Practical DWR 2 Projects

Frank W. Zammetti

Apress®

Practical DWR 2 Projects

Copyright © 2008 by Frank W. Zammetti

All rights reserved. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without the prior written permission of the copyright owner and the publisher.

ISBN-13 (pbk): 978-1-59059-941-9

ISBN-10 (pbk): 1-59059-941-1

ISBN-13 (electronic): 978-1-4302-0556-2

ISBN-10 (electronic): 1-4302-0556-3

Printed and bound in the United States of America 9 8 7 6 5 4 3 2 1

Trademarked names may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, we use the names only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

Java™ and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc., in the US and other countries. Apress, Inc., is not affiliated with Sun Microsystems, Inc., and this book was written without endorsement from Sun Microsystems, Inc.

Lead Editor: Steve Anglin
Technical Reviewer: Herman van Rosmalen
Editorial Board: Steve Anglin, Ewan Buckingham, Tony Campbell, Gary Cornell, Jonathan Gennick, Kevin Goff, Jonathan Hassell, Matthew Moodie, Joseph Ottinger, Jeffrey Pepper, Ben Renow-Clarke, Dominic Shakeshaft, Matt Wade, Tom Welsh
Project Manager: Beth Christmas
Copy Editor: Ami Knox
Associate Production Director: Kari Brooks-Copony
Production Editor: Kelly Winquist
Compositor: Dina Quan
Proofreader: April Eddy
Indexer: Julie Grady
Artists: Anthony Volpe, April Milne
Cover Designer: Kurt Krames
Manufacturing Director: Tom Debolski

Distributed to the book trade worldwide by Springer-Verlag New York, Inc., 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax 201-348-4505, e-mail `orders-ny@springer-sbm.com`, or visit `http://www.springeronline.com`.

For information on translations, please contact Apress directly at 2855 Telegraph Avenue, Suite 600, Berkeley, CA 94705. Phone 510-549-5930, fax 510-549-5939, e-mail `info@apress.com`, or visit `http://www.apress.com`.

The information in this book is distributed on an "as is" basis, without warranty. Although every precaution has been taken in the preparation of this work, neither the author(s) nor Apress shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in this work.

The source code for this book is available to readers at `http://www.apress.com`.

Let's see . . . this is my third book now . . . the first I dedicated to my wife, kids, mom, dad, and John Sheridan for keeping the Shadows off our backs. The second I dedicated to all the animals I've eaten, a batch of childhood friends who helped shape my early life, Denny Crane, and my wife and kids once more. So, I've covered everyone that counts at least once. So, who to dedicate this one to? Who's worthy of my adulation and respect?

Oh, oh! I know . . . ME! I dedicate this book to ME!

OK, fine, I guess I can't do that.

So, I instead dedicate this book to my wife and kids. AGAIN.

I dedicate this book to my sister because I just realized I ***didn't*** *cover everyone that counts, but now I have, sis!*

I dedicate this book to the folks at Harmonix and Bungie because Guitar Hero, Rock Band, and Halo just flat-out rule. I need some Dream Theater, Queensryche, Shadow Gallery, Fates Warning, and Enchant to make my life complete though, so get on that, OK guys?

And lastly, I dedicate this book to all the alien species we have yet to meet. I just hope those dudes need books on web programming, because my kids eat like you wouldn't believe! I'll even take Quatloos!

Contents at a Glance

Foreword xiii
About the Author xv
About the Technical Reviewer xvii
About the Illustrator xix
Acknowledgments xxi
Introduction xxiii

PART 1 Setting the Table

CHAPTER 1 An Introduction to Ajax, RPC, and Modern RIAs 3
CHAPTER 2 Getting to Know DWR 43
CHAPTER 3 Advanced DWR 95

PART 2 The Projects

CHAPTER 4 InstaMail: An Ajax-Based Webmail Client 129
CHAPTER 5 Share Your Knowledge: DWiki, the DWR-Based Wiki 189
CHAPTER 6 Remotely Managing Your Files: DWR File Manager 259
CHAPTER 7 Enter the Enterprise: A DWR-Based Report Portal 329
CHAPTER 8 DWR for Fun and Profit (a DWR Game!) 419
CHAPTER 9 Timekeeper: DWR Even Makes Project Management Fun! 457

