PeepCode press

$9

Rails2

New features for your applications

by Ryan Daigle

(y!) Factorial

CONTENTS

INTRODUCTION

Those who saw David Heinemeier Hansson's (http://www.loudthinking.com) keynote (http://www.bestechvideos.com/2007/10/02/david-heinemeier-hansson-keynote-at-railsconf-2007) at RailsConf 2007 know that Rails 2.0 doesn't contain waves of extraordinary new features and paradigm shifts. Instead, it focuses on further supporting RESTful web architectures, some small feature maturations and the removal of libraries and features that have been determined not to be essential to Rails.

I've been covering new Rails features as they've been introduced to Rails in the What's New in Edge Rails (http://ryandaigle.com) series of articles for several releases now. Most major features have been discussed there, but some have not and some have even changed slightly since the initial feature commit. My hope is that this mini-book will provide developers with a timely handbook that will clearly outline new features and how to use them.

Ryan Daigle, *Founder*, yFactorial, LLC (http://yfactorial.com)

ActiveRecord

CHAPTER 1

ActiveRecord is the biggest part of Rails, so it's no surprise that it has received some enhancements for Rails 2.0.

Validations

Numerical Validation Becomes Useful

The `validates_numericality_of` method sounds like a really useful validation, but has failed me more often than not. I expect that no longer to be the case with the new options supported in Rails 2.

Here's a quick run-down of these options:

```
validates_numericality_of :salary,
                          :greater_than => 39999

validates_numericality_of :salary,
                          :greater_than_or_equal_to => 40000

validates_numericality_of :ten,
                          :equal_to => 10

validates_numericality_of :bonus,
                          :less_than => 5000

validates_numericality_of :bonus,
                          :less_than_or_equal_to => 4999

validates_numericality_of :prime,
                          :odd => true

validates_numericality_of :squared,
                          :even => true
```

You can even stack the conditions (just make sure they're not in conflict with each other):

```
validates_numericality_of :salary,
                          :greater_than_or_equal_to => 8000,
                          :even => true
```

:allow_blank

Have you ever used the `:allow_nil` option to skip validation if the particular attribute in question is nil?

```
class Post < ActiveRecord::Base
  validates_numericality_of :rating, :allow_nil => true
end

Post.new(:rating => nil).valid?
#=> true
```

This is useful for optional attributes or for those attributes whose existence is already being validated with `validates_presence_of`. But, for optional attributes that come in from your standard web request, the value is often set to be the empty string (since the form field was submitted with an empty value). The only way to deal with that case used to be by crafting your own `:if` proc:

```
class Post < ActiveRecord::Base

  validates_numericality_of :rating,
    :if => Proc.new { |post| not post.rating.blank? }

end
```

Now that pattern has been factored out into the `:allow_blank` option, which will skip validation if the attribute is nil, empty or a whitespace string.

```
class Post < ActiveRecord::Base

  validates_numericality_of :rating, :allow_blank => true

end
```

The default value for `:allow_blank` is `false`, as it is for the existing `:allow_nil`.

Query Caching

Prior to Rails 2, ActiveRecord didn't have a built-in caching mechanism. You were forced to roll your own instance variable cache like:

```
@cached_all ||= find(:all)
```

or bust out some big guns like memcached (http://www.danga.com/memcached).

Starting with Rails 2, ActiveRecord has been enabled with a very simple, but effective, query cache. It's automatically enabled, so you can just sit back and enjoy the speed-up.

How it works is that all select statement results are cached, keyed by the statement itself, so subsequent identical statements don't need to hit the database. The cache is cleared anytime an INSERT, UPDATE or DELETE operation is executed even if it doesn't affect the results of cached select statements This is not a robust, handle every case solution. However, it is a low impact and easy way to provide

CACHE SCOPE

The query cache is a short-lived beast. Because Rails is single-threaded, the cache only lives as long as its host model instance lives. In most cases this is only the scope of the incoming HTTP request. As a result, identical select statements across different requests won't be able to take advantage of the query cache.

caching that won't get in your way.

Disable Caching

In the rare instance when you want to be sure you're going out and fetching the latest results from the database (maybe you have an external process that you know has affected your dataset outside the scope of the ActiveRecord model), you can disable the cache within an `uncached` block. Here's an easy way to provide a `find` method that will always bypass the cache:

```
class Post < ActiveRecord::Base

  class << self

    # Force to go out and get the latest from the db
    def force_find(*args)
      uncached { find(*args) }
    end

  end
end
```

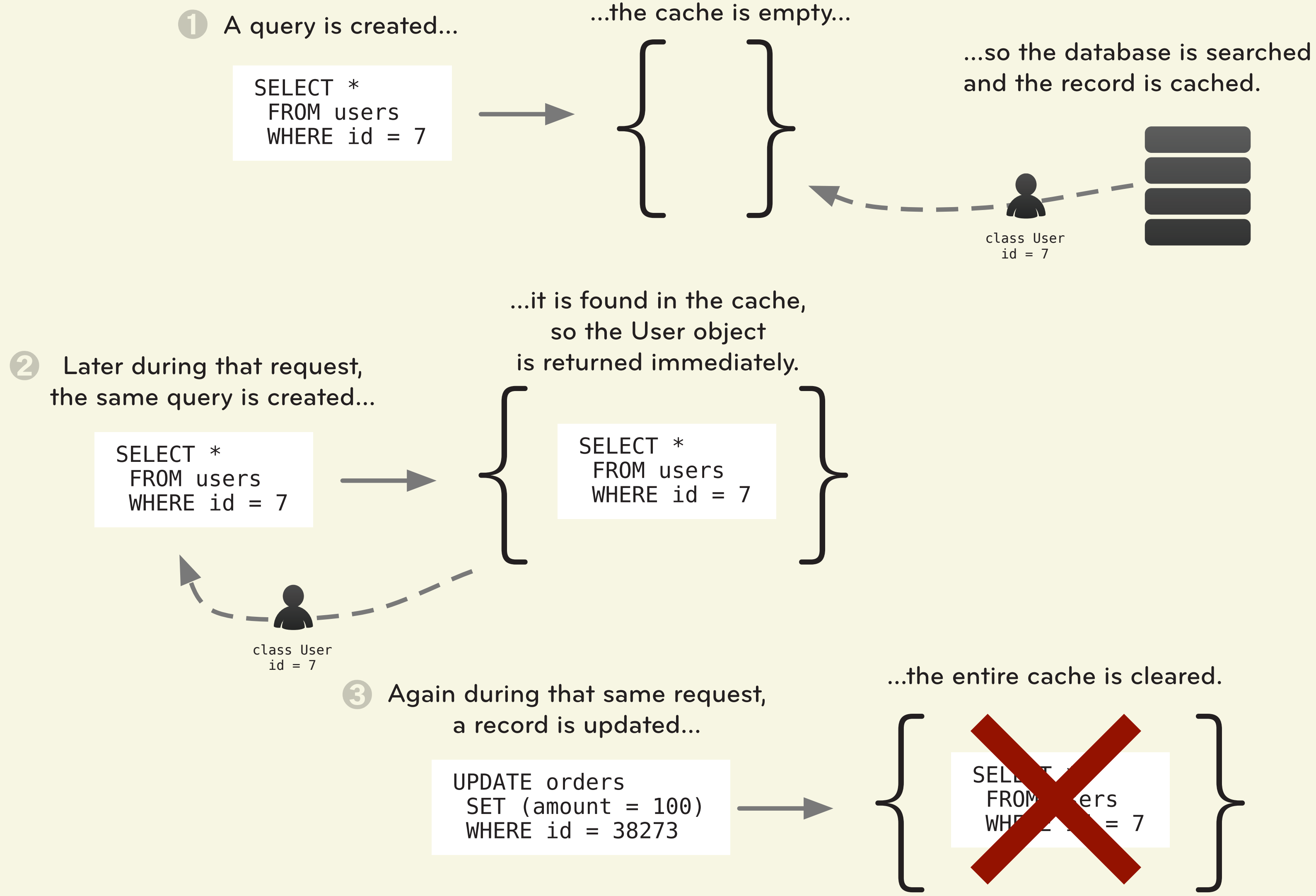
Hits on the query cache are clearly denoted in the log files with the CACHE line prefix.

With this `Post` definition you could call `force_find` the same way you call `find`:

```
Post.force_find(:all,
                :limit => 5,
                :order => 'created_at DESC')
```

While the query cache is mostly a transparent feature, it's good to know what's happening under the covers of your ActiveRecord models.

THE QUERY CACHE



1 A query is created...

```
SELECT *
 FROM users
 WHERE id = 7
```

...the cache is empty...

...so the database is searched and the record is cached.

...it is found in the cache, so the User object is returned immediately.

2 Later during that request, the same query is created...

```
SELECT *
 FROM users
 WHERE id = 7
```

```
SELECT *
 FROM users
 WHERE id = 7
```

3 Again during that same request, a record is updated...

```
UPDATE orders
 SET (amount = 100)
 WHERE id = 38273
```

...the entire cache is cleared.

```
SELECT *
 FROM users
 WHERE id = 7
```

Sexy Migrations

Migrations have their fans and their detractors. Me? Well, I like 'em. And I really like the Sexy Migrations plugin (http://errtheblog.com/post/2381) that let you condense your migrations and provide for some syntactical shortcuts. While not a direct port of the plugin, Rails 2 has taken most of the functionality of the plugin and made it part of Rails. Here's the lowdown.

Condensed Column Declarations

Chances are many of your migrations have a lot of duplication between similarly defined columns:

```
# The old way
class CreatePosts < ActiveRecord::Migration

  # Create a table holding blog posts
  create_table :posts do |t|

    t.column :user_id,     :integer, :null => false
    t.column :category_id, :integer, :null => false
    t.column :body, :text

    # Standard auto-magic columns
    t.column :created_at, :datetime
    t.column :updated_at, :datetime
  end

  def self.down
    drop_table :posts
  end
end
```

With the new sexy migrations syntax you can now declare the type of the column first, allowing you to declare multiple columns of the same type in a single line:

```
class CreatePosts < ActiveRecord::Migration

  # Create a table holding blog posts
  create_table :posts do |t|
    t.integer :user_id, :category_id, :null => false
    t.text    :body
    t.timestamps
  end

  def self.down
    drop_table :posts
  end
end
```

The definition of the `user_id` and `category_id` columns can now be contained in the same line since they have the same definition. You can also get both auto-magic timestamp columns with a single `t.timestamps` call.

And instead of declaring the foreign key columns directly, as in this example, we can use the `references` method for a more intuitive syntax:

```
class CreatePosts < ActiveRecord::Migration

  # Create a table holding blog posts
  create_table :posts do |t|
    t.references :user, :category, :null => false
    t.text    :body
    t.timestamps
  end

  def self.down
    drop_table :posts
  end
end
```

You can also use `belongs_to` instead of `references`:

```
  t.belongs_to :user, :category, :null => false
```

The `references` method also supports a `:polymorphic` option which will create both the foreign key column to the polymorphic type and the `type` column itself.

So `t.references :taggable, :polymorphic => true` will generate both the `taggable_id` and `taggable_type` column of the current table. You can also use `t.references :taggable, :polymorphic => { :default => 'Post' }` format to specify a default polymorphic type.

That's tight!

Foxy Fixtures

Fixtures are a mixed bag. They provide an easy way to provision the database for each test case and they provide convenient accessor methods for the records. However, they're also notorious for breaking down as the test dataset grows due to the need to manually build the various associations between records.

Users of Rails 1.2 may be familiar with the Foxy Fixtures (http://svn.geeksomnia.com/rathole/trunk/README) plugin (originally called `rathole` in some circles) which adds some very helpful auto-generation of foreign key references and other common columns. John Barnette (http://geeksomnia.com) managed to get this functionality added to Rails for the 2.0 release, making all of our lives much easier.

Here's how you use it. Assume a basic company to employees one to many association:

```
  class Company < ActiveRecord::Base
    has_many :employees
  end
```

```
class Employee < ActiveRecord::Base
  belongs_to :company
end
```

Without Rathole, there's a lot of non-essential stuff. It's easy to mess up IDs since we have to manually specify them. All this gets exponentially hairier as the data-set increases.

```
# companies.yml
yfactorial:
  id: 1
  name: yFactorial, LLC
  created_at: <%= Time.now %>
  updated_at: <%= Time.now %>


# employees.yml
ryan:
  id: 1
  name: Ryan Daigle
  company_id: 1
  created_at: <%= Time.now %>
  updated_at: <%= Time.now %>
```

With Rathole's features, you no longer need to manually reconcile the cross referenced IDs or set the auto-magic date column values. Here is what your fixtures can look like in Rails 2:

```
# companies.yml
yfactorial:
  name: yFactorial, LLC


# employees.yml
ryan:
  name: Ryan Daigle
  company: yfactorial
```

Take special note of the following changes between the two fixture-sets:

- You no longer have to specify the ID for each record. It is automatically generated from the hash of the name of the record (so the integer hash of `yfactorial` becomes that record's ID).
- There's no need to specify the auto-magic date field values anymore. All the `created_xx` and `updated_xx` fields will be automatically populated with `Time.now`.
- When referencing other record IDs (as in `employees.company_id`), you can just use the fixture name and it will resolve that column to the primary key value of the referenced record. This is what allowed me to change the `company_id: 1` line in `employees.yml` to be `company: yfactorial`.

> Since the primary keys of each record are dynamically calculated (as a hash of the fixture name), the fixture doesn't need to be loaded before knowing what its primary key value will be. This resolves a lot of the recursive dependency and ordering issues you may have otherwise encountered with an approach such as this.

Many to Many Associations

But there's more! Foreign key references get especially twisted when you have join tables that govern your `has_and_belongs_to_many` associations. So assume a company `has_and_belongs_to_many` `industry_associations`:

```ruby
class Company < ActiveRecord::Base
  has_and_belongs_to_many :industry_associations,
        :join_table => 'company_industry_associations'
end
```

```
class IndustryAssociation < ActiveRecord::Base
  has_and_belongs_to_many :companies,
        :join_table => 'company_industry_associations'
end
```

Instead of having to write our own `company_industry_associations.yml` fixture file, we can just reference the collection of associations a company belongs to inline in `companies.yml`:

```
# companies.yml
yfactorial:
  name: yFactorial, LLC
  industry_associations: ruby, webservices
```

Assuming there's an `industry_associations.yml` fixture file that specifies records named `ruby` and `webservices`, the habtm join table will be automatically populated for you.

Polymorphic Relationships

Say you have a domain model with a polymorphic association:

```
class Address < ActiveRecord::Base
  belongs_to :addressable, :polymorphic => true
end
```

Instead of having to specify the type and id of the polymorphic association in your fixture file:

```
# addresses.yml
home:
  address: 142 My St.
  addressable_type: Location
  addressable_id: 1
```

```
# locations.yml
office:
  id: 1
```

you can just reference the symbol of the fixture identifying the relationship and enclose the expected type in parentheses:

```
# addresses.yml
home:
  address: 142 My St.
  addressable: office (Location)

# locations.yml
office:
  id: 1
```

Other Tidbits

Need some other shortcuts?

How about the fact that you can use the `$LABEL` string to reference the name of the fixture within the fixture itself? So the following fixture file will create a record in the database whose `name` column will be `yfactorial`:

```
# companies.yml
yfactorial:
  name: $LABEL
```

More? You want more of this? How about the fact that you can factor out default values with this syntax:

```
DEFAULTS: &DEFAULTS
  state: nc
  industry: software
```

```
yfactorial:
  name: yFactorial, LLC
  <<: *DEFAULTS

terralien:
  name: Terralien, Inc.
  <<: *DEFAULTS
```

Overall, this is a huge win for those of us who use fixtures.

