SOFTWARE DEVELOPMENT

ELSEVIER
DIGITAL
PRESS

Open Source Software

Implementation and Management

Paul Kavanagh

Open Source Software

This Page Intentionally Left Blank

Open Source Software: Implementation and Management

Paul Kavanagh

ELSEVIER
DIGITAL
PRESS

AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO•
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO

Elsevier Digital Press
200 Wheeler Road, Burlington, MA 01803, USA
Linacre House, Jordan Hill, Oxford OX2 8DP, UK

Copyright © 2004, Elsevier Inc. All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher.

Permissions may be sought directly from Elsevier's Science & Technology Rights Department in Oxford, UK: phone: (+44) 1865 843830, fax: (+44) 1865 853333, e-mail: permissions@elsevier.com.uk. You may also complete your request on-line via the Elsevier homepage (http://elsevier.com), by selecting "Customer Support" and then "Obtaining Permissions."

∞ Recognizing the importance of preserving what has been written, Elsevier prints its books on acid-free paper whenever possible.

Library of Congress Cataloging-in-Publication Data
Application submitted.

ISBN: 1-55558-320-2

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library.

For information on all Digital Press publications
visit our Web site at www.digitalpress.com and www.bh.com/digitalpress

04 05 06 07 08 09 10 9 8 7 6 5 4 3 2 1

Printed in the United States of America

To John, Jessica, Danny, and Allison.

This Page Intentionally Left Blank

Contents

Preface **xvii**

Intended Audience for This Book xviii

How This Book Is Structured xix

Acknowledgements **xxi**

1 Open Source Software: Definitions and History **1**

1.1 Definition of Terms 1

1.1.1 What Is Free Software? 2

1.1.2 What Are Good Examples of Open Source? 3

1.1.3 Is It Necessary to Adopt Open Source Wholesale? 3

1.1.4 Does "Open Source" Mean Linux? 4

1.1.5 Does Open Source Require Different Business Methods? 4

1.1.6 Will All Systems Be Open Source One Day? 5

1.1.7 Is Open Source a Fad That Will Go Away? 5

1.2 A Brief History of Software 6

1.2.1 Early Years 6

1.2.2 Software Companies 6

1.2.3 UNIX 7

1.2.4 BSD 8

1.2.5 GNU and FSF 9

1.2.6 Linux 10

1.2.7 The Personal Computer 11

1.2.8 The Internet 12

1.2.9 The World Wide Web 13

1.3 Summary 14

2 Where Open Source Is Successful 19

2.1 Analytical Framework 19
2.1.1 Disruptive Innovations 19
2.1.2 The Technology Adoption Curve 23
2.1.3 The Open Source Stack 25
2.1.4 Adoption of Specific Open Source Technologies 26
2.2 Open Source Is in Widespread Successful Use 29
2.2.1 Open Source Is the Heart of the Internet 29
2.2.2 Linux Is Shipping a Lot 30
2.2.3 Open Source Appliances Are Everywhere 31
2.2.4 New Companies and New Businesses Use Open Source 32
2.2.5 Open Source Is Broadly Adopted 34
2.3 Examples of Open Source Systems 36
2.4 Summary 39

3 Open Source: The Good, the Bad, and the Ugly 41

3.1 What Is Good about Open Source 41
3.1.1 Why Your Right to View Code Matters 41
3.1.2 Why Your Right to Change and Redistribute Code Matters 43
3.1.3 You Can Buy from Different Vendors and Adopt New Platforms 44
3.1.4 Open Source Avoids Proprietary Information Formats 46
3.1.5 Open Source Allows Integration between Products 46
3.1.6 Open Source Licensing Is Simpler and Less Expensive 48
3.1.7 Open Source is a Good Solution for International Companies 50
3.1.8 There Is a Large Pool of Skilled Open Source Professionals 51
3.2 Open Source Is Not Enough by Itself 52
3.2.1 Deployment Platform 52
3.2.2 Database Platform 53
3.2.3 Software Language, Architecture, and Implementation 53
3.2.4 Data Architecture 54
3.3 How Choosing Open Source Is More Difficult for You 56

3.3.1 Open Source Has a Less Complete Level of Sales Support 56
3.3.2 Specific Product Reviews Will Not Favor Open Source 58
3.3.3 Open Source Products Are Not Bundled, Branded, or Integrated 60
3.4 What Others Say about Open Source 63
3.5 Summary 64

