

[Read free ebook] 3D Game Engine Architecture: Engineering Real-Time Applications with Wild Magic (The Morgan Kaufmann Series in Interactive 3d Technology)

# 3D Game Engine Architecture: Engineering Real-Time Applications with Wild Magic (The Morgan Kaufmann Series in Interactive 3d Technology)

David H. Eberly

ebooks | Download PDF | \*ePub | DOC | audiobook



[Download](#)

[Read Online](#)

#1213479 in Books 2004-12-17 Original language: Spanish PDF # 1 9.50 x 7.75 x 1.50l, 3.24 #File Name: 012229064X756 pages | File size: 30.Mb

David H. Eberly : 3D Game Engine Architecture: Engineering Real-Time Applications with Wild Magic (The Morgan Kaufmann Series in Interactive 3d Technology) before purchasing it in order to gage whether or not it would be worth my time, and all praised 3D Game Engine Architecture: Engineering Real-Time Applications with Wild Magic (The Morgan Kaufmann Series in Interactive 3d Technology):

1 of 1 people found the following review helpful. Great Book For Engine Developers. Didn't Like the Author's Coding Style. By A. M. Hernandez In preparation for my 3D game engine, I have been trying to read all I can on 3D engine design and architecture. Although there are some good books out there, it's very difficult to find a text that will walk

you through everything you need to know. That said, 3D Game Engine Architecture: Engineering Real-Time Applications with Wild Magic by David H. Eberly gives it a good attempt. The book covers the author's Wild Magic engine, and discusses certain choices he made when developing the engine. It briefly touches on OpenGL, discusses abstracting away platform-specific details, 3D mathematics (and there is a lot of math in this book), an object system, scene-graphs, level of detail, render states, sorting, terrain, animation, collision detection, physics, and more. A lot of ground is covered in less than 800 pages. However, I found much of the book difficult to follow and still feel like I could have a better picture of the "architecture" of an engine. When I think of "architecture" I think about a broad 500 foot view of a project. I think of flow charts or UML. I expect discussion on how all these disparate elements come together to form a whole. Sadly, that is mostly missing from this book. What the author provides is a good insight into his particular engine, and certain specific aspects of that engine. While this is still a great example to look at, I feel the text could have been more robust in terms of painting the big picture. Some of the things that I found missing were an event system, which seems crucial to an object-oriented engine, or a component architecture, really any type of structure that allows communication between classes. Additionally, I found myself getting lost multiple times while reading the book. The author would frequently put in dense mathematical equations and proofs, sometimes spanning multiple pages, and by the end you would be left to wonder what the purpose of the equation even was. I feel like having proofs of equations was not really relevant to the architecture, and surely there are many books on straight math if the reader needed that. Some math is necessary, of course, for a 3D engine but the space could have been used for more important topics. Not really a jab at the book so much as it is the author's coding conventions, I really did not like his style. I realize this is somewhat of a holy-war with programmers, but I guess we all have a style that is comfortable for us. Personally I found the author's style to be really obtuse, and made reading the code snippets more difficult. For example, for a camera's forward vector, he would use something like: `m_pkFVecWhere` "m\_" was a member variable, "p" is a pointer, "k" is of a class type, and "FVec" for forward vector. Personally I would use simply: `forwardVector` just glancing through the code, which one is more apparent to what it is? This really bothered me to no end, but I guess you can chock it up to personal taste. All-in-all it may sound like I am putting down on this book, but I actually did find it useful in a lot of ways. Certainly if you are aiming to create a 3D game engine from scratch, you will need any and all the help you can get. So yes, still read this book. However, I had much higher expectations and I feel it was a missed opportunity for the author. While it is still a decent resource, this should not be your first stop in engine development.

3 of 4 people found the following review helpful. Good software engineering book  
By Rodrigo Damazio Well, needless to say, Eberly rocks. This is a great book on game engine architecture - even though it's less mathematically intensive than Eberly's other books (which, for many, is a good thing), it has a very good coverage of software engineering aspects of game engines, as it explains in detail how (and why) the Wild Magic engine is organized. Do keep in mind, though, that the book's purpose is not teaching how to develop games or algorithms for that - it's a software engineering book, and a good one at that.

29 of 30 people found the following review helpful. Finally!  
By Dave Amy Astle One of the criticisms I had of Dave Eberly's previous 3D Game Engine Design is that it didn't really say much about how to design an engine. Rather, it focused mostly on the programming/implementation details, which was disappointing to some. This new book is what many people expected from the older book. In this book, the author walks through the design and architecture of a 3D game engine, using his Wild Magic engine as an example, but also drawing on his experience developing NDL's NetImmerse. Throughout, he describes why each design decision was made, and in many cases alternative solutions are discussed as well. This isn't just a high level discussion, however, as ample source code, figures, equations, and sample applications are included to get you started with implementation. The topics covered include the core engine systems, scene graphs, renderers, cameras, LOD, animation, terrain, special effects, physics and collision detection. Numerous sample applications and tools are also included. Dave's writing style is clear and minimally conversational, and he's kept the math to a minimum, making this a remarkably easy read. This isn't a complete treatment of a game engine, since some important topics (e.g. scripting, audio) aren't included, but the material it does cover is worth it. Whether you're currently working on a game engine, planning to start one, or just want to have a better understanding of how they work, you'll be happy with this book.

