

## Symbols

---

`$$()` 335, 440  
    and CSS rules 336  
    and RJS 440  
`$()` 20, 318, 384  
    defined 334  
    flexibility of argument  
        type 335  
    and Insertion objects 343  
    multiple arguments 335  
`$()` function 76  
`$_POST`  
    and raw POST data 377  
`$_REQUEST` 380  
    and raw POST data 377  
`$A()` 63, 318, 391  
    and the DOM 329  
    and Strings 319  
`$F()` 352  
    form element types 352  
`$H()` 23, 320, 379  
    and the Hash Object 320  
`$HTTP_RAW_POST_DATA`  
    377, 495  
`$R()` 323  
`<<` method  
    of RJS page object 441

## A

---

absolute positioning 40  
    of DOM elements 345  
absolutize() 346  
Abstract.TimedObserver 354

ActionController 492  
ActionPack 492  
ActiveRecord 432  
ActiveX 5  
    component 19  
addClassName() 340  
addHeader() 493  
Adept 479  
    package manager 480  
Ajax 4, 143  
    adding new programming  
        languages to a project 86  
    benefits to network traffic 91  
    different styles of 72  
    impact on network traffic 87  
    moving a classic web app  
        over to 48  
Ajax helper classes  
    in Prototype 72  
Ajax helper objects 50  
Ajax libraries 25  
Ajax request 379  
Ajax Responder 370, 377, 409  
    as notification system 370  
Ajax.Autocompleter  
    167–168, 428  
    afterUpdateElement  
        option 171, 173  
    frequency option 171–172  
    indicator option 171, 173  
    minChars option 171–172  
    parameters option 171–172  
    paramName option 171  
    tokens option 171, 173  
    updateElement option  
        171, 173

Ajax.Base 50  
Ajax.InPlaceCollectionEditor  
    163  
    ajaxOptions option 166  
    cancelLink option 165  
    cancelText option 165  
    clickToEditText option 165  
    collection option 163, 165  
    externalControl option 166  
    formClassName option 165  
    formId option 165  
    highlightcolor option 165  
    highlightendcolor option 165  
    loadTextUrl option 166  
    okButton option 165  
    okText option 165  
    onComplete option 165  
    onFailure option 165  
    savingClassName option 165  
    savingText option 165  
Ajax.InPlaceEditor 143  
    ajaxOptions option 148, 153  
    callback option 148, 152, 154  
    cancelLink 150  
    cancelLink option 147  
    cancelText option 147, 150  
    clickToEditText option 147  
    cols option 147, 150  
    externalControl option 148  
    formClassName option 147  
    formId option 147, 151  
    highlightcolor option  
        147, 150  
    highlightendcolor  
        option 147, 150  
    loadingText option 147, 150

- Ajax.InPlaceEditor (*continued*)
  - loadTextURL option
    - 148, 150, 153
  - okButton option 147, 150
  - okText option 147
  - onComplete option
    - 148, 152, 158
  - onFailure option
    - 148, 152, 158
  - and Rails 426
  - rows option 147
  - savingClassName option
    - 147, 151
  - savingText option
    - 147, 150–151
  - size option 147, 150
- Ajax.PeriodicalUpdater
  - 73, 83, 422
  - potential dangers of 74
- Ajax.Request 22, 27, 51, 72, 252, 376, 392, 409, 490
  - and HTML Forms 353
- Ajax.Responders 74, 82, 401, 435
- Ajax.Updater 72–73, 76, 78, 421
  - container element 73
  - limitations of 434
  - in Rails 414
  - and Rails
    - remote\_function 423
- Alex P 381
- aliases
  - for methods of
    - Enumerable 306
- all() 304
- animation 387
  - effects 98
  - timeline 389
- anonymous function
  - 16, 279, 285
  - and closures 285
  - and Prototype's event system 350
- any() 304
  - and detect() 306
- Apache 474
- Apache Tomcat 478
- Apache Web Server 478
  - versions 478
- Apache web server 491
- apache\_request\_headers() array
  - and PHP 377
- Appear effect 114
- appendChild() 332, 342
- Apple iTunes 187
- application
  - context 472
  - workflow 142
- application.js
  - in Rails apps 413
- application.rhtml 414
- application/x-www-form-urlencoded 19
- arguments 281
- Array 253
- Array methods
  - and RJS 440
- Array.all() 304
- Array.any() 304
- Array.clear() 301
- Array.collect() 311, 378, 391
- Array.compact() 301
- Array.concat() 296
- Array.detect() 306
- Array.each() 303
- Array.findAll() 306
- Array.first() 300
- Array.flatten() 302
- Array.grep() 308
  - optional iterator function 308
- Array.include() 270, 305
- Array.indexOf() 301
- Array.inject() 313
- Array.invoke() 312
- Array.join() 299, 378
- Array.last() 300
- Array.max() 316
- Array.min() 316
- Array.partition() 307
- Array.pluck() 310, 392
- Array.pop() 295
- Array.prototype 300
- Array.push() 295
- Array.reject() 306
- Array.reverse() 298, 316
- Array.shift() 295
- Array.slice() 296
  - optional arguments 296
- Array.sort() 298, 316
  - providing a comparator function 298
- Array.sortBy() 317
- Array.splice() 296–297
  - to add or substitute elements 297
  - to remove elements 297
- Array.unshift() 295
- Array.without() 301
- Array.zip() 315
- array-like syntax
  - and reflection 261
- arrays
  - associative 293
  - as fundamental language feature 293
  - numerically indexed 293
  - types of 293
- Arrays and Strings 299
- asynchronous 5
  - property of Ajax.Request 54
  - request 27, 46
- auto\_complete\_for 427
- autocomplete feature 426
- Autocompleter 265
- autocomplete control 168
  - in action 169
  - callbacks 173
  - creation 168
  - example 168
  - multiple choices 173
  - options 171
  - server-side code 174
  - server-side control 171
  - styling 179
  - triggering 172
- Autocompleter.Local 181
  - choices option 184–185
  - fullSearch option 184, 186
  - ignoreCase option 184, 186
  - indicator option 184
  - minChars option 185
  - parameters option 184
  - paramName option 184
  - partialChars option 184–186
  - partialSearch option 184–185

## B

---

- balance of code size
  - between client and server 85
- bandwidth
  - consumes 88

