

# Index

## SYMBOLS

**& (ampersand), 63**  
**\$args array, 119**  
**@ (array character), 203**  
**\* (asterisk), 18, 302**  
**\ (back slash), 18**  
**^ (caret character), 17–18**  
**\$ (dollar sign), 17–18**  
**"" (double quotes), 34**  
**/ (forward slash), 17**  
**@include directive, 244**  
**. (period), 18**  
**+ (plus sign), 18**  
**? (question mark), 18, 63**  
**[] (square brackets), 18**

## A

**accessing third party, 248**  
**account, creating**  
     NIPSA, 35  
     NSID, 23  
     types of accounts, 24  
     URL, setting, 24  
**ActivePerl (ActiveState), 57**  
**Add a tag link, 33**  
**Add to Set button, 30**  
**Add/Edit Application Extension Mapping**  
     **dialog box, 51**

## adding

authentication to photo gallery, 161–167  
 contacts, 35, 36, 344–345  
 controls to map page, 264–266  
 data  
     to database, 301–302  
     to map, 270–271  
 geodata to photos, 266–267  
 headers and navigation to web pages, 136–139  
 maps to photo gallery, 271–274  
 More info link to photo gallery, 131–136  
 navigation to photo gallery, 324–328  
 tags to photos, 339–340

## Additional Information section of photo page, 248

**addLink function, 253**

## aggregator, 87

## AJAX (Asynchronous Javascript and XML)

description of, 12  
 prototype.js library, 131–136

## All Rights Reserved license, 39

## All Sizes button, 27

## Allowed Sites-Software Installation dialog box, 180

## allowing user to add tag to photo, 34

## ampersand (&), 63

## Apache Service Monitor, 54

## Apache Web Server

configuring, 55  
 description of, 14  
 document root, finding, 54–55

## Apache Web Server (continued)

- installing, 53–54
- PHP installing, 55
- testing setup, 55–56

## API (application programming interface), 61

### API call

- minimizing, 83
- REST, 62–65
- SOAP, 66–68
- XML-RPC, 68–71

### API endpoint

- description of, 61
- REST, 62
- SOAP, 66
- XML-RPC, 68

## API Explorer, 79–82

### API key

- configuring, 148–150
- getting, 43–45
- Google Maps, getting, 260–261
- Technorati
  - getting, 242
  - protecting, 254
- terms of use attached to, 290

## API Key Authentication Setup page, 149, 150

### API kit

- authentication and, 153
- description of, 82, 116

### API methods

- authentication, 379
- blogs, 379
- contacts, 380
- favorites, 380
- `flickr.auth.checkToken`, 158
- `flickr.auth.getFrob`, 167
- `flickr.auth.getToken`, 153
- `flickr.people.getInfo`, 74–75
- `flickr.people.getPublicGroups`, 292–293
- `flickr.photos.getInfo`, 72–73
- `flickr.photos.getWithGeoData`, 267, 269
- `flickr.photos.search`, 227, 267–268
- `flickr.photos.setMeta`, 161, 166
- `flickr.photos.upload.checkTickets`, 176
- group pools, 380
- groups, 380, 395–397
- interestingness, 381
- overview of, 71

- people, 381, 391–394
- photo comments, 382, 397–398
- photo licenses, 382, 397
- photo notes, 382
- photo transforms, 383
- photo upload, 383
- photos, 381–382, 385–391
- photoset comments, 383
- photosets, 383, 394–395
- reflection, 384
- responses and, 71–75
- tags, 384
- test, 384
- URLs, 384

`api_key` parameter, 61–62, 64, 151

`apikey.php` file, 124–125, 149

`api_sig` parameter, 151, 153

## Application Configuration dialog box, 50

## application programming interface (API), 61

## applying for API key, 43–45

### architecture

- client-side, 5–6
- server-side, 6–7

`$args` array, 119

`array (@)` character, 203

array key, 119

asterisk (\*), 18, 302

## Asynchronous Javascript and XML (AJAX)

- description of, 12
- `prototype.js` library, 131–136

## asynchronous uploading, 176

## Atom format, 89, 92–93

## Attribution license type, 40

`authenticate` method, 157, 160

authenticated call, 153

authenticating self, 159–160

### authentication

- adding to photo gallery, 161–167
- API kits and, 153
- configuring API key, 148–150
- desktop application, 167
- methods for, 379
- mobile application, 167–168
- need for, 147–148
- typical sequence for, 151–153

authenticator, creating, 154–159

`auth_token` parameter, 153

**B****back slash (\), 18****badge**

- client-side, using JavaScript, 110–112
- description of, 99
- designing, 99–100
- server-side, using PHP, 100–109