Dave Eberly's 3D Game Engine Design was the first professional guide to the essential concepts and algorithms of real-time 3D engines and quickly became a classic of game development. Dave's new book 3D Game Engine Architecture continues the tradition with a comprehensive look at the software engineering and programming of 3D engines. This book is a complete guide to the engineering process, starting with a walk-through of the graphics pipeline showing how to construct the core elements of 3D systems, including data structures, the math system, and the object system. Dave explains how to manage data with scene graphs, how to build rendering and camera systems, and how to handle level of detail, terrain, and animation. Advanced rendering effects such as vertex and pixel shaders are also covered as well as collision detection and physics systems. The book concludes with a discussion of application design, development tools, and coding standards for the source code of the new version of the Wild Magic engine included on the CD-ROM. Wild Magic is a commercial-quality game engine used by many companies and is a

unique resource for the game development community.

"Readers of Eberly's previous books, 3D Game Engine Design and Geometric Tools for Computer Graphics, asked for a volume with more code samples and fewer equations. This book will please and aid them greatly.. That key points are sometimes underscored with a dry wit is characteristic of how a great teacher makes studying a difficult topic personally rewarding for the student." -Joseph Goldstone, Lilliputian Pictures

"Credo Interactive has been using the WildMagic API for the past 3 years in various commercial projects. 3D Game Engine Architecture provides an excellent source of theoretical background and practical usage information for the API. Together the textbook and the WildMagic sourcecode form a comprehensive and well designed foundation for any 3D application." -Lars Wilke, Director of Development, Credo Interactive Inc.

"For those that have searched for a commercial quality rendering library available at virtually no cost and with extensive clear documentation. Let them look no further. From low level structures, to high level application design, Eberly has laid out everything necessary for commercial quality game development. May the projects inspired by his writings be plentiful and prosperous." -Timothy Prepscius, DimensionDoor, Inc.

From the Back Cover

Dave Eberly's 3D Game Engine Design was the first professional guide to the essential concepts and algorithms of real-time 3D engines and quickly became a classic of game development. Dave's new book 3D Game Engine Architecture continues the tradition with a comprehensive look at the software engineering and programming of 3D engines. This book is a complete guide to the engineering process, starting with a walk-through of the graphics pipeline showing how to construct the core elements of 3D systems, including data structures, the math system, and the object system. Dave explains how to manage data with scene graphs, how to build rendering and camera systems, and how to handle level of detail, terrain, and animation. Advanced rendering effects such as vertex and pixel shaders are also covered as well as collision detection and physics systems. The book concludes with a discussion of application design, development tools, and coding standards for the source code of the new version of the Wild Magic engine included on the CD-ROM. Wild Magic is a commercial-quality game engine used by many companies and is a unique resource for the game development community.

Features

- \*CD-ROM with the complete C++ source code for Wild Magic version 3, a commercial-quality game engine for Windows, Linux, and OS X.
- \*A comprehensive, practical guide to all the steps necessary to build professional-quality real-time simulations with just minimal mathematics required.
- \*Emphasizes the application of software engineering principles and describes the architecture of large libraries.

About the Author

Dave Eberly is the president of Geometric Tools, Inc. ([www.geometrictools.com](http://www.geometrictools.com)), a company that specializes in software development for computer graphics, image analysis, and numerical methods. Previously, he was the director of engineering at Numerical Design Ltd. (NDL), the company responsible for the real-time 3D game engine, NetImmerse. He also worked for NDL on Gamebryo, which was the next-generation engine after NetImmerse. His background includes a BA degree in mathematics from Bloomsburg University, MS and PhD degrees in mathematics from the University of Colorado at Boulder, and MS and PhD degrees in computer science from the University of North Carolina at Chapel Hill. He is the author of 3D Game Engine Design, 2nd Edition (2006), 3D Game Engine Architecture (2005), Game Physics (2004), and coauthor with Philip Schneider of Geometric Tools for Computer Graphics (2003), all published by Morgan Kaufmann. As a mathematician, Dave did research in the mathematics of combustion, signal and image processing, and length-biased distributions in statistics. He was an associate professor at the University of Texas at San Antonio with an adjunct appointment in radiology at the U.T. Health Science Center at San Antonio. In 1991, he gave up his tenured position to re-train in computer science at the University of North Carolina. After graduating in 1994, he remained for one year as a research associate professor in computer science with a joint appointment in the Department of Neurosurgery, working in medical image analysis. His next stop was the SAS Institute, working for a year on SAS/Insight, a statistical graphics package. Finally, deciding that computer graphics and geometry were his real calling, Dave went to work for NDL (which is now Emergent Game Technologies), then to Magic Software, Inc., which later became Geometric Tools, Inc. Dave's participation in the newsgroup [comp.graphics.algorit](http://comp.graphics.algorit)