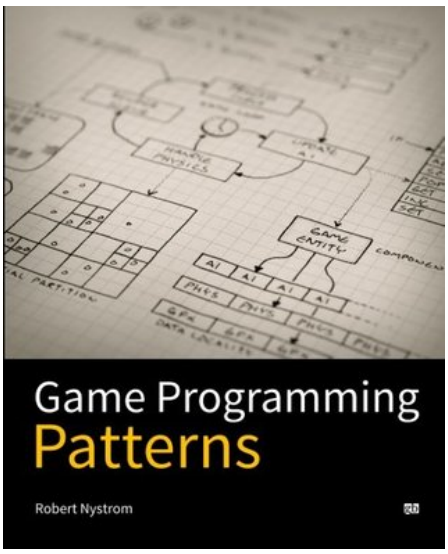

Game Programming Patterns

Robert Nystrom

Game Programming Patterns

Robert Nystrom

Game Programming Patterns Robert Nystrom
The biggest challenge facing many game programmers is completing their game. Most game projects fizzle out, overwhelmed by the complexity of their own code. Game Programming Patterns tackles that exact problem. Based on years of experience in shipped AAA titles, this book collects proven patterns to untangle and optimize your game, organized as independent recipes so you can pick just the patterns you need.

You will learn how to write a robust game loop, how to organize your entities using components, and take advantage of the CPUs cache to improve your performance. You'll dive deep into how scripting engines encode behavior, how quadtrees and other spatial partitions optimize your engine, and how other classic design patterns can be used in games.

Game Programming Patterns Details

Date	:	Published November 2nd 2014 by Genever Benning (first published January 1st 2011)
ISBN	:	9780990582908
Author	:	Robert Nystrom
Format	:	Paperback 354 pages
Genre	:	Computer Science, Programming, Science, Games, Game Design, Sports and Games, Nonfiction, Technical

From Reader Review Game Programming Patterns for online ebook

Neeraj Adhikari says

If you have already been doing game programming for some time, chances are you already know the patterns described in this book. But if you are just starting out, this book is a must-read. Robert Nystrom has done a great job with analogies, humor and good explanation. Due to that, this book is more fun to read than the original Gang of Four book.

Diogo Muller says

Very good book on Design Patterns for Games. If you are a game developer, you probably know and use a few, if not most of those patterns already. However, the author explains quite a few use cases, optimizations, and reasons why it's better to use one pattern instead of another, depending on what tradeoffs you can accept. Not only that, but the author favors showing things by example, step by step, with a lot of humor in-between, which works better in this case than just dumping a lot of theory at the reader and showing one example or two. Oh, while the focus is on game development, most of the designs in this book are also useful for other types of applications.

Even if you know a bit about Game Programming, I feel this is a good read - it's not a long book, and you may learn a few things. Some of the most useful lessons on this book are not when to use a pattern or when to do something, but when NOT to use a certain pattern or when it's not worth doing some optimization, heavy change or something like that. I feel most programmers could benefit from learning a bit about that too.

Gaber says

This book is well-written and will give you a lot of 'recipes', applicable not just to game development.
As a web developer I find quite some chapters(State pattern, Component pattern) very beneficial.
The author adds a few humor bits(sorry for the pun) from time to time, so don't expect it to be very dry(again sorry).

Ben Rand says

Author has a good writing style. I liked that he pointed out drawbacks to each of the patterns he described, something that's often lacking in discussions about design patterns.

Jamie Jansen says

Fantastic read!

Concise, engaging, and applicable to real-world situations. I'd consider this as the first stop for any game programmers looking to optimize their software architecture and efficiency.

Elias Daler says

This book made me a better programmer and saved me from writing a messy engine. It's very fun to read, contains amazing examples and each chapter will likely cause you to think about improving one or another aspect of your game engine as you read this book.

This is one of my favorite programming books in general as well.

Mikko McMenamin says

Deep dive into the most powerful game programming patterns. Important and useful information for game programmers and other developers who want to deepen their understanding of software architecture.