- bandwidth (*continued*)
    - overhead of multiple Ajax requests 77
    - shaping using Fiddler 462
  - bar chart 91
  - benefit visual effects
    - benefit of 23
  - bind() 23
  - bindAsEventListener()
    - 20, 23, 351
  - BlindDown effect 115
  - BlindUp effect 115
  - Boo 492
  - breadcrumb 57, 64
    - trail 28, 30–31, 34, 361
  - browser event model 8
    - classic versus W3C 349
  - bubble sort algorithm 298
  - BufferedReader 495
  - buffering of the response
    - on the server 492
  - bugs, cross-canceling 227
  - Builder object 372, 392
  - built-in transitions 126
  - business logic
    - and multiple update problem 77
    - reusing 75
    - separating from presentation 37
- C**
- 
- C# 254, 269, 281, 284, 393
  - caching proxies 459
  - callback 51
  - callback function 284, 290
    - and browser events 349
    - and closures 286
    - for form element observers 354
    - in Prototype and Scriptaculous 286
    - to Image.onload() 392
    - writing by hand 72
    - X-JSON header as argument of 82
  - callback handler 19, 23
  - camel-case convention 439
  - Cancel button 133
  - canceling effects 133
  - carriage returns
    - and HTML parsing 329
  - cascading style sheet. *See* CSS
  - Catalina. *See* Tomcat
  - CGI parameters 50
  - checkbox elements 352
  - checked history
    - of browser event model 349
  - child nodes 391
  - childNodes 299, 318
  - childNodes property
    - of DOM elements 328
    - and pseudo-arrays 329
  - children property of DOM elements (IE only) 329
  - clash in namespaces 266
  - class definitions in JavaScript
    - lack of 258
  - class inheritance
    - in JavaScript 265
  - class methods
    - and JavaScript Functions 278
  - Class object 263
  - Class Proxies
    - generic form of 440
    - in RJS 439
  - Class.create() 263, 269
    - as ‘syntactic sugar’ 265
  - classic web application 29, 327
    - and the DOM 327
    - and HTML forms 351
    - shortcomings of 27
  - classic web application model 43, 47
    - limitations of 41
  - className property 339
    - assigning multiple values XE 340
  - classNames() 341
  - cleanWhitespace() 342
  - clear() 301
  - click and drag. *See* drag and drop
  - client-side
    - domain objects 255
    - logic 60
  - clientX and clientY 290
  - clone() 347
    - optional arguments 348
    - See also* Position.clone()
  - close button 386
  - close-up 27
  - close-up view
    - hiding, in QuickGallery 76
  - closure 16, 201, 278, 284
    - accidental creation of 285
    - and anonymous functions 285
    - as solution to function context switching in objects 287
    - definition 284
    - in JavaScript 278
    - and JavaScript Functions 278
    - and the keyword this 287
    - lack of syntax 285
    - practical use of 278, 284, 286
    - problems in coding 285
    - and scope of variables 285
    - simplifying in Prototype 288
    - and XMLHttpRequest object 19
  - code size, content-versus data-centric Ajax 86
  - coding conventions
    - marking private methods 270
  - collect() 311, 318, 320, 378, 391
  - Combination Effects 114
  - combining words and pictures 361
  - Combo Box 202
  - common functionality
    - abstracting into a mixin 274
  - compact() 301
  - comparator functions 298
    - bubblesort 298
  - comparing styles of Ajax coding 85
  - comparison function
    - for max() and min() 317
  - complex Array
    - manipulations 296
  - complex data structures
    - and arrays 293
  - complex object model 375
  - composite effects 388
  - concat() 296
  - conditional logic
    - applying to arrays 304
  - configuring effects
    - in Rails 424
  - CONNECT 447

- constant strings 177
  - constructor 258
    - for Ajax.Updater 76
  - containment option for Scriptaculous Sortables 370
  - content-centric 47–48, 72, 84
  - content-centric Ajax
    - HTTP traffic generation 91
    - and Prototype 87
    - reducing client-side coding burden 72
  - ContentType 494
  - contentType attribute
    - of JSP page directives 493
  - Content-type header 492
  - context object 283
  - context path 154, 472
    - programmatically rendering 154
  - Control.Slider 188
    - axis option 193–194
    - axis range 193
    - disabled option 194, 196
    - onChange option 194, 197
    - onSlide option 194, 197
    - range option 195
    - setEnabled() 196
    - setEnabled() 196
    - setValue() 196
    - sliderValue option 194
    - values option 194–196
  - controller 175, 178, 419
  - controller class in Ruby on Rails 412
  - controls 141
    - extending with Ajax 141
    - HTML set 141
    - in-place collection editor 162
    - in-place text editor 142
  - cookies 446
  - coordinate space and DOM elements 345
  - core effects 109, 388
  - core JavaScript classes
    - extending 252
  - Costello, Eric 6
  - criteria for evaluating styles of Ajax coding 85
  - Crockford, Doug 6
  - cross-browser 20
    - and DOM parsing 329
    - formatted HTML 329
    - and W3C event model 349
  - CRT television set effect 116
  - CSS 4, 37, 141, 143, 170, 179, 188, 207, 220, 235
    - defining look and feel 382
    - delegating visual layout to 39
    - and the Element object 339
    - fine-grained modifications 341
    - float property 40
    - hover modifier 383
    - overflow 232, 241
    - rule 40, 373
    - selector expression 336
  - CSS display property 383
    - block versus inline 383
  - CSS opacity as modal backdrop 384
  - CSV file 463
  - cumulativeOffset() 345
  - custom HTML attributes
    - and form\_remote\_tag 419
  - custom HTTP headers 376
  - custom JavaScript using RJS 433
  - custom Scriptaculous Effects 386
  - Cygwin 471, 484
- D**
- 
- DarwinPorts 481
  - data- and content-centric Ajax
    - and internet folklore 92
  - data reformatting 160
  - database primary keys
    - and DOM element IDs 328
  - data-centric 47, 49, 84
    - approach to Ajax 80
  - declaring a function as a top level object 279
  - declaring Functions 279
  - defaults argument of javascript\_include\_tag 413
  - defining look and feel using CSS 382
  - defining Object types using prototypes 258
  - DELETE 447, 452
  - deployment descriptor 155, 474
  - design issues 360
  - desktop controls 141
  - detect() 306
  - DHTML 141, 143
    - tooltip libraries 382
  - dialog box 142
  - Dickens, Charles 444
  - Dictionaries associative arrays 293
  - digital cameras 28
  - document base 472
  - Document Object Model. *See* DOM
  - document.forms 299
  - document.getElementById() 328
  - doGet() 491
  - DOM 4–5, 10, 318, 326
    - and Array-like objects 318
    - inspector 327
    - manipulation 86, 121
    - node 75
    - and Prototype 326, 334
    - as structural definition 39
    - as a tree structure 327
  - DOM element 73, 392
  - childNodes property 299
  - getElementsByTagName() 299
  - and Prototype's Element object 336
    - target for content-centric Ajax 76
    - types 331
  - DOM methods 48
    - using to create new nodes 331
  - Donne, John 446
  - doPost() 491
  - dot notation for JavaScript Objects 254
  - drag and drop 10, 12, 205, 367–368
    - customizing, in Rails 430
    - delete operation 205
    - desktop metaphor 205
    - drop target 206
    - example 223
    - for editing slideshows 367
    - in action 223
    - and Rails 428