4 Five Immediate Open Source Opportunities 67

4.1 Create an Open Source Lab 68
4.1.1 Review Existing Work 69
4.1.2 Train Developers to Program in
Open Source Languages 70
4.2 Migrate Infrastructure to Samba
and OpenLDAP71
4.2.1 File and Print Servers 71
4.2.2 Manage Use of Windows Proprietary Features 75
4.2.3 Train Administrators in Linux and Samba 76
4.3 Build Some LAMP Applications 76
4.4 Bring New Desktop Systems to
the Underserved79
4.4.1 New and Small Businesses 80
4.4.2 Franchises 81
4.4.3 Call Centers 82
4.4.4 Retail, Food Service, and Hospitality 83
4.4.5 Government, Healthcare, and Education 85
4.4.6 Unlicensed Software 88
4.4.7 International Opportunities 89
4.5 Migrate Applications and Databases to Open Source 90
4.5.1 Evaluate Open Source Databases 92
4.5.2 Replace Small and Old Database Applications with Open Source 92
4.5.3 Migrate UNIX to Linux 93
4.5.4 Evaluate and Purchase Packages on Linux 94
4.5.5 Enterprise Application Software 95
4.6 Summary 96

5 Five More Open Source Opportunities 99

5.1 Introduction 99

5.1.1 Customization and Integration 100
5.1.2 Organization Size 101
5.2 Directory Services 102
5.2.1 Migration and Interoperability 103
5.3 Email 103
5.3.1 UNIX Mail Systems 103
5.3.2 Migration 104
5.3.3 PC-Based Mail Systems 104
5.3.4 Replacing Exchange 105
5.3.5 Integrated Exchange Replacements 107
5.4 Groupware and Collaboration 107
5.4.1 Wiki 107
5.4.2 Other Community Software 109
5.4.3 Weblogs 111
5.4.4 Instant Messaging 111
5.5 Complex Web Publishing 112
5.5.1 Portal Components 114
5.5.2 Open Source Content Portals 116
5.6 Manage User Desktops 117
5.6.1 Analyze Desktop Use and Licensing 120
5.7 Other Possibilities 121
5.8 Summary 123

6 Operating Systems 125

6.1 Contents of the Operating System 125
6.1.1 FreeBSD 128
6.1.2 The Value of Alternative Operating Systems? 129
6.1.3 Using the Shell . . . 129
6.1.4 Recent Linux Improvements 131
6.1.5 Scaling Linux up and Down 132
6.1.6 Security 134
6.2 Linux Distribution Vendors 134
6.2.1 The Many Versions of Linux 134
6.3 Enterprise Distribution Vendors 138
6.4 Community-Supported Distribution Vendors 138
6.4.1 Debian 138
6.4.2 Fedora 139
6.5 International Alternatives 139
6.5.1 Consumer Linux Choices 140
6.5.2 Booting from a CD 140
6.6 Summary 142

7 Open Source Server Applications 145

7.1 Infrastructure Services 145
7.1.1 File and Print Services 146
7.1.2 Directory Services 147
7.2 Web Servers 148
7.2.1 Apache 148
7.2.2 Other Web Servers 149
7.3 Database Servers 150
7.3.1 Classes of Database Servers 151
7.3.2 Analysis of Database System Sizes 151
7.3.3 Open Source Database Choices 162
7.3.4 Database Performance Is Good Enough 164
7.3.5 Competing with Closed Code Databases 166
7.4 Mail Servers 167
7.5 Systems Management 168
7.6 Summary 170

8 Open Source Desktop Applications 173

8.1 Introduction 173
8.1.1 The Open Source Desktop 173
8.1.2 Linux Desktop Share 174
8.1.3 Limitations to Desktop Linux Adoption 174
8.2 Graphical Desktops 175
8.3 Web Browsers 180
8.3.1 Deploying Browsers 180
8.4 The Office Suite 182
8.4.1 OpenOffice.org 183
8.4.2 Competition in the Office Suite Market 185
8.4.3 Comparison of Microsoft Office to OpenOffice 188
8.4.4 Migration from Microsoft Office to OpenOffice 189
8.4.5 Lock-in and Complexity 191
8.4.6 When You Don't Need an Office Suite 194
8.5 Mail and Calendar Clients 195
8.5.1 Professional Applications 196
8.5.2 Drawing and Image Management 197
8.6 Personal Software 198
8.6.1 Running Windows Applications 199
8.7 Summary 201