**batch organize mode (Organizr), 31****BBC News RSS feed, 219****Blog This button, 241****Bloglines, 87****blogosphere**

- description of, 241
- querying to find where photos are blogged and embedding search results in photo page, 248–254
- Technorati and, 241

**blogs, methods for, 379****boundary marker, 170****bounding box, 268****browser, 16****buddy icon, 35****buddy icon URL, building, 78****building URL**

- buddy icon, 78
- group icon, 79
- image, 77–78
- photo page, 76–77
- photoset, 78
- photostream, 75
- profile, 75

**C****cache**

- of RSS feeds, setting up, 221
- using with gallery application, 320–323

**caching data**

- case study, 291–296
- deciding what to cache, 290
- deciding where to cache, 291
- description of, 289
- to minimize API calls, 83
- purpose of, 289–290

**callback URL, 150–151****caret character (^), 17–18****cascading style sheets (CSS), 8–9****CDF (Channel Definition Format), 98–99****changing order of photos in photostream, 38****channel and RSS format, 92****Channel Definition Format (CDF), 98–99****character encoding, 62****checkAuthenticatedUser method, 166****citizenship**

- commercial use and, 82
- minimizing API calls, 83

**client-side badge, creating with JavaScript, 110–112****client-side mashup, 5–6****closure, 278****collaboration and Web 2.0, 3–4****column, 300****Comma-Separated Values (CSV) feed, 96****comment**

- highlighting, 190–194
- methods for, 382

**comments author tag, 398****comments authorname tag, 398****comments id tag, 398****comments permalink tag, 398****commercial API key, 44****commercial use, 82****Comprehensive Perl Archive Network (CPAN), 17, 60****configuring**

- Apache Web Server, 55
- API key, 148–150
- IIS, 49–51

**contact**

- adding, 35, 36, 344–345
- methods for, 380

**contact data structure, 391****Contacts menu, 35****content**

- ease of getting, 5
- separation of style from, 8–9

**Content-Disposition header, 171****Content-Type header, 15****Control-click (select noncontiguous items), 32****controls, adding to map page, 264–266****Cookie header, 15****copying API key, 44**

## copyright issues

- All Rights Reserved license, 39
- Creative Commons license, 40
- CPAN (Comprehensive Perl Archive Network), 17, 60**
- `CREATE TABLE` **SQL command, 300–301, 317–318**
- `createMarker` **function, 276, 278**
- Creative Commons license, 40**
- CSS (cascading style sheets), 8–9**
- CSV (Comma-Separated Values) feed, 96**
- Curl, 22**

## D

### data

- adding to database, 301–302
- sharing and enhancement of, 3–4

### database

- adding data to, 301–302
- creating, 300
- deleting data from, 306
- modifying data in, 305–306
- relational, benefits of, 291
- searching data in, 303–304
- sorting data in, 305
- Utata web site, 295–296
- viewing data, 302–303

### database management system, 21

### Date Posted tab, 28–39

### Default Web Site Properties dialog box, 47, 49, 51 degree, 255

### DELETE SQL command, 306

### deleting data from database, 306

### desktop application authentication, 167

### detailed format, 72

### determining user's NSID, 183–185

### development environment, setting up, 45–46

### directory, setting up, 56–57

### displaying

- client-side badge, 112
- information on map, 279–282
- NSID, 185–188
- photo gallery, 122–125
- photograph, linking back to Flickr when, 40
- server-side badge, 104–109
- Smart Set, 144–145

### Dive Into Greasemonkey web site, 199

### Document Object Model (DOM), 11–12, 185

### document root, finding

- Apache Web server, 54–55
- Microsoft IIS, 46–47

### `document.evaluate` **method, 189–190**

### Documents tab (Default Web Site Properties dialog box), 51

### dollar sign (\$), 17–18

### DOM (Document Object Model), 11–12, 185

### `doMap()` **function, 263**

### DOMParser **object (Firefox), 245**

### double quotes (“”), 34

### dragging map, 256

### DROP TABLE **SQL command, 301**

### dynamic HTML, 11

## E

### ECMAScript, 11

### Edit Date screen, 28–39

### Edit link, creating, 163–165

### Edit System Variable dialog box, 48

### editing upload date, 38

### encapsulation boundary, 170

### Environment Variables dialog box, 48

### error response

- REST, 65–66
- SOAP, 68
- XML-RPC, 71

### event handler, 198

### Everyone's Photos feed, 89

### `execute_method` **method, 204**

### Explore section, 38

### Extensible Hypertext Markup Language (XHTML), 8

### Extensible Markup Language (XML), 9–10

### `extractKeywords` **method, 237**

## F

### family, marking photo as visible to, 36–37

### favorites, methods for, 380

### feed

- common, available, 89
- description of, 87
- formats
  - Atom, 89, 92–93
  - CDF, 98–99

