

社交应用编程 (影印版)



Programming

Social Applications

O'REILLY®

東南大學出版社

YAHOO! PRESS

Jonathan LeBlanc 著

TP393

1083

KD00921890

社交应用编程 (影印版)

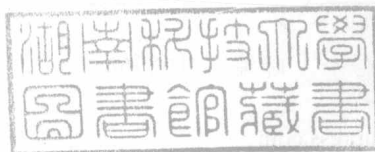
Programming Social Applications

Jonathan LeBlanc

湖南科技大学图书馆



KD00921890



O'REILLY®

Beijing • Cambridge • Farnham • Köln • Sebastopol • Tokyo

O'Reilly Media, Inc. 授权东南大学出版社出版

东南大学出版社

图书在版编目 (CIP) 数据

社交应用编程: 英文 / (美) 勒布朗 (LeBlanc, J.) 著. —影印本. —南京: 东南大学出版社, 2012.4

书名原文: Programming Social Applications

ISBN 978-7-5641-3393-1

I. ①社… II. ①勒… III. ①计算机网络—程序设计—英文 IV. ①TP393

中国版本图书馆 CIP 数据核字 (2012) 第 052772 号

江苏省版权局著作权合同登记

图字: 10-2011-408 号

©2011 by O'Reilly Media, Inc.

Reprint of the English Edition, jointly published by O'Reilly Media, Inc. and Southeast University Press, 2012. Authorized reprint of the original English edition, 2011 O'Reilly Media, Inc., the owner of all rights to publish and sell the same.

All rights reserved including the rights of reproduction in whole or in part in any form.

英文原版由 O'Reilly Media, Inc. 出版 2011。

英文影印版由东南大学出版社出版 2012。此影印版的出版和销售得到出版权和销售权的所有者——O'Reilly Media, Inc. 的许可。

版权所有, 未得书面许可, 本书的任何部分和全部不得以任何形式重制。

社交应用编程 (影印版)

出版发行: 东南大学出版社

地 址: 南京四牌楼 2 号 邮编: 210096

出 版 人: 江建中

网 址: <http://www.seupress.com>

电子邮件: press@seupress.com

印 刷: 扬中市印刷有限公司

开 本: 787 毫米 × 980 毫米 16 开本

印 张: 34

字 数: 666 千字

版 次: 2012 年 4 月第 1 版

印 次: 2012 年 4 月第 1 次印刷

书 号: ISBN 978-7-5641-3393-1

定 价: 72.00 元 (册)

本社图书若有印装质量问题, 请直接与营销部联系。电话 (传真): 025-83791830

Preface

I first began developing social applications when Facebook opened up its developer platform in 2007, giving people like me a taste of the extensive social data that an application can use to improve growth and target personalization settings. At the time, I was building social fantasy sports applications for *CBSSports.com*, pulling user information to enrich that fantasy sports data into a highly personalized state.

It wasn't until 2008, when I joined the partner integrations team in the Yahoo! Developer Network, that I got my first peek at an open source approach to social application development through OpenSocial. What attracted me to OpenSocial was not the fact that you could build an application once and deploy to numerous OpenSocial containers (which proved to be a faulty notion), but rather that through an open source approach I could build social applications on a container and understand how these platforms worked from a core level. I developed a deep drive to explore how the relationships that people form on the Web can enrich and personalize their online lives. This was the starting point of my career advocating open source social technologies.

OpenSocial was the gateway specification for me, leading me to explore the Shindig OpenSocial container, OpenID and OAuth (for authentication and authorization, respectively), the third-party code security technologies Caja and ADSafe, and newer distributed web framework specifications like Activity Streams, PubSubHubbub, and the Open Graph protocol. I quickly came to realize that there was a wide range of open source technologies to enable the construction of rich social frameworks. These technologies and specifications built rich layers of functionality in a simple way using very open methodologies.

These social technologies and specifications are what this book is about. Each chapter uncovers a new layer in the construction of highly viral social applications and platforms. We start by exploring the concepts behind social applications and containers, and then dive into the technologies used to build them. With the application basics down, we look at technologies to secure third-party code on a container, and follow with a discussion of how to secure user information and develop a standard login architecture for platforms. After exposing all of those complex layers, we take an in-depth look at distributed web frameworks that showcase standardization techniques for syndicating activities, discovering rich web and user data from sites and email

addresses. And finally, we explore some wonderful upcoming standards in the social application world.