- drag and drop (*continued*)
    - refreshing 371
    - sorting using 228
    - to reorganize elements 368
    - and trash can 368
    - using Scriptaculous 369
  - Draggable 206–208
    - axis 213
    - callbacks 215
    - change option 211, 215, 217
    - constraint option 211, 213
    - constraints 212–213
    - controlling movement 213
    - current position 216
    - effects 216
    - endeffect option
      - 211, 216–217
    - example 208
    - ghosting 214
    - ghosting option 211, 214, 226
    - grid 212
    - handle option 211, 215
    - handles 215
    - options 210
    - positioning 211
    - revert option
      - 211, 216–217, 226
    - revert zIndex 211
    - reverteffect option
      - 211, 216–217
    - snap option 211–213
    - stacking order 214
    - starteffect option 211, 216
    - zIndex option 214
  - draggable objects 428
  - draggable\_element helper 429
  - drop target 206, 218, 370
  - drop zone. *See* drop target
  - drop\_receiving\_element
    - support for Ajax 430
  - drop\_receiving\_element helper 429
  - DropOut effect 116
  - droppable 206–207
    - objects 428
  - Dropposables 228
    - accept option 219–221
    - add() 218
    - callbacks 221–222
    - constraints 220
    - containment option 219, 221
    - creation 218
    - drop notification 222
    - drop threshold 219
    - example 218
    - hoverclass option 219–221, 223, 226
    - onDrop option 219, 222–223
    - onHover option 219, 221–223
    - options 218
    - overlap option 219, 222
    - removal 218
    - remove() 218
    - selectivity 220
    - visual cues 219
  - dropping 217
  - duration of Effects 388
- E**
- 
- each() 303, 318, 335
  - ease of development 85
    - measuring 85
  - effect toggling, example 119
  - Effect.Highlight 417
  - Effect.SlideShow object 392
  - effects 98
    - Appear 114
    - BlindDown 115
    - BlindUp 115
    - callback options 131
    - canceling 133
    - code triggers 131
    - Combination effects 114
    - common options 107
    - Core effects 109
    - diagnostics 132
    - DropOut 116
    - duration 107
    - Fade 114
    - Fold 117
    - frames per second 108
    - global queue 135
    - Grow 116
    - Highlight 110
    - irregular pacing 134
    - Laboratory 104
    - MoveBy 113
    - multiple 134
    - named queues 136
    - Opacity 109
    - pairs 114
    - Puff 116
    - Pulsate 117
    - queuing 135
    - quick example 99
    - running in parallel 134
    - running serially 135
    - Scale 111
    - Shake 117
    - Shrink 116
    - simultaneous 134
    - SlideDown 115
    - SlideUp 115
    - Squish 117
    - start and end points 108
    - SwitchOff 116
    - table of 103
    - toggling 117
    - transitions 125
    - types 103
  - Effects Laboratory
    - General 104
    - Parallel 135
    - Queuing 137
  - efficiency 85
  - element
    - grouping 217
    - mapping 217
    - ordering 217
    - rearranging 217
    - sorting 228
  - Element object 365
    - and CSS 339
    - flexible calling semantics 336
    - as mixin 336
    - and Prototype 336
  - element proxy in RJS 439
  - Element.addClassName() 340
  - Element.classNames() 341
  - Element.cleanWhitespace() 342
  - Element.extend.cache
    - object 337
  - Element.getStyle() 341
  - Element.hasClassName() 341
  - Element.hide() 336
  - Element.remove() 319, 337
  - Element.removeClassName()
    - 340
  - Element.replace() 337, 342
  - Element.setStyle() 341
  - Element.show() 336
  - Element.toggle() 336

Element.update() 337, 342, 436  
   and tags 338  
 email address validation 159  
 encodeURI() 19, 23  
 encodeURIComponent() 172  
 encoding 172  
 Enumerable 300, 303  
   and Hash 320  
   in the Prototype library 300  
 Enumerable mixin  
   in Ruby 273  
 Enumerable Object  
   and Ruby 303  
 Enumerable object 303  
   aliases for common  
     methods 306  
     and hashes 303  
     and ObjectRanges 303  
 Enumerable.all() 304  
 Enumerable.any() 304  
 Enumerable.collect() 311  
   iterator function for 311  
 Enumerable.detect() 306  
 Enumerable.each() 303  
   as foundation of Array  
   extensions 304  
 Enumerable.find(). *See* Enumerable.detect()  
 Enumerable.findAll() 306  
 Enumerable.grep() 308  
   optional iterator function 308  
 Enumerable.include() 305  
 Enumerable.inject() 313  
   and iterator functions 314  
   versus collect() 313  
 Enumerable.invoke() 312  
   passing extra arguments  
   to 313  
 Enumerable.max() 316  
 Enumerable.min() 316  
 Enumerable.partition() 307  
 Enumerable.pluck() 310, 322  
 Enumerable.reject() 306  
 Enumerable.select(). *See* Enumerable.findAll()  
 Enumerable.sortBy() 317  
   ranking function for 317  
 Enumerable.zip() 315  
   and multiple arguments 315  
 Enumeration 9  
 /etc/init.d/apache2 480

eval() 48  
 evalScripts  
   property of Ajax.Updater 78  
 event handlers  
   programmatically  
     assigning 16  
   using JavaScript 349  
 event model  
   classic versus W3C 349  
   and Prototype 349  
 Event observers  
   and window.onload 383  
 event.closures  
   and event model 16  
 Event.observe() 21, 349, 384  
 event-driven 286  
   programming 286, 291  
 event-handler functions  
   and Prototype 350  
 explode() 31, 34  
 extended controls 142  
 extending Object  
   dangers of 262  
 extending the native Array  
   class 261  
 extra network traffic 42