- CSV, 96
- JSON, 95–96
- overview of, 89–91
- PHP, 94
- RSS, 91–92, 219–221
- serialized PHP, 94–95
- SQL, 97–98
- YAML, 96–97
- getting, 88–89
- illustrated news, creating
  - code, 227–235
  - code explanation, 236–239
- retrieving with Magpie, 222–225
- FeedDemon, 87**
- Fielding, Roy (author of HTTP specification), 19**
- Filter Properties dialog box, 50**
- `findImages` **method, 238**
- finding**
  - elements in web page, 184–185
  - images, 225–227
  - photos
    - public keyword and, 35
    - tags and, 33, 34
  - tags in web page, 188–190
- `findRecentPhotos()` **method, 124, 323**
- Firefox web browser. See also Greasemonkey**
  - description of, 16
  - `document.evaluate` method, 189–190
  - DOMParser object, 245
  - opening script in, 181
- Flickr API Group, 84**
- Flickr Developer Support Group, 84**
- Flickr Organizr, 31–33**
- Flickr Paginator, enhancing, 194–198**
- Flickr Settings window (Uploadr), 26–27**
- Flickr Uploadr, 25, 26–27**
- Flickr web site**
  - API Explorer page, 268
  - API keys page, 43
  - API pages, 72
  - authentication page, 151, 167
  - content made available by, 5
  - Explore section, 38
  - feed page, 90
  - Help, 82
  - home page, 23
  - official badge page, 99
  - photostream page, 339
  - Replace API, 177
  - search box drop-down list, 340–341
  - search results page, 341–344
  - Services Page, 83
  - sets page, 29–30
  - tools page, 25
  - Upload API, 169
  - Upload page, 24–25
  - Welcome screen, 338
  - Your API Keys page, 148–149
- `Flickr::API` **object, 204**
- `flickr.auth.checkToken` **method, 158**
- `FlickrAuthenticator` **class, 154, 157, 161**
- `flickr.auth.getFrob` **method, 167**
- `flickr.auth.getToken` **method, 153**
- FlickrCentral, 37, 38, 345–346**
- `FlickrGallery` **class, 124, 161, 318**
- `flickr.people.getInfo` **method, 74–75**
- `flickr.people.getPublicGroups` **method, 292–293**
- `flickr.photo.search` **method, 65**
- `flickr.photos.getInfo` **method, 72–73**
- `flickr.photos.getWithGeoData` **method, 267, 269**
- `flickr.photos.search` **method, 227, 267–268**
- `flickr.photos.setMeta` **method, 161, 166**
- `flickr.photos.upload.checkTickets` **method, 176**
- `FlickrResponse` **tag, 68**
- `flushDBPhotos()` **method, 318–319**
- folksonomy, 34**
- format**
  - Atom, 89, 92–93
  - CDF, 98–99
  - CSV, 96
  - for feeds, 89–91
  - JSON, 95–96
  - for photograph, 27
  - PHP, 94
  - request
    - REST, 62–65
    - SOAP, 66–68
    - XML-RPC, 68–71
  - RSS, 91–92, 219–221

### format (continued)

- serialized PHP, 94–95
- SQL, 97–98
- YAML, 96–97

### forward slash (/), 17

### free account, limitations on, 24, 25

### friend, marking photo as visible to, 36–37

### frob, 153

## G

### gallery web site

- AJAX, using, 131–136
  - authentication, adding to, 161–167
  - cache, using, 320–323
  - displaying photos, 122–125
  - headers and navigation, adding, 136–139
  - map
    - adding to, 271–274
    - displaying information on, 279–282
    - populating, 274–278
    - sidebar, creating, 282–287
  - navigation, adding, 324–328
  - overview of, 115–116
  - photo page, creating, 127–130
  - photo table, creating, 311–320
  - setting up, 116–122
  - Smart Set
    - defining, 139–144
    - displaying, 144–145
    - identifying, 328–329
    - storing, 329–331
    - viewing, 331–335
  - styling results, 125–127
  - upload page, creating, 171–176
- GDownloadUrl function, 274–278**
- ### geodata
- adding to photos, 266–267
  - description of, 255–256
  - processing, 269–271
  - retrieving from photos, 267–269
- ### Geotagging interface, 32
- geoview.php **script, 281–282**
- ### GET request, 14
- getConnection() **method, 318**
- getItems **method, 238**
- getKeywords **method, 236**

