Table of Contents

Preface xv

1: Introduction **to Regular Expressions** 1

Solving Real Problems 2
Regular Expressions as a Language 4
The Filename Analogy 4
The Language Analogy 5
The Regular-Expression Frame of Mind 6
Searching Text Files: Egrep 7
Egrep Metacharacters 8
Start and End of the Line 8
Character Classes 9
Matching Any Character-Dot 11
Alternation 12
Word Boundaries 14
In a Nutshell 15
Optional Items 16
Other Quantifiers: Repetition 17
Ignoring Differences in Capitalization 18
Parentheses and Backreferences 19
The Great Escape 20
Expanding the Foundation 21
Linguistic Diversification 21
The Goal of a Regular Expression 21
A Few More Examples 22

Regular Expression Nomenclature 24
Improving on the Status Quo 26
Summary 28
Personal Glimpses 30

2: Extended Introductory Examples 31

About the Examples 32
A Short Introduction to Perl 33
Matching Text with Regular Expressions 34
Toward a More Real-World Example 36
Side Effects of a Successful Match 36
Intertwined Regular Expressions 39
Intermission 43
Modifying Text with Regular Expressions 45
Automated Editing 47
A Small Mail Utility 48
That Doubled-Word Thing 54

3: Overview of Regular Expression Features and Flavors . 59

A Casual Stroll Across the Regex Landscape 60
The World According to Grep 60
The Times They Are a Changin' 61
At a Glance 63
POSIX 64
Care and Handling of Regular Expressions 66
Identifying a Regex 66
Doing Something with the Matched Text 67
Other Examples 67
Care and Handling: Summary 70
Engines and Chrome Finish 70
Chrome and Appearances 71
Engines and Drivers 71
Common Metacharacters 71
Character Shorthands 72
Strings as Regular Expressions 75
Class Shorthands, Dot, and Character Classes 77
Anchoring 81
Grouping and Retrieving 83
Quantifiers 83

Alternation 84
Guide to the Advanced Chapters 85
Tool-Specific Information 85

4: The Mechanics of Expression Processing 87

Start Your Engines! 87
Two Kinds of Engines 87
New Standards 88
Regex Engine Types 88
From the Department of Redundancy Department 90
Match Basics 90
About the Examples 91
Rule 1: The Earliest Match Wins 91
The "Transmission" and the Bump-Along 92
Engine Pieces and Parts 93
Rule 2: Some Metacharacters Are Greedy 94
Regex-Directed vs. Text-Directed 99
NFA Engine: Regex-Directed 99
DFA Engine: Text-Directed 100
The Mysteries of Life Revealed 101
Backtracking 102
A Really Crummy Analogy 102
Two Important Points on Backtracking 103
Saved States 104
Backtracking and Greediness 106
More About Greediness 108
Problems of Greediness 108
Multi-Character "Quotes" 109
Laziness? 110
Greediness Always Favors a Match. 110
Is Alternation Greedy? 112
Uses for Non-Greedy Alternation 113
Greedy Alternation in Perspective 114
Character Classes vs. Alternation 115
NFA, DFA, and POSIX 115
"The Longest-Leftmost" 115
POSIX and the Longest-Leftmost Rule 116
Speed and Efficiency 118
DFA and NFA in Comparison 118

Practical Regex Techniques 121
Contributing Factors 121
Be Specific 122
Difficulties and Impossibilities 125
Watching Out for Unwanted Matches 127
Matching Delimited Text 129
Knowing Your Data and Making Assumptions 132
Additional Greedy Examples 132
Summary 136
Match Mechanics Summary 136
Some Practical Effects of Match Mechanics 137

5: Crafting a Regular Expression 139

A Sobering Example 140
A Simple Change-Placing Your Best Foot Forward 141
More Advanced-Localizing the Greediness 141
Reality Check 144
A Global View of Backtracking 145
More Work for a POSIX NFA 147
Work Required During a Non-Match 147
Being More Specific 147
Alternation Can Be Expensive 148
A Strong Lead 149
The Impact of Parentheses 150
Internal Optimizations 154
First-Character Discrimination 154
Fixed-String Check 155
Simple Repetition 155
Needless Small Quantifiers 156
Length Cognizance 157
Match Cognizance 157
Need Cognizance 157
String/Line Anchors 158
Compile Caching 158
Testing the Engine Type 160
Basic NFA vs. DFA Testing 160
Traditional NFA vs. POSIX NFA Testing 161
Unrolling the Loop 162
Method 1: Building a Regex From Past Experiences 162

The Real "Unrolling the Loop" Pattern 164
Method 2: A Top-Down View 166
Method 3: A Quoted Internet Hostname 167
Observations 168
Unrolling C Comments 168
Regex Headaches 169
A Naive View 169
Unrolling the C Loop 171
The Freeflowing Regex 173
A Helping Hand to Guide the Match 173
A Well-Guided Regex is a Fast Regex 174
Wrapup 176
Think'. 177
The Many Twists and Turns of Optimizations 177

6: **Tool-Specific Information** 181

Questions You Should Be Asking 181
Something as Simple as Grep 181
In This Chapter 182
Awk 183
Differences Among Awk Regex Flavors 184
Awk Regex Functions and Operators 187
Tcl 188
Tcl Regex Operands 189
Using Tcl Regular Expressions 190
Tcl Regex Optimizations 192
GNU Emacs 192
Emacs Strings as Regular Expressions 193
Emacs's Regex Flavor 193
Emacs Match Results 196
Benchmarking in Emacs 197
Emacs Regex Optimizations 197

7: **Perl Regular Expressions** 199

The Perl Way 201
Regular Expressions as a Language Component 202
Perl's Greatest Strength 202
Perl's Greatest Weakness 203
A Chapter, a Chicken, and The Perl Way 204

An Introductory Example: Parsing **CSV** Text 204
Regular Expressions and The Perl Way 207
Perl Unleashed 208
Regex-Related Perlisms 210
Expression Context 210
Dynamic Scope and Regex Match Effects 211
Special Variables Modified by a Match 217
"Doublequotish Processing" and Variable Interpolation 219
Perl's Regex Flavor 225
Quantifiers-Greedy and Lazy 225
Grouping 227
String Anchors 232
Multi-Match Anchor 236
Word Anchors 240
Convenient Shorthands and Other Notations 241
Character Classes 243
Modification with \Q and Friends: True Lies 245
The Match Operator 246
Match-Operand Delimiters 247
Match Modifiers 249
Specifying the Match Target Operand 250
Other Side Effects of the Match Operator 251
Match Operator Return Value 252
Outside Influences on the Match Operator 254
The Substitution Operator 255
The Replacement Operand 255
The /e Modifier 257
Context and Return Value 258
Using /g with a Regex That Can Match Nothingness 259
The Split Operator 259
Basic Split 259
Advanced Split 261
Advanced Split's Match Operand 262
Scalar-Context Split 264
Split's Match Operand with Capturing Parentheses 264
Perl Efficiency Issues 265
"There's More Than One Way to Do It" 266
Regex Compilation, the /o Modifier, and Efficiency 268
Unsociable $& and Friends 273

The Efficiency Penalty of the /i Modifier 278
Substitution Efficiency Concerns 281
Benchmarking 284
Regex Debugging Information 285
The Study Function 287
Putting It All Together 290
Stripping Leading and Trailing Whitespace 290
Adding Commas to a Number 291
Removing C Comments 292
Matching an Email Address 294
Final Comments 304
Notes for Perl4 305

A: Online Information 309

B: Email Regex Program 313