## F

Fade effect 114  
 Fedora 481  
 fgets() 380  
 Fiddler 462  
   differences from  
     LiveHTTPHeaders 462  
   and Internet Explorer 462  
   manual configuration of  
     client 463  
   and other web browsers 462  
 filesystem 29, 360  
   functions  
     PHP 30  
 Filters 167  
 findAll() 65, 306, 309, 321  
   and complex objects 309  
 Fink 481  
 Firebug 436  
 Firefox 451, 456  
   extensions 460  
 first() 300  
 firstChild  
   problems with whitespace 62  
 fixed-length arrays 293  
 flatten() 302  
 Flickr 408  
 flushing the browser's cache 88  
 Fold effect 117  
 folklore about Ajax network  
   performance 92  
 Form class 439  
 form element types  
   and \$( ) 352  
 Form object  
   in Prototype 351  
 Form.disable() 439  
 Form.Element.Observer  
   354, 421  
 Form.enable() 439  
 Form.focusFirstElement() 439  
 Form.getElements() 355  
 Form.Observer 354  
 Form.reset() 352  
 Form.serialize() method 420  
 form\_remote\_tag 415, 419, 421  
 formatted HTML 329  
 Foster, Mike 6  
 frames per second (fps)  
   property 389  
 frequency  
   of form element  
     observers 354  
   property of  
     Ajax.PeriodicalUpdater 84  
 frequency parameter  
   to observe\_field 420  
 full-page refresh 29, 46, 72  
 Function arguments 281  
   mutability of 281  
 function context 283  
   and the browser event  
   model 287  
   and browser events 350  
   object 282  
   objects to be called with  
     arbitrary contexts 17  
   reassigning 283  
   switching  
     inside objects 287  
 function declaration  
   inline 279  
   top-level 279

Function object  
 as an Object 280  
 inherently anonymous 280  
 methods of 280  
 properties of 280  
 referencing versus  
 invoking 280  
 function() keyword 279  
 Function, as first-class  
 object 278  
 Function.apply() 281  
 Function.bind() 289  
 and closures 289  
 example of 289  
 Function.bindAsEventListener()  
 290  
 and event handlers 290  
 as special case of bind() 290  
 Function.call() 281  
 functionality 88  
 functions  
 as arguments to other  
 functions 303  
 declaring 279  
 are first-class objects 303  
 invoking 279  
 and Prototype 288

**G**

---

Garrett, Jesse James 4  
 GD  
 graphics libraries  
 installing 479  
 graphics library for PHP 33  
 and XAMPP 482  
 generate the X-JSON header 81  
 generated JavaScript, on  
 server 79  
 generating arbitrary JavaScript  
 in RJS 441  
 generating HTML fragments 75  
 generating JavaScript  
 benefits 412  
 using Ruby on Rails 411  
 GET 447, 452  
 example message 448  
 GET and POST  
 and forms in Rails 419  
 getElements() 355

getElementsByTagName()  
 62, 299, 331  
 and \$A() 331  
 and white space 331  
 getStyle() 341  
 ghosting 206, 214  
 GIF animation 173  
 global layout in Rails apps 414  
 global variable 63  
 Gnome 479  
 good practices 40  
 Google Suggest 12  
 graceful degradation of Ajax  
 features in Rails 419  
 graphs  
 of HTTP traffic 459  
 to display HTTP traffic 88  
 graying out the UI 385  
 grep() 308  
 and iterator functions 308  
 Grow effect 116  
 GUI drag and drop 205

## H

---

halting effects 133  
 hasClassName() 341  
 Hash Object 319  
 and Enumerable 320  
 filtering with findAll() 321  
 and Hash object and POST  
 body 379  
 as mixin 319  
 Hash.keys() 322  
 Hash.toQueryString() 323, 379  
 Hash.values() 322  
 Hashes associative arrays 293  
 Hatcher, Erik 6  
 HEAD 447  
 header() function in PHP 493  
 headers array in Rails 494  
 helper objects for Ajax 415  
 helpers in Rails  
 and Prototype and  
 Scriptaculous 413  
 hidden iframes 47  
 hide() 336  
 hiding DOM elements  
 CSS versus DOM  
 manipulation 332

hierarchical classification 360  
 high-level API 80  
 to avoid tight coupling 80  
 Highlight effect 110, 150  
 hover modifier  
 in CSS 383  
 HTML 4, 141, 188, 206–207  
 block elements 188  
 button 145  
 controls 141  
 elements 326  
 form elements 162  
 Forms 10  
 forms 41, 351, 494  
 fragment 419, 434  
 link 145  
 list element 170  
 select element 164, 167  
 text entry control 145  
 HTML markup 326  
 adding behavior 378  
 HTTP 175, 490  
 description of 444  
 GET 160, 177, 447  
 HEAD 447  
 headers 19  
 measuring toolkit 459  
 message format 448  
 method 54, 76  
 POST 160, 177, 353, 447  
 protocol 17, 444  
 statelessness 446  
 status 66  
 status code 417  
 verb 50, 61  
 HTTP 1.1 specification  
 location of 445  
 HTTP traffic 74, 459  
 breakdown by mime type 90  
 graphs 88  
 results of measuring 88  
 HTTP traffic data  
 parsing and analyzing 463  
 HttpServlet 177  
 HttpServletRequest 491  
 hyperlinks 41  
 Hypertext Transfer Protocol.  
*See* HTTP