Love this functionality but haven't yet made the leap to Rails 2.0? Download the original foxy fixtures plugin (http://svn.geeksomnia.com/plugins/trunk/foxy_fixtures) to give your 1.2 apps some foxy lovin'.

ActionController & ActionView

CHAPTER 2

ActionController and ActionView are packaged together as the `actionpack` gem.

Asset Servers

It's highly likely that your application references more than two of the following:

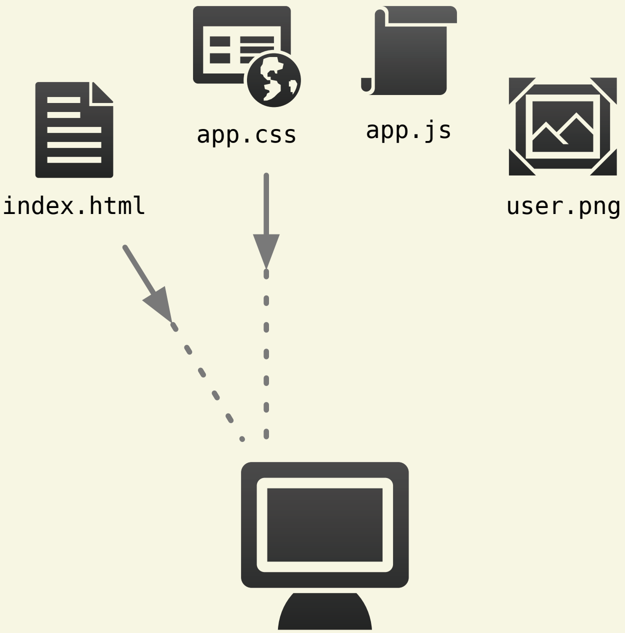
- HTML content
- Javascript include files
- Stylesheets
- Images

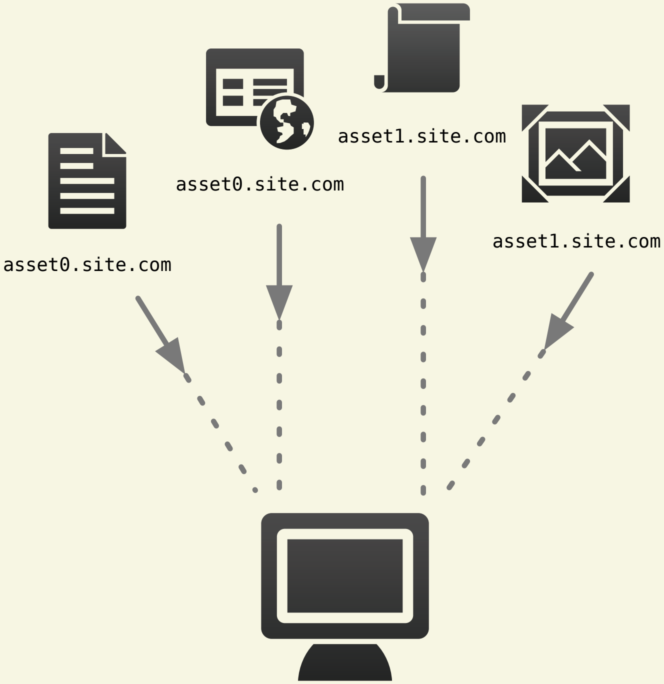
The problem is that modern browsers limit themselves to making only two connections to any single domain. Even if your users have high-speed broadband connections to the internet, they will have to wait until all the assets have been sequentially downloaded.

Rails 2 offers a solution to this problem based on ideas from Cal Henderson (http://www.thinkvitamin.com/features/webapps/serving-javascript-fast), Ezra Zygmuntowicz (http://brainspl.at/articles/2006/12/29/speed-up-page-loads), and other research (http://www.die.net/musings/page_load_time). The solution is to use several servers to host assets (optimally 4). However, you can take advantage of this even if you only have one server.

First, add this line to `config/environments/production.rb`:

ASSET SERVERS





A browser will only open 2 connections to any domain at one time.

If there are more than 2 assets to fetch, the browser waits until the first two have been fully downloaded.

Naming several asset servers tricks the browser into opening many connections at once, even if they are on the same physical server.

The result is faster delivery of the assets to the browser.

```
config.action_controller.asset_host =
                      "http://asset%d.site.com"
```

The important part is the %d, which will be replaced with the numbers 0-3. The rest should match the actual domain name of your site.

- http://asset0.site.com
- http://asset1.site.com
- http://asset2.site.com
- http://asset3.site.com

This isn't random...the hash of the filename is used to generate a consistent number for a given asset every time it is requested.

SERVER CONFIGURATION

The second problem is how to point those subdomains back at your server.

First, you need to setup the DNS for the subdomains. The easiest way is to use a wildcard DNS record. This will catch all subdomain requests and send them to the IP address of your server.

```
*.site.com. 10000 IN A 1.2.3.4
```

If you use SSL, you will also need to buy a wildcard SSL certificate. Standard SSL certificates are only good for a single domain, including the subdomain.

Next, your webserver needs to accept requests for the asset subdomains.

In Apache, you can use a `ServerAlias` directive. You may need to list this as the last virtual host in your config file if you hope to serve other subdomains, too.

```
ServerAlias *.site.com
```

In Nginx, you could list the asset subdomains explicitly, or just comment out the `server_name` directive to accept all subdomains.

```
server {
  listen 80;
  server_name site.com asset0.site.com asset1.site.com
asset2.site.com asset3.site.com;
}
```

And that's it! In the development environment, assets will be served locally. In the production environment, assets will be served from the four subdomains, providing a much faster browsing experience.

Asset Caching

Serving assets from multiple subdomains isn't the only way to speed up the delivery of content to your visitors. You can also benefit from serving fewer assets. Even though this results in larger individual files, it will require fewer total requests from the browser, which may speed up the delivery of assets.

Rails 2 provides a convenient way to do this. The built-in stylesheet and Javascript tags now take an extra `:cache` option that will combine related assets into a single file. As you might expect, Rails does this in an intelligent way that caches the files only in the production environment (or any cached environment).

For example, this is a common link tag that you might find in an application layout.

```
stylesheet_link_tag "application", "forms", :cache => true
```

In the development environment, the `application.css` and `forms.css` files will be served independently. In production, they will be combined into a single file and written to `public/stylesheets/all.css`.

> You don't have to do anything in order to take advantage of the aggregated `all.css` stylesheet. Just add the `:cache` option and Rails will emit the appropriate HTML include tags in the production environment.

You can change the name of the combined file by passing it as an argument to the `:cache` option. This is useful for grouping assets into separately aggregated files.

```
javascript_include_tag :defaults, :cache => true
# => Creates public/javascripts/all.js with prototype.js,
#    effects.js, controls.js, dragdrop.js,
#    and application.js.
```

```
javascript_include_tag "cal", "game", :cache => "widgets"
# => Creates public/javascripts/widgets.js with
#    cal.js and game.js.
```

Why would you want to split assets into different files? Whenever a file changes, the browser will need to download the entire file again. Libraries such as Prototype and Script.aculo.us don't change very often, but your `application.js` probably will. So you may benefit from packaging external libraries separately from application-specific libraries.

OTHER OPTIONS

Although there is a benefit to combining assets in this way, there is more that can be done.

The :defaults package listed above weighs in at 165 KB. It can be compressed significantly by enabling GZIP compression on your webserver. This will dynamically compress textual content, including the actual HTML pages generated by your Rails application.

For Apache, add this to your VirtualHost section:

```
# Deflate
AddOutputFilterByType DEFLATE text/html text/xml text/plain
text/css application/x-javascript text/javascript;
BrowserMatch ^Mozilla/4 gzip-only-text/html
BrowserMatch ^Mozilla/4\.0[678] no-gzip
BrowserMatch \bMSIE !no-gzip !gzip-only-text/html
```

> You may also need to load the deflate_module
> if it wasn't already included with your server.

For Nginx, add this to your http section:

```
gzip on;
gzip_http_version 1.0;
gzip_comp_level 2;
gzip_proxied any;
gzip_types text/plain text/html text/css application/x-
javascript text/xml application/xml application/xml+rss
text/javascript;
```

Even further compression can be achieved by removing blank lines and whitespace from your files. Many people feel that this is unnecessary, but it's undeniable that it does result in even smaller files.

Scott Becker's asset_packager plugin (http://synthesis.sbecker.net/pages/asset_packager) can do this for you.

Default Cookie-based Sessions

Sessions are a sticky subject. Sessions can cause a lot of scalability problems and be a bottleneck to your application's growth if they're not properly accounted for. You can now rest easy as the Rails team has decided the cookie-based sessions are a reasonable default.

Cookie-based sessions are an improvement because they're very fast. The time it takes to process a cookie from the client is an order of magnitude faster than having to retrieve it from the database, the other reasonable default storage option.

While cookie-based sessions are now the default, you still have to provide some configuration settings for them within the `Rails::Initializer.run` block in your `environment.rb` file. Just uncomment or add the following to your initializer block to get your new sessions up and running.

```
# Your secret key for verifying cookie-based
# session data integrity.
# If you change this key, all existing sessions
# will become invalid!
config.action_controller.session = {
   :session_key => '_myapp_session',
   :secret      => 'getyourownsecretkey'
}
```

CAVEAT

These cookie based sessions are limited to 4K in size, so if you're overloading your sessions with data you'll need to look at the other session storage options or trim back a little on what you're putting in your sessions. After all, do you really need more than the user id and the flash? If you violate this size constraint then you'll be seeing a CGI::Session::CookieStore::CookieOverflow error rear its ugly head.

Security Concerns

For those of you a bit uncomfortable with the thought of putting your valuable session data out there, keep in mind that the data is digested using OpenSSL::HMAC which avoids any potential data tampering. If you're still not happy with that level of security, you can provide your own digest method for the cookies:

```
class CGI::Session::CookieStore
  def generate_digest(data)
    # Provide your own uber-strength encryption here
    Digest::SHA512.hexdigest "#{data}#{@secret}"
  end
end
```

Have at it, tiger.

Partial Layouts

Partials are a great way to extract out common bits and pieces of your views. However, there are sometimes structural commonalities that govern your partials that, prior to Rails 2, were forced to exist separately in each partial. Now you can use partial layouts to provide structure to partials with common characteristics.

Say you have the proverbial blog application that has both a detailed way to display an article and a simple way to view an article. While the base bit of article view is somewhat static, the structure and supporting information around it varies depending on which layout is in play. In the main index page we want to list each article without a lot of extraneous information, so we invoke the rendering of the `_article` partial with the `_article_simple` layout:

```
<% for post in @posts do %>
  <%= render :partial => 'article',
             :layout => 'article_simple',
             :locals => {:article => post} %>
<% end %>
```

The local variables will be made available to both the partial and the layout. So within the `_article_simple` layout you can still reference the `article` local:

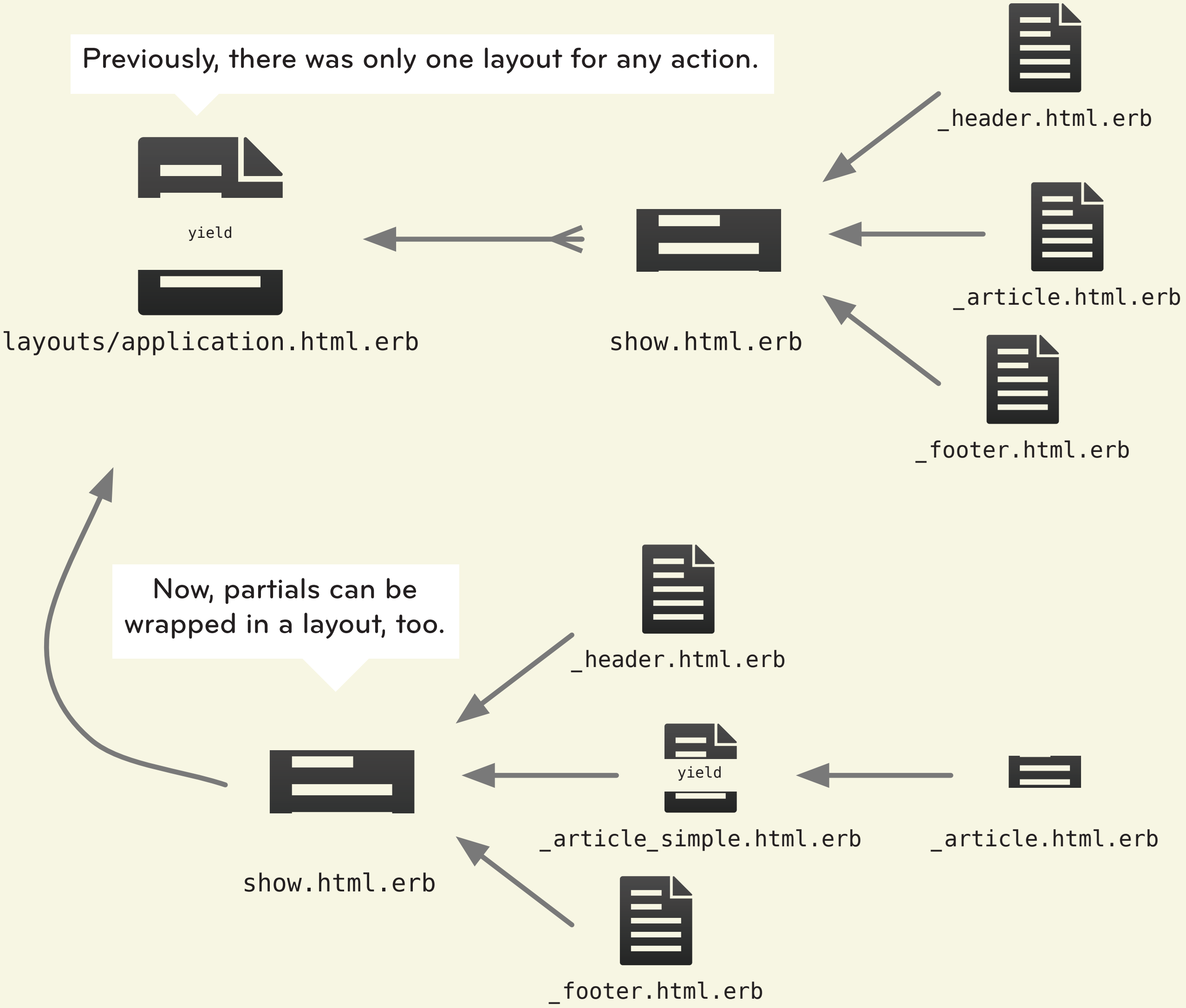
```
<div id="article_<%= article.id %>">
  <%= yield %>
</div>
```

> Partial layouts are stored in the controller's view directory, not the global layouts directory. So our `_article_simple` partial is stored in `app/views/posts/_article_simple.html.erb`.

The partial itself in this example is pretty standard, containing the common bit of an article view:

```
<h2 id="article_<%= article.id %>_title">
  <%= article.title %>
</h2>
<div id="article_<%= article.id %>_body">
  <%= article.body %>
</div>
```

PARTIAL LAYOUTS



Rendering this simple article partial produces this html:

```
<div id="article_1">
  <h2 id="article_1_title">
    Partial Layouts
  </h2>
  <div id="article_1_body">
    If you've ever taken the time to factor out the various
commonalities that often exist in your Rails views into
partials, you've probably noticed that there are still
opportunities for DRYing up those partials. Well now you
have a tool in your toolbox...
  </div>
</div>
```

However, when viewing an individual article it's appropriate to display more detail about the article. In that case you could use a more detailed article layout. Here's the `show.html.erb` view:

```
<%= render :partial => 'article',
           :layout => 'article_detailed',
           :locals => {:article => @post} %>
```

And the more detailed layout (`_article_detailed.html.erb`):

```
<div id="article_<%= article.id %>">
  <div class="ads"><%= get_ads %></div>
  <ul class="meta">
    <li>Posted by <%= article.author %></li>
    <li>on <%= article.published_at %></li>
  </ul>
  <%= yield %>
</div>
```

WHEN TO USE?

