

Contents

Acknowledgments	xi
Introduction	xix
Chapter 1: Emergence(y) of the New Web	1
Rails, Art, and the New Web	2
Art and Engineering	3
The New Web	3
The Truth about Web Applications	5
Patient History: The World Wide Web	5
From Documents to Interfaces	8
The Decline of Semantics	9
Hello, Web Applications	12
Emergence of the New Web	19
Chapter 2: The Rails Concept	21
One Part Framework	22
The Configuration View	23
The Code View	25
The Process View	26
One Part Language	27
Models on Rails	28
Web-Specific Tasks	30
JavaScript	31
Plug-Ins	32
Two Parts Mindset	32
Web Sites Are MVC Applications	32
Web Applications Are Ecosystems	33
Convention over Configuration	34
A Little Assumption Goes a Long Way	34
Aesthetics Matter	36
Restraint Can Be Liberating	37
You Repeat Yourself Too Much	38

Contents

Testing Isn't a Choice	40
The Web Is a Set of Resources, Not Services	41
Summary	42
No Silver Bullet	43
Optimize Your Coding Before You Optimize Your Code	43
Chapter 3: The Server as an Application	45
Model-View-Controller: The Abbreviated Version	47
MVC and the Web	48
The MVC Design Process	50
The View Is Your Spec	51
Example: Social Networking for Cooks	51
Managing the Application Lifecycle	55
Think Agile, Not Engineered	55
Think Controlled, Not Organic	56
Beware of Open-Heart Surgery	56
Summary	57
Chapter 4: Getting the Most from M, V, and C	59
The Best API Documentation Is Free	60
The Model	61
Model Objects Should Understand Each Other	62
Use Exceptions to Make Code Cleaner	64
Mapping from Rows to Objects	67
Polymorphic Associations	68
The World Outside of ActiveRecord	70
The View	70
The Variable Problem	70
Rails-Style JavaScript	72
Partials as Atoms and Molecules	73
Picking the Proper Partials	74
The View Isn't Just HTML	77
The Controller	77
Reusable CRUD	78
Rails Scaffolding Is a Bunch of CRUD	79
Dealing with Two-Step Actions	80
Knowing When to Outsource	83
Knowing When to Refactor	87
Conclusion	88

Chapter 5: Beautiful Web APIs	89
Two Big Ideas for Web APIs	91
The New URL: Addressing Concepts, Not Files	91
The Application Is the API	93
Routing	93
Anatomy of the Web API Call	96
Overlaying the API	97
The respond_to Method	97
Writing a Non-HTML Result	99
Adding Custom MIME Types	103
Registering Types with Rails	104
Creating Your Own MIME Type	105
API-Metering, the Rails Way	105
Authenticating the User	105
The Metering Algorithm	107
Applying Metering via Filters	108
What about SOAP/XML-RPC Services?	109
Summary	112
Chapter 6: Resources and REST	115
A Web of Resources	116
Identifying Resources	117
Talking About Resources	117
Representing Resources	118
Representational State Transfer	118
HTTP: The Resource CRUD	119
Defining an Application in Terms of Resources	122
Communicating with the Client: Resources as Your API	126
Put Another Way: The Network Is the Computer	127
REST and Rails	128
Mapping Resources in the Router	129
But It's the Real World: Named Routes are Still Needed	130
Resource Scaffolds	133
Nested Resources	133
Singleton Resources versus Regular Resources	137
Summary	138
Chapter 7: The Five Styles of AJAX	139
The Big Secrets	141
AJAX Isn't Necessarily the Hard Part	141

Contents

AJAX Introduces Tough Design Issues	141
You Have Your Pick of JavaScript Frameworks, Even in Rails	143
The Five Styles of AJAX	144
Proxy Style	146
Partial Style	148
Puppet Style	149
Compiled-to-Web Style	151
In-Place Application Style	153
AJAX as Just Another API	155
Rails-Style AJAX	157
Partial-Style AJAX Controllers (and AJAX CRUD)	157
Puppet-Style AJAX Controllers (and RJS)	160
Elegant Degradation	162
Moving Backward from the Partial-Style	163
Moving Backward from Rich User Interfaces	164
Summary	165
Chapter 8: Playing with Blocks	167
The Block Mindset	169
Comparing Methods, Procs, and Blocks	173
Methods	173
Procs	177
Blocks	179
Moving Between Blocks and Procs	180
The Big Scope Experiment	181
Experiment 1: Blocks Are Affected by Changes in Their Source Environment	182
Experiment 2: Blocks Can Affect the Environment from Which They Came	184
Block Patterns, Blocks in Rails	186
Iteration	186
Aspect-Oriented Programming	187
Building Output in HTML and XML	192
Dual-Use Functions	194
The Callback	194
Summary	195
Chapter 9: Mixins and Monkey Patching	197
Mixins	199
Organizing Code into Modules	199
Methods in Modules	201
Mixing Modules into Classes	202
Mixins in Rails	205

Monkey Patching	210
Eval: The Interpreter's Back Door	211
Eval's Two Siblings	213
Good Monkey Patching Technique	219
Summary	222
Chapter 10: Code That Writes Code (That Writes Code)	225
Dynamic Code and DSLs Revisited	227
Code-Writing Macros	228
Creating Methods on the Fly with <code>define_method</code>	228
<code>define_method</code> Example: The Pentagon and the Kremlin	229
Scope and <code>define_method</code>	230
Using <code>define_method</code> for Rails Macros	232
Macro Summary	234
Calling Methods That Don't Exist: Objects That Adapt to the Way You Use Them	234
Some Basic Examples	236
Example: A Shortcut for <code>Array.each</code>	237
Beware of Catching Everything	240
<code>method_missing</code> Patterns	241
Implementing <code>method_missing</code> Patterns	242
Reflection	248
Variables and Constants	249
Methods	250
Modules	251
Summary	252
Chapter 11: How I Learned to Stop Worrying and Love the Schema	253
Bringing the Database into the Picture: The LAMP Stack	254
Thinking in Migrations	256
Writing Migrations	257
Performing Schema Migrations	259
Team Schema Development	260
Seeding Data for Production	262
Small Datasets: Seed Migrations	262
Medium Datasets: Seed Fixtures	263
Large Datasets: Dumpfiles	264
When a Database Isn't Enough	266
Model Object Hierarchies	266
Storing Lists, Hashes, and Other Fun Things	271
Custom Getters and Setters	272
Summary	273

Contents

Chapter 12: Behavior-Driven Development and RSpec	275
Behavior-Driven Development	276
RSpec: BDD for Ruby and Rails	279
The Spec Development Cycle	279
Writing the Spec	280
Implementing Examples	281
Matchers	282
Custom Matchers	285
Before and After	287
An Example Trip through the Development Cycle	288
Part 1: Writing The Story	288
Part 2: Writing the Specs	288
Part 3: Initializing and Writing a Basic Test	290
Part 4: Writing Behavior Tests That Motivate Development	291
Part 5: Completing the Behavioral Test Implementations	293
But Wait, There's More	295
Summary	296
Index	297