**I**


---

- id attribute
  - and database primary keys 328
  - of DOM elements 328
  - of HTML Form elements 353
- IFrames 6
- IIS and PHP 478
- image gallery 27
- Image object 392
- implementing slideshow
  - as an Effect object 386
- implode() 29
- In place editing 425
- in\_place\_editor\_field
  - helper 425
- inactivity 41
- include() 305
- inconsistent state of user
  - interface 77
- indentation, readability 54
- indexOf() 301
- information architecture
  - and Web 2.0 360
- information overload 167
- informed decision
  - in web architecture 85
- inheritance
  - in Ajax helper classes 76
  - single 269, 272
  - single and multiple 269
- initialize() method 263, 388
- inject() 313
  - and the Hash Object 321
- innerHTML 16, 48, 73, 342, 392
  - and content-centric Ajax 72
  - as simple persistence
    - format 375
    - property defined 334
- in-place collection editor 162
  - creation 162
  - in action 164
  - options 165
- in-place editing 143
- in-place text editor
  - activating 146
  - callbacks 152
  - control 142, 150
  - creation 143
  - CSS class names 151
  - dispose() 146
  - disposing 146
  - DOM IDs 151
  - enterEditMode() 146
  - example 144
  - failure 153
  - in action 144
  - loading 153
  - options 146
  - script control 146
  - usage examples 154
  - validations 155
- InPlaceEditor 373
  - configuring 373
- insert\_html 435
- insertBefore() 333, 342
- Insertion object 342
- Insertion.Bottom 343
  - and appendChild() 343
- Insertion.Top 343
  - and \$( ) 343
- installing Python 468
- instanceof operator 261
- interacting with the DOM 327
- interactive JavaScript
  - interpreter 252
- interactivity and UI design 366
- interface 4, 273
- Internet Explorer 123, 192, 233, 235, 456
  - version 5 5
  - version 7 19
- invoke() 312
- IronPython 492
- is\_dir() 30
- iterating over
  - Arrays 294
  - JavaScript Objects and extending objects 262
  - properties of a JavaScript object 261
- iterator function 303, 391
  - and each() 303
  - and Enumerable.inject() 314
  - external 303
  - and the Hash Object 321
  - internal 303
- iterators as anonymous
  - functions 304

**J**


---

- Java 393, 470, 490
  - classpath 157
  - Collection Classes 178
  - Collections and iterators 303
  - Collections API 493
  - regular expression 159
  - Servlets 141, 470
  - StringBuilder 162
- java.io.InputStream 495
- java.io.Reader 495
- JAVA\_HOME 471, 473
- JavaScript 207
  - dynamic loading
    - technique 102
  - in the web browser 326
  - wide but not deep knowledge
    - of 278
- JavaScript Array 9
  - as stacks 295
  - filtering 306
  - iterating over 294
  - length property 294
  - as linked list 293
  - object 293
  - and Prototype 300
  - and Strings 299
  - tidying, using Prototype 302
  - truncating 294
- JavaScript Callbacks
  - in Rails 417
- JavaScript event model 16
- JavaScript Event object 289
- JavaScript Events 10
- JavaScript Function object
  - 8, 278
- JavaScript Object
  - as associative array 293
  - extensibility 254
- JavaScript object declaration
  - simple 255
- JavaScript Object Notation 67, 257
- JavaScript Object type 278
- JavaScript Objects
  - as associative arrays 256, 319
- JavaScript reflection and the
  - Scratchpad application 261
- javascript\_include\_tag 413, 427
- JavaScriptGenerator 436

- JavaServer Pages Standard Tag Library 178
  - JavaServer Pages. *See* JSP
  - javax.servlet.http.HttpServlet-Request 491
  - javax.servlet.http.HttpServlet-Response 493
  - join() 299, 378
  - JPEG images 31
  - JScript.NET 492
  - JSON 49, 67, 81, 309
    - as alternative to XML 70, 258
    - core rules 257
    - expression evaluating 69, 82
    - for defining object prototypes 259
    - header 377
    - and Prototype 258
    - syntax 280
  - JSP 145, 154, 160, 178, 470, 493
    - Expression language 155
    - page directive 493
    - Specification 155
  - JSTL 178
- K**
- 
- KDE 479
  - key presses 290
  - keys() 322
  - key-value pairs 293, 322
  - Kock, Piers-Paul 6
- L**
- 
- Laboratory
    - Ajax Auto-completer 171
    - drag and drop 219
    - draggables 211, 214
    - HTTP 454
    - in-place text editor 148
    - local autocompleter 184–185
    - slider control 194
    - sortable 232, 236, 241
  - language
    - extensions 7
    - features 252
  - large data sets 167
  - last() 300
  - Lawrence, Eric 462
  - learning curve 252
  - legacy app and code reuse 86
  - length property 299
    - Array object 294
    - is read/write 294
  - lenient parsing of HTML
    - by web browsers 328
  - Ley, Jim 6
  - limitations of classic web application model 41
  - linear transition 389
  - link\_to and link\_to\_remote
    - url argument 414
  - link\_to method 413
  - link\_to\_remote 414–415, 418
  - link\_to\_remote helper 436
  - link\_to\_remote method 413
  - linked list 293, 295
  - lists, sorting 228
  - LiveHTTPHeaders 460
    - differences from Fiddler 462
  - LiveHttpHeaders 88
  - local autocompleter control
    - appearance 185
    - case control 186
    - creation 181
    - example 182
    - in action 183
    - matching rules 185
    - options 183
  - locally scoped variable 80
  - loose coupling 49
  - loosely coupling
    - notification system 370
  - loosely typed languages
    - and reflection 261
  - low-level
    - callbacks 66
    - capability
      - of Prototype 252
  - low-level UI manipulation 326
- M**
- 
- Mac OS X 205, 479
    - and Apache 481
    - and GD libraries 481
    - System Preferences 187
    - third-party package managers 481
    - and XAMPP 482
  - Mac OS X stop
    - switching off default Apache server 482
  - Macintosh 205
  - MacPorts 481
  - Mandriva 481
  - manually written
    - constructors 269
  - map(). *See* Enumerable.collect()
  - Maps 293
    - associative arrays 293
  - max() 316
  - measuring HTTP traffic
    - from the client 460
  - memo object 314
  - memo pattern XE 314
  - methods 8
  - methods of Objects
    - implementing in JavaScript 256
  - Microsoft Excel 468
  - MIME type 23, 376, 491
    - setting 58
  - min() 316
  - mixin 273
    - and the Element object 336
    - applying to base classes 276
  - mixin object and object types 274
  - mixture of content-centric and script-centric Ajax 80
  - Model 2 Pattern 175
  - modern OO languages
    - and multiple inheritance 269
  - monolithic 41
  - mouse
    - coordinates 290
    - movement 290
  - mouseover 381
  - MoveBy effect 113
  - Mozilla 460
    - DOM inspector 327
  - multiple effects 134
  - multiple inheritance 273
  - multiple listeners
    - on UI events 383
  - multiple update 77, 81
    - and breadcrumb trail 77
    - limits of Prototype's solutions 93
    - problem 48

multiple update (*continued*)  
 and Rails 434  
 and RJS 435, 437  
 MVC Pattern 175  
 MVC web application 154