It's easy to get confused about how to use partial layouts as opposed to just several partials or even nested partials. I like to think of partial layouts as common view elements wrapped *around* the main partial content – they control the structure. Multiple or nested partials, on the other hand, just iterate through common view content. Ask yourself whether you have a common structure wrapped around your view bits or if you just have several common, independent bits.

RESTful Routing Updates

Rails' march towards the day when REST rules the world continues in this release with a few notable updates to RESTful routing.

Custom Action Delimiter

Prior to Rails 2, any custom actions defined in your routes were appended to the URI with a semicolon as the delimiter. So, for example, a `deactivate` action that could only be accessed via POST would be defined in `config/routes.rb` like this:

```
map.resources :posts, :member => {:deactivate => :post}
```

It would be matched with a semicolon with `/posts/:id;deactivate`.

Other built-in actions also used the semicolon, such as `edit`.

Like communism, it seemed like a good idea at the time. However, the semicolon made page caching nearly impossible since various webservers had different understandings of the role of the semicolon.

With Rails 2 the semicolon will be replaced by your standard, less obnoxious forward-slash: `/posts/:id/deactivate`.

But wait! Doesn't this create the possibility for custom collection actions to override member actions? If I have a custom collection action named `/articles/fruit`, how will Rails distinguish that from an article I wrote about fruit which also resides at `/articles/fruit`? The answer is that the collection action will always take precedence. But honestly, if you run into this rare situation, you probably need to reorganize your controllers to eliminate this possibility.

Automatic Named Route Prefixing

Previously, when you had a series of nested resources you had to explicitly set the `:name_prefix` to avoid conflict with other top-level named routes:

```
map.resources :comments
map.resources :posts do |post|
  post.resources :comments, :name_prefix => 'post_'
end
```

This gave you several named routes for comments within posts:

- post_comments_path(post) → /posts/:post_id/comments
- new_post_comment_path(post) → /posts/:post_id/comments/new

etc...

The name prefix is more than just automatic... it's required! You'll see a warning in older applications that use the shorter version of the URL helper, such as `comments_url`. You must use the fully qualified `post_comments_url` now.

As of Rails 2 this `:name_prefix` is automatically set for you on nested resources, eliminating the need to specify the prefix at all. You can always disable the automatic prefix by setting it to nil: `:name_prefix => nil`.

Namespaces

You can now declare a routing namespace using in a RESTful manner with `map.namespace`. The most common use of this is for establishing an administrative section of an application.

```
map.namespace(:admin) do |admin|
  admin.resources :posts
end
```

With this routing definition, you get the standard host of `post` routes and URL helpers, all prefixed with `admin`:

- `admin_posts_path` → `/admin/posts`
- `admin_post_path(post)` → `/admin/posts/:id`
- `formatted_admin_post_path(post, format)` → `/admin/posts/:id.:format`
- `new_admin_post_path` → `/admin/posts/new`
- `edit_admin_post_path(post)` → `/admin/posts/:id/edit`
- etc.

With this namespace, the routing mechanism will try to find a controller named `Admin::PostsController` to satisfy these URLs.

has_many and has_one Routing

REST in Rails is all about thinking in terms of resources and the relationships between the resources. In doing that, you begin to think in terms of an object model even at the routing level.

With Rails 2 the routing DSL provides a `has_many` association that brings a simple way to declare a one to many relationship between resources. It can replace nested sets of `map.resources` for simple scenarios that don't need to declare any custom actions:

```
map.resources :posts,
  :has_many => [:comments, :tags],
  :has_one => :user
```

Althought it is indented here, these are hash arguments, not method calls in a block.

The has_many association above is equivalent to this in previous Rails releases:

```
# The old way
map.resources :posts do |post|
  post.resources :comments
  post.resources :tags
end
```

The has_one declaration is used to define, obviously, singular associations and is reflected in the URI that matches the route. In this case a post's authoring user is found at /posts/:id/user. (Note the singular user).

rescue_from Exception Handlers

Exceptions happen...and that's OK. What's not OK is letting the end user see these gross displays of your application's innards. The standard way of making sure no exceptions go un-handled is by providing your own rescue_action_in_public method in application.rb, which forces you to handle specific exceptions in a sort of procedural way:

```
class PostsController < ApplicationController

  def rescue_action_in_public(exception)
    case(exception)
      when ActiveRecord::RecordNotFound
        render :file => '/bad_record'
      when NoMethodError
        render :file => '/no_method'
      else
        render :file => '/error'
    end
  end

end
```

Ok, that works, but it's pretty clear that this method will quickly grow too big and too procedural for our beautiful language, Ruby. Well we now have a much cleaner, DSL-ish way to map specific exceptions to the logic that should handle them – the `rescue_from` class method:

```ruby
class PostsController < ApplicationController

  # Declare exception to handler methods
  rescue_from ActiveRecord::RecordNotFound,
              :with => :bad_record

  rescue_from NoMethodError,
              :with => :show_error

  # Define handler methods

  def bad_record
    render :file => '/bad_record'
  end

  def show_error(exception)
    render :text => exception.message
  end

end
```

The `:with` option is where you specify the symbol of the instance method that should be invoked when the exception of the given type is raised. That handler method can be a no-arg method, or can take the exception that's being handled as its only argument.

Handler methods can also be defined in the `ApplicationController` and reused in multiple sub-controllers.

You can also use a block or a proc to handle the exception inline.

```
class PostsController < ApplicationController

  rescue_from(ActiveRecord::RecordNotFound) do |exception|
    render :file => '/bad_record'
  end

  rescue_from NoMethodError, :with => proc do |e|
    render :text => e.message
  end

end
```

The `rescue_from` method lets you cleanly separate the exception (and the logic that handles it) from the controller where the exception is encountered. It's a much more flexible way to structure your error handling code.

HTTP Authentication

HTTP authentication feels like a dinosaur in today's age of flashy, Ajaxy login forms and whiz-bang visual effects. But, it's a well-supported authentication mechanism and is keenly suited for web-services authentication.

> PeepCode uses HTTP authentication to serve authenticated, personalized feeds at https://peepcode.com/purchases.rss.

With Rails 2 you have a very easy way to perform http authentication, though it's currently only wired for basic authentication (not digest). To get basic authentication functionality, `authenticate_or_request_with_http_basic` is your man.

```
class AdminController < ApplicationController

  before_filter :authenticate

  def authenticate
    authenticate_or_request_with_http_basic do |name, pass|
      User.admin?(name, pass)
    end
  end
end
```

The `authenticate_or_request_with_http_basic` method passes in the username and password submitted via http authentication and is where you put the logic that determines if those credentials are authorized or not. In this example, `User.admin?` contains that logic and returns a `boolean` value representing the success of the authentication. The final value of the block is what is returned from `authenticate_or_request_with_http_basic` (keep that in mind when used in a controller filter as it will determine the execution of the rest of the action chain).

If you need to have a little more control over the authentication process, you have access to both `authenticate_with_http_basic` and `request_http_basic_authentication`.

The `authenticate_with_http_basic` method is very much like `authenticate_or_request_with_http_basic` in that it yields the username and password to a block to determine authorization status.

The `request_http_basic_authentication` method just sends a response back that tells the client to prompt the user to re-enter their credentials.

TESTING

To test the submission of a set of user credentials via http basic authentication you can use the `ActionController::HttpAuthentication::Basic.encode_credentials` method to encode the values. Set these encoded values in the 'HTTP_AUTHORIZATION' header of the test request in your functional test:

```
def test_authentication

  # Setup authentication values
  @request.env['HTTP_AUTHORIZATION'] =
    ActionController::HttpAuthentication:
:Basic.encode_credentials({
      users(:ryan).name,
      users(:ryan).password
    })

  get :index
  assert_response :success
end
```

These two methods can be used in tandem in a more finer grained manner on authentication.

```
class AdminController < ApplicationController

  before_filter :authenticate

  # This method restricts a user to only
  # three failed login attempts.
  def authenticate
    # Protect from attacks with low session expiration
    return false if (session[:login_tries].to_i > 3)

    # Attempt to authenticate
    if(@user = authenticate_with_http_basic { |name, pass|
                 User.authenticate(name, pass)
               })
      session[:login_tries] = nil
      return true
    else
      session[:login_tries] = session[:login_tries].to_i + 1
      request_http_basic_authentication
    end
  end

end
```

In this example we use the two authentication methods to ensure that a user only gets 3 chances to login in any one sitting. This example will probably not blow your mind, but it shows you how to access and utilize the separate authentication methods provided for you.

Convenient Access to Helpers

Helpers let you extract complicated view logic out of the views and into a more appropriate, reusable form. However, by default, they're only accessible within their related views. For instance, the `posts_` helper is accessible by /posts views. You can always include helpers of other models by using the helper class method within your controller:

```
helper :other_helper_name
```

This is nice, but often times you rely on several helpers being available, especially in complicated views that reference several models. In these cases, instead of picking and choosing which helpers you need, you can use helper :all to include all helpers within the /app/helpers directory.

> If you create a Rails 2 application with the rails generator you'll get all helpers included by default. This can be changed by commenting out the helper :all line in your application.rb.

Miscellaneous

CHAPTER 3

De-Cruft Your Environment File

When your project started, the Rails `environment.rb` file was a nice little `Rails::Initializer.run` block. As your project grew, however, chances are that file became a random configuration receptacle – holding any bit of configuration code you didn't know where else to put. Here's what one of my project's `environment.rb` file looks like (condensed for readability):

```
require File.join(File.dirname(__FILE__), 'boot')

Rails::Initializer.run do |config|
end

# A bunch of non-environment related configuration
# snippets, quickly growing...
ActionMailer::Base.delivery_method = :sendmail
ActiveSupport::CoreExtensions::Date::Conversions::DATE_
FORMATS.merge!(:concise => "%d.%b.%y")

# Bunch more...
```

Ugly? No, but quickly growing beyond the bounds of what an *environment* file should contain.

With Rails 2, you're now given a directory (`config/initializers`) to place these separate, discreet bits of configuration as their own files. These are then automatically loaded *after* plugins are loaded in the Rails bootstrap lifecycle (giving you the ability to tweak plugins as well as all other Ruby and Rails classes.)

As a rudimentary example, here are the two files in `config/initializers` extracted from the `environment.rb` file above.

```
# config/initializers/mail.rb
ActionMailer::Base.delivery_method = :sendmail
ActionMailer::Base.default_charset = "utf-8"


# config/initializers/date_formats.rb
custom_date_formats = {
  :concise => "%d.%b.%y",
  :medium => "%b %e, %Y"
}
ActiveSupport::CoreExtensions::Date::Conversions::DATE_
FORMATS.merge!(custom_date_formats)
```

TIPS

If the order in which your configuration snippets are loaded is important, prefix each file in `config/initializers` with a number. The files are sorted prior to being loaded, so this will force a specific order. Without numerical file names, normal alphabetical ordering will be used to determine load sequence.

Bootstrap Initializer

The files in `config/initializers` are loaded *after* Rails' environment is loaded (including `environment.rb`). However, there's a pre-environment hook provided for you should you need to do things prior to Rails' environment getting in the mix: just create a `config/preinitializer.rb` file.

This file is loaded after the Rails classes are loaded but before the environment file and the environment configuration.

Collection Fixtures

Fixtures provide an easy way to provision test data for your various unit and functional tests. Along with fixtures comes an easy way to reference the provisioned data via the familiar `fixture_type(fixture_name)` notation:

```
# The old way
```

```
  def test_post_find
    assert_equal [posts(:rails2), posts(:peepcode)],
                 Post.find(:all)
  end
```

It's common to have to reference a series of fixture items in an array, as this example shows, by creating your own array structure. However, with Rails 2, you can now pass in a collection of symbols to the fixture method to get back the array of fixtures with those given names:

```
  def test_post_find
    assert_equal posts(:rails2, :peepcode), Post.find(:all)
  end
```

This cuts down on unnecessary fixture references and is just a more concise way of denoting the same thing.

Rake Tasks

While Rake is separate from Rails, Rails ships with a rich library of rake tasks that helps you with various development tasks. Rails 2 ships with a number of useful additions to this library.

Route Listing Task

Routing can get complex, especially with the host of named routes that Rails creates for you as part of its RESTful routing framework.

To help you figure out what routes are recognized within your application you now have a `routes` task that will list all routes along with their name, HTTP method, path and what controller and action they resolve to.

To invoke this task simply run `rake routes` to see a listing similar to this one:

```
# Indented for readability
messages           GET /messages
                       {:action=>"index",
                        :controller=>"messages"}

formatted_messages GET /messages.:format
                       {:action=>"index",
                        :controller=>"messages"}

                   POST /messages
                       {:action=>"create",
                        :controller=>"messages"}
```

The first column contains the name of the route, if it is a named route. Subsequent columns contain the HTTP method recognized for the route, the URL path and lastly, the controller and action the route resolves to.

Another debugging benefit of this output is the order in which the routes are displayed. Rails evaluates routes in order and will use the first route that a URL matches.

> The default output keeps each request on the same line, which may wrap in your terminal.

Database Provisioning Tasks

The `rake db:migrate` task is one of the more useful rake tasks that Rails provides. With Rails 2, however, we've been given additional tasks that support database lifecycle actions such as `create` and `drop`.

DB:CREATE

The `rake db:create` task creates the database specified in `config/databases.yml` for the current `RAILS_ENV`. Applying `db:create` to a non-development environment is easy as specifying `RAILS_ENV` on the command line:

```
rake db:create RAILS_ENV=test
```

This task will *not* overwrite the database if it already exists.

DB:CREATE:ALL

If you want a quick way to create all the database specified in your `databases.yml` file, then `db:create:all` is your man. It traverses all the database definitions and creates the database for each definition *if the database is local*. That is, only if the `host` value is either blank or `localhost` will the database be created. This is to ensure you don't do anything too catastrophic on a remote database.

DB:DROP AND DB:DROP:ALL

The `db:drop` and `db:drop:all` tasks work much like their `create` siblings – dropping the database for the current `RAILS_ENV` and dropping all local databases, respectively.

DB:RESET

The `db:reset` task is a great way to quickly recycle your database. It drops the database, creates the database and then migrates the db. This ensures a fresh state and it useful for making sure your migrations will execute properly in a new environment. This task only executes on the database for the existing `RAILS_ENV` and takes the same VERSION argument as `db:migrate` if you want to specify a specific version to migrate up to.

```
rake db:reset RAILS_ENV=test VERSION=23
```

DB:ROLLBACK

Rolling back to a previous version of the database schema is a pretty common occurrence when you're doing heavy development – and when you do you usually rollback one version. To assist you with this common task you can now run the `rollback` task with an optional STEP argument to specify how many versions to rollback (the default being just one version)

```
rake db:rollback STEP=2
```

DB:VERSION

Have you ever wondered what migration version your database is at? You no longer have to go fishing in the `schema_info` table now that this task is available:

```
rake db:version
Current version: 53
```

Source Code Annotations

I like to litter my source code comments with TODO annotations that I use to mark where I need to circle back around and give some more attention to. This is great, but is only as useful as it is easy to list these todos. In Rails 2 there is now a rake task that will pull out all TODO, FIXME and OPTIMIZE annotations.

The `rake notes` task lists all files that are annotated, which line they're annotated on and which annotation was used in a format similar to this:

```
app/controllers/ratings_controller.rb:
  *  11 (TODO) Fill out error handling here
app/helpers/profiles_helper.rb:
  *  18 (FIXME) This breaks unit test
```

If you're only interested in listing a specific annotation you can call the appropriate task of these finer grained commands:

```
rake notes:fixme
rake notes:optimize
rake notes:todo
```

> TextMate users can view a similar report with `ctrl-shift-T`.

script/generate Update

With Rails 1.2 you were able to run the `scaffold_resource` generator to create a REST-based application.

```
script/generate scaffold_resource model_name property:type
```

This creates a model, a controller, migration and a set of view templates that's ready to use as the starting point for your REST-based, resource-oriented application.