INDEX 523

Contents

Foreword xiii
About the Author xv
About the Technical Reviewer xvii
About the Illustrator xix
Acknowledgments xxi
Introduction xxiii

PART 1 Setting the Table

CHAPTER 1 An Introduction to Ajax, RPC, and Modern RIAs 3

A Brief History of Web Development: The "Classic" Model 3
Dawn of a Whole New World: The PC Era 5
Yet Another Revolution: Enter the Web 7
What's So Wrong with the Classic Web? 11
Enter Ajax 14
Why Is Ajax a Paradigm Shift? On the Road to RIAs 18
The Flip Side of the Coin 23
Let's Get to the Good Stuff: Our First Ajax Code, the Manual Way 25
A Quick Postmortem 29
Hey, I Thought This Was Ajax?!? 30
Cutting IN the Middle Man: Ajax Libraries to Ease Our Pain 33
Alternatives to Ajax 36
Hmm, Are We Forgetting Something? What Could It Be?
Oh Yeah, DWR! 39
Summary 40

CHAPTER 2 Getting to Know DWR 43

First Things First: Why DWR at All? 43
DWR: RPC on Steroids for the Web 45
DWR Architectural Overview 47
Getting Ready for the Fun: Your DWR Development Environment 49

A Simple Webapp to Get Us Started 52
Getting the Lay of the Land: Directory Structure 52
From Code to Executable: Ant Build Script 53
Application Configuration: web.xml 54
The Markup: index.jsp 55
On the Server Side: MathServlet.java 56
The Workhorse: MathDelegate.java 58
It's Alive: Seeing It in Action 60
Adding DWR to the Mix 61
The DWR Test/Debug Page 65
Configuring DWR Part 1: web.xml 67
Configuring DWR Part 2: dwr.xml 70
Built-in Creators and Converters 75
The <init> Section 76
The <allow> Section 76
The <signatures> Section 79
Interacting with DWR on the Client 81
Basic Call Syntax 81
Call Metadata Object Approach 82
A Word on Some Funky Syntax 83
Setting Beans on a Remote Object 83
Extended Data Passing to Callbacks 87
Interacting with DWR on the Server 88
DWR Configuration and Other Concepts: The engine.js File 90
Call Batching 92
A Quick Look at util.js, the DWR Utility Package 92
Summary 94

CHAPTER 3 Advanced DWR 95

Locking the Doors: Security in DWR 95
Deny by Default 96
J2EE Security and DWR 98
When Perfection Is Elusive: Error Handling in DWR Applications 101
Handling Warnings 102
Handling Errors 102
Handling Exceptions 102
Edge Cases: Improper Responses 102
The Mechanics of Handling Exceptional Situations 105
Another Word on Exceptions 106
Help from Elsewhere: Accessing Other URLs 107

Turning the Tables: Reverse Ajax 109
Polling 111
Comet 112
Piggybacking 114
The Code of Reverse Ajax 115
Don't Go It Alone: Integration with Frameworks and Libraries 117
Spring 118
JSF 119
WebWork/Struts 2 119
Struts "Classic" 120
Beehive 121
Hibernate 122
Something Old, Something New: Annotations 122
Summary 125

PART 2 The Projects

CHAPTER 4 InstaMail: An Ajax-Based Webmail Client 129

Application Requirements and Goals 129
Dissecting InstaMail 130
Configuration Files 132
The Client-Side Code 134
The Server-Side Code 163
Suggested Exercises 187
Summary 188

CHAPTER 5 Share Your Knowledge: DWiki, the DWR-Based Wiki 189

Application Requirements and Goals 189
FreeMarker 191
Apache Derby 194
Spring JDBC 195
Dissecting DWiki 197
Configuration Files 199
The Client-Side Code 205
The Server-Side Code 230
Suggested Exercises 257
Summary 258