## N

name and id attributes  
 of HTML Form elements 353  
 named arguments 51  
 alternative to, in JavaScript 52  
 named effect queues 136  
 NameValueCollection 491, 494  
 native object 19  
 natural effect pairs  
 toggling 117  
 navigate 28  
 navigation 360  
 nesting Objects within one  
 another 256  
 .NET 490–491  
 network  
 latency 172, 177, 181  
 load 459  
 load, measuring 459  
 unpredictable and  
 unreliable 77  
 network traffic, content- versus  
 data-centric Ajax 91  
 nextSibling 330  
 normal browsing mode  
 QuickGallery v2 364  
 notation, using JSON 256  
 notification system 370  
 Nuvola icon set 14

## O

Object 8  
 class, in JavaScript 252  
 object  
 composition 273  
 hierarchy 265  
 literal 51  
 [object Object] 310  
 Object with methods  
 simulating in JavaScript 280  
 Object.extend() 263, 265, 388  
 and Functions 267  
 and inheritance 271

as simple object merging  
 tool 265  
 for modeling object  
 hierarchies 269  
 object-oriented hierarchies 8  
 object-oriented JavaScript  
 techniques 6  
 object-oriented programming  
 and JavaScript 254  
 expectations of 278  
 ObjectRange 319, 323  
 Objects  
 extending the behavior 255  
 in the Scratchpad  
 application 253  
 observe() 349  
 observe\_field 415, 420  
 without invoking Ajax 420  
 observe\_form 415, 421  
 Observer pattern 349  
 observers and HTML forms 353  
 offset of DOM elements  
 and Prototype 345  
 offsetLeft  
 option to clone() 348  
*See also* Position.clone()  
 offsetTop  
 option to clone() 348  
*See also* Position.clone()  
 onchange event handler 420  
 onclick event handler 373  
 onComplete 55  
 and Ajax.Responders 82  
 onException 55  
 onFailure 55, 392  
 onInteractive 55  
 onkeypress event handler 416  
 onload 144, 157  
 onload event handler  
 on Image objects 392  
 onLoad 55  
 onLoading 54  
 onsubmit handler  
 of form\_remote\_tag 419  
 onSuccess 55, 392  
 opacity and CSS 385  
 Opacity Effect 109, 114, 216  
 OpenOffice Calc 468  
 OPTIONS 447  
 options object 52, 73, 388  
 preparing 388

options object in Effects classes  
 mapping to Ruby code 424  
 order of returning HTTP  
 responses 77  
 outerHTML 339  
 over-lapping effects 132

## P

page contract 178  
 page object in RJS 436  
 page.replace\_html 437  
 page.select() method 440  
 Parallel Effect 134  
 parent and a child class 271  
 parent node 329  
 parent property  
 of DOM elements 329  
 parent-child relationships 39  
 partial page template 412, 438  
 partition() 307  
 performance 85, 87  
 PeriodicalExecuter 422  
 periodically\_call\_remote  
 415, 422  
 Perl 393  
 phone number, formatting 160  
 photo thumbnails 223  
 PHP 28, 393, 490  
 and GD libraries 479  
 installing 478  
 and operating systems 479  
 variation between installed  
 sites 478  
 versions of 478  
 and web servers 478  
 pie charts 90  
 pluck() 310, 392  
 polling the server 83  
 pop() 295  
 popup 142  
 port  
 80 445, 474  
 8080 445, 472  
 Position object 344  
 position parameter  
 of page.insert\_html 438  
 of Rails Ajax helpers 417  
 Position.absolutize() 346  
 Position.clone() 347  
 Position.cumulativeOffset() 345

- Position.positionedOffset() 345
  - Position.realOffset() 345
  - Position.relativize() 346
  - Position.within() 346
  - positionedOffset() 345
  - positive difference of Ajax on
    - HTTP traffic 89
  - POST 447
    - body format 449
    - encoding 449
    - encoding example 449
  - POST body 420
  - POST request 376, 494
    - and HTML forms 494
    - and Java servlets 495
    - and querystrings 495
    - reading raw request body 495
  - POST variables 53
  - postBody 54
  - presentation, separating from
    - business logic 37
  - previousSibling 330
  - primary and secondary
    - resources
      - in HTTP traffic analysis 467
  - programmatically reference
    - to DOM elements 76
  - Prototype 4, 7, 143, 153, 248
    - \$A() 226
    - \$R() function 195
    - Ajax helper classes 46
    - Ajax.Request 153
    - and Arrays 293
    - and arrays of complex
      - elements 309
    - and closures 288
    - and content-centric Ajax 87
    - and DOM 326
    - and Functions 288
    - version 1.4 89
    - version 1.5 89
  - Prototype and
    - Scriptaculous 343, 356
    - and the \$( ) function 335
    - as standalone JavaScript
      - libraries 411
    - house style 311
    - in a production setting 27
    - in real-world setting 360
    - and Ruby on Rails 411
    - when to use 360
  - Prototype Array methods
    - and RJS 440
    - chaining into single line of
      - code 311
  - Prototype house style and DOM
    - element creation 372
  - Prototype library
    - and core JavaScript
      - objects 261
    - focus on ease of use 334
  - Prototype ObjectRange 195
  - Prototype.js version 1.5 74
  - Prototype's Array helpers 378
  - Prototype's Class object 263
  - Prototype's Event observers
    - and window.onload 383
  - prototype-based inheritance 7
  - prototypes 258
    - of core JavaScript objects 260
  - proxy server 459, 462
  - public and private members 270
  - Puff effect 116
  - Pulsate effect 117, 365, 370
  - push() 295
  - PUT 447
  - Python 393, 463
    - installing 468
  - Python's lambdas
    - and closures 284
- ## Q
- 
- QueryString 322–323
    - encoding using Hash
      - Object 322–323
  - querystring 19, 53
    - and HTML forms 352
  - queueing effects 135
    - example 137
  - quick wins with Prototype and
    - Scriptaculous 252
  - QuickGallery 27, 72, 360,
    - 435, 459, 467, 490
    - as an Ajax application 47
    - application 483
    - configuring 29, 486
    - content-centric 72, 74
    - content-centric
      - implementations 83
    - page layout 361
    - v2 393
    - version 2 360
- ## R
- 
- radio buttons 162
  - Rails 411
    - integrating Prototype and
      - Scriptaculous 411
  - Rails Ajax helpers, specifying
    - multiple target nodes 417
  - Rails helpers, generating server-
    - side code with 426
  - Rating object 14
  - Ratings example 13
  - raw POST body 380
  - raw POST data 377
  - raw request body 495
  - readyState 23, 66
  - readystate value 19
  - realOffset() 345
  - reassigning function
    - context 283
  - recording HTTP traffic 459
  - Recycle Bin 205
  - refactoring 20, 374
  - reflection 261
  - reformatting 160
  - register() method
    - of Ajax.Responders 82
  - regular expression 159, 308, 467
  - reject() 306–307
  - relative positioning
    - of DOM elements 345
  - relativize() 346
  - remote\_function 415, 423
  - remove() 319, 337
  - removeChild() 332
  - removeClassName() 340
  - render() method of Rails,
    - ApplicationController 412
  - rendering differences
    - across browsers 346
  - rendering engine 334
  - rendering partials in Rails 435
  - replace() 337
  - replace\_html 435

- replace\_html method 437
  - replaceChild() 333
  - report data 167
  - request 444
    - body and headers 460
    - CONNECT 447
    - DELETE 447
    - GET 447
    - HEAD 447
    - headers 449, 490
    - Host header 449
    - maintaining state 446
    - message format 448
    - methods 447
    - OPTIONS 447
    - POST 447
    - PUT 447
    - TRACE 447
    - types of 447
    - User-Agent header 449
    - variable 29
  - request.raw\_post 423, 496
  - requestHeaders 54
  - reset() and HTML forms 352
  - response 444
    - error code 158
    - header 158
    - message example 450
    - message format 450
    - status codes 450
  - responseText 69
  - responseXML 492
  - responsiveness of polling 84
  - reuse 86
    - and content-centric Ajax 86
    - of client-side code 80
    - of code 37
  - reverse() 298, 316
  - Rico 7
  - RJS 433
    - inlining in the controller 441
  - .rjs file 436
  - root folder. *See* document base
  - Ruby 7, 273, 393, 490
    - blocks 284
    - closures 284
  - Ruby Enumerable mixin 440
  - Ruby on Rails 411, 492
    - and content-centric Ajax 87
    - and data-centric Ajax 87
    - and script-centric Ajax 87
- S**
- 
- Safari 456
  - saving HTTP traffic data
    - using Fiddler 462
    - using LiveHTTPHeaders 460
  - Scale effect 111, 115
  - scaling algorithm 32
  - scheduling 387
  - scope 285
    - and closures 285
  - scoped variable 178
  - Scratchpad application 252, 294
    - and the DOM 326
    - highlight() method 328
    - how to use 252
  - <script> tags 78
    - and Ajax.Updater 78
    - and innerHTML 78
  - Scriptaculous 4, 97, 141, 248
    - and Rails 424
    - autocompleter controls 167
    - controls 141
    - download location 101
    - drag and drop 205
    - Draggable 208
    - Droppables 217
    - effects 97
    - files 101
    - in-place collection editor
      - control 162
    - library 100
    - loading 102
    - local autocompleter
      - control 181
    - Opacity Effect 216
    - setting up 100
    - slider control 187
    - Sortable 228
  - Scriptaculous Effect 24, 319, 386, 408
    - managing multiple
      - instances 370
  - Scriptaculous InPlaceEditor 373
  - script-centric 47–48
    - writing against an API 49
  - scrolling and DOM element
    - offset 345
  - searching Arrays 301
  - select element 162
  - sensible default 389
  - sensible default values
    - providing 51
  - separating business logic and presentation 37
  - Separation of Concerns 177
  - serial effects 135
  - serialize()
    - and HTML forms 352
    - and name attribute 353
  - serializing Objects as
    - QueryStrings 323
  - server load 77, 84
    - and polling 84
  - server response as a string 73
  - server-relative addressing 154
  - server-side
    - languages 490
    - latency 172
    - reformatting 160
    - validation 155, 162
    - well-factored 75
  - servlet 154, 175, 491
  - servlet path 157
  - session 446
    - storing calculation results 77
  - setHeader() 493
  - setHeight
    - option to clone() 348
    - See also* Position.clone()
  - setLeft
    - option to clone() 348
    - See also* Position.clone()
  - setStyle() 341
  - setTimeout() 83
  - setTop
    - option to clone() 348
    - See also* Position.clone()
  - setWidth
    - option to clone() 348
    - See also* Position.clone()
  - Shake effect 117
  - shift() 295
  - show() 336
  - show() function and Arrays 307
  - show().scratchpad application
    - show() method 253
  - Shrink effect 116
  - simplifying code base 77
  - simultaneous effects 134
  - sinusoidal transition 389

- Sites folder, XAMPP for
  - Mac OS X 482
- sizing edit box with Rails
  - `in_place_editor_field` 426
- `skip_style` option on Rails auto-complete helper 428
- Skype and port 80 485
- `slice()` 296
- SlideDown effect 115
- slider control 187
  - axis 194
  - callbacks 197
  - creation 188
  - enabling and disabling 196
  - example 188
  - from desktop
    - applications 187
  - handle 187
  - in action 190
  - in-depth example 197
  - operation 187
  - options 193
  - orientation 194
  - styling 200
  - track 187
  - using images with 191
  - value display 201
  - value range 195
- slideshow 360
  - editing mode 364
  - editor 361
  - editor impact on layout 362
  - retrieving from server 378
  - storing on the server 375
- SlideShow object 369, 402
- slideshow player 361, 384
  - implementing 386
- `slideshow.js` 365
- SlideUp effect 115
- Smalltalk's compiled blocks
  - and closures 284
- `sort()` 298, 316
- Sortable 228
  - axis 238
  - callbacks 243
  - constraint option
    - 232, 238–239
  - constraints 235
  - containment option
    - 231, 236–237, 243
  - `create()` 231, 238, 243
  - creation 228, 234
  - drag behavior 235
  - `dropOnEmpty` option
    - 231, 238
  - element type 232
  - enabling 232
  - example 228, 233
  - format option 232
  - ghosting option 231, 235
  - handle option 232
  - hoverclass option 231, 235
  - in action 236
  - moving between 236
  - multiple targets 236
  - nesting 241
  - `onChange` option 232, 243
  - only option 231, 235–236
  - `onUpdate` option 232, 243
  - options 231
  - orientation 238
  - overlap option 232, 240
    - and Rails 428
  - scroll option 232, 242
  - scrolling 241
  - `scrollSensitivity` option
    - 232, 242
  - `scrollSpeed` option 232, 242
  - `serialize()` 243–244, 246
  - serializing elements 243
  - tag option 231–232, 234
  - tree option 232, 241
  - `treeTag` option 232, 241
- Sortable object refreshing 370
- `Sortable.create()` 370–371, 432
- `Sortable.serialize()` 433
- `sortable_element`
  - helper 431
  - parameters 431
- sortables in Rails convention for
  - element IDs 431
- `sortBy()` 317
- sorting 228
- special folders 30
- `splice()` 296–297
- `split()` 299, 319
- spreadsheet 88
  - analysis of HTTP data 463
  - to analyze HTTP traffic 88
- square bracket notation
  - for JavaScript Objects 254
- Squish effect 117
- stack
  - definition of 295
  - JavaScript Arrays as 295
- `start()`, method of
  - Ajax.PeriodicalUpdater 74
- stateless protocol 445
- static functions of Element
  - object 336
- static methods 266
- Steinberg, Dan 6
- Stephenson, Sam 87, 433
- stock ticker, automatically
  - updating 73
- `stop()`, method of
  - Ajax.PeriodicalUpdater 74
- stop-start
  - in QuickGallery 42
  - interactions 77
  - pattern of interaction 5
  - pattern of user interaction 41
- storing and retrieving
  - information 360
- `String.split()` 299, 319
- strongly typed OO-languages 276
- structured data 56, 72
- style property of DOM
  - elements 341
- styles of Ajax, evaluating and
  - comparing 72
- style-sheet and Rails auto-complete helper 428
- subclassing `Effect.Base` 386
- subfolders 28, 57
- `sudo` command 480
  - and Mac OS X 482
- Sun 470
- super-user privileges
  - on Ubuntu Linux 480
- supporting libraries 87
- Suse 481
- switching function contexts 284
- SwitchOff effect 117
- symbolic links and Windows 485
- synaptic 479
- `System.Web.HttpRequest` 491, 495
  - and raw POST body 495
  - `ContentType` property 491
- `System.Web.HttpResponse` 494