Rails 2 makes no apologies for its all-in attitude about RESTful development and replaces the standard `script/generate scaffold` with the `scaffold_resource` functionality from 1.2. The general attitude seems to be "Why wouldn't you want to create RESTful scaffolding?". There's not much not to like with the RESTful generation – giving you RESTful forms in the view templates and the appropriate RESTful structure in your controllers.

There was once a `scaffold` method that would dynamically generate all the CRUD methods for a controller and model. It's been removed, but third party solutions are available. See make_resourceful (http://svn.hamptoncatlin.com/make_resourceful/trunk) or ActiveScaffold (http://activescaffold.com).

Debugging

Debugging in Rails (http://wiki.rubyonrails.org/rails/pages/HowtoDebugWithBreakpoint) used to rely on `breakpointer`, which was subsequently broken (or fixed depending on your perspective) in Ruby 1.8.5. With Rails 2 we can now use ruby-debug for first class debugging. To install ruby-debug (on a Mac), do the following:

```
sudo gem install -y ruby-debug
# Place a 'debugger' statement in your code where
# you want debugging to start.
# Start the rails server script with
# the '-u' or '--debugger' option
script/server --debugger
```

Then, when you hit the application code where you've placed the `debugger` statement, you'll get a debug prompt in the terminal where you started the server. You can also drop directly into an `irb` shell by typing `irb` at the debug prompt.

Let's go through a quick example of how you would use the debugger from `script/console`.

Here's an example of a brief debugging session where I want to poke around the index action of my `posts_controller.rb` from the console:

```
trunk ryan$ script/console
```

```
Loading development environment

# Explicitly load the ruby-debug library when in the console
>> require 'ruby-debug'
=> []

# Invoke the index action of my controller
>> app.get 'posts?limit=5'

# 'debugger' statement encountered so you're automatically
# dropped into the ruby-debug shell (rdb) and the line
# you're on is printed out
/.../trunk/app/controllers/posts_controller.rb:7 @posts =
Post.find(:all, :limit => params[:limit])

# List the lines around where I am right now
(rdb:1) list
[2, 10] in /.../trunk/app/controllers/posts_controller.rb
   2
   3    resources_controller_for :posts
   4
   5    def index
   6      debugger
=> 7      @posts = Post.find(:all, :limit => params[:limit])
   8      render :xml => @posts
   9    end
   10

# Print out the value of the :limit parameter
(rdb:1) p params[:limit]
"5"

# Go to the next line
(rdb:1) next
/.../trunk/app/controllers/posts_controller.rb:8 render :xml
=> @posts

# List where I am again?
(rdb:1) list
[3, 10] in /.../trunk/app/controllers/posts_controller.rb
   3    resources_controller_for :posts
   4
   5    def index
```

A FEW DEBUGGER COMMANDS

- `app.get 'posts?limit=5'` → Get a page
- `list` → Show code around current breakpoint
- `p @variable` → Inspect a variable
- `next` → Go to next line
- `step` → Enter a method
- `up` → Leave a method
- `cont` → Exit debugger

```
   6      debugger
   7      @posts = Post.find(:all, :limit => params[:limit])
=> 8      render :xml => @posts
   9    end
   10

# Step into the render statement
(rdb:1) step
/.../trunk/vendor/rails/actionpack/lib/action_controller/
benchmarking.rb:45 unless logger

# Now where am I?
(rdb:1) list
[40, 49] in /.../trunk/vendor/rails/actionpack/lib/action_
controller/benchmarking.rb
   40         end
   41       end
   42
   43       protected
   44         def render_with_benchmark(options = nil,
deprecated_status = nil, &block)
=> 45           unless logger
   46             render_without_benchmark(options, &block)
   47           else
   48             db_runtime = ActiveRecord::Base.connection.
reset_runtime if Object.const_defined?("ActiveRecord") &&
ActiveRecord::Base.connected?
   49

# Ahhh, I'm in Rails code - get me out of here!
# Go up a frame (step out)
(rdb:1) up
#1 /.../trunk/app/controllers/posts_controller.rb:8 in
'index'

# Am I back home yet?
(rdb:1) list
[3, 10] in /.../trunk/app/controllers/posts_controller.rb
   3    resources_controller_for :posts
   4
   5    def index
   6      debugger
   7      @posts = Post.find(:all, :limit => params[:limit])
```

```
=> 8      render :xml => @posts
   9    end
   10

# Show me a little bit about the @posts collection
(rdb:1) p @posts.size
1
(rdb:1) p @posts
[#<Post id: 26, title: "Debugging with Ruby!", ...>]

# Ok, I've seen enough - let's bolt
(rdb:1) cont
=> 200

# Now I'm back in the rails console irb shell - whew!
>> exit
```

When you're running `script/server` with the `–debugger` flag, you don't have to explicitly require the `ruby-debug` library as you have to do within the console.

To debug within your tests, place a `debugger` statement in your tests and execute them with `rdebug -n`:

```
rdebug -n test/unit/post_test.rb
```

You can also explicitly require `ruby-debug` as you did in the above `console` example and run your tests as you normally would (i.e. without the help of `rdebug`).

People seem to have varying opinions on the usefulness of a debugger as opposed to a robust test suite, but it's a utility that is always nice to have in your toolbox. If you're not yet using a debugger I suspect you will quickly see its usefulness once your comfort with it increases. With this release of Rails, you now have no roadblocks to using a first-class debugger, so have at it.

Here is a list (http://blog.nanorails.com/articles/2006/07/14/a-better-rails-debugger-ruby-debug) of all the commands available to you from the ruby-debug shell

Deprecated Features

CHAPTER 4

Since Rails 1.1, the core team has marked features for deletion. You've probably seen some of these messages in your logs or in the output while running tests:

```
DEPRECATION WARNING: server_settings has been renamed
smtp_settings, this warning will be removed with rails 2.0
See http://www.rubyonrails.org/deprecation for details.
(called from /Users/topfunky/git/railsapp/config/../config/
environment.rb:59)
```

If you're as curious as I am, you tried to visit the official deprecation page (http://www.rubyonrails.org/deprecation) and found a sparse site that lists a few older issues. The easiest way to upgrade is to watch your logs and fix issues as they appear.

Indeed, these features have now been completely removed (or significantly changed) in Rails 2.

Sometimes it isn't clear how an application should be rewritten if a deprecated feature is still needed. Here are a few major deprecated features and possible replacements.

Deprecations from Previous Versions

Some syntactical changes have been encouraged since Rails 1.1. If you still have any of this code in your current application, it can be easily fixed.

@params becomes params

Form parameters, session variables, flash messages, and other variables were once accessed locally with instance variables.

```
# The wrong way
@params[:comment]
```

```
@session[:user_id_]
@flash[:notice] = "Alert!"
@request.subdomains.first
@env['PATH_INFO']
```

The proper way is to drop the at sign.

```
# The right way
params[:comment]
session[:user_id_]
flash[:notice] = "Alert!"
request.subdomains.first
env['PATH_INFO']
```

find_all becomes find(:all)

Originally, ActiveRecord provided method name shortcuts for common queries:

```
# The wrong way
Article.find_all
Article.find_first
```

These have been removed in favor of an argument.

```
# The right way
Article.find(:all)
Article.find(:first)
```

Dynamic finders still use 'all' in the method name. For example, `Article.find_all_by_title('Chunky Bacon')`.

Components

Components were once a great way to package a controller together with related view templates. Unforunately, they were proved to be slow and have been removed.

REPLACEMENT

The most common use of components is for sidebar widgets or free-standing parts of a page. The embedded_actions (http://dev.notso.net/svn/rails/plugins/embedded_actions/trunk) plugin recreates and enhances some of the functionality of components for this purpose and may be able to replace your existing component-based code.

Forms

Every Rails project once used `start_form_tag` and `end_form_tag`, thanks to the built-in scaffolding.

That has been deprecated in favor of the block-based `form_for`. The functionality is the same, but the syntax has changed.

```
<% form_for :article,
            @article,
            :url => articles_path do |f| %>
  <%= f.text_field :title %>
  <%= submit_tag "Save" %>
<% end %>
```

Major changes here are:

- The `form_for` method is inside a non-printing ERB tag.
- The form fields are wrapped in a block. The contents can be passed to a partial if the `f` variable is passed along with it as a `local`.

- Fields use the same helper names but only need to provide the field name as an argument (other arguments such as `:value` can also be passed).
- No `end_form_tag` is needed...the form automatically closes itself when the `end` is reached.

Overall, the new `form_for` communicates the some content with less code and less repetition.

REPLACEMENT

The good news is that `start_form_tag` isn't really gone...it's still there as `form_tag`! If you need to quickly convert an application to Rails 2 and don't have time to rewrite every form with a `form_tag` block, you can use this syntax:

```
<%= form_tag articles_path %>
  <%= text_field :article, :title %>
  <%= submit_tag "Save" %>
</form>
```

After your application is running and deployed, you can go back and convert your forms to use the new syntax.

Pagination Removed

If more than a few records are returned by a query, you should split the results so they can be viewed on multiple pages. The built-in pagination helper was deemed slow and has been removed. There is no longer any built-in way to split a list of records into batches.

REPLACEMENT

The will_paginate (http://rock.errtheblog.com/will_paginate) plugin uses a different syntax that you may like even better.

In your controller, call the dynamic `paginate` method:

```
@articles = Post.paginate(:page => params[:page])
```

You can also filter dynamically with conditions. To paginate only the records authored by Jim, use this syntax:

```
@author = Author.find_by_name 'jim'
@articles =
    Post.paginate_by_author_id(@author.id,
                               :page => params[:page])
```

In your view, call the helper:

```
<%= will_paginate @articles %>
```

The paging links can be styled with CSS. See the documentation site listed above for the details.

> The paginator (http://paginator.rubyforge.org) gem is another option for paging with Rails.

acts_as Plugins

One of the original appeals of Rails was the instant functionality available with `acts_as` model behaviors. In Rails 2, all `acts_as` behaviors have been removed from the core.

REPLACEMENT

Fortunately, they are all available separately. You can install them from the core plugin repository (http://svn.rubyonrails.org/rails/plugins). The functionality is the same, but you will need to explicitly install them in your application.

```
script/plugin install
    http://svn.rubyonrails.org/rails/plugins/acts_as_list
```

Other useful `acts_as` plugins:

- acts_as_versioned (http://svn.techno-weenie.net/projects/plugins/acts_as_versioned) – Archives a copy of database records when they are edited.
- acts_as_paranoid (http://svn.techno-weenie.net/projects/plugins/acts_as_paranoid) – Soft-delete so no records are ever deleted permanently.
- acts_as_state_machine (http://elitists.textdriven.com/svn/plugins/acts_as_state_machine/trunk) – Stores state with records and executes callback methods when the state changes.

Database Adapters Removed

Non-standard database adapters have been removed from the core distribution.

REPLACEMENT

If you're using a database other than MySQL, Sqlite, or PostgreSQL, you'll need to install the database adapter separately. You can download the adapter code (http://svn.rubyonrails.org/rails/adapters) or install the relevant gem:

```
sudo gem install activerecord-oracle-adapter
```

At the time of writing, these gems had not yet been published to RubyForge. However, you can install them gem from the gem repository at rubyonrails.com:

```
% sudo gem install activerecord-oracle-adapter --source
http://gems.rubyonrails.com
```

Alternately, you could checkout the source from Subversion and build the gem locally:

```
% svn co http://svn.rubyonrails.org/rails/adapters/oracle
% rake gem
% sudo gem install pkg/activerecord-oracle-adapter-1.0.0.gem
```

smtp_settings

ActionMailer needs to know your mail server domain name, username, and password. These settings have been renamed from `server_settings` to `smtp_settings`.

It's an easy fix:

```
ActionMailer::Base.smtp_settings = my_hash_of_options
```

While you're at it, make life easier on yourself by splitting this out into a YAML file. I store mine in `config/mailer.yml`.

```
---
:address: "smtp.example.com"
:port: 587
:domain: "smtp.example.com"
:user_name: "test@topfunky.com"
:password: "not-the-real-password"
:authentication: :plain
```

I read the settings into my application with this line in `config/environment.rb`:

```
yaml_contents = File.open("#{RAILS_ROOT}/config/mailer.yml")
mailer_options = YAML.load(yaml_contents)
ActionMailer::Base.smtp_settings = mailer_options
```

Renamed Routes

Nested routes automatically gain a prefix that matches the singular form of their parent's name.

This is a benefit because helper methods are more explicit, but it also forces you to use the new name in your controllers and views.

For example, here is a nested route:

```
map.resources :products do |product|
  product.resources :bugs
end
```

Previously, you would have had access to these methods:

- bugs_path → /products/:product_id/bugs
- bug_path → /products/:product_id/bugs/:id
- edit_bug_path → /products/:product_id/bugs/:id;edit

Now these same route methods include `product`.

- bugs_path → product_bugs_path
- bug_path → product_bug_path
- edit_bug_path → edit_product_bug_path

Changelogs

CHAPTER 5

For your reference, here are the changelogs from the major parts of Rails (syntax-highlighted where relevant).

ActionMailer

SVN

- Fix silent failure of rxml templates. #9879 [jstewart]

2.0.0 [PREVIEW RELEASE] (SEPTEMBER 29TH, 2007)

- Fix attachment decoding when using the TMail C extension. #7861 [orangechicken]
- Increase mail delivery test coverage. #8692 [kamal]
- Register alternative template engines using ActionMailer::Base.register_template_extension('haml'). #7534 [cwd, Josh Peek]
- Only load ActionController::UrlWriter if ActionController is present [Rick Olson]
- Make sure parsed emails recognized attachments nested inside multipart parts. #6714 [Jamis Buck]
- Allow mailer actions named send by using *send* internally. #6467 [iGEL]
- Add assert_emails and assert_no_emails to test the number of emails delivered. #6479 [Jonathan Viney]

```
# Assert total number of emails delivered:
assert_emails 0
ContactMailer.deliver_contact
assert_emails 1
```

```
# Assert number of emails delivered within a block:
assert_emails 1 do
  post :signup, :name => 'Jonathan'
end
```

ActionPack

SVN

- caches_page uses a single after_filter instead of one per action. #9891 [lifofifo]
- Update Prototype to 1.6.0_rc1 and script.aculo.us to 1.8.0 preview 0. [sam, madrobby]
- Dispatcher: fix that to_prepare should only run once in production. #9889 [Nathaniel Talbott]
- Memcached sessions: add session data on initialization; don't silently discard exceptions; add unit tests. #9823 [kamk]
- error_messages_for also takes :message and :header_message options which defaults to the old "There were problems with the following fields:" and "<count> errors prohibited this <object_name> from being saved". #8270 [rmm5t, zach-inglis-lt3]
- Make sure that custom inflections are picked up by map.resources. #9815 [mislav]
- Changed SanitizeHelper#sanitize to only allow the custom attributes and tags when specified in the call [DHH]
- Extracted sanitization methods from TextHelper to SanitizeHelper [DHH]
- rescue_from accepts :with => lambda { |exception| ... } or a normal block. #9827 [lifofifo]
- Add :status to redirect_to allowing users to choose their own

response code without manually setting headers. #8297 [codahale, chasgrundy]