The content of this book comes from years of direct partner integration work emphasizing the power and features behind open source technologies while collaborating with other developers and companies to create rich social integrations with Yahoo!. This book is a labor of love, as I have both taught and learned from seeing firsthand how social integration technologies are applied to real-world applications and interactions.

Audience

Since this book touches on many different areas of social web application development, container specifications, architecture, and standards, the audience that it will appeal to includes a wide breadth of fields and proficiencies, including (but not limited to):

- Social web application developers who are building applications for Facebook, iGoogle, Orkut, YAP, or any other social networking site that hosts third-party applications
- Application platform architects and server-side engineers who are building products to host a socialized experience
- Frontend engineers who wish to leverage the customization and direct targeting afforded by the massive social graph derived from these technologies
- Hackers and part-time developers who are building small-scale personal projects off of the social web
- Followers of open source technology who want to understand how these technologies are being used to promote social sharing and standards
- Web developers and company teams who wish to develop membership systems and authentication security
- Security gurus and engineers who want to learn about security within online social experiences

Contents of This Book

This book covers many technologies and tools for working with the social web, from container and application development to building highly engaging social graphs.

Each chapter builds on the fundamentals you've learned in the preceding chapters' social explorations. Here are the overarching topics covered throughout the book, broken down by chapter:

Chapter 1

Takes you through an overview of applications, systems, and open source fundamentals to give you a good foundation for implementing the technologies in the remainder of the book.

Chapter 2

Explores the concepts behind the social graph, breaking it down into its fundamental properties.

Chapter 3

This chapter forms the base of our social application development, walking you through the construction of a social container to host third-party applications.

Chapter 4

Examines extensions and features built into the OpenSocial JavaScript libraries.

Chapters 5 and 6

These chapters offer a deeper exploration of the OpenSocial specification. We will look at the core social aspects of a social platform, from the social graph implementation to the data architecture model.

Chapter 7

Our final OpenSocial chapter will dive into advanced OpenSocial topics such as templating, data pipelining methods, and the future of OpenSocial.

Chapter 8

Covers third-party code security models and how a container can protect itself and its users against malicious code using frontend security systems.

Chapter 9

Explores user and application authorization through OAuth, diving into both OAuth 1 and the newer OAuth 2 specification.

Chapter 10

Details experimental and new technologies being developed for constructing social graphs, activities, and distributed web frameworks.

Chapters 11 and 12 (Chapter 12 available online)

These final chapters look at user authentication and authentication security through the use of OpenID and the OpenID OAuth hybrid extension.

Chapter 12, the Glossary, and the Appendix are available on this book's website (<http://www.oreilly.com/catalog/9781449394912>).

Using an Open Source Technology Stack

Since this book's major focus is teaching the fundamentals of social application, container, and graph development using an open source stack, it is only prudent that I outline the technologies we will examine.

The major set of open source technologies we will explore in this book includes:

- OpenSocial for exploring the social graph and application development
- Shindig and Partuza as container implementations using OpenSocial
- OAuth for secure application and user authorization
- OpenID for user authentication, including the hybrid OpenID OAuth extension
- Caja and ADsafe for securing frontend code
- The Open Graph protocol to explore social web entities
- Activity Streams as a foundation for delivering activity content
- WebFinger as a means of discovering public user data using email addresses
- OExchange as a means of sharing any URL with any other web service on the Web
- PubSubHubbub as a means of syndicating user conversations from a root provider to multiple subscribers
- The Salmon protocol for taking the foundation of PubSubHubbub and unifying conversations between publishers and subscribers

As we explore this open stack, we will compare the technologies with many of the current proprietary standards used in the industry today. This will give you a good overview of both the potential and the implications of using open source fundamentals.

Conventions Used in This Book

The following typographical conventions are used in this book:

Plain text

Indicates menu titles, menu options, menu buttons, and keyboard accelerators (such as Alt and Ctrl).

Italic

Indicates new terms, URLs, email addresses, filenames, file extensions, pathnames, directories, and Unix utilities.

Constant width

Indicates commands, options, switches, variables, attributes, keys, functions, types, classes, namespaces, methods, modules, properties, parameters, values, objects, events, event handlers, XML tags, HTML tags, macros, the contents of files, or the output from commands.

Constant width bold

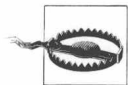
Shows commands or other text that should be typed literally by the user.

Constant width italic

Shows text that should be replaced with user-supplied values.



This icon signifies a tip, suggestion, or general note.



This icon indicates a warning or caution.