- getNextPhoto() **method, 324, 327–328**
- getPhoto() **method, 324, 327**
- getPhotoInfo() **method, 130**
- getPreviousPhoto() **method, 324, 327–328**
- getSmartSet() **method, 142**
- getWithGeoData() **method, 322, 323**
- GLatLng **class, 263**
- global\_nsid **variable, 191**
- GMap2 **class, 263**
- GMarker **class, 275**
- GM\_xmlHttpRequest **function**
  - code, 243–244
  - code explanation, 244–245
  - parsing response to, 245–247

### Google Maps

- API key, getting, 260–261
- map page
  - controls, adding, 264–266
  - creating, 261–264
- working with, 256–260

### Greasemonkey

- description of, 179
- determining user's NSID, 183–185
- displaying NSID, 185–188
- Dive Into Greasemonkey web site, 199
- finding tags using XPath, 188–190
- highlighting comment, 190–194
- installing, 179–181
- paginator, enhancing, 194–198
- querying Technorati with, 248–254
- script, creating, 181–183
- Technorati and, 243–247

### Greenwich meridian, 255

### group

- description of, 37–38
- methods for, 380

### group badge, creating with ImageMagick, 205–213

### group icon URL, building, 79

### group pool

- description of, 38
- methods for, 380

### groups admin tag, 396

### groups description tag, 397

### groups eighteenplus tag, 396

### groups iconserver tag, 396

### groups members tag, 397

**groups** name tag, 396  
**groups** nsid tag, 396  
**groups** photos tag, 396  
**groups** privacy tag, 396  
**groups** throttle tag, 397  
 GXml class, 274–278

## H

**header, adding to photo gallery, 136–139**  
**Help options (Flickr web site), 82**  
**highlighting comment, 190–194**  
**Home Directory tab (Default Web Site Properties dialog box), 47**  
 Host header, 15  
**HTML (Hypertext Markup Language), 7**  
**HTTP (Hypertext Transfer Protocol), 14**

## I

**identifying Smart Set, 328–329**  
**IIS. See Internet Information Services (IIS, Microsoft)**  
**IIS Management Console, 46**  
**illustrated news feed, creating**  
   code, 227–235  
   code explanation, 236–239  
**image**  
   building, 215  
   resizing, 25, 366  
**image sizes, 77**  
**image URL, building, 77–78**  
**ImageMagick**  
   better badge, creating, 213–215  
   description of, 201–202  
   group badge, creating, 205–213  
   pseudo-image, 205  
   using Perl to interface with, 202–205  
 Image::Magick object, 204–205, 214  
 @include directive, 244  
 include statement, 138  
**Info Window, populating, 278–282**  
 INSERT SQL command, 301–302, 319  
**installing**  
   Apache Web Server, 53–54  
   Greasemonkey, 179–181  
   ImageMagick, 201  
   Magpie, 221  
   Microsoft IIS, 46

MySQL  
   on Microsoft Windows, 296–298  
   on Unix, 298  
 Perl  
   under Microsoft Windows, 57–59  
   on Unix systems, 59–60  
 PerlMagick, 201  
 PHP  
   for Apache Web Server, 55  
   for Microsoft IIS, 47–49  
   phpFlickr library, 116  
**interestingness, 38–39, 381**  
**Internet Explorer (Microsoft), 16**  
**Internet Information Services (IIS, Microsoft)**  
   configuring, 49–51  
   description of, 14  
   document root, finding, 46–47  
   installing, 46  
   PHP, installing, 47–49  
   testing setup, 52–53

**invitation-only group, 38**  
**ISAPI Filters tab (Default Web Site Properties dialog box), 49**  
 isGoodKeyword method, 237  
 isUtatan() method, 293–295

## J

**Jamieson, Catherine (writer and photographer), 291**  
**JavaScript**  
   client-side badge, creating with, 110–112  
   description of, 11–12  
**JavaScript function closure, 278**  
**join, 331**  
**join condition, 332**  
**joining FlickrCentral, 345–346**  
**JPEG format, 27**  
**JSON (JavaScript Object Notation)**  
   description of, 12–14  
   feed format, 95–96, 110

## K

keypress event handler, 198  
**keyword**  
   public, 35  
   retrieving, 236  
   searching with, 225–227