- Add link_to :back which uses your referrer with a fallback to a javascript link. #7366 [eventualbuddha, tarmo]
- error_messages_for and friends also work with local variables. #9699 [Frederick Cheung]
- Fix url_for, redirect_to, etc. with :controller => :symbol instead of 'string'. #8562, #9525 [Justin Lynn, Tarmo Tanav, shoe]
- Use #require_library_or_gem to load the memcache library for the MemCache session and fragment cache stores. Closes #8662. [Rick]
- Move ActionController::Routing.optimise_named_routes to ActionController::Base.optimise_named_routes. Now you can set it in the config. [Rick]

```
config.action_controller.optimise_named_routes = false
```

- ActionController::Routing::DynamicSegment#interpolation_chunk should call #to_s on all values before calling URI.escape. [Rick]
- Only accept session ids from cookies, prevents session fixation attacks. [bradediger]

2.0.0 [Preview Release] (September 29th, 2007)
[Includes duplicates of changes from 1.12.2 – 1.13.3]

- Fixed that render template did not honor exempt_from_layout #9698 [pezra]
- Better error messages if you leave out the :secret option for request forgery protection. Closes #9670 [rick]
- Allow ability to disable request forgery protection, disable it in test mode by default. Closes #9693 [lifofifo]
- Avoid calling is_missing on LoadErrors. Closes #7460. [ntalbott]

- Move Railties' Dispatcher to ActionController::Dispatcher, introduce before_ and after_dispatch callbacks, and warm up to non-CGI requests. [Jeremy Kemper]
- The tag helper may bypass escaping. [Jeremy Kemper]
- Cache asset ids. [Jeremy Kemper]
- Optimized named routes respect AbstractRequest.relative_url_root. #9612 [danielmorrison, Jeremy Kemper]
- Introduce ActionController::Base.rescue_from to declare exception-handling methods. Cleaner style than the case-heavy rescue_action_in_public. #9449 [norbert]
- Rename some RequestForgeryProtection methods. The class method is now #protect_from_forgery, and the default parameter is now 'authenticity_token'. [Rick]
- Merge csrf_killer plugin into rails. Adds RequestForgeryProtection model that verifies session-specific _tokens for non-GET requests. [Rick]
- Secure #sanitize, #strip_tags, and #strip_links helpers against xss attacks. Closes #8877. [Rick, lifofifo, Jacques Distler] This merges and renames the popular white_list helper (along with some css sanitizing from Jacques Distler version of the same plugin). Also applied updated versions of #strip_tags and #strip_links from #8877.
- Remove use of & logic operator. Closes #8114. [watson]
- Fixed JavaScriptHelper#escape_javascript to also escape closing tags #8023 [rubyruy]
- Fixed TextHelper#word_wrap for multiline strings with extra carrier returns #8663 [seth]
- Fixed that setting the :host option in url_for would automatically turn off :only_path (since :host would otherwise not be shown) #9586 [Bounga]

- Added FormHelper#label #8641 [jcoglan]
- Added AtomFeedHelper (slightly improved from the atom_feed_helper plugin) [DHH]
- Prevent errors when generating routes for uncountable resources, (i.e. sheep where plural == singluar). map.resources :sheep now creates sheep_index_url for the collection and sheep_url for the specific item. [Koz]
- Added support for HTTP Only cookies (works in IE6+ and FF 2.0.5+) as an improvement for XSS attacks #8895 [lifo/Spakman]
- Don't warn when a path segment precedes a required segment. Closes #9615. [Nicholas Seckar]
- Fixed CaptureHelper#content_for to work with the optional content parameter instead of just the block #9434 [sandofsky/wildchild].
- Added Mime::Type.register_alias for dealing with different formats using the same mime type [DHH]. Example:

```
class PostsController < ApplicationController
  before_filter :adjust_format_for_iphone

  def index
    @posts = Post.find(:all)

    respond_to do |format|
      format.html
      # => renders index.html.erb and uses "text/html"
      #    as the content type
      format.iphone
      # => renders index.iphone.erb and uses "text/html"
      #    as the content type
    end
  end


  private

  def adjust_format_for_iphone
    if request.env["HTTP_USER_AGENT"] && request.env["HTTP_
USER_AGENT"][/iPhone/]
      request.format = :iphone
    end
  end
end
```

- Added that render :json will automatically call .to_json unless it's being passed a string [DHH].
- Autolink behaves well with emails embedded in URLs. #7313 [Jeremy McAnally, tarmo]
- Fixed that default layouts did not take the format into account #9564 [lifofifo]
- Fixed optimized route segment escaping. #9562 [wildchild, Jeremy Kemper]
- Added block acceptance to JavaScriptHelper#javascript_tag. #7527 [Bob Silva, Tarmo Tanav, rmm5t]
- root_path returns '/' not ''. #9563 [lifofifo]
- Fixed that setting request.format should also affect respond_to blocks [DHH]
- Add option to force binary mode on tempfile used for fixture_file_upload. #6380 [Jonathan Viney]
- Fixed that resource namespaces wouldn't stick to all nested resources #9399 [pixeltrix]
- Moved ActionController::Macros::AutoComplete into the auto_complete plugin on the official Rails svn #9512 [lifofifo]
- Moved ActionController::Macros::InPlaceEditing into the in_place_editor plugin on the official Rails svn #9513 [lifofifo]
- Removed deprecated form of calling xml_http_request/xhr without the first argument being the http verb [DHH]
- Removed deprecated methods [DHH]:

```
* ActionController::Base#keep_flash (use flash.keep instead)
* ActionController::Base#expire_matched_fragments (just call
expire_fragment with a regular expression)
* ActionController::Base.template_root/= methods (use
ActionController#Base.view_paths/= instead)
* ActionController::Base.cookie (use ActionController#Base.
cookies[]= instead)
```

- Removed the deprecated behavior of appending ".png" to image_tag/image_path calls without an existing extension [DHH]
- Removed ActionController::Base.scaffold—it went through the whole idea of scaffolding (card board walls you remove and tweak one by one). Use the scaffold generator instead (it does resources too now!) [DHH]
- Optimise named route generation when using positional arguments. [Koz] This change delivers significant performance benefits for the most common usage scenarios for modern rails applications by avoiding the costly trip through url_for. Initial benchmarks indicate this is between 6 and 20 times as fast.
- Explicitly require active_record/query_cache before using it. [Jeremy Kemper]
- Fix layout overriding response status. #9476 [lotswholetime]
- Add field_set_tag for generating field_sets, closes #9477. [djanowski]
- Allow additional parameters to be passed to named route helpers when using positional arguments. Closes #8930 [ian.w.white@gmail.com]
- Make render :partial work with a :collection of Hashes, previously this wasn't possible due to backwards compatibility restrictions. [lifofifo]
- request.host works with IPv6 addresses. #9458 [yuya]
- Fix bug where action caching sets the content type to the Action-

CachePath object. Closes #9282 [mindforge]

- Find layouts even if they're not in the first view_paths directory. Closes #9258 [caio]
- Major improvement to the documentation for the options / select form helpers. Closes #9038 [kampers, jardeon, wesg]
- Fix number_to_human_size when using different precisions. Closes #7536. [RichardStrand, mpalmer]
- Added partial layouts (see example in action_view/lib/partials.rb) [DHH]
- Allow you to set custom :conditions on resource routes. [Rick]
- Fixed that file.content_type for uploaded files would include a trailing \r #9053 [bgreenlee]
- url_for now accepts a series of symbols representing the namespace of the record [Josh Knowles]
- Make :trailing_slash work with query parameters for url_for. Closes #4004 [nov]
- Make sure missing template exceptions actually say which template they were looking for. Closes #8683 [dasil003]
- Fix errors with around_filters which do not yield, restore 1.1 behaviour with after filters. Closes #8891 [skaes] After filters will **no longer** be run if an around_filter fails to yield, users relying on this behaviour are advised to put the code in question after a yield statement in an around filter.
- Allow you to delete cookies with options. Closes #3685 [josh, Chris Wanstrath]
- Allow you to render views with periods in the name. Closes #8076 [norbert]

```
render :partial => 'show.html.erb'
```

- Improve capture helper documentation. #8796 [kampers]
- Prefix nested resource named routes with their action name, e.g. new_group_user_path(@group) instead of group_new_user_path(@group). The old nested action named route is deprecated in Rails 1.2.4. #8558 [David Chelimsky]
- Allow sweepers to be created solely for expiring after controller actions, not model changes [DHH]
- Added assigns method to ActionController::Caching::Sweeper to easily access instance variables on the controller [DHH]
- Give the legacy X-POST_DATA_FORMAT header greater precedence during params parsing for backward compatibility. [Jeremy Kemper]
- Fixed that link_to with an href of # when using :method will not allow for click-through without JavaScript #7037 [stevenbristol/josh]
- Fixed that radio_button_tag should generate unique ids #3353 [BobSilva/rebecca/josh]
- Fixed that HTTP authentication should work if the header is called REDIRECT_X_HTTP_AUTHORIZATION as well #6754 [mislaw]
- Don't mistakenly interpret the request uri as the query string. #8731 [lifofifo, Jeremy Kemper]
- Make ActionView#view_paths an attr_accessor for real this time. Also, don't perform an unnecessary #compact on the @view_paths array in #initialize. Closes #8582 [dasil003, julik, rick]
- Tolerate missing content type on multipart file uploads. Fix for Safari 3. [Jeremy Kemper]
- Deprecation: remove pagination. Install the classic_pagination plugin for forward compatibility, or move to the superior will_paginate plugin. #8157 [Josh Peek]
- Action caching is limited to GET requests returning 200 OK status.

#3335 [tom@craz8.com, halfbyte, Dan Kubb, Josh Peek]

- Improve Text Helper test coverage. #7274 [Rob Sanheim, Josh Peek]
- Improve helper test coverage. #7208, #7212, #7215, #7233, #7234, #7235, #7236, #7237, #7238, #7241, #7243, #7244 [Rich Collins, Josh Peek]
- Improve UrlRewriter tests. #7207 [Rich Collins]
- Resources: url_for([parent, child]) generates /parents/1/children/2 for the nested resource. Likewise with the other simply helpful methods like form_for and link_to. #6432 [mhw, Jonathan Vaught, lotswholetime]
- Assume html format when rendering partials in RJS. #8076 [Rick]
- Don't double-escape url_for in views. #8144 [Rich Collins, Josh Peek]
- Allow JSON-style values for the :with option of observe_field. Closes #8557 [kommen]
- Remove RAILS_ROOT from backtrace paths. #8540 [Tim Pope]
- Routing: map.resource :logo routes to LogosController so the controller may be reused for multiple nestings or namespaces. [Jeremy Kemper]
- render :partial recognizes Active Record associations as Arrays. #8538 [kamal]
- Routing: drop semicolon and comma as route separators. [Jeremy Kemper]
- request.remote_ip understands X-Forwarded-For addresses with nonstandard whitespace. #7386 [moses]
- Don't prepare response when rendering a component. #8493 [jsierles]
- Reduce file stat calls when checking for template changes. #7736

[alex]

- Added custom path cache_page/expire_page parameters in addition to the options hashes [DHH]. Example:

```
def index
  caches_page(response.body, "/index.html")
end
```

- Action Caching speedup. #8231 [skaes]
- Wordsmith resources documentation. #8484 [marclove]
- Fix syntax error in code example for routing documentation. #8377. [norbert]
- Routing: respond with 405 Method Not Allowed status when the route path matches but the HTTP method does not. #6953 [Josh Peek, defeated, Dan Kubb, Coda Hale]
- Add support for assert_select_rjs with :show and :hide. #7780 [dchelimsky]
- Make assert_select's failure messages clearer about what failed. #7779 [dchelimsky]
- Introduce a default respond_to block for custom types. #8174 [Josh Peek]
- auto_complete_field takes a :method option so you can GET or POST. #8120 [zapnap]
- Added option to suppress :size when using :maxlength for FormTagHelper#text_field #3112 [rails@tpope.info]
- catch possible WSOD when trying to render a missing partial. Closes #8454 [Catfish]
- Rewind request body after reading it, if possible. #8438 [s450r1]
- Resource namespaces are inherited by their has_many subresources. #8280 [marclove, ggarside]

- Fix filtered parameter logging with nil parameter values. #8422 [choonkeat]
- Integration tests: alias xhr to xml_http_request and add a request_method argument instead of always using POST. #7124 [Nik Wakelin, Francois Beausoleil, Wizard]
- Document caches_action. #5419 [Jarkko Laine]
- Update to Prototype 1.5.1. [Sam Stephenson]
- Allow routes to be declared under namespaces [Tobias Luetke]:

```
map.namespace :admin do |admin|
  admin.root :controller => "products"
  admin.feed 'feed.xml',
             :controller => 'products',
             :action => 'feed',
             :format => 'xml'
end
```

- Update to script.aculo.us 1.7.1_beta3. [Thomas Fuchs]
- observe_form always sends the serialized form. #5271 [manfred, normelton@gmail.com]
- Parse url-encoded and multipart requests ourselves instead of delegating to CGI. [Jeremy Kemper]
- select :include_blank option can be set to a string instead of true, which just uses an empty string. #7664 [Wizard]
- Added url_for usage on render :location, which allows for record identification [DHH]. Example:

```
# ...expands the location to person_url(person).
render :xml => person,
       :status => :created,
       :location => person
```

- Introduce the request.body stream. Lazy-read to parse parameters rather than always setting RAW_POST_DATA. Reduces the memory footprint of large binary PUT requests. [Jeremy Kemper]

- Add some performance enhancements to ActionView. Cache base_paths, Cache template extensions, Remove unnecessary rescues.

- Assume that rendered partials go by the HTML format by default

```
def my_partial
  render :update do |page|
    # in this order
    # _foo.html.erb
    # _foo.erb
    # _foo.rhtml
    page.replace :foo, :partial => 'foo'
  end
end
```

- Added record identifications to FormHelper#form_for and PrototypeHelper#remote_form_for [DHH]. Examples:

```
<% form_for(@post) do |f| %>
  ...
<% end %>

<%# This will expand to be the same as: -%>

<% form_for :post,
            @post,
            :url => post_path(@post),
            :html => {
                :method => :put,
                :class => "edit_post",
                :id => "edit_post_45"
            } do |f| %>
  ...
<% end %>

<%# And for new records: -%>

<% form_for(Post.new) do |f| %>
```

```
  ...
<% end %>

<%# This will expand to be the same as: -%>

<% form_for :post,
            @post,
            :url => posts_path,
            :html => {
                :class => "new_post",
                :id => "new_post"
            } do |f| %>
  ...
<% end %>
```

- Rationalize route path escaping according to RFC 2396 section 3.3. #7544, #8307. [Jeremy Kemper, chrisroos, begemot, jugend]
- Added record identification with polymorphic routes for ActionController::Base#url_for and ActionView::Base#url_for [DHH]. Examples:

```
redirect_to(post)
# => redirect_to(posts_url(post))
# => Location: http://example.com/posts/1

link_to(post.title, post)
# => link_to(post.title, posts_url(post))
# => <a href="/posts/1">Hello world</a>

# Any method that calls url_for on its parameters
# will automatically benefit from this.
```

- Removed deprecated parameters_for_method_reference concept (legacy from before named routes) [DHH]
- Add ActionController::Routing::Helpers, a module to contain common URL helpers such as polymorphic_url. [Nicholas Seckar]
- Included the HttpAuthentication plugin as part of core (ActionCont

roller::HttpAuthentication::Basic) [DHH]

- Modernize documentation for form helpers. [jeremymcanally]
- Add brief introduction to REST to the resources documentation. [fearoffish]
- Fix various documentation typos throughout ActionPack. [Henrik N]
- Enhance documentation and add examples for url_for. [jeremymcanally]
- Fix documentation typo in routes. [norbert, pam]
- Sweep flash when filter chain is halted. [Caio Chassot <lists@v2studio.com>]
- Fixed that content_tag with a block will just return the result instead of concate it if not used in a ERb view #7857, #7432 [michael.niessner]
- Replace the current block/continuation filter chain handling by an implementation based on a simple loop. #8226 [Stefan Kaes]
- Update UrlWriter to accept :anchor parameter. Closes #6771. [octopod]
- Added RecordTagHelper for using RecordIdentifier conventions on divs and other container elements [DHH]. Example:

```
<% div_for(post) do %>
  <%= post.body %>
<% end %>

<%# Becomes... -%>

<div id="post_45" class="post">
  What a wonderful world!
</div>
```