Using Code Examples

This book is here to help you get your job done. In general, you may use the code in this book in your programs and documentation. You do not need to contact us for permission unless you're reproducing a significant portion of the code. For example, writing a program that uses several chunks of code from this book does not require permission. Selling or distributing a CD-ROM of examples from O'Reilly books does require permission. Answering a question by citing this book and quoting example code does not require permission. Incorporating a significant amount of example code from this book into your product's documentation does require permission.

We appreciate, but do not require, attribution. An attribution usually includes the title, author, publisher, and ISBN. For example: "*Programming Social Applications* by Jonathan LeBlanc (O'Reilly). Copyright 2011 Yahoo! Inc., 978-1-449-39491-2."

If you feel your use of code examples falls outside fair use or the permission given above, feel free to contact us at permissions@oreilly.com.

Safari® Books Online



Safari Books Online is an on-demand digital library that lets you easily search over 7,500 technology and creative reference books and videos to find the answers you need quickly.

With a subscription, you can read any page and watch any video from our library online. Read books on your cell phone and mobile devices. Access new titles before they are available for print, and get exclusive access to manuscripts in development and post feedback for the authors. Copy and paste code samples, organize your favorites, download chapters, bookmark key sections, create notes, print out pages, and benefit from tons of other time-saving features.

O'Reilly Media has uploaded this book to the Safari Books Online service. To have full digital access to this book and others on similar topics from O'Reilly and other publishers, sign up for free at <http://my.safaribooksonline.com>.

How to Contact Us

Please address comments and questions concerning this book to the publisher:

O'Reilly Media, Inc.
1005 Gravenstein Highway North
Sebastopol, CA 95472
800-998-9938 (in the United States or Canada)
707-829-0515 (international or local)
707-829-0104 (fax)

We have a web page for this book, where we list errata, examples, and any additional information. You can access this page at:

<http://www.oreilly.com/catalog/9781449394912>

To comment or ask technical questions about this book, send email to:

bookquestions@oreilly.com

For more information about our books, courses, conferences, and news, see our website at <http://www.oreilly.com>.

Find us on Facebook: <http://facebook.com/oreilly>

Follow us on Twitter: <http://twitter.com/oreillymedia>

Watch us on YouTube: <http://www.youtube.com/oreillymedia>

Acknowledgments

First and foremost, my heartfelt thanks go out to my wife, Heather, for “putting up with me” throughout these many months of obsession and late nights, and for the constant support she has given me.

Thank you also to Mary Treseler of O'Reilly for being a sounding board for my many questions and for helping to guide me through this process.

To Rachel Monaghan, the copyeditor for this book, I am grateful for the wonderful tone and flow that you have provided in these chapters.

Next, I want to express my gratitude to all of the reviewers of this book: Matthew Russell, Bill Day, Henry Saputra, Mark Weitzel, and Joseph Catera. Thank you all for catching issues before they became immortalized in print, for suggesting wonderful improvements to this text, and for calling me out on content that was simply not good enough to be a part of this book.

My appreciation goes out to my parents and sister for always standing by me and for teaching me that with hard work I can accomplish anything.

A final big thanks goes out to Havi Hoffman, who runs the Yahoo! Press program at Yahoo!. Without her help and support, this book could have never happened.

Table of Contents

Preface	xv
1. Social Application Container Core Concepts	1
What Is a Social Application Container?	2
The User Profile	3
User Friends and Connections	4
The User Activity Stream	4
Implementing Proprietary Versus Open Standards	5
Proprietary Implementation	5
Open Source Implementation	6
Why This Book Covers Open Standards	7
The Embedded Application: Building in a Black Box	7
Embedded Application Security	9
Cross-Site Scripting	10
Same-Origin Policy and Older Browsers	10
Drive-by Downloads	11
Securing Applications	11
The External Application: Integrating Social Data	
Outside the Container	11
Application Views	12
The Home View (Small View)	13
The Profile View (Small View)	14
The Canvas View (Large View)	15
The Default View (Any View)	16
Application Permission Concepts	17
Client-Side Versus Server-Side Applications	19
Using Template Systems for the Markup Layer	19
Using a Blended Server and Client Environment	19
Deferring the Loading of Noncritical Content	20
When Good Applications Go Bad	21
The Portable Flash Application	21
The Underdeveloped View	22