## L

**latitude**, 255, 260

**license, methods for**, 382

**licenses** `id` tag, 397

**licenses** `name` tag, 397

**licenses** `url` tag, 397

**linking to Flickr**, 40

**Linux operating system**

ImageMagick, PerlMagick, and, 202

installing MySQL on, 298

setting up web server, 56

**loc.alize.us**, 267

**longitude**, 255, 260

## M

**Mac OS X**

ImageMagick and, 202

setting up web server, 56

**Magpie**

illustrated news feed, creating using, 227–239

installing and setting up, 221

retrieving feed with, 222–225

**managing set**, 32

**map page**

adding data to map, 270–271

controls, adding, 264–266

creating, 261–264

displaying information on map, 279–282

populating map, 274–278

sidebar, creating, 282–287

**Map tab (Organizr)**, 32–33

**Mappings tab (Application Configuration dialog box)**, 50

**marking photo as public or private**, 36–37

**mashable system**

east of getting content and, 5

permissions, restrictions, and, 4–5

**mashup architecture**

client-side, 5–6

server-side, 6–7

**mashup, definition of**, 4

**MD5 hash**, 329

**membership, tracking**, 292–293

method **parameter**, 62, 64, 153

methodName **tag**, 70

**methods**

API Explorer and, 79–82

authenticate, 157, 160

authentication, 379

blogs, 379

checkAuthenticatedUser, 166

contacts, 380

document.evaluate, 189–190

execute\_method, 204

extractKeywords, 237

favorites, 380

findImages, 238

findRecentPhotos(), 124, 323

flickr.auth.checkToken, 158

flickr.auth.getFrob, 167

flickr.auth.getToken, 153

flickr.people.getInfo, 74–75

flickr.people.getPublicGroups, 292–293

flickr.photos.getInfo, 72–73

flickr.photos.getWithGeoData, 267, 269

flickr.photos.search, 227, 267–268

flickr.photos.setMeta, 161, 166

flickr.photos.upload.checkTickets, 176

flushDBPhotos(), 318–319

getConnection(), 318

getItems, 238

getKeywords, 236

getNextPhoto(), 324, 327–328

getPhoto(), 324, 327

getPhotoInfo(), 130

getPreviousPhoto(), 324, 327–328

getSmartSet(), 142

getWithGeoData(), 322, 323

group pools, 380

groups, 380, 395–397

interestingness, 381

isGoodKeyword, 237

isUtatan(), 293–295

overview of, 71

people, 381, 391–394

photo comments, 382, 397–398

photo licenses, 382, 397

photo notes, 382

photo transforms, 383

photo upload, 383

photos, 381–382, 385–391

photoset comments, 383

photosets, 383, 394–395  
 phpFlickr library  
     async\_upload method, 176  
     Flickr API methods and, 119  
     replace method, 177  
 populateDBPhotos(), 319  
 print\_r, 119, 122  
 reflection, 384  
 responses and, 71–75  
 setToken, 258  
 showSmartSetThumbnail(), 142  
 splitIntoWords, 236–237  
 tags, 384  
 test, 384  
 updateCredentials, 159, 160  
 uploadPhoto(), 175  
 URLs, 384  
**Microsoft**  
   Internet Explorer, 16  
   Internet Information Services (IIS)  
     configuring, 49–51  
     description of, 14  
     document root, finding, 46–47  
     installing, 46  
     PHP, installing, 47–49  
     testing setup, 52–53  
   Windows  
     ImageMagick, installing, 201–202  
     MySQL, installing, 296–298  
     Perl, installing, 57–59  
     PHP MySQL extension, 306  
**minimizing API calls, 83**  
**mini-token, 168**  
**mobile application authentication, 167–168**  
**modifying data in database, 305–306**  
**More info link, adding to photo gallery, 131–136**  
**Moveable Type system (Six Apart), 292**  
**moving in Google Maps, 256–257**  
**Mozilla Developer Center, 188**  
**MySQL**  
   checking installation, 299  
   database  
     adding data to, 301–302  
     creating, 300  
     deleting data from, 306  
     modifying data in, 305–306  
     searching data in, 303–304

    sorting data in, 305  
     viewing data, 302–303  
   description of, 21  
   installing  
     on Microsoft Windows, 296–298  
     on Unix, 298  
   PHP and, 306–310  
   table, creating, 300–301  
   working with, 298  
**MySQL Server Instance Configuration Wizard, 297, 298**  
 mysqladmin tool, 299

## N

**naming conventions, 123, 181, 318**  
**navigation, adding to photo gallery, 136–139, 324–328**  
**Network Services ID (NSID), 23**  
**news feed, illustrated, creating**  
   code, 227–235  
   code explanation, 236–239  
 NewsItem class, 236  
 NewsVisualizer class, 237  
**Next link, adding to photo page, 324–328**  
**NIPSA (Not in Public Site Areas), 35**  
**NoDerivs license type, 40**  
**noncommercial API key, 44**  
**NonCommercial license type, 40**  
**Not in Public Site Areas (NIPSA), 35**  
**NSID (Network Services ID)**  
   description of, 23  
   determining, 183–185  
   displaying, 185–188  
   @ (Perl array character) and, 203