- Added page[record] accessor to JavaScriptGenerator that relies on

RecordIdentifier to find the right dom id [DHH]. Example:

```
format.js do
  # Calls: new Effect.fade('post_45');
  render(:update) { |page|
    page[post].visual_effect(:fade)
  }
end
```

- Added RecordIdentifier to enforce view conventions on records for dom ids, classes, and partial paths [DHH]
- Added map.namespace to deal with the common situation of admin sections and the like [DHH]

```
# Before:

map.resources :products,
              :path_prefix => "admin",
              :controller => "admin/products",
              :collection => { :inventory => :get },
              :member => { :duplicate => :post }

map.resources :tags,
              :name_prefix => 'admin_product_',
              :path_prefix => "admin/products/:product_id",
              :controller => "admin/product_tags"

map.resources :images,
              :name_prefix => 'admin_product_',
              :path_prefix => "admin/products/:product_id",
              :controller => "admin/product_images"

map.resources :variants,
              :name_prefix => 'admin_product_',
              :path_prefix => "admin/products/:product_id",
              :controller => "admin/product_variants"

# After:

map.namespace(:admin) do |admin|
  admin.resources :products,
    :collection => { :inventory => :get },
    :member     => { :duplicate => :post },
    :has_many   => [ :tags, :images, :variants ]
end
```

- Added :name_prefix as standard for nested resources [DHH]. WARNING: May be backwards incompatible with your app

```
# Before:

map.resources :emails do |emails|
  emails.resources :comments,
                   :name_prefix => "email_"

  emails.resources :attachments,
                   :name_prefix => "email_"
end

# After:

map.resources :emails do |emails|
  emails.resources :comments
  emails.resources :attachments
end

# This does mean that if you intended to have
# comments_url go to /emails/5/comments,
# then you'll have to set :name_prefix to nil explicitly.
```

- Added :has_many and :has_one for declaring plural and singular resources beneath the current [DHH]

```
# Before:

map.resources :notes do |notes|
  notes.resources :comments
  notes.resources :attachments
  notes.resource :author
end

# After:
```

```
map.resources :notes,
              :has_many => [ :comments, :attachments ],
              :has_one => :author
```

- Added that render :xml will try to call to_xml if it can [DHH]. Makes these work:

```
render :xml => post
render :xml => comments
```

- Added :location option to render so that the common pattern of rendering a response after creating a new resource is now a 1-liner [DHH]

```
render :xml => post.to_xml,
       :status => :created,
       :location => post_url(post)
```

- Ensure that render_text only adds string content to the body of the response [DHH]
- Return the string representation from an Xml Builder when rendering a partial. Closes #5044 [tpope]
- Fixed that parameters from XML should also be presented in a hash with indifferent access [DHH]
- Tweak template format rules so that the ACCEPT header is only used if it's text/javascript. This is so ajax actions without a :format param get recognized as Mime::JS. [Rick]
- The default respond_to blocks don't set a specific extension anymore, so that both 'show.rjs' and 'show.js.rjs' will work. [Rick]
- Allow layouts with extension of .html.erb. Closes #8032 [Josh Knowles]
- Change default respond_to templates for xml and rjs formats.

[Rick] Default xml template goes from #{action_name}.rxml => #{action_name}.xml.builder. Default rjs template goes from #{action_name}.rjs => #{action_name}.js.rjs. You can still specify your old templates:

```
respond_to do |format|
  format.xml do
    render :action => "#{action_name}.rxml"
  end
end
```

- Fix WSOD due to modification of a formatted template extension so that requests to templates like 'foo.html.erb' fail on the second hit. [Rick]
- Fix WSOD when template compilation fails [Rick]
- Change ActionView template defaults. Look for templates using the request format first, such as "show.html.erb" or "show.xml.builder", before looking for the old defaults like "show.erb" or "show.builder" [Rick]
- Highlight helper highlights one or many terms in a single pass. [Jeremy Kemper]
- Dropped the use of ; as a separator of non-crud actions on resources and went back to the vanilla slash. It was a neat idea, but lots of the non-crud actions turned out not to be RPC (as the ; was primarily intended to discourage), but legitimate sub-resources, like /parties/recent, which didn't deserve the uglification of /parties;recent. Further more, the semicolon caused issues with caching and HTTP authentication in Safari. Just Not Worth It [DHH]
- Added that FormTagHelper#submit_tag will return to its original state if the submit fails and you're using :disable_with [DHH]
- Cleaned up, corrected, and mildly expanded ActionPack documentation. Closes #7190 [jeremymcanally]

- Small collection of ActionController documentation cleanups. Closes #7319 [jeremymcanally]
- Make sure the route expiry hash is constructed by comparing the to_param-ized values of each hash. [Jamis Buck]
- Allow configuration of the default action cache path for #caches_action calls. [Rick Olson]

```
class ListsController < ApplicationController
  caches_action :index, :cache_path => Proc.new {
|controller|
    controller.params[:user_id] ?
      controller.send(:user_lists_url, c.params[:user_id]) :
      controller.send(:lists_url) }
end
```

- Performance: patch cgi/session/pstore to require digest/md5 once rather than per #initialize. #7583 [Stefan Kaes]
- Cookie session store: ensure that new sessions doesn't reuse data from a deleted session in the same request. [Jeremy Kemper]
- Deprecation: verification with :redirect_to => :named_route shouldn't be deprecated. #7525 [Justin French]
- Cookie session store: raise ArgumentError when :session_key is blank. [Jeremy Kemper]
- Deprecation: remove deprecated request, redirect, and dependency methods. Remove deprecated instance variables. Remove deprecated url_for(:symbol, *args) and redirect_to(:symbol, *args) in favor of named routes. Remove uses_component_template_root for toplevel components directory. Privatize deprecated render_partial and render_partial_collection view methods. Remove deprecated link_to_image, link_image_to, update_element_function, start_form_tag, and end_form_tag helper methods. Remove deprecated human_size helper alias. [Jeremy Kemper]
- Consistent public/protected/private visibility for chained methods.

#7813 [Dan Manges]

- Prefer MIME constants to strings. #7707 [Dan Kubb]
- Allow array and hash query parameters. Array route parameters are converted/to/a/path as before. #6765, #7047, #7462 [bgipsy, Jeremy McAnally, Dan Kubb, brendan]
- Add a #dbman attr_reader for CGI::Session and make CGI::Session::CookieStore#generate_digest public so it's easy to generate digests using the cookie store's secret. [Rick]
- Added Request#url that returns the complete URL used for the request [DHH]
- Extract dynamic scaffolding into a plugin. #7700 [Josh Peek]
- Added user/password options for url_for to add http authentication in a URL [DHH]
- Fixed that FormTagHelper#text_area_tag should disregard :size option if it's not a string [Brendon Davidson]
- Set the original button value in an attribute of the button when using the :disable_with key with submit_tag, so that the original can be restored later. [Jamis Buck]
- session_enabled? works with session :off. #6680 [Catfish]
- Added :port and :host handling to UrlRewriter (which unified url_for usage, regardless of whether it's called in view or controller) #7616 [alancfrancis]
- Allow send_file/send_data to use a registered mime type as the :type parameter #7620 [jonathan]
- Allow routing requirements on map.resource(s) #7633 [quixoten]. Example:

```
map.resources :network_interfaces,
              :requirements => {
                :id => /^\d+\.\d+\.\d+\.\d+$/
              }
```

- Cookie session store: empty and unchanged sessions don't write a cookie. [Jeremy Kemper]

- Added helper(:all) as a way to include all helpers from app/helpers/*/.rb in ApplicationController [DHH]

- Integration tests: introduce methods for other HTTP methods. #6353 [caboose]

- Routing: better support for escaped values in route segments. #7544 [Chris Roos]

- Introduce a cookie-based session store as the Rails default. Sessions typically contain at most a user_id and flash message; both fit within the 4K cookie size limit. A secure message digest is included with the cookie to ensure data integrity (a user cannot alter his user_id without knowing the secret key included in the digest). If you have more than 4K of session data or don't want your data to be visible to the user, pick another session store. Cookie-based sessions are dramatically faster than the alternatives.   [Jeremy Kemper] Example `config/environment.rb`:

```
# Use an application-wide secret key
# and the default SHA1 message digest.
config.action_controller.session = {
    :secret => "can't touch this"
  }

# Store a secret key per user
# and employ a stronger message digest.
config.action_controller.session = {
  :digest => 'SHA512',
  :secret => Proc.new { User.current.secret_key }
}
```

- Added .erb and .builder as preferred aliases to the now deprecated .rhtml and .rxml extensions [Chad Fowler]. This is done to separate the renderer from the mime type. .erb templates are often used to render emails, atom, csv, whatever. So labeling them .rhtml doesn't make too much sense. The same goes for .rxml, which can be used to build everything from HTML to Atom to whatever. .rhtml and .rxml will continue to work until Rails 3.0, though. So this is a slow phasing out. All generators and examples will start using the new aliases, though.

- Added caching option to AssetTagHelper#stylesheet_link_tag and AssetTagHelper#javascript_include_tag [DHH]. Examples:

```
stylesheet_link_tag :all, :cache => true
# when ActionController::Base.perform_caching is false...
# <link href="/stylesheets/style1.css" ...
# <link href="/stylesheets/styleB.css" ...
# <link href="/stylesheets/styleX2.css" ...

stylesheet_link_tag :all, :cache => true
# when ActionController::Base.perform_caching is true...
# <link href="/stylesheets/all.css" ...

# ...when caching is on, all.css is the concatenation
# of style1.css, styleB.css, and styleX2.css.
# Same deal for JavaScripts.
```

- Work around the two connection per host browser limit: use asset%d.myapp.com to distribute asset requests among asset[0123].myapp.com. Use a DNS wildcard or CNAMEs to map these hosts to your asset server. See http://www.die.net/musings/page_load_time/ for background. [Jeremy Kemper]
- Added default mime type for CSS (Mime::CSS) [DHH]
- Added that rendering will automatically insert the etag header on 200 OK responses. The etag is calculated using MD5 of the response body. If a request comes in that has a matching etag, the response will be changed to a 304 Not Modified and the response body will be set to an empty string. [DHH]
- Added X-Runtime to all responses with the request run time [DHH]

- Add Mime::Type convenience methods to check the current mime type. [Rick]

```
request.format.html? # => true if Mime::HTML
request.format.jpg?  # => true if Mime::JPG
# ActionController sample usage:
# the session will be disabled for non html/ajax requests
session :off, :if => Proc.new { |req| !(req.format.html? ||
req.format.js?) }
```

- Performance: patch cgi/session to require digest/md5 once rather than per #create_new_id. [Stefan Kaes]
- Add a :url_based_filename => true option to ActionController::Streaming::send_file, which allows URL-based filenames. [Thomas Fuchs]
- Fix that FormTagHelper#submit_tag using :disable_with should trigger the onsubmit handler of its form if available [DHH]
- Fix #render_file so that TemplateError is called with the correct params and you don't get the WSOD. [Rick]
- Fix issue with deprecation messing up #template_root= usage. Add #prepend_view_path and #append_view_path to allow modification of a copy of the
- superclass' view_paths. [Rick]
- Allow Controllers to have multiple view_paths instead of a single template_root. Closes #2754 [John Long]
- Add much-needed html-scanner tests. Fixed CDATA parsing bug. [Rick]
- improve error message for Routing for named routes. Closes #7346 [Rob Sanheim]
- Added enhanced docs to routing assertions. Closes #7359 [Rob Sanheim]
- fix form_for example in ActionController::Resources documentation.

Closes #7362 [gnarg]

- Make sure that the string returned by TextHelper#truncate is actually a string, not a char proxy—that should only be used internally while working on a multibyte-safe way of truncating [DHH]

- Added FormBuilder#submit as a delegate for FormTagHelper#submit_tag [DHH]

- Allow Routes to generate all urls for a set of options by specifying :generate_all => true. Allows caching to properly set or expire all paths for a resource. References #1739. [Nicholas Seckar]

- Change the query parser to map empty GET params to "" rather than nil. Closes #5694. [Nicholas Seckar]

- date_select and datetime_select take a :default option. #7052 [nik.wakelin]

```
date_select "post",
            "written_on",
            :default => 3.days.from_now
date_select "credit_card",
            "bill_due",
            :default => { :day => 20 }
```

- select :multiple => true suffixes the attribute name with [] unless already suffixed. #6977 [nik.kakelin, ben, julik]

- Improve routes documentation. #7095 [zackchandler]

- mail_to :encode => 'hex' also encodes the mailto: part of the href attribute as well as the linked email when no name is given. #2061 [Jarkko Laine, pfc.pille@gmx.net]

- Resource member routes require :id, eliminating the ambiguous overlap with collection routes. #7229 [dkubb]

- Remove deprecated assertions. [Jeremy Kemper]

- Change session restoration to allow namespaced models to be autoloaded. Closes #6348. [Nicholas Seckar]

- Fix doubly appearing parameters due to string and symbol mix-ups. Closes #2551. [aeden]
- Fix overly greedy rescues when loading helpers. Fixes #6268. [Nicholas Seckar]
- Fixed NumberHelper#number_with_delimiter to use "." always for splitting the original number, not the delimiter parameter #7389 [ceefour]
- Autolinking recognizes trailing and embedded . , : ; #7354 [Jarkko Laine]
- Make TextHelper::auto_link recognize URLs with colons in path correctly, fixes #7268. [imajes]
- Update to script.aculo.us 1.7.0. [Thomas Fuchs]
- Modernize cookie testing code, and increase coverage (Heckle++) #7101 [Kevin Clark]
- Improve Test Coverage for ActionController::Routing::Route#matches_controller_and_action? (Heckle++) #7115 [Kevin Clark]
- Heckling ActionController::Resources::Resource revealed that set_prefixes didn't break when :name_prefix was munged. #7081 [Kevin Clark]
- Fix #distance_of_time_in_words to report accurately against the Duration class. #7114 [eventualbuddha]
- Refactor #form_tag to allow easy extending. [Rick]
- Update to Prototype 1.5.0. [Sam Stephenson]
- RecordInvalid, RecordNotSaved => 422 Unprocessable Entity, StaleObjectError => 409 Conflict. #7097 [dkubb]
- Allow fields_for to be nested inside form_for, so that the name and id get properly constructed [Jamis Buck]
- Allow inGroupsOf and eachSlice to be called through rjs. #7046 [Cody Fauser]

- Allow exempt_from_layout :rhtml. #6742, #7026 [Dan Manges, Squeegy]
- Recognize the .txt extension as Mime::TEXT [Rick]
- Fix parsing of array[] CGI parameters so extra empty values aren't included. #6252 [Nicholas Seckar, aiwilliams, brentrowland]
- link_to_unless_current works with full URLs as well as paths. #6891 [Jarkko Laine, manfred, idrifter]
- Lookup the mime type for #auto_discovery_link_tag in the Mime::Type class. Closes #6941 [Josh Peek]
- Fix bug where nested resources ignore a parent singleton parent's path prefix. Closes #6940 [Dan Kubb]
- Fix no method error with error_messages_on. Closes #6935 [nik.wakelin Koz]
- Slight doc tweak to the ActionView::Helpers::PrototypeHelper#replace docs. Closes #6922 [Steven Bristol]
- Slight doc tweak to #prepend_filter. Closes #6493 [Jeremy Voorhis]
- Add more extensive documentation to the AssetTagHelper. Closes #6452 [Bob Silva]
- Clean up multiple calls to #stringify_keys in TagHelper, add better documentation and testing for TagHelper. Closes #6394 [Bob Silva]
- [DOCS] fix reference to ActionController::Macros::AutoComplete for #text_field_with_auto_complete. Closes #2578 [Jan Prill]
- Make sure html_document is reset between integration test requests. [ctm]
- Set session to an empty hash if :new_session => false and no session cookie or param is present. CGI::Session was raising an unrescued ArgumentError. [Josh Susser]