9 How Open Source Software Is Developed 203

9.1 Methodology 203
9.1.1 Open Source Compared with Closed Code 204
9.1.2 Open Source Compared with Corporate Development 207
9.1.3 Open Source Development Tools 207
9.1.4 Managing People 208
9.2 Languages Used to Develop Open Source Products 209
9.2.1 C and C++ 210
9.2.2 Perl 213
9.2.3 PHP 213
9.2.4 Python 214
9.2.5 Java 214
9.2.6 Other Languages 215
9.3 Cross-Platform Code 215
9.4 Summary 218

10 Managing System Implementation 221

10.1 Implementation Roles 221
10.1.1 Customer Management 223
10.1.2 Program Management 223
10.1.3 Development 223
10.1.4 Testing 223
10.1.5 Communication 223
10.1.6 Deployment 224
10.2 Open Source Impact on Team Issues 224
10.3 Implementation Process 226
10.3.1 Releases 226
10.3.2 Team Roles during the Process 227
10.4 Implementation Principles 228
10.4.1 Resource Trade-offs 228
10.4.2 Frequent Releases 229
10.4.3 Support Elements 230
10.4.4 Watching for Problems 230
10.5 Key Documents 231
10.5.1 Project Definition 231
10.5.2 Risk Management 231
10.5.3 Example of a Risk Assessment 233
10.5.4 Functional Specification 235
10.5.5 Technical Specification 236
10.6 Migration 236

10.6.1 Migration Approaches 237
10.6.2 Assessing the Current System 237
10.7 Interacting with the Open Source Community 239
10.7.1 Hiring from the Community 239
10.7.2 Employee Agreements 240
10.7.3 Repaying the Community 240
10.8 Support 241
10.9 Summary 243

11 Application Architecture 245

11.1 Types of Systems 245
11.1.1 Extreme Systems 246
11.1.2 Transactional Systems 248
11.1.3 Knowledge Management 248
11.2 Tiered Design 249
11.3 Managing Performance and Scalability 251
11.3.1 State Management 251
11.3.2 Queuing 254
11.3.3 Database Design 256
11.3.4 Application Servers 259
11.4 Interoperability 260
11.4.1 Shared Data 261
11.4.2 Process Communication 262
11.4.3 Application Integration Engines 262
11.4.4 Web Services 263
11.4.5 Data Formats 265
11.5 Development Platform Choices 266
11.5.1 Java 266
11.5.2 .Net 268
11.5.3 LAMP 269
11.6 Summary 272

12 The Cost of Open Source Systems 275

12.1 Total Cost of Ownership 276
12.1.1 Staffing Costs 277
12.1.2 Hardware Costs 279
12.1.3 Software Costs 279
12.1.4 Using Third-Party Application and Database Servers 281
12.1.5 Pricing Open Source Software 282
12.1.6 Pricing Closed Code Software 283

12.1.7 Pricing Windows Software 284
12.2 Types of Costs 285
12.2.1 Fixed Costs 286
12.2.2 Off-Budget Costs 286
12.2.3 Sunk Costs 286
12.2.4 Switching Costs 287
12.3 Scenarios 289
12.3.1 Small Organization: Web Site 289
12.3.2 Large Organization: Internal Use 293
12.4 Summary 295

13 Licensing 297

13.1 Types of Licenses 297
13.1.1 Relicensing Only Matters If You Distribute 298
13.1.2 Reciprocal Licenses Are Similar to Commercial Licenses 298
13.2 Licenses in Use 298
13.2.1 Reciprocal Licenses 299
13.2.2 Nonreciprocal Licenses 299
13.2.3 Which License to Use 300
13.3 Mixing Open and Closed Code 300
13.4 Dual Licensing 302
13.5 Other Intellectual Property Issues 303
13.5.1 Provenance 304
13.6 Summary 305

A Resources 307

A.1 Managing an Open Source Lab 307
A.2 Installing an Evaluation Linux System 309
A.2.1 Setting up Interoperability 312
A.2.2 Dual Boot 312
A.2.3 Running Linux on Windows 313
A.2.4 Running Windows on Linux 314
A.3 Next Steps 315
A.4 Top Ten Reasons to Use Open Source Software 315
A.5 Web Links 316

B The Open Source Definition 321

C Examples of Open Source Licenses 323

C.1 GPL 323
C.2 Mozilla Public License 332
C.3 The BSD License 343

Bibliography 345

About the Author 351

Writing Environment 352

Index 353

This Page Intentionally Left Blank

Preface

Open source software is now under serious consideration in many organizations. The success of several open source products, particularly Apache and Linux, in the enterprise has come surprisingly quickly, given that they were not commercially available ten years ago. It is important for managers responsible for adopting technology to be fully aware of the issues that open source represents to their organizations.

A set of new technologies is nothing new to the information technology professional, of course. There are always new technologies and business structures, and the steps that we take to evaluate and adopt them are not so different. But each new set of innovations can also bring new kinds of challenges. Open source offers, in addition to a very rich set of technologies with long histories, a set of new ways to look at certain problems. Issues that are new now include:

- A variety of new licensing options, and claims from some quarters that some of these introduce new business risks
- Opportunities to deal with the loosely structured community that creates open source, from selecting distributions and buying products to hiring and retaining staff
- The possibility that open source software is built and maintained in a different way, and the implications this has for our organizations

The biggest challenge we all face at this time is the difficult climate our industry is in. This is a very tight time for spending, and every technology decision needs to be justified by carefully analyzing costs and risks. Fortunately, open source may be able to help us here, and this book will stick closely to the areas where we can achieve a clear and reasonably rapid return on investment.

Intended Audience for This Book

This book is written for professional managers and implementers of information technology, who are not currently experts in open source software, but who will, over the next year or so, evaluate it and then, in many cases, adopt the technologies, tools, and practices for themselves and their teams.

The book is aimed, in a sense, at generalists. That is partly because most open source software is a set of disruptive technologies rather than in direct line with what is in place. When a set of technologies is mature, we often react to these technologies in specialized, rigid ways. As a new set of technologies is introduced, they can often combine differently. The Web, for instance, changed the boundaries of what many people do. Also, with new ideas, it is often necessary to do everything from building the business case to doing hands-on experimentation. With each component, we will look at the business case, at the technology from an overview, and at the way it works in practice to install and use.

This is a self-teaching guide to the issues faced in transitioning to this new set of technologies. The issues range from business and social concerns through technology and architecture to cookbook-style details. The book should:

- Convince you that there are roles for open source software in your organization
- Place open source development in the context of software history
- Enumerate the various scenarios involving open source that are more (or less) appropriate for most organizations over the next couple of years
- Give you enough examples and successful references for the constituent parts so you feel comfortable entering a process of selecting and using them
- Contrast and compare open source products among themselves and with products for the Windows systems and others that are most widely used today so you know what they are for and which ones you'll need

- Guide the process of getting an open source lab off the ground quickly and without errors and wasted time to the point where you can run it yourself.

This book is aimed at the group known to technology marketers as the "early majority." These are not the very first people to adopt technology, who often do it for its own sake. It is the group of people who, in their professional life, take new ideas in technology and practice and have the vision and practical leadership to put them in widespread business use in their organizations.

The emphasis of the book is on developing new uses for systems using open source software rather than just considering the migration of existing systems. In my experience, it is difficult to cost justify migrations and to achieve customer satisfaction. Migration and interoperability always play a part in system introduction and are thoroughly covered, but only as a tactical element.

How This Book Is Structured

First, given the independent history of the open source community, it is useful to deal with the people, organizations, and programs that have brought us to this point; however, we will do that briefly, because there are many good sources of this information available.

Next, we cover the question of whether open source systems are ready for use in any circumstance, particularly in large conventional organizations (including commercial businesses and government organizations). We will look at the advantages open source has offered to others and can offer to you, as well as the challenges you will likely face in attempting to adopt it and introduce it in your organization.

After this, the book looks at the systems that can be implemented now from the perspective of a business decision maker. My emphasis is the systems that will work and deliver value quickly, the measures used to determine that, and the road maps for deployment. We separate the systems, where open source is already a preferred choice that will work in almost every organization, from those where a careful evaluation must be made or where there is more work to be done before a pragmatic decision would select open source.