## O

**opening script in Firefox, 181**  
**options dialog, 25, 26–27**  
**ORDER BY clause (SQL), 305**  
**Organize feature, 266**  
**organizing photographs**  
   Organizr, 31–33  
   photostream, 28–29  
   sets, 29–30  
   tags, 33–34  
**Organizr, 31–33**

### P

**paginator, enhancing, 194–198**

**people contact tag, 393**

**people family tag, 393**

**people friend tag, 393**

**people gender tag, 392**

**people iconserver tag, 392**

**people ignored tag, 392**

**people isadmin tag, 392**

**people ispro tag, 392**

**people location tag, 394**

**people mbx\_shalsum tag, 394**

**people, methods for, 381**

**people nsid tag, 392**

**people photos tag, 394**

**people photosurl tag, 394**

**people profileurl tag, 394**

**people realname tag, 393**

**people revcontact tag, 393**

**people revfamily tag, 393**

**people revfriend tag, 393**

**people username tag, 393**

**period (.), 18**

**Perl**

description of, 16, 17

installing

under Microsoft Windows, 57–59

on Unix systems, 59–60

mantra, 365

**Perl Package Manager (ActivePerl), 58, 59**

**PerlMagick library, 201**

**Perl-style regular expression, 17–19**

**perms parameter, 151**

**per\_page parameter, 64**

**person data structure, 391–392**

**photo comment. See comment**

**photo comments tag, 389**

**photo dates tag, 388–389**

**photo description tag, 387**

**photo details, retrieving, 72–73**

**photo editability tag, 389**

**photo extras tag, 390–391**

**photo gallery application**

AJAX, using, 131–136

authentication, adding to, 161–167

cache, using, 320–323

displaying photos, 122–125

headers and navigation, adding, 136–139

map

adding to, 271–274

displaying information on, 279–282

populating, 274–278

sidebar, creating, 282–287

navigation, adding, 324–328

overview of, 115–116

photo page, creating, 127–130

photo table, creating, 311–320

setting up, 116–122

Smart Set

defining, 139–144

displaying, 144–145

identifying, 328–329

storing, 329–331

viewing, 331–335

styling results, 125–127

upload page, creating, 171–176

**photo ID, 27**

**photo id tag, 386**

**photo isfamily tag, 387**

**photo isfavorite tag, 388**

**photo isfriend tag, 388**

**photo ispublic tag, 387**

**photo license tag, 388**

**photo note, methods for, 382**

**photo notes tag, 390**

**photo originalformat tag, 388**

**photo owner tag, 387**

**photo page**

Additional Information section of, 248

creating, 127–130

querying to find where photos are blogged and

embedding search results in, 248–254

View blogs link, 252

**photo page URL, building, 76–77**

**photo privacy settings, modifying, 34**

**photo rotation tag, 388**

**photo secret tag, 387**

**photo server tag, 387**

**photo tags tag, 390**

**photo title tag, 387**

**photo transform, methods for, 383**

**photo upload, methods for, 383**

**photo urls tag, 390**

**photographs**

- geodata, adding to, 266–267
- methods for, 381–382
- organizing
  - Organizr, 31–33
  - photostream, 28–29
  - sets, 29–30
  - tags, 33–34
- retrieving geodata from, 267–269
- tags, adding to, 339–340
- uploading
  - asynchronous, 176
  - formats, 27
  - overview of, 24–26
  - photo IDs and URLs, 27
  - replacing when, 176
  - with Upload API, 169–171

**Photos from Your Contacts page, 35**

photos **tag, 64–65**

**photoset**

- description of, 29–30
- methods for, 383

**photoset comment, methods for, 383****photoset description tag, 395****photoset id tag, 395****photoset photos tag, 395****photoset primary tag, 395****photoset secret tag, 395****photoset server tag, 395****photoset title tag, 395****photoset URL, building, 78****photostream, 28–29****photostream page, 339****photostream URL, building, 75****PHP**

- compatibility with Apache Web Server, 53
- description of, 16–17
- as feed format, 94
- installing
  - Apache Web Server, 55
  - Microsoft IIS, 47–49
  - MySQL and, 306–310
- server-side badge, creating with, 100–109
- usort function, 371

**phpFlickr library**

- async\_upload method, 176
- authentication and, 153

- description of, 82
- installing and using, 116–122
- methods, 119
- replace method, 177

**ping, 241****plus sign (+), 18**

populateDBPhotos() **method, 319**

**populating map, 274–278****POST request, 15, 170****Previous link, adding to photo page, 324–328****primary key, 318****prime meridian, 255**

print\_r **method, 119, 122**

**Privacy page, 36–37****private group, 38****private, marking photo as, 36–37, 147****pro account, limitations on, 24, 25****processing geodata, 269–271****profile URL, building, 75****Prototype JavaScript framework, 12, 131–136****pseudo-image, 205****public group, 38****public keyword, 35****Q****query**

- to find where photos are blogged and embed search results in photo page, 248–254

- Technorati, 242

- XPath, 10

**query string, REST, 63–64****question mark (?), 18, 63****R****Really Simple Syndication or Rich Site Summary (RSS) format, 91–92, 219–221****reflection, methods for, 384****Refresh Photo Table link, 316****regular expression (regexp), 17–19****relational database, benefits of, 291****relevance, sorting images by, 226–227****remix, 4****remote procedure call, 20–21****reordering photostream, 28–29****Replace API, 176**

# Representational State Transfer (REST)

---

## **Representational State Transfer (REST), 19–20, 62–65**

### **request format**

REST, 62–65  
SOAP, 66–68  
XML-RPC, 68–71

### **resizing uploaded image, 25, 366**

### **resources, 83–84**

### **response data structures**

groups, 395–397  
people, 391–394  
photo comments, 397–398  
photo licenses, 397  
photos, 385–391  
photosets, 394–395

### **response object, 245**

## **REST (Representational State Transfer), 19–20, 62–65**

### **result set, 310**

### **retrieving**

API key, 44  
feed with Magpie, 222–225  
geodata from photos, 267–269  
images, and sorting by relevance, 226–227  
photo details, 72–73  
user details, 74–75

### **root account, 298**

### **rsp tag, 64**

## **RSS (Really Simple Syndication or Rich Site Summary) format, 91–92, 219–221**

## **S**

### **Safari web browser, 16**

### **satellite imagery, viewing, 259–260**

### **saving API key, 44**

### **screen name, 23**

### **search box drop-down list, 340–341**

### **search results page, 341–344**

### **searching**

data in database, 303–304  
Google Maps, 258–259  
for image, 225–227  
public keyword and, 35  
tags and, 33, 34  
Technorati cosmos, 242

### **SELECT SQL statement**

ORDER BY clause, 305  
use of, 302–303  
WHERE clause, 303–304

### **selecting photo in Organizr, 32**

### **serialized PHP feed format, 94–95**

### **server-side badge**

creating with PHP, 100–103  
displaying, 104–109

### **server-side mashup, 6–7**

### **set, 29–30**

### **set ID, 30**

### **setToken method, 258**

### **ShareAlike license type, 40**

### **Shift-click (select range), 32**

### **showBlogs function, 253**

### **showSmartSetThumbnail() method, 142**

### **sidebar, creating, 282–287**

### **signature for method call, 151**

### **Six Apart Moveable Type system, 292**

### **sizes, image, 77**

### **slider control (Google Maps), 258**

### **Smart Set**

defining, 140–144  
description of, 139  
displaying, 144–145  
identifying, 328–329  
storing, 329–331  
viewing, 331–335