- Routing uses URI escaping for path components and CGI escaping for query parameters. [darix, Jeremy Kemper]
- Fix assert_redirected_to bug where redirecting from a nested to to a top-level controller incorrectly added the current controller's nesting. Closes #6128. [Rick Olson]
- Singleton resources: POST /singleton => create, GET /singleton/new => new. [Jeremy Kemper]
- Use 400 Bad Request status for unrescued ActiveRecord::RecordInvalid exceptions. [Jeremy Kemper]
- Silence log_error deprecation warnings from inspecting deprecated instance variables. [Nate Wiger]
- Only cache GET requests with a 200 OK response. #6514, #6743 [RSL, anamba]
- Add a 'referer' attribute to TestRequest. [Jamis Buck]
- Ensure render :json => ... skips the layout. Closes #6808 [Josh Peek]
- Fix HTML::Node to output double quotes instead of single quotes. Closes #6845 [mitreandy]
- Correctly report which filter halted the chain. #6699 [Martin Emde]
- Fix a bug in Routing where a parameter taken from the path of the current request could not be used as a query parameter for the next. Closes #6752. [Nicholas Seckar]
- Unrescued ActiveRecord::RecordNotFound responds with 404 instead of 500. [Jeremy Kemper]
- Improved auto_link to match more valid urls correctly [Tobias Luetke]
- Add singleton resources. [Rick Olson]

```
map.resource :account
```

```
# GET /account
# GET /account;edit
# UPDATE /account
# DELETE /account
```

- respond_to recognizes JSON. render :json => @person.to_json automatically sets the content type and takes a :callback option to specify a client-side function to call using the rendered JSON as an argument. #4185 [Scott Raymond, eventualbuddha]

```
# application/json response
# with body 'Element.show({:name: "David"})'
respond_to do |format|
  format.json {
    render :json => { :name => "David" }.to_json,
           :callback => 'Element.show' }
end
```

- Makes :discard_year work without breaking multi-attribute parsing in AR. #1260, #3800 [sean@ardismg.com, jmartin@desertflood.com, stephen@touset.org, Bob Silva]
- Adds html id attribute to date helper elements. #1050, #1382 [mortonda@dgrmm.net, David North, Bob Silva]
- Add :index and @auto_index capability to model driven date/time selects. #847, #2655 [moriq, Doug Fales, Bob Silva]
- Add :order to datetime_select, select_datetime, and select_date. #1427 [Timothee Peignier, patrick@lenz.sh, Bob Silva]
- Added time_select to work with time values in models. Update scaffolding. #2489, #2833 [Justin Palmer, Andre Caum, Bob Silva]
- Added :include_seconds to select_datetime, datetime_select and time_select. #2998 [csn, Bob Silva]
- All date/datetime selects can now accept an array of month

names with :use_month_names. Allows for localization. #363 [tomasj, Bob Silva]

- Adds :time_separator to select_time and :date_separator to select_datetime. Preserves BC. #3811 [Bob Silva]
- Added map.root as an alias for map.connect '' [DHH]
- Added Request#format to return the format used for the request as a mime type. If no format is specified, the first Request#accepts type is used. This means you can stop using respond_to for anything else than responses [DHH]. Examples:

```
GET /posts/5.xml   | request.format => Mime::XML
GET /posts/5.xhtml | request.format => Mime::HTML
GET /posts/5       | request.format => request.accepts.first
                     (usually Mime::HTML for browsers)
```

- Added the option for extension aliases to mime type registration [DHH]. Example (already in the default routes):

```
Mime::Type.register "text/html",
                    :html,
                    %w( application/xhtml+xml ),
                    %w( xhtml )
# ...will respond on both .html and .xhtml.
```

- @response.redirect_url works with 201 Created responses: just return headers['Location'] rather than checking the response status. [Jeremy Kemper]
- Added CSV to Mime::SET so that respond_to csv will work [Cody Fauser]
- Fixed that HEAD should return the proper Content-Length header (that is, actually use @body.size, not just 0) [DHH]
- Added GET-masquarading for HEAD, so request.method will return :get even for HEADs. This will help anyone relying on case request.method to automatically work with HEAD and map.resources will

also allow HEADs to all GET actions. Rails automatically throws away the response content in a reply to HEAD, so you don't even need to worry about that. If you, for whatever reason, still need to distinguish between GET and HEAD in some edge case, you can use Request#head? and even Request.headers["REQUEST_METHOD"] for get the "real" answer. Closes #6694 [DHH]

- Update Routing to complain when :controller is not specified by a route. Closes #6669. [Nicholas Seckar]
- Ensure render_to_string cleans up after itself when an exception is raised. #6658 [Rob Sanheim]
- Extract template_changed_since? from compile_template? so plugins may override its behavior for non-file-based templates. #6651 [Jeff Barczewski]
- Update to Prototype and script.aculo.us [5579]. [Thomas Fuchs]
- simple_format helper doesn't choke on nil. #6644 [jerry426]
- Update to Prototype 1.5.0_rc2 [5550] which makes it work in Opera again [Thomas Fuchs]
- Reuse named route helper module between Routing reloads. Use remove_method to delete named route methods after each load. Since the module is never collected, this fixes a significant memory leak. [Nicholas Seckar]
- ActionView::Base.erb_variable accessor names the buffer variable used to render templates. Defaults to _erbout; use _buf for erubis. [Rick Olson]
- assert_select_rjs :remove. [Dylan Egan]
- Always clear model associations from session. #4795 [sd@notso.net, andylien@gmail.com]
- Update to Prototype 1.5.0_rc2. [Sam Stephenson]
- Remove JavaScriptLiteral in favor of ActiveSupport::JSON::Variable. [Sam Stephenson]

- Sync ActionController::StatusCodes::STATUS_CODES with http://www.iana.org/assignments/http-status-codes. #6586 [dkubb]
- Multipart form values may have a content type without being treated as uploaded files if they do not provide a filename. #6401 [Andreas Schwarz, Jeremy Kemper]
- assert_response supports symbolic status codes. #6569 [Kevin Clark]

  ```
  assert_response :ok
  assert_response :not_found
  assert_response :forbidden
  ```

- Cache parsed query parameters. #6559 [Stefan Kaes]
- Deprecate JavaScriptHelper#update_element_function, which is superseeded by RJS [Thomas Fuchs]
- pluralize helper interprets nil as zero. #6474 [pope]
- Fix invalid test fixture exposed by stricter Ruby 1.8.5 multipart parsing. #6524 [Bob Silva]
- Set ActionView::Base.default_form_builder once rather than passing the :builder option to every form or overriding the form helper methods. [Jeremy Kemper]
- Deprecate expire_matched_fragments. Use expire_fragment instead. #6535 [Bob Silva]
- Update to latest Prototype, which doesn't serialize disabled form elements, adds clone() to arrays, empty/non-string Element.update() and adds a fixes excessive error reporting in WebKit beta versions [Thomas Fuchs]
- Deprecate start_form_tag and end_form_tag. Use form_tag / '</form>' from now on. [Rick]
- Added block-usage to PrototypeHelper#form_remote_tag, document block-usage of FormTagHelper#form_tag [Rick]

- Add a 0 margin/padding div around the hidden _method input tag that form_tag outputs. [Rick]
- Added block-usage to TagHelper#content_tag [DHH]. Example:

```
<% content_tag :div, :class => "strong" %>
 Hello world!
<% end %>

<%# Will output: -%>
<div class="strong">Hello world!</div>
```

- Deprecated UrlHelper#link_to_image and UrlHelper#link_to :post => true #6409 [BobSilva]
- Upgraded NumberHelper with number_to_phone support international formats to comply with ITU E.123 by supporting area codes with less than 3 digits, added precision argument to number_to_human_size (defaults to 1) #6421 [BobSilva]
- Fixed that setting RAILS_ASSET_ID to "" should not add a trailing slash after assets #6454 [BobSilva/chrismear]
- Force *_url named routes to show the host in ActionView [Rick]

```
<%= url_for ... %> # no host
<%= foo_path %>    # no host
<%= foo_url %>     # host!
```

- Add support for converting blocks into function arguments to JavaScriptGenerator#call and JavaScriptProxy#call. [Sam Stephenson]
- Add JavaScriptGenerator#literal for wrapping a string in an object whose #to_json is the string itself. [Sam Stephenson]
- Add <%= escape_once html %> to escape html while leaving any currently escaped entities alone. Fix button_to double-escaping issue. [Rick]

- Fix double-escaped entities, such as &amp;amp;, &amp;#123;, etc. [Rick]
- Fix deprecation warnings when rendering the template error template. [Nicholas Seckar]
- Fix routing to correctly determine when generation fails. Closes #6300. [psross].
- Fix broken assert_generates when extra keys are being checked. [Jamis Buck]
- Replace KCODE checks with String#chars for truncate. Closes #6385 [Manfred Stienstra]
- Make page caching respect the format of the resource that is being requested even if the current route is the default route so that, e.g. posts.rss is not transformed by url_for to '/' and subsequently cached as '/index.html' when it should be cached as '/posts.rss'. [Marcel Molina Jr.]
- Use String#chars in TextHelper::excerpt. Closes #6386 [Manfred Stienstra]
- Install named routes into ActionView::Base instead of proxying them to the view via helper_method. Closes #5932. [Nicholas Seckar]
- Update to latest Prototype and script.aculo.us trunk versions [Thomas Fuchs]
- Fix relative URL root matching problems. [Mark Imbriaco]
- Fix filter skipping in controller subclasses. #5949, #6297, #6299 [Martin Emde]
- render_text may optionally append to the response body. render_javascript appends by default. This allows you to chain multiple render :update calls by setting @performed_render = false between them (awaiting a better public API). [Jeremy Kemper]
- Rename test assertion to prevent shadowing. Closes #6306.

[psross]

- Fixed that NumberHelper#number_to_delimiter should respect precision of higher than two digits #6231 [phallstrom]
- Fixed that FormHelper#radio_button didn't respect an :id being passed in #6266 [evansj]
- Added an html_options hash parameter to javascript_tag() and update_page_tag() helpers #6311 [tzaharia]. Example:

```
update_page_tag :defer => 'true' { |page| ... }

# Gives:
# <script defer="true" type="text/javascript">...</script>
#
# Which is needed for dealing with the IE6 DOM
# when it's not yet fully loaded.
```

- Fixed that rescue template path shouldn't be hardcoded, then it's easier to hook in your own #6295 [mnaberez]
- Fixed escaping of backslashes in JavaScriptHelper#escape_javascript #6302 [sven@c3d2.de]
- Fixed that some 500 rescues would cause 500's themselves because the response had not yet been generated #6329 [cmselmer]
- respond_to :html doesn't assume .rhtml. #6281 [Hampton Catlin]
- Fixed some deprecation warnings in ActionPack [Rick Olson]
- assert_select_rjs decodes escaped unicode chars since the Javascript generators encode them. #6240 [japgolly]
- Deprecation: @cookies, @headers, @request, @response will be removed after 1.2. Use the corresponding method instead. [Jeremy Kemper]
- Make the :status parameter expand to the default message for that status code if it is an integer. Also support symbol statuses.

[Jamis Buck]. Examples:

```
head :status => 404        # expands to "404 Not Found"
head :status => :not_found # expands to "404 Not Found"
head :status => :created   # expands to "201 Created"
```

- Add head(options = {}) for responses that have no body. [Jamis Buck]. Examples:

```
head :status => 404
# => return an empty response with a 404 status
head :location => person_path(@person), :status => 201
```

- Fix bug that kept any before_filter except the first one from being able to halt the before_filter chain. [Rick Olson]
- strip_links is case-insensitive. #6285 [tagoh, Bob Silva]
- Clear the cache of possible controllers whenever Routes are reloaded. [Nicholas Seckar]
- Filters overhaul including meantime filter support using around filters + blocks. #5949 [Martin Emde, Roman Le Negrate, Stefan Kaes, Jeremy Kemper]
- Update RJS render tests. [sam]
- Update CGI process to allow sessions to contain namespaced models. Closes #4638. [dfelstead@site5.com]
- Fix routing to respect user provided requirements and defaults when assigning default routing options (such as :action => 'index'). Closes #5950. [Nicholas Seckar]
- Rescue Errno::ECONNRESET to handle an unexpectedly closed socket connection. Improves SCGI reliability. #3368, #6226 [sdsykes, fhanshaw@vesaria.com]
- Added that respond_to blocks will automatically set the content type to be the same as is requested [DHH]. Examples:

```
respond_to do |format|
  format.html { render :text => "I'm being sent as text/
html" }
  format.rss  { render :text => "I'm being sent as
application/rss+xml" }
  format.atom { render :text => "I'm being sent as
application/xml", :content_type => Mime::XML }
end
```

- Added utf-8 as the default charset for all renders. You can change this default using ActionController::Base.default_charset=(encoding) [DHH]

- Added proper getters and setters for content type and charset [DHH]. Example of what we used to do:

```
response.headers["Content-Type"] = "application/atom+xml;
charset=utf-8"
# ...now:
response.content_type = Mime::ATOM
response.charset      = "utf-8"
```

- Updated prototype.js to 1.5.0_rc1 with latest fixes. [Rick Olson] XPATH support, Make Form.getElements() return elements in the correct order, Fix broken Form.serialize return

- Declare file extensions exempt from layouts. #6219 [brandon]

```
ActionController::Base.exempt_from_layout 'rpdf'
```

- Add chained replace/update support for assert_select_rjs [Rick Olson]

```
# Given RJS like...
page['test1'].replace %(<div id="test1">foo</div>)
page['test2'].replace_html %(<div id="test2">foo</div>)
# Test it with...
assert_select_rjs :chained_replace
assert_select_rjs :chained_replace, "test1"
assert_select_rjs :chained_replace_html
assert_select_rjs :chained_replace_html, "test2"
```

- Load helpers in alphabetical order for consistency. Resolve cyclic javascript_helper dependency. #6132, #6178 [choonkeat@gmail.com]
- Skip params with empty names, such as the &=Save query string from <input type="submit"/>. #2569 [manfred, raphinou@yahoo.com]
- Fix assert_tag so that :content => "foo" does not match substrings, but only exact strings. Use :content => /foo/ to match substrings. #2799 [Eric Hodel]
- Add descriptive messages to the exceptions thrown by cgi_methods. #6091, #6103 [Nicholas Seckar, Bob Silva]
- Update JavaScriptGenerator#show/hide/toggle/remove to new Prototype syntax for multiple ids, #6068 [petermichaux@gmail.com]
- Update UrlWriter to support :only_path. [Nicholas Seckar, Dave Thomas]
- Fixed JavaScriptHelper#link_to_function and JavaScriptHelper#button_to_function to have the script argument be optional [DHH]. So what used to require a nil, like this:

```
link_to("Hider", nil, :class => "hider_link") { |p|
p[:something].hide }
# can be written like this:
link_to("Hider", :class => "hider_link") { |p| p[:something].
hide }
```

- Update to script.aculo.us 1.6.3 [Thomas Fuchs]
- Update to Prototype 1.5.0_rc1 [sam]
- Added access to nested attributes in RJS #4548 [richcollins@

gmail.com]. Examples:

```
page['foo']['style']
# => $('foo').style;
page['foo']['style']['color']
# => $('blank_slate').style.color;
page['foo']['style']['color'] = 'red'
# => $('blank_slate').style.color = 'red';
page['foo']['style'].color = 'red'
# => $('blank_slate').style.color = 'red';
```