CHAPTER 6 Remotely Managing Your Files: DWR File Manager 259

Application Requirements and Goals 259
dhtmlx UI Components 261
Jakarta Commons IO 268
Jakarta Commons FileUpload 269
Dissecting Fileman 270
Configuration Files 273
The Client-Side Code 278
The Server-Side Code 314
Suggested Exercises 326
Summary 326

CHAPTER 7 Enter the Enterprise: A DWR-Based Report Portal 329

Application Requirements and Goals 329
Spring Dependency Injection (IoC) 331
DataVision 333
Quartz 336
script.aculo.us 337
A Sample Database to Report Against 340
Dissecting RePortal 341
Configuration Files 346
The RePortal Database 351
The Client-Side Code 352
The Server-Side Code 385
Suggested Exercises 416
Summary 417

CHAPTER 8 DWR for Fun and Profit (a DWR Game!) 419

Application Requirements and Goals 419
DWR Annotations 420
Reverse Ajax in Action 421
Anything Else, or Can We Get Goin' Already?!? 422
Dissecting InMemoria 423
Configuration Files 424
The Client-Side Code 426
The Server-Side Code 441
Suggested Exercises 456
Summary 456

CHAPTER 9 Timekeeper: DWR Even Makes Project Management Fun! 457

Application Requirements and Goals 457
HSQLDB 458
Hibernate 459
Ext JS 461
Dissecting Timekeeper 463
Configuration Files 465
The Client-Side Code 471
The Server-Side Code 507
Suggested Exercises 520
Summary 521

INDEX 523

Foreword

The funny thing about getting heavily involved in an open source project is the roller coaster ride you embark on. There's the buzz from seeing the hits to the web server and reading what people think of your project. There's the gnawing feeling of responsibility when you discover very large web sites using your code, and you're worried about bugs you might have created. There's the total flat feeling when a friend tells you he or she is taking your code out of a project because he or she prefers an alternative; and there's the burnout when you just can't keep up with the volume of work and realize that a huge percentage of what you do is not directly development related.

My experiences with open source have opened a huge number of doors. I've met people whom I wouldn't have met otherwise and had job offers that I wouldn't have dreamed of before. There really is a magic buzz to open source.

Marc Andreeson, one of the minds behind Netscape and Ning, wrote recently about how to hire good developers. To paraphrase Marc: "Hire someone that has worked on open source software" (`http://blog.pmarca.com/2007/06/how_to_hire_the.html`).

Some companies rate candidates using trick questions: they get the developers who are good at typing "interview questions" into Google. Some companies rate candidates using industry certifications (MCSD, SCJD, etc.): they get people that work at rich companies that depend on training, and not talent. Some companies rate candidates using CVs/resumes: they end up hiring "talent embroiderers." Some companies rate candidates using interviews: they get the people who sound good and look good.

Unsurprisingly, these selection techniques don't get you the best candidates. So how do you find the developers who love writing good code, who get a buzz from solving the problem in a neat way, and who do take pride in their work?

The answer according to Marc, and according to my experience, is to hire people who love their work enough to get involved with a project that was optional.

So here's your invitation to get a leg up on getting a job with people who hire great developers: get into open source development. It doesn't have to be DWR, although we'd love to have the extra help. Just pick something that excites you and get involved.

The problem with getting started is a typical crossing-the-chasm problem. The first few minutes are easy. You've used a project, liked it, and maybe joined the mailing list. You might even have found something you would like to work on. When you are involved in a project, you know what you are doing and can contribute. But there is a chasm between these places where you are learning the code, learning how the project does things, learning the process, and so on. While you are crossing the chasm, you are unproductive because you are in unfamiliar territory.

So here are a few hints about how to cross the chasm. First, find somewhere that the chasm isn't too wide—start by fixing something small. The chance of any IT project failing is inversely proportional to the size of the project. Start with a simple feature that makes something better. Almost all IT projects have these in abundance.

Second, don't think that because it's tricky, you must be stupid, or that the project must be misguided. There are always reasons why things are tricky. The answer could be historic: when the code was written, people didn't expect the code to be used in this way. Or maybe there is some refactoring that needs doing that hasn't been completed. DWR's code is fairly good because the code is young and we're fanatical about refactoring, but some projects have more history to them.

The difference between those who can cross the chasm and those who can't is drive. You don't need to be a genius, have a brilliant CV, or look good at an interview. Even the ability to type "interview questions" into Google is optional. The people who can cross the chasm are those with the drive to succeed.

Getting involved can come in many forms, and sometimes it's even sort of tangential to the project itself, such as writing a book about the project. Sometimes the tangential help is some of the most valuable. The things developers leave out are often things they are bad at. For years, I've wanted there to be a DWR book but known I'm the wrong person to write it, so I'm particularly pleased to see Frank step forward to write the first DWR book. Thanks for having the drive to get involved, Frank.

Joe Walker
Creator of DWR

About the Author

FRANK W. ZAMMETTI is a developer/architect/whatever is called for at any particular moment in time for a large mutual fund servicing company in the United States by day, and a multi-project open source contributor by night. And, after three books, he can probably finally say "author by night" legitimately too!

Frank has been involved, in one form or another, with computers for 25+ years, which is about 75 percent of his life thus far (only counting the natural part anyway) and has been programming right from the start. When other kids were out playing tag and riding bikes, Frank was . . . well, he was out there with them because his mom wouldn't let him sit in the house all day. But at night, ah at night, Frank was hacking away at code all through the Hour of the Wolf, when he wasn't blowing up his dad's workbench with some wacky electronics contraption at least. About 15 of those 25 years have been "professional" years, which simply means he was (and still is) being paid to pretend he knows what he's talking about.

Frank has written two other books which have garnered rave reviews. One guy named Bob said of his first book, "Frank's writing is more tolerable than most," and Jimbo McMalmun from Piedmont wrote, "Of all the books on this topic I've read, Frank's was the first." Lastly, *Online Computer Magazine of America* said: "The bright yellow cover made us think of bees, which is the same grade we give Mr. Zammetti's efforts." Seriously folks, you flatter me!

Frank lives in Pennsylvania with his wife of 13 years, his two children Andrew and Ashley, his dog Belle, and two guinea pigs that apparently have not heard the phrase "Never bite the hand that feeds you." When Frank isn't writing or coding, he can most usually be found getting his rear end handed to him by Andrew in Guitar Hero, or trying to get yet another game to work on Ashley's outdated PC (she's only four and yet already knows more about computers than many of the adults Frank has met over the years!).

Frank has a deep set of personal beliefs that guide his life. Among them are the belief that the Daleks are redeemable, the belief that tacos were handed down from God herself, and the belief that *Friends* is singularly the most overrated show of all time.

And in case you haven't guessed by now, Frank truly believes that next to a healthy colon, little in life is more important than laughter, so in all seriousness, he hopes you have a good time reading this book **as well as** learn a thing or two!

About the Technical Reviewer

HERMAN VAN ROSMALEN works as a developer/software architect for De Nederlandsche Bank N.V., the central bank of the Netherlands. He has more than 20 years of experience in developing software applications in a variety of programming languages. Herman has been involved in building mainframe, PC, and client-server applications. Since 2000, however, he has been involved mainly in building J2EE web-based applications. After working with Struts for years (pre-1.0), he got interested in Ajax and joined the Java Web Parts open source project in 2005. Besides this book, Herman has also served as technical editor for the Apress titles *Practical Ajax Projects with Java Technology* and *Practical Javascript, DOM Scripting, and Ajax Projects*. Herman lives in a small town, Pijnacker, in the Netherlands with his wife Liesbeth and their children Barbara, Leonie, and Ramon. You can reach him via e-mail at herros@gmail.com.

About the Illustrator

ANTHONY VOLPE did the illustrations for this book. He has worked on several video games with author Frank Zammetti, including Invasion: Trivia!, IO Lander, K&G Arcade, and Ajax Warrior. Anthony lives in Collegeville, Pennsylvania, and works as a graphic designer and front-end web developer. His hobbies include recording music, writing fiction, making video games, and going to karaoke bars to make a spectacle of himself.

Acknowledgments

There are quite a few people who played a role in making this book happen, so I'll do my best to remember everyone, but I probably won't, so apologies in advance.

First and foremost, I'd like to thank Joe Walker, creator of DWR, for, well, **creating DWR**! I'd also like to thank him for his support in writing this book, writing the foreword, and asking me to copresent a session with him at The Ajax Experience in Boston (next time we can actually rehearse and I can earn my pay!).

I'd like to acknowledge all the folks at Apress who worked on this book and gave me a third authoring experience that was an absolute pleasure. Beth Christmas, Ami Knox, Steve Anglin, Kelly Winquist, April Eddy, and all the folks I don't even know by name, thank you!

Anthony Volpe once again worked his graphic/art magic on the illustrations in this book and even on some diagrams this time around. Thanks, man!

I have to, of course, throw the usual thanks at Herman Van Rosmalen for again taking on technical review duties for me. After doing a technical review on Ian Roughley's book *Practical Apache Struts 2 Web 2.0 Projects* myself, I now understand exactly the effort that work involves, and you've been with me for three books now, and your efforts have never been appreciated more, my friend!

Last but certainly not least, I'd like to thank **YOU** for buying and reading this book! A book without a reader is like a donut without a police officer (I kid, I love the folks in blue!), peanut butter without chocolate, or a politician without a scandal: some things just can't exist without the other!

As I said, I know I'm almost certainly forgetting to acknowledge someone here, so how about I just thank anyone who has ever, is now, or will ever draw breath and be done with it? If I had the technology, I could be like Wowbagger the Infinitely Prolonged, but handing out "Thanks!" instead of insults. Alas, I've been busy writing this book so have not yet completed my Grand Unified Theory that would allow for that, so this'll have to do!

Introduction

DWR. Three little letters that possess more power than General Zod under a yellow sun. Three little letters that amount to a whole book's worth of text.

You know, this is my third authoring gig, and the first two were kind of similar in that they covered a variety of libraries and toolkits, and all sorts of different approaches to Ajax, RIA, and Web 2.0 development. This time around though, I'm writing about a single topic, and when I was first asked to do it, I was a little worried frankly . . . Was there enough to write about? Would I get bored? How much different would it be to stay, essentially, on one topic for 500+ pages?

If it had been any other library, I'm not sure what the answer would be, but with DWR, as it turns out, it was both easy and enjoyable the whole way through!

You see, DWR makes developing advanced webapps almost **too** easy sometimes! It makes it effortless to do some things that can at times be difficult to do, and of course do well. It puts so much power in the palm of your hand without it being a heavy lift that you sometimes have to stop and realize just what kind of magic it must be doing under the covers to make it look so easy at the level you interact with it on.

Now, one way in which this book is just like my previous two is that it takes a very pragmatic approach to learning. I know personally, I need to see things in action, and more than that, I need it to be explained to me. Don't just throw code in front of me; that's often more trouble than it's worth. Instead, put code in front of me that's commented well, that's constructed in a consistent manner, and that has some explanation to go along with it. For me, that's how I learn best, and I know I'm not alone. Also, don't waste my time with contrived, overly simplistic examples that don't go deep enough to be of any real use. No, show me a real, working, practical application and tear it apart, that's what I like. And the fact that someone has been willing to publish three such books from me proves there must be like-minded people out there!

Another way this book is like my previous efforts is that I have a guiding philosophy that says that life is pretty darned tough under the best of circumstances, so why not lighten the load a little and bring laughter to things any chance we get? That comes through in my writing. My friends and coworkers would tell you that I'll make a joke about just about anything at just about any time. It's my way, for better or worse (and I'd be lying if I didn't say it was for worse sometimes!). That's also the way I write. The greatest compliment I've been paid, a couple of times by various folks, is that I write like I'm sitting there speaking to you. It's not on purpose; it's just the way I naturally write. Enough people seem to like it and think it works to allow me to keep getting books published, so I surmise it's not a bad thing.

That's what this book is: a series of practical applications, torn apart and described, in a (hopefully!) entertaining style that will, with a little luck, keep you smiling as you learn about a truly great library in DWR.

