Fusebox: Developing ColdFusion Applications
Synopsis

Developers face a constant struggle to launch projects on time and under budget, especially without pulling all-nighters. Fusebox helps ensure successful projects by providing a framework that serves as a base for applications. It’s a standard process that makes projects more manageable and simplifies maintenance and requests for change during development. With this book, you'll learn to make ColdFusion applications reliably successful by following a standardized system. In addition, relying on the Fusebox framework to help plan and organize your ColdFusion code will allow you to write increasingly complex and specialized applications. Jeff Peters and Nat Papovich, both members of the Fusebox Council, share their extensive experience in this book. They'll teach you to use Fusebox with your ColdFusion applications and develop a set of best practices for managing web projects. Read this book if you want to eliminate frustrations and roadblocks in your projects, such as unmanageable complexity, wasteful redundancy of effort, time-consuming code maintenance, and slow development speed.

Book Information

Paperback: 456 pages
Publisher: New Riders; 1 edition (June 28, 2002)
Language: English
ISBN-10: 0735712697
Product Dimensions: 6.9 x 1.2 x 8.9 inches
Shipping Weight: 1.6 pounds (View shipping rates and policies)
Average Customer Review: 4.0 out of 5 stars  See all reviews (22 customer reviews)
Best Sellers Rank: #3,664,270 in Books (See Top 100 in Books)  #41 in Books > Computers & Technology > Programming > Web Programming > Cold Fusion  #2686 in Books > Textbooks > Computer Science > Database Storage & Design  #9143 in Books > Textbooks > Computer Science > Programming Languages

Customer Reviews

This book goes from theory to nuts-and-bolts and back again on how to design, implement, use, manipulate, and manage code written in the Fusebox method. While I particularly like Wireframes and FLiP, my favorite Fusebox piece by far are FuseDocs. The book spends a fair amount of time explaining each piece of the Fusebox methodology and gives plenty of examples. There is a companion website, which is more than handy. The book is broken into two sections: coding and
the life cycle of the code. Each section seems to have a bit different focus, and I imagine that the
Life Cycle Process section is more immediately attractive to managers and independent developers.
Yet I am glad to find both pieces here, juxtaposed. It keeps me from losing the forest for the trees in
the coding section, and from getting too far removed from the actual work at hand in the FLiP
section.According to the authors, Fusebox was developed for use by small teams with a theoretical
manager somewhere. I can see how this methodology would bring focus to and demystify any
application development. But I have to write that while teams might have been the target audience,
contractors will come to love this. Fusebox and especially its FuseDoc element empower the novice
contractor to tackle the Big Project that has been landed at long last and produce results without
becoming overwhelmed. Far more importantly, though, it allows a developer familiar with FuseBox
to speak a fluid language about time, task, and integration. As someone who finds herself after
lunch looking over code written before breakfast and wondering what the heck I was trying to do,
FuseDocs are a godsend.Fusebox methodology, while written for ColdFusion, isn't limited to the
ColdFusion world.

Whether you want to learn about the intricacies of the Fusebox core files; or you want to understand
how Fusebox can help you become a better ColdFusion® developer, the book Fusebox:
Developing ColdFusion® Applications by Jeff Peters and Nat Papovich is well worth the
read. This book covers the basics of the Fusebox framework and FLiP (Fusebox Lifecycle Process)
methodology to the more advanced features of Fusebox such as nested circuits and
layouts. Beginners will appreciate the authors' abilities to convey complex ideas through examples
that relate to real-world development experiences. Through their examples, it is easy to see how
using FLiP and Fusebox can ease and even eliminate the common pitfalls that developers go
through when coding their applications. The book starts out by explaining why a framework and
methodology are beneficial to creating successful applications, and will give the reader some insight
into the relationship between the developer and the client. This is a recommended read for all
managers. The expertise of the authors really shows in chapters 3 and 4 as they delve into the
complexities of the Fusebox core files. This section may be overly complex for the Fusebox
beginner, but is an insightful read for those who already have experience creating Fusebox
applications. If you were ever curious about the functionality of the code that makes up the Fusebox
core files, this section will give you greater understanding of how Fusebox works. Chapters 5-10
walk the reader through all of the parts that a developer is responsible for when creating a Fusebox
application. The basics such as the fuses (ColdFusion templates) and XFA’s (eXit FuseActions’s or
Triggers) are covered in chapters 5 and 6.

Jeff Peters and Nat Papovich have both been major forces in the development of Fusebox -- a methodology for developing web based applications in an efficient and organized fashion. While build and primarily used with ColdFusion, Fusebox can also be used with PHP or JSP code. [There is nothing available for ASP - arguably the most popular web scripting environment - due to limitations in the ASP language. ASP is unable to dynamically include files.] This book provides a good introduction to Fusebox 3, as well as the FLiP methodology. They spend the first 100 pages or so walking through every line of code that is executed to initiate a fuse request (the 300+ line core file, the support files, etc.), then get into details of how to write your own fuse code, and how it interacts with the FB3 files. They then spend a decent amount of time with FLiP -- the Fusebox Lifecycle Process. This I found less compelling, because much of FLiP is really oriented towards contract web site development. While some is applicable to the corporate environment in which I work, other parts really don’t fit all that well. Nonetheless, I feel that Peters & Papovich do a good job explaining the concepts behind FLiP, as well as the various software tools which are available now to support it. They provide quick demonstrations of wireframing & prototyping tools, a visual mind mapping tool, and a tool to build fuse-level unit testing harnesses. I found I really liked the approach of this book -- diving right into the deep end of the pool and working through the core files line by line. I’d initially gotten into Fusebox about a year ago, learning it on my own from what was available online. A few months later, I was fortunate enough to have Hal Helms teach a class in my company.

Download to continue reading...