- Fixed that AssetTagHelper#image_tag and others using compute_public_path should not modify the incoming source argument (closes #5102) [eule@space.ch]
- Deprecated the auto-appending of .png to AssetTagHelper#image_tag calls that doesn't have an extension [DHH]
- Fixed FormOptionsHelper#select to respect :selected value #5813
- Fixed TextHelper#simple_format to deal with multiple single returns within a single paragraph #5835 [moriq@moriq.com]
- Fixed TextHelper#pluralize to handle 1 as a string #5909 [rails@bencurtis.com]
- Improved resolution of DateHelper#distance_of_time_in_words for better precision #5994 [Bob Silva]
- Changed that uncaught exceptions raised any where in the application will cause RAILS_ROOT/public/500.html to be read and shown instead of just the static "Application error (Rails)" [DHH]
- Integration tests: thoroughly test ActionController::Integration::Ses sion. #6022 [Kevin Clark] (tests skipped unless you `gem install mocha`)
- Added deprecation language for pagination which will become a plugin by Rails 2.0 [DHH]

- Added deprecation language for in_place_editor and auto_complete_field that both pieces will become plugins by Rails 2.0 [DHH]
- Deprecated all of ActionController::Dependencies. All dependency loading is now handled from Active Support [DHH]
- Added assert_select* for CSS selector-based testing (deprecates assert_tag) #5936 [assaf.arkin@gmail.com]
- radio_button_tag generates unique id attributes. #3353 [Bob Silva, somekool@gmail.com]
- strip_tags passes through blank args such as nil or "". #2229, #6702 [duncan@whomwah.com, dharana]
- Cleanup assert_tag :children counting. #2181 [jamie@bravenet.com]
- button_to accepts :method so you can PUT and DELETE with it. #6005 [Dan Webb]
- Update sanitize text helper to strip plaintext tags, and <img src="javascript:bang">. [Rick Olson]
- Update routing documentation. Closes #6017 [Nathan Witmer]
- Add routing tests to assert that RoutingError is raised when conditions aren't met. Closes #6016 [Nathan Witmer]
- Deprecation: update docs. #5998 [jakob@mentalized.net, Kevin Clark]
- Make auto_link parse a greater subset of valid url formats. [Jamis Buck]
- Integration tests: headers beginning with X aren't excluded from the HTTP_ prefix, so X-Requested-With becomes HTTP_X_REQUESTED_WITH as expected. [Mike Clark]
- Tighten rescue clauses. #5985 [james@grayproductions.net]
- Fix send_data documentation typo. #5982 [brad@madriska.com]
- Switch to using FormEncodedPairParser for parsing request

parameters. [Nicholas Seckar, DHH]

- respond_to .html now always renders #{action_name}.rhtml so that registered custom template handlers do not override it in priority. Custom mime types require a block and throw proper error now. [Tobias Luetke]
- Deprecation: test deprecated instance vars in partials. [Jeremy Kemper]
- Add UrlWriter to allow writing urls from Mailers and scripts. [Nicholas Seckar]
- Clean up and run the Active Record integration tests by default. #5854 [kevin.clark@gmail.com, Jeremy Kemper]
- Correct example in cookies docs. #5832 [jessemerriman@warpmail.net]
- Updated to script.aculo.us 1.6.2 [Thomas Fuchs]
- Relax Routing's anchor pattern warning; it was preventing use of [^/] inside restrictions. [Nicholas Seckar]
- Add controller_paths variable to Routing. [Nicholas Seckar]
- Fix assert_redirected_to issue with named routes for module controllers. [Rick Olson]
- Tweak RoutingError message to show option diffs, not just missing named route significant keys. [Rick Olson]
- Invoke method_missing directly on hidden actions. Closes #3030. [Nicholas Seckar]
- Require Tempfile explicitly for TestUploadedFile due to changes in class auto loading. [Rick Olson]
- Add RoutingError exception when RouteSet fails to generate a path from a Named Route. [Rick Olson]
- Replace Reloadable with Reloadable::Deprecated. [Nicholas Seckar]

- Deprecation: check whether instance variables have been monkeyed with before assigning them to deprecation proxies. Raises a RuntimeError if so. [Jeremy Kemper]
- Add support for the param_name parameter to the auto_complete_field helper. #5026 [david.a.williams@gmail.com]
- Deprecation! @params, @session, @flash will be removed after 1.2. Use the corresponding instance methods instead. You'll get printed warnings during tests and logged warnings in dev mode when you access either instance variable directly. [Jeremy Kemper]
- Make Routing noisy when an anchor regexp is assigned to a segment. #5674 [francois.beausoleil@gmail.com]
- Added months and years to the resolution of DateHelper#distance_of_time_in_words, such that "60 days ago" becomes "2 months ago" #5611 [pjhyett@gmail.com]
- Short documentation to mention use of Mime::Type.register. #5710 [choonkeat@gmail.com]
- Make controller_path available as an instance method. #5724 [jmckible@gmail.com]
- Update query parser to support adjacent hashes. [Nicholas Seckar]
- Make action caching aware of different formats for the same action so that, e.g. foo.xml is cached separately from foo.html. Implicitly set content type when reading in cached content with mime revealing extensions so the entire onous isn't on the webserver. [Marcel Molina Jr.]
- Restrict Request Method hacking with ?_method to POST requests. [Rick Olson]
- Fix bug when passing multiple options to SimplyRestful, like :new => { :preview => :get, :draft => :get }. [Rick Olson, Josh Susser, Lars Pind]

- Dup the options passed to map.resources so that multiple resources get the same options. [Rick Olson]
- Fixed the new_#{resource}_url route and added named route tests for Simply Restful. [Rick Olson]
- Added map.resources from the Simply Restful plugin [DHH]. Examples (the API has changed to use plurals!):

```
map.resources :messages
map.resources :messages, :comments
map.resources :messages, :new => { :preview => :post }
```

- Fixed that integration simulation of XHRs should set Accept header as well [Edward Frederick]
- TestRequest#reset_session should restore a TestSession, not a hash [Koz]
- Don't search a load-path of '.' for controller files [Jamis Buck]
- Update integration.rb to require test_process explicitly instead of via Dependencies. [Nicholas Seckar]
- Fixed that you can still access the flash after the flash has been reset in reset_session. Closes #5584 [lmarlow@yahoo.com]
- Allow form_for and fields_for to work with indexed form inputs. [Jeremy Kemper, Matt Lyon]

```
<% form_for 'post[]', @post do |f| -%>
<% end -%>
```

- Remove leak in development mode by replacing define_method with module_eval. [Nicholas Seckar]
- Provide support for decimal columns to form helpers. Closes #5672. [dave@pragprog.com]
- Update documentation for erb trim syntax. #5651 [matt@mattmargolis.net]

- Pass :id => nil or :class => nil to error_messages_for to supress that html attribute. #3586 [olivier_ansaldi@yahoo.com, sebastien@goetzilla.info]
- Reset @html_document between requests so assert_tag works. #4810 [jarkko@jlaine.net, easleydp@gmail.com]
- Update render :partial documentation. #5646 [matt@mattmargolis.net]
- Integration tests behave well with render_component. #4632 [edward.frederick@revolution.com, dev.rubyonrails@maxdunn.com]
- Added exception handling of missing layouts #5373 [chris@ozmm.org]
- Fixed that real files and symlinks should be treated the same when compiling templates #5438 [zachary@panandscan.com]
- Fixed that the flash should be reset when reset_session is called #5584 [shugo@ruby-lang.org]
- Added special case for "1 Byte" in NumberHelper#number_to_human_size #5593 [murpyh@rubychan.de]
- Fixed proper form-encoded parameter parsing for requests with "Content-Type: application/x-www-form-urlencoded; charset=utf-8" (note the presence of a charset directive) [DHH]
- Add route_name_path method to generate only the path for a named routes. For example, map.person will add person_path. [Nicholas Seckar]
- Avoid naming collision among compiled view methods. [Jeremy Kemper]
- Fix CGI extensions when they expect string but get nil in Windows. Closes #5276 [mislav@nippur.irb.hr]
- Determine the correct template_root for deeply nested components. #2841 [s.brink@web.de]

- Fix that routes with *path segments in the recall can generate URLs. [Rick]
- Fix strip_links so that it doesn't hang on multiline <acronym> tags [Jamis Buck]
- Remove problematic control chars in rescue template. #5316 [Stefan Kaes]
- Make sure passed routing options are not mutated by routing code. #5314 [Blair Zajac]
- Make sure changing the controller from foo/bar to bing/bang does not change relative to foo. [Jamis Buck]
- Escape the path before routing recognition. #3671
- Make sure :id and friends are unescaped properly. #5275 [me@julik.nl]
- Fix documentation for with_routing to reflect new reality. #5281 [rramdas@gmail.com]
- Rewind readable CGI params so others may reread them (such as CGI::Session when passing the session id in a multipart form). #210 [mklame@atxeu.com, matthew@walker.wattle.id.au]
- Added Mime::TEXT (text/plain) and Mime::ICS (text/calendar) as new default types [DHH]
- Added Mime::Type.register(string, symbol, synonyms = []) for adding new custom mime types [DHH]. Example: Mime::Type.register("image/gif", :gif)
- Added support for Mime objects in render :content_type option [DHH]. Example: render :text => some_atom, :content_type => Mime::ATOM
- Add :status option to send_data and send_file. Defaults to '200 OK'. #5243 [Manfred Stienstra m.stienstra@fngtps.com]
- Routing rewrite. Simpler, faster, easier to understand. The published

API for config/routes.rb is unchanged, but nearly everything else is different, so expect breakage in plugins and libs that try to fiddle with routes. [Nicholas Seckar, Jamis Buck]

```
map.connect '/foo/:id', :controller => '...', :action =>
'...'
map.connect '/foo/:id.:format', :controller => '...',
:action => '...'
map.connect '/foo/:id', ..., :conditions => { :method =>
:get }
```

- Cope with missing content type and length headers. Parse parameters from multipart and urlencoded request bodies only. [Jeremy Kemper]
- Accept multipart PUT parameters. #5235 [guy.naor@famundo.com]
- Added interrogation of params[:format] to determine Accept type. If :format is specified and matches a declared extension, like "rss" or "xml", that mime type will be put in front of the accept handler. This means you can link to the same action from different extensions and use that fact to determine output [DHH]. Example:

```
class WeblogController < ActionController::Base
  def index
    @posts = Post.find :all

    respond_to do |format|
      format.html
      format.xml { render :xml => @posts.to_xml }
      format.rss { render :action => "feed.rxml" }
    end
  end
end

# returns HTML when requested by a browser,
# since the browser has the HTML mimetype
# at the top of its priority list
Accept: text/html
GET /weblog
```

```
# returns the XML
Accept: application/xml
GET /weblog

# returns the HTML
Accept: application/xml
GET /weblog.html

# returns the XML
Accept: text/html
GET /weblog.xml

# All this relies on the fact that you have a route
# that includes .:format.
```

- Expanded :method option in FormTagHelper#form_tag, FormHelper#form_for, PrototypeHelper#remote_form_for, PrototypeHelper#remote_form_tag, and PrototypeHelper#link_to_remote to allow for verbs other than GET and POST by automatically creating a hidden form field named _method, which will simulate the other verbs over post [DHH]
- Added :method option to UrlHelper#link_to, which allows for using other verbs than GET for the link. This replaces the :post option, which is now deprecated. Example: link_to "Destroy", person_url(:id => person), :method => :delete [DHH]
- follow_redirect doesn't complain about being redirected to the same controller. #5153 [dymo@mk.ukrtelecom.ua]
- Add layout attribute to response object with the name of the layout that was rendered, or nil if none rendered. [Kevin Clark kevin.clark@gmail.com]
- Fix NoMethodError when parsing params like &&. [Adam Greenfield]
- Fix flip flopped logic in docs for url_for's :only_path option. Closes #4998. [esad@esse.at]

- form.text_area handles the :size option just like the original text_area (:size => '60x10' becomes cols="60" rows="10"). [Jeremy Kemper]
- Excise ingrown code from FormOptionsHelper#options_for_select. #5008 [anonymous]
- Small fix in routing to allow dynamic routes (broken after [4242]) [Rick]

```
map.connect '*path',
            :controller => 'files',
            :action => 'show'
```

- Replace alias method chaining with Module#alias_method_chain. [Marcel Molina Jr.]
- Replace Ruby's deprecated append_features in favor of included. [Marcel Molina Jr.]
- Use #flush between switching from #write to #syswrite. Closes #4907. [Blair Zajac blair@orcaware.com]
- Documentation fix: integration test scripts don't require integration_test. Closes #4914. [Frederick Ros sl33p3r@free.fr]
- ActionController::Base Summary documentation rewrite. Closes #4900. [kevin.clark@gmail.com]
- Fix text_helper.rb documentation rendering. Closes #4725. [Frederick Ros]
- Fixes bad rendering of JavaScriptMacrosHelper rdoc (closes #4910) [Frederick Ros]
- Allow error_messages_for to report errors for multiple objects, as well as support for customizing the name of the object in the error summary header. Closes #4186. [andrew@redlinesoftware.com, Marcel Molina Jr.]

```
error_messages_for :account,
                   :user,
                   :subscription,
                   :object_name => :account
```

- Enhance documentation for setting headers in integration tests. Skip auto HTTP prepending when its already there. Closes #4079. [Rick Olson]
- Documentation for AbstractRequest. Closes #4895. [kevin.clark@gmail.com]
- Refactor various InstanceTag instance method to class methods. Closes #4800. [skaes@web.de]
- Remove all remaining references to @params in the documentation. [Marcel Molina Jr.]
- Add documentation for redirect_to :back's RedirectBackError exception. [Marcel Molina Jr.]
- Update layout and content_for documentation to use yield rather than magic @content_for instance variables. [Marcel Molina Jr.]
- Fix assert_redirected_to tests according to real-world usage. Also, don't fail if you add an extra :controller option: [Rick]

```
redirect_to :action => 'new'
assert_redirected_to :controller => 'monkeys',
                     :action => 'new'
```

- Cache CgiRequest#request_parameters so that multiple calls don't re-parse multipart data. [Rick]
- Diff compared routing options. Allow #assert_recognizes to take a second arg as a hash to specify optional request method [Rick]

```
assert_recognizes({:controller => 'users', :action =>
'index'}, 'users')
assert_recognizes({:controller => 'users', :action =>
'create'}, {:path => 'users', :method => :post})
```

- Diff compared options with #assert_redirected_to [Rick]
- Add support in routes for semicolon delimited "subpaths", like /books/:id;:action [Jamis Buck]
- Change link_to_function and button_to_function to (optionally) take an update_page block instead of a JavaScript string. Closes #4804. [zraii@comcast.net, Sam Stephenson]
- Fixed that remote_form_for can leave out the object parameter and default to the instance variable of the object_name, just like form_for [DHH]
- Modify routing so that you can say :require => { :method => :post } for a route, and the route will never be selected unless the request method is POST. Only works for route recognition, not for route generation. [Jamis Buck]
- Added :add_headers option to verify which merges a hash of name/value pairs into the response's headers hash if the prerequisites cannot be satisfied. [Sam Stephenson]

```
verify :only => :speak, :method => :post,
       :render => { :status => 405,
                    :text => "Must be post" },
       :add_headers => { "Allow" => "POST" }
```

- Added ActionController.filter_parameter_logging that makes it easy to remove passwords, credit card numbers, and other sensitive information from being logged when a request is handled #1897 [jeremye@bsa.ca.gov]

ActiveRecord

SVN

- Qualified column names work in hash conditions, like :conditions => { 'comments.created_at' => ... }. #9733 [danger]

- Fix regression where the association would not construct new finder SQL on save causing bogus queries for "WHERE owner_id = NