Pro Drupal Development

John K. VanDyk and
Matt Westgate

Apress®

Pro Drupal Development

ISBN-13 (pbk): 978-1-59059-755-2

ISBN-10 (pbk): 1-59059-755-9

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

Lead Editor: Matt Wade
Technical Reviewers: Steve Potts, Robert Douglass
Editorial Board: Steve Anglin, Ewan Buckingham, Gary Cornell, Jason Gilmore, Jonathan Gennick, Jonathan Hassell, James Huddleston, Chris Mills, Matthew Moodie, Jeff Pepper, Paul Sarknas, Dominic Shakeshaft, Jim Sumser, Matt Wade
Project Manager: Elizabeth Seymour
Copy Edit Manager: Nicole Flores
Copy Editors: Nicole Flores, Heather Lang, Susannah Davidson Pfalzer
Assistant Production Director: Kari Brooks-Copony
Production Editor: Janet Vail
Compositor: Susan Glinert
Proofreader: April Eddy
Indexer: John Collin
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 2560 Ninth Street, Suite 219, Berkeley, CA 94710. 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.

For the Great Architect and to my incredibly patient wife and children.
—John VanDyk

To my wife, who is my best friend and my true love, and to our ferrets,
who bring me more joy than any other quadrupeds I know.
—Matt Westgate

Contents at a Glance

Contents

Foreword

Some things just happen in life—you don't plan them. You might go the grocery store and be unable to find the item you are looking for but just happen to meet the love of your life. It's not something you plan on.

When I was a student, I needed a small web-based tool to communicate with my friends. Because I wasn't satisfied by existing tools, I set out to create one myself. That tool has since transformed into the content management framework called Drupal.

What started as a hobby project with a small codebase now powers thousands of web sites, including some of the world's most popular ones. The growth of Drupal is not exactly what I planned, and it's a scary thought—not because of a flaw in Drupal as a platform or a project. No, Drupal is a great system with a wonderful community of people who contribute to its growth. Every day, Drupal is proving to be a viable system for building powerful web applications that are stable, extensible, and easy to use. It's scary simply because I never planned for Drupal to become this successful. Drupal's life has been a chain of unexpected and often incredible events that keep taking me by surprise.

When I began work on Drupal, I spent days and days behind the computer, striving to create a better web-based tool. Fewer lines of code and pure elegance were the goals. Completely and utterly focused on Drupal's code and architecture, I aimed to make great software, not necessarily popular software. It turns out that if something is worth using, it will attract attention and be used.

When I graduated, I set about transforming Drupal from being a small communication tool into something more extensive. I used it to run drop.org, a group weblog dedicated to tracking interesting web technologies. In addition, drop.org acted as an experimental platform that let me explore things like RSS feeds, content moderation, forums, and so on. By 2000, drop.org had attracted numerous followers, and people were genuinely interested in my experiments; they started making suggestions and began wanting to take part in the development process. Shortly after that, on January 15, 2001, I made Drupal available as Free Software.

Since then, everyone has been able to download Drupal free of charge. Anyone can run, copy, and modify Drupal and even redistribute modified versions of it—as long other people are given the exact same rights, as spelled out by the GNU General Public License.

Making Drupal freely available was a great decision. The key benefit of using Drupal is not its ease of use or its functionality, though these are important. Drupal's unique value lies in the facts that the project is open and accessible and that there are very few limitations on what you can do with Drupal. What sets Drupal apart from other systems is its thriving community, a direct product of the openness and transparency. The Drupal community is what makes Drupal tick, and the way we, as a community, develop Drupal is what makes it so successful.

Think about this for a moment: what are your chances of becoming the world's foremost expert on a proprietary content management system (CMS) or on proprietary software in general? Unless you're working for the company owning the software and you get access to proprietary documents or high-level meetings within corporate walls, your chances are slim—you simply won't get access to all the internal information.

Contrast this scenario with Drupal development. As a developer, you have access to Drupal's complete source code. You can read up on all the discussion that led to any design decisions, and you can tap right into the brains of the best Drupal developers in the world. In fact, there is nothing that stops you from becoming the best Drupal developer in the world. The only limitation is your willingness to learn.

While these ideas are not new—Free Software has been around for a while—it does explain why I'm so excited about this book for Drupal developers. *Pro Drupal Development* will help more people cut their teeth on Drupal. If anything was missing in the Drupal community until now, it was a great developer book, and by writing this book, John and Matt have made a legendary contribution to Drupal. I didn't plan for that either.

Dries Buytaert
Drupal founder and project lead

About the Authors

JOHN VANDYK began his work with computers on a black Bell & Howell Apple II by printing out and poring over the BASIC code for Little Brick Out in order to increase the paddle width. Before discovering Drupal, John was involved with the UserLand Frontier community and wrote his own content management system (with Matt Westgate) using Ruby.

John's day job is in the entomology department at Iowa State University of Science and Technology where he is a systems analyst and adjunct assistant professor. His master's thesis focused on cold tolerance of deer ticks, and his doctoral dissertation was on the effectiveness of photographically created three-dimensional virtual insects on undergraduate learning.

John lives with his wife Tina in Ames, Iowa. They homeschool their six children, who have become used to bedtime stories like "The Evil HTTP Request and the Heroic Drupal Session Handler."

MATT WESTGATE has been disassembling anything he could get his hands on since he discovered he had opposable thumbs, so it was a natural transition for Matt to enter the world of computers and start hacking open source software.

Matt is a cofounder of Lullabot, an education and consulting firm dedicated to helping people learn how to build and architect community-driven web sites. Lullabot has helped the BBC, Participant Productions, Sony, MTV, and the George Lucas Educational Foundation shine a little brighter online.

Matt lives with his wife and two ferrets and is currently studying to become a Big Mind facilitator.

About the Technical Reviewers

STEVE POTTS graduated from Manchester University, England with a bachelor's degree in applied computing and then went on to a master's degree at the Open University in computing for commerce and industry.

Even before his start in higher education, he was working hard in the defense industry to squeeze an immense amount of failure-resistant software into a remarkably small footprint that digital watches would find miniature now. His work to date has involved hundreds of applications for defense, handheld devices, mobile internet, and the Web.

Given his obvious disposition for being meticulous (his friends have other words to describe this), he is an accomplished technical editor having worked on Java, XHTML, PHP, and wireless publications including Apress's own *Building Online Communities with Drupal, phpBB, and WordPress* (Douglass, Robert T., Mike Little, and Jared W. Smith. Berkeley: 2005)

Steve founded his own technical consultancy outfit, Free Balloon, and holds the rewarding position of chief technical officer at Hawdale Associates, an invigorating usability and design customer experience company operating out of Manchester, England.

ROBERT DOUGLASS, coauthor of *Building Online Communities with Drupal, phpBB, and WordPress*, is a member of the Drupal Association and a consultant for Lullabot. He is the author and maintainer of numerous Drupal modules and a regular contributor to Drupal core.

Acknowledgments

First of all, thanks to our families for their understanding and support during the writing of this book, especially as the true extent of the commitment became apparent.

Drupal is essentially a community-based project. This book could not have happened without the selfless gifts of the many people who write documentation, submit bug reports, create and review improvements, and generally help Drupal to become what it is today.

But among the many, we'd like to thank those few who went above and beyond what could have been expected.

Those include the members of the #drupal internet relay chat channel, who put up with the constant questioning of how things worked, why things were written a certain way, or whether or not a bit of code was brilliant or made no sense at all. Also, we'd like to thank those whose arms we twisted to provide feedback on drafts or figures to increase the usefulness of this book. Among them are Bert Boerland, Larry Crell, Robert Douglass, Druplicon, Kevin Hemenway, Chris Johnson, Rowan Kerr, Bèr Kessels, Gerhard Killesreiter, Jonathan Lambert, Kjartan Mannes, Tim McDorman, Allie Micka, Earl Miles, David Monosov, Steven Peck, Chad Phillips, Adrian Rossouw, James Walker, Aaron Welch, Moshe Weitzman, and Derek Wright. Apologies to those who contributed but whose names we have missed here.

A special thanks to Károly Négyesi, Steven Wittens, Angela Byron, Heine Deelstra, John Resig, Ted Serbinski, Nathan Haug, Jeff Eaton, Gábor Hojtsy, and Neil Drumm for their critical review of parts of the manuscript.

Thanks to Jon Tollefson at Iowa State University and Jeff Robbins at Lullabot for believing that this book was a worthwhile investment of time.

Thanks to the Apress team for showing grace when code examples needed to be changed yet again and for magically turning our drafts into a book.

And of course, thanks to Dries Buytaert for sharing Drupal with the world.

Introduction

The journey of a software developer is an interesting one. It starts with taking things apart and inspecting the isolated components to try to understand the whole system. Next, you start poking at and hacking the system in an attempt to manipulate its behavior. This is how we learn—we hack.

You follow that pattern for some time until you reach a point of confidence where you can build your own systems from scratch. You might roll your own content management system, for example, deploy it on multiple sites, and think you're changing the world.

But there comes a critical point, and it usually happens when you realize that the maintenance of your system starts to take up more time than building the features. You wish that you knew back when you started writing the system what you know now. You begin to see other systems emerge that can do what your system can do and more. There's a community filled with people who are working together to improve the software, and you realize that they are, for the most part, smarter than you. And even more, the software is free.

This is what happened to us and maybe even you. It's a common journey with a happy ending—hundreds of developers working together on one simultaneous project. You make friends; you make code; and you are still recognized for your contributions just as you were when you were flying solo.

This book was written for three levels of understanding. First and most importantly, there are pretty pictures in the form of diagrams and flowcharts; those looking for the big picture of how Drupal works will find them quite useful. At the middle level are code snippets and example modules. This is the hands-on layer, where you get your hands dirty and dig in. We encourage you to install Drupal, work along with the examples (preferably with a good debugger) as you go through the book, and get comfortable with Drupal. The last layer is the book as a whole: the observations, tips, and explanations between the code and pictures. This provides the glue between the other layers.

If you're new to Drupal, we suggest reading this book in order, as chapters are prerequisites for those that follow.

Lastly, you can download this book's code examples as well as the flowcharts and diagrams from `http://drupalbook.com` or `www.apress.com`.

Good luck and welcome to the Drupal community!