An Overview of This Book

This book is broken down into two sections. The first section, which consists of the first three chapters, is the more "academic" section. Here's a rundown of the chapters:

- Chapter 1 is an introduction to Ajax. It covers the history of application development, how it's evolved onto the Web, and how Ajax is used in modern webapp development. It even touches on some basic Ajax code.
- Chapter 2 formally introduces DWR, describing what it is, how it works (at a high level), and gets into some basic usages of it.
- Chapter 3 goes into more depth on DWR, including more advanced features like reverse Ajax.

After this section comes the second section, which is the projects. Here we have a series of six chapters presenting one application in each:

- Chapter 4 is where we develop InstaMail, an Ajax-based webmail client.
- Chapter 5 shows how to develop a wiki application using DWR. This chapter includes usage of the Freemarker templating library and the Derby database, a pure Java embeddable database.
- Chapter 6 shows how to develop a decent file manager application, à la Windows Explorer. It uses the dhtmlx GUI widget components to make a good-looking and highly functional user interface, as well as various Jakarta Commons libraries.
- Chapter 7 is the chapter in which we develop an enterprise reporting portal. This application utilizes the DataVision open source reporting tool, as well as script.aculo.us for UI effects, Spring for database access, and Derby once again for a data store.
- Chapter 8 shows how to build a game using DWR. I admit it's not going to rival World of Warcraft or anything, but it *does* show how reverse Ajax, sometimes called Comet, can be used with DWR with little effort.
- Chapter 9, the final chapter, is where we construct a time-tracking system for managing projects and booking time. This project uses the fantastic Ext JS library to create a UI that has a very native feel to it. This project also uses Hibernate for data access and HSQLDB for its data store.

As you can see, the projects cover a lot of territory and mix in a number of supporting libraries that, when combined with the power of DWR, make for some pretty decent applications without having to kill ourselves writing every last bit of code.

Obtaining This Book's Source Code

One thing that's important to me, and this may sound a little gross at first, is that you should be able to, by and large, read this book in the bathroom. What I mean is, you generally shouldn't **have** to be sitting in front of a computer to get the full benefit from this book. To that end, I try to show as much of the code as possible, as well as screenshots of each application, enough so that if you never play with the real thing, it won't kill the experience entirely.

That being said, when you write a book, you sometimes in fact do have to make sacrifices and not show everything. So, even though I strive to make this book "bathroom-ready™," it will be assumed at various times that you have in fact downloaded all the source code and are running the application or can refer to the full source that won't be on the printed page. To that end, the source code for all the projects in this book, as well as WAR files you can drop in your servlet container of choice immediately to run, can be downloaded from the Apress web site at `www.apress.com`. Click the Source Code/Download link and then find *Practical DWR 2 Projects* in the list. From the book's home page, you can download a single ZIP file that contains everything, all code, WAR files, and anything else that may be required. Each chapter has its own directory, so it's easy to find what you need at any point.

Obtaining Updates for This Book

Look, I'm purfect, and hence you willn't find not a sinngle error anywhare in this bok.

AHEM.

Seriously though, writing a book is a lot of hard work and late, tired nights when you have a full-time job and two kids after that. Even with a diligent author, a competent technical reviewer, an editor, copy editor, layout editor, project manager, and maybe others, mistakes are bound to creep in. I will therefore apologize in advance and beg your forgiveness for anything you may find!

That being said, a current errata list is available on this book's home page on the Apress web site, along with information on how to submit reports of anything you find. I very much appreciate any such feedback you would like to offer and want to assure you that as thanks, I will put in a good word for you in the afterlife (of course, you'd better hope I go to the right place!).

Contacting the Author

If for some bizarre reason you would like to contact me, feel free to e-mail me at `fzammetti@omnytex.com`. I loathe spam though, so if you're trying to sell me any sort of "medication," want to refinance my mortgage, have some pictures I **have** to see, or otherwise just want to annoy me, please note that I retain the right to launch a small nuclear volley in your general direction (and as powerful as DWR is, I wouldn't be surprised if control of a nuclear arsenal is a feature I just didn't notice!).