume, I hoped it would expand my basic understanding of JavaScript and make my transition from object-oriented world into function-based programming a bit easier. This initial objective wasn't reached, for a very simple reason - 'JavaScript Patterns' is a book that tells you 'what and how', but not 'why and when'. Also, it assumes a deeper understanding of JavaScript as a language that I possess after two years of its use. I don't think it's fair to blame the book for not being what I hoped for it to be. But at the same time I must say that the presentations of different patterns in this book are really very brief and shallow. The coverage of modules in particular was in my opinion deeply disappointing - a couple of generic code snippets and references to patterns previously discussed was deeply unsatisfactory coverage of this fundamental pattern to me.

Don't get me wrong, I understand perfectly the reason this book is presented in this way - the reader is supposed to take what's shown in the book, apply it to the real life program and build on his own from there. Fine, I get it. But the brevity of the 'demonstrations' in this particular book means that the reader will have to do a lot of guesswork and probably find additional sources of information BEFORE understanding a particular technique and isn't providing this understanding the actual goal of this book? Would it be so hard for the author to expand his examples, perhaps apply the techniques he discusses to a couple of commonly occurring real life problems?

It may be the brevity of content of this book, it may be my lacking knowledge of more complex aspects of function-based programming in general and JavaScript in particular. Whatever the reason may be, I didn't find this book very informative.