The Copycat View Application	23
The Oversharing Application	24
The Unmonetized Application	24
The Feed Application	25
Application Model Case Studies	26
Case Study: Friendship-Based Social Gaming	26
Case Study: Product Sales Applications	30
Case Study: Location-Based Applications	32
Quick-Start Tips	36
Understand Your Audience	36
Build Social Integration Points Early	37
Build with Monetization in Mind	37
Create Comprehensive Views That Play Off One Another	37
2. Mapping User Relationships with the Social Graph	39
The Online Social Graph	39
Applying the Real-Life Social Graph Online	41
Clustering Users Automatically	41
Privacy and Security	42
Establishing Trust	42
Sharing Private User Data: Opt-in Versus Opt-out	43
The Opt-in Sharing Model	43
The Opt-out Sharing Model	44
Understanding Relationship Models	44
The Follower Model	45
The Connection Model	46
The Group Model	47
Relationships Versus Entities	50
Building Social Relevance: Exploring the Facebook Social Graph	51
Building Upon Real Identity	51
Understanding the Viral Channels	52
Building User Groups	53
Avoiding Irrelevant Social Graphs	53
Defining Entity Likes and Dislikes Through the OpenLike Protocol	54
Integrating the OpenLike Widget	54
How the Shared Likes Appear	55
Conclusion	56
3. Constructing the Foundation of a Social Application Platform	57
What You'll Learn	57
Apache Shindig	57
Setting Up Shindig	58
Installing Shindig on Mac OS X (Leopard)	59

Installing Shindig on Windows	62
Testing Your Shindig Installation	65
Partuza	66
Requirements	66
Installing Partuza on Mac OS X (Leopard)	67
Installing Partuza on Windows	69
Testing the Partuza Installation	75
The OpenSocial Gadget XML Specification	75
Configuring Your Application with ModulePrefs	76
Require/Optional	76
Preload	77
Icon	77
Locale	78
Link	79
Defining User Preferences	81
Enum Data Types	82
Application Content	82
Defining Content Views	83
Inline Versus Proxy Content	89
Putting It All Together	91
4. Defining Features with OpenSocial JavaScript References	95
What You'll Learn	95
Including the OpenSocial Feature JavaScript Libraries	96
Dynamically Setting the Height of a Gadget View	96
Inserting Flash Movies in Your Gadget	98
Displaying Messages to Your Users	100
Creating a Message	100
Positioning the Message Windows	103
Styling the Message and Window	105
Saving State with User Preferences	108
Setting Your Gadget Title Programmatically	110
Integrating a Tabbed Gadget User Interface	111
The Basic Gadget	112
Creating a Tab from Markup	112
Creating a Tab from JavaScript	113
Getting and Setting Information About the TabSet	114
Extending Shindig with Your Own JavaScript Libraries	117
Putting It All Together	121
Building the Gadget XML File	121
Displaying the Gadget Using Shindig	125

5. Porting Applications, Profiles, and Friendships	127
What You'll Learn	127
Evaluating OpenSocial Container Support	127
Core Components of the OpenSocial Specification	129
Core API Server Specification	130
Core Gadget Container Specification	130
Social API Server Specification	131
Social Gadget Container Specification	132
OpenSocial Container Specification	132
Cross-Container Development and Porting	132
Use a Blended Client-Server Environment	133
Decouple Social Features from Mainstream Application Code	133
Avoid Using Container-Specific Tags	133
Porting Applications from Facebook to OpenSocial	134
Employ iframes for Non-Social-Application Constructs	134
Abstract Facebook Function Logic	135
Separate Visual Markup from Programming Logic	135
Use REST Endpoints, Not FQL	135
Employ a Server-Side Heavy Code Implementation	135
Personalizing Applications with Profile Data	136
The Person Object	136
Person Data Extraction Methods	136
Fields Available Within the Person Object	141
Extending the Person Object	162
Capturing the User Profile	168
Using Friendships to Increase Your Audience	170
Making a Request to Capture User Friendships	171
Putting It All Together	171
The Gadget Specification	172
The Content Markup	172
The JavaScript	174
Running the Gadget	175
 6. OpenSocial Activities, Sharing, and Data Requests	 177
What You'll Learn	177
Promoting Your Applications with OpenSocial Activities	178
Personalizing an Application Experience by Consuming Activity Updates	179
Driving Application Growth by Producing Activity Updates	180
Direct Sharing Versus Passive Sharing	183
Direct Sharing	184
Passive Sharing	185
Balanced Sharing	186
Making AJAX and External Data Requests	187

Making Standard Data Requests	188
Pushing Content with Data Requests	190
Using Signed Requests to Secure a Data Connection	191
Putting It All Together	199