### **SOAP**

API calls, 66–68  
description of, 20

### **Software Installation dialog box, 180**

### **sorting**

data in database, 305  
images by relevance, 226–227

### **splitIntoWords method, 236–237**

### **sprintf() function (PHP), 319**

### **SQL (Structured Query Language)**

description of, 21  
feed format, 97–98

### **square brackets ([ ]), 18**

### **storing Smart Set, 329–331**

### **style of web site, 8–9**

### **styleElement function, 253**

**styling photo gallery, 125–127**

**summary format, 72**

**syndication, 87**

## T

**table, creating, 300–301**

**tag**

adding to photo, 339–340

finding, in web page, 188–190

methods for, 384

overview of, 33–34

**tag search, 225–227**

**tag spammer, 292**

**tag URI scheme, 92**

**tagging project, 292**

**talking to Flickr**

character encoding, 62

overview of, 61–62

request format

REST, 62–65

SOAP, 66–68

XML-RPC, 68–71

**taxonomy, 34**

**Technorati**

API key

getting, 242

protecting, 254

description of, 241

Greasemonkey and, 243–247

querying with Greasemonkey, 248–254

searching cosmos, 242–243

**Terms of Use, 40, 290**

**test, methods for, 384**

**testing setup**

Apache Web server, 55–56

Microsoft IIS, 52–53

**third party, accessing, 248**

**third-party application, permissions given to, 148**

**ticket number, 176**

**token, 153, 168**

**tools and technologies**

AJAX, 12, 131–136

Curl, 22

databases, 21, 291, 295–296, 300–306

HTML, XHTML, and CSS, 7–9

JavaScript and DOM, 11–12

JSON, 12–14

MySQL, 299, 300–306

overview of, 7

PHP and Perl, 16–17

regular expressions, 17–19

REST, 19–20, 62–65

SOAP, 20, 66–68

Web browsers, 16

Web servers and HTTP, 14–15

XML and XPath, 9–11

XML-RPC, 20–21, 68–71

**tracking membership, 292–293**

## U

**Unicode Transformation Format, 62**

**Unix**

installing MySQL on, 298

PHP MySQL extension, 306

`unsafeWindow` **variable, 191**

`UPDATE` **SQL command, 305–306**

`updateCredentials` **method, 159, 160**

**Upload API, 169–171**

**upload page, creating, 171–176**

**uploading photograph**

asynchronous uploading, 176

formats, 27

overview of, 24–26

photo IDs and URLs, 27

replacing when, 176

with Upload API, 169–171

`uploadPhoto()` **method, 175**

**Uploadr, 25, 26**

**URL**

for account, setting, 24

API Explorer, 79

building

buddy icon URL, 78

group icon URL, 79

image URL, 77–78

photo page URL, 76–77

photoset URL, 78

photostream URL, 75

profile URL, 75

callback, 150–151

## URL (continued)

---

### URL (continued)

- for group, 38
- methods for, 384
- for photo, 27
- for photoset, 30

### URL-encoded parameter, 64

### user

- allowing to add tag to photo, 34
- details about, retrieving, 74–75
- determining NSID of, 183–185
- displaying NSID of, 185–188

### user data structure, 391

### user\_id parameter, 64

### .user.js file extension, 181

### usort function (PHP), 371

### Utata group, 206, 214

### Utata web site, 291–296

### UTF-8 character encoding, 62

## V

### View blogs link, 252

### view count, increasing, 292

### viewing

- data in database, 302–303
- resized image, 27
- satellite imagery, 259–260
- Smart Set, 331–334

### View→Page Source, 183

### virtual host, 15

### Visible to Friends or Visible to Family option, 37

### visualizing-news/index.php page, 235

## W

### Watson, John (creator of mashups and remixes), 215

### web browser, 16. *See also* Firefox web browser

### web server

- description of, 14
- setting up
  - Apache, 53–56
  - directories, 56–57
  - Linux-based, 56
  - Mac OSX, 56
  - Microsoft IIS, 46–53

### Web Services Description Language (WSDL), 20

### web sites. *See also* Flickr web site

- ActivePerl, 57
- aggregators, 87
- Apache, 53
- BBC News RSS feed, 219
- CPAN, 17
- Creative Commons, 40
- Curl, 22
- Dive Into Greasemonkey, 199
- feeds, 89
- Flickr Organizr, 31
- geotagging photos, 267
- Google Maps, 256
- Greasemonkey, 179
- HTML 4 Specification, 170
- ImageMagick, 201, 213
- JSON, 14
- Magpie, 221
- MD5 hash, 329
- Mozilla Developer Center, 188
- MySQL, 21, 296
- PHP, 17
- phpFlickr library, 82, 116
- Prototype JavaScript framework, 12
- Prototype library, 131
- regular expression pattern syntax, 19
- tag URI scheme, 92
- Technorati, 241
- Unicode Consortium, 62
- Utata, 291–296
- web servers, 14
- World Wide Web Consortium, 7
- YAML, 94

### Web 2.0, 3–4

### Welcome screen, 338

### WHERE clause (SQL), 303–304, 332

### wildcard character (\*), 302

### Windows (Microsoft)

- ImageMagick, installing, 201–202
- MySQL, installing, 296–298
- Perl, installing, 57–59
- PHP MySQL extension, 306

### World Wide Web Consortium (W3C), 7

### WSDL (Web Services Description Language), 20

### W3C (World Wide Web Consortium), 7

**X**

**XHTML (Extensible Hypertext Markup Language), 8**

**XML (Extensible Markup Language), 9–10**

**XML Path Language (XPath), 10–11**

**XML-RPC**

API calls, 68–71

description of, 20–21

method parameter and, 153

**XPath (XML Path Language)**

description of, 10–11

using to find tags, 188–190

**Y**

**YAML feed format, 96–97**

**Yuan.CC Maps, 267**

**Z**

**zooming in on Google Maps, 257**

**zooming out of Google Maps, 258**

