Katie Cunningham says

Just finished this and I can't recommend it enough. Even if you know programming patterns, I feel like this is a good resource to dive into. It goes into when some patterns are bad (a topic that I've found most books skimp on), and how certain patterns apply specifically to game development.

Its most stunning accomplishment was being the first programming patterns book that made me really want to code rather than question my life choices.

Uroš says

Good read for every developer or engineer. No necessary to be used for games only, I can see many parallels in embedded also. But I sure would like to find some time to test all this in some obscure open source indie game.

Cen Rao says

Very informative and well written

This is a very good book for developer, not only for game developer but also general software developers. I will try to read it as many times as I can.

Nathan Glenn says

I'm not a game programmer, but I still enjoyed reading through all of the patterns, which are useful in many contexts. His writing style is extremely clear, and he presents each pattern with an example use-case, example code, and my favorite, discussions of the various design trade-offs. I understood for the first time what a union (C/C++) is good for!

Matt Hertel says

Game Programming Patterns was a well-written, comprehensive book that was easy to understand, but still expressed complex concepts that are useful to a broad range of programming skill levels. I will continue to use the book as a reference, as the concepts and examples are easy to apply to any situation where a program needs order or optimization. I would recommend this book to anyone who is interested in learning the basic programming patterns that operate the games we play, and applying them to their own projects.

Codepoetz says

Robert examines several classic Gang-of-Four design patterns and describes different ways that each pattern can be used to solve specific problems that you often encounter during game programming. Personally, I was already familiar with most of the content in this book but still enjoyed it as a nice summary. Note: the author provides this book for free online if you don't want to purchase the paper version.

Gwern says

(I read the online version.)

This book follows the standard pattern for design pattern books: short chapters on each particular style, with definition, pros & cons, simple pedagogic example, comparisons with other design patterns, and possibly some discussion of real-world implementations & game engines. I haven't heard of any other place where one could find this sort of game-oriented programming design advice, and in that respect, this book is unrivaled.

While far from encyclopedic, the chosen design patterns all seem like reasonable choices for video game programmers: there are some architectural ones like Flyweight & Singleton which everyone needs to know, and then a good helping of high-performance or game-specific patterns (eg. Data Locality/Object Pool/Dirty Bit & Double Buffer/Game Loop/Spatial Partition, respectively), as well as a few fairly exotic patterns which often show up in games but not that many other areas (for example, Byte Code shows up in a lot of games to support modding, but you'll otherwise spot such things only in a few extremely-extensible applications like text editors or programming languages).

The actual chapters are surprisingly brief (from my non-game-programming perspective, anyway), and come

across more as unusually long blog posts than meaty book chapters. Nystrom mentions several times he was trying for brevity, but I would say he's taken simplicity to simplemindedness. I would have liked to see more of everything: more examples, more real-world applications, more benchmarks, more comparisons to other languages (*especially* to higher-level languages like Python/JavaScript/Haskell, which as hardware resources increase, game programmers will increasingly use; us Haskellers say that design patterns are simply flaws in the language, and I'm curious how much of Nystrom's writing is simply due to working around C and Java issues).

The web presentation is in standard 2.0-style: big font, lots of whitespace. Not too bad for reading, although the sidebar notes were annoying since a lot of them would have been better off incorporated into the text or simply axed, and their relation to the text could be very confusing if there were 2 or 3 on the same page. The web presentation also omits a major advantage of being online: *comments*! I started reading the site solely because I saw one of the chapters submitted to Hacker News & found it interesting. By a quick count, there's 18 relevant discussions on Hacker News and another 37 on Reddit, yet someone reading it has no idea. There were many interesting comments and suggestions in those submissions, why not excerpt the best? Or at least link them at the end of each chapter as a "further reading" or something?

Maksim Ryzhikov says

I think it's an awesome book. I'm not a game programmer but:

First: Patterns from this book could be applied to any other domains
Second: It's very interesting and fun to know about game programming
Third: I like author's writing style. It's clear, lightweight and fun

---