**T**

- 
- target element 15
  - TCP/IP 445
  - template 75
    - file 57
  - test function 306
  - test script for HTTP traffic
    - measurement 88
  - text entry control 145
  - text input box 365
  - text node and the DOM 331
  - text/javascript mime type 492
  - text\_field\_with\_auto\_complete 427
  - this as a reserved keyword 282
  - thumbnail 27, 223
    - image 31
    - images 28, 75, 361
      - generating 374
    - images and GD library 374
  - tight coupling 49, 370
    - and script-centric Ajax 80
  - timer object 73
  - title attribute 382
  - toArray() 319
    - and the \$A() function 319
  - Toggle Pane Widget 119
  - toggle utility function 118
  - toggle() 336
  - tooggling effects 117
  - Tomcat 470
    - application server 141
    - downloading 470
    - location of 470
    - manager application 474
    - managing applications 474
    - roles 475
    - running 473
    - user setup 475
    - welcome page 473
  - toolbar
    - in QuickGallery v2 362
    - UI element 381
  - tooltip 381
    - and DHTML 382
    - and title attribute 382
  - toQueryString() 23, 323, 379
  - toString() method 8
  - TRACE 447
  - traffic, HTTP 88
  - transitions 125
    - built-in 126
    - charting 128
    - charts of 128
    - mapper tool 127
    - table of 126
    - writing custom 130
  - Transmission Control Protocol/  
Internet Protocol. *See* TCP/  
IP
  - trash can
    - implementing drag and  
drop 370
    - in QuickGallery v2 363
  - Triffids 185
  - tuning polling frequency 84
- 
- U**
  - Ubuntu Linux 479
    - and PHP 479
  - underscore as scope
    - delimiter 270
  - Universal Resource Locator.
    - See* URL
  - Unix command-line tool
    - grep 308
  - unshift() 295
  - update parameter
    - for Rails Ajax helpers 416
  - update() 337
    - and tags 338
  - update() method
    - for custom Effects 389
  - update\_attribute
    - and ActiveRecord 426
  - updateUI()Rating object
  - updateUI() 15
  - updating user interface
    - programmatically 327
  - URL 76, 445
    - encoding 172, 353, 449
    - query parameter format 172
  - url parameter for Rails Ajax
    - helpers 416
  - usability 5, 46
    - drag and drop 205, 228
    - guidelines 167
    - information overload 167
    - visual feedback 206, 219
  - use cases
    - of Ajax.PeriodicalUpdater 83
  - user agent 445
  - UTF-8 character set 493
- 
- V**
  - values() 322
  - VB.NET 492
  - VB.Net 393
  - view 178
    - in Rails app 412
  - Vignoni, David 14
  - virtual class 273
  - Virtual folders 360
  - visibility
    - tooggling 119
  - visual consistency
    - when using Effects 370
  - visual feedback 11, 23
  - visual jerkiness 42
  - visual layout of elements
    - and Prototype 344
  - visual\_effect helper 424
- 
- W**
  - W3C event model
    - and Prototype 349
  - W3C standard event model 349
  - Web 2.0 Logo Creator 381
  - web applications 141–142, 167
    - controller 175
    - limitations 141
    - Model 2 Pattern 174
    - proper separation of  
concerns 174
  - web browser
    - JavaScript interpreter 326
  - web developers 248
  - web server logs 459
  - web servers and PHP 478
  - web.xml 474
  - WEB-INF 175
  - weight of images
    - in HTTP traffic profile 89

- weight of JavaScript
  - in an Ajax app 89
- whitespace
  - and DOM parsing 329, 380
  - and DOM solution 330
  - in DOM and Prototype 342
- widget creation 121
- window.onload 401
  - and Event Observers 383
  - attaching multiple callbacks 383
- window.onload event 383
- Windows 484
- Windows XP 479
  - Sound Panel 187
- within() 346

- without() 301
- workflow 59

## X

---

- XAMPP 482
  - coexisting with Skype 485
  - control panel 484
  - and GD 482
  - for Mac OS X 482
  - port scanning 484
  - troubleshooting 484
  - for Windows 484
- XHR object 278
  - and closures 278
- X-JSON header 81, 83, 370–371, 374, 409, 435

- XML 49, 56
  - bloat, and whether it matters 92
  - format 375
- XMLHttpRequest 5, 47, 153, 158
  - readystates, in Rails 417
  - wrapper object 72
- XMLHttpRequest object
  - and request headers 490
- X-SLIDENAME custom header 376
- XSLT style sheet 453

## Z

---

- zip() 315