7. **Advanced OpenSocial and OpenSocial Next** **203**

What You'll Learn	203
Data Pipelining	203
Data Request Types	206
Making Data Available to Proxied Data Requests	211
Working with Pipelined Data on the Client	212
Handling Errors Produced by the Data Pipe	215
Dynamic Parameters	216
OpenSocial Templating	218
A Different Approach to Markup and Data	219
Rendering Templates	222
Expressions	225
Special Variables	226
Conditionals	229
Looping Content	231
Marrying Data Pipelining and Templating	236
Other Special Tags	238
Template Libraries	240
JavaScript API	244
A Few More Tags: The OpenSocial Markup Language	249
Displaying a Person's Name: os:Name	250
Creating a Person Selector: os:PeopleSelector	250
Display a Person's Badge: os:Badge	251
Loading External HTML: os:Get	251
Localization Support with Message Bundles	251
The OpenSocial REST API Libraries	254
Which Libraries Are Available	254
OpenSocial Next: Areas of Exploration	254
Enterprise Containers	255
Mobile Transitions	255
Distributed Web Frameworks	256
OpenSocial and Distributed Web Frameworks	256
Activity Streams	256
PubSubHubbub	257
Salmon Protocol	258
Open Graph Protocol	258
Putting It All Together	259

8. Social Application Security Concepts	265
What You'll Learn	265
Hosting Third-Party Code Through iframes	266
A Secure Approach: The Caja Project	266
Why Use Caja?	267
Attack Vectors: How Caja Protects	267
Redirecting Users Without Their Consent	267
Mining a User's Browser History	268
Arbitrary Code Execution with document.createElement	269
Logging the User's Keystrokes	269
Setting Up Caja	271
Cajoling Scripts from the Command Line	273
Cajoling HTML and JavaScript	273
Modifying the Cajoler Rendering Format	278
Running Caja from a Web Application	279
Running Caja with an OpenSocial Gadget	281
Adding Caja to a Gadget	282
A Practical Example	282
Using JSLint to Spot JavaScript Issues Early	284
Playing in the Caja Playground	285
Tips for Working in a Caja Environment	286
Implement Code Modularity: Don't Cajole an Entire Project	286
Use Precajoled JavaScript Libraries	286
Don't Rely on Firebug or the Cajoled JavaScript Source Code	288
Don't Embed Events in Markup	288
Centralize JavaScript: Request Data and Markup Only	289
A Lighter Alternative to Caja: ADSafe	290
ADSafe Versus Caja: Which One Should You Use?	291
How to Implement ADSafe	292
Setting Up the ADSafe Object	292
The DOM Object	294
DOM Selection with the Query Method	295
Working with Bunch Objects	299
Attaching Events	306
Defining Libraries	307
Putting It All Together	309
The Data Source	309
The Head: Script Includes and Styles	310
The Body: Markup Layer	311
The Body: JavaScript Layer	312
The Final Result	313
Conclusion	314

9. Securing Social Graph Access with OAuth	315
Beyond Basic Auth	315
Basic Auth Implementation: How It Works	316
The Reasons Against Using Basic Authentication	317
The OAuth 1.0a Standard	318
OAuth 1.0a Workflow	319
The End-User Experience	327
Two-Legged Versus Three-Legged OAuth	329
Three-Legged OAuth Implementation Example	332
Tools and Tips for Debugging Signature Issues	348
OAuth 2	352
OAuth 2 Workflow	352
Implementation Example: Facebook	361
Implementation Example: Requesting More User Information in the Facebook OAuth Process	372
Implementation Example: End-User Experience	375
Tips for Debugging Request Issues	376
Conclusion	380
 10. The Future of Social: Defining Social Entities Through Distributed Web Frameworks ..	 381
What You'll Learn	381
The Open Graph Protocol: Defining Web Pages As Social Entities	382
The Rise and Fall of Metadata	382
How the Open Graph Protocol Works	383
Implementing the Open Graph Protocol	384
A Real-World Example: The Facebook Open Graph	390
Practical Implementation: Capturing Open Graph Data from a Web Source	392
The Shortcomings of the Open Graph Protocol	400
Activity Streams: Standardizing Social Activities	401
Why Do We Need to Define a Standard for Activities?	401
Implementing Activity Streams	402
Object Types	406
Verbs	407
WebFinger: Expanding the Social Graph Through Email Addresses	410
Finger to WebFinger: The Origin of WebFinger	410
Implementing WebFinger	411
The Shortcomings of the WebFinger Protocol	413
OExchange: Building a Social Sharing Graph	414
How Does OExchange Work?	414
The Uses of OExchange	415
Implementing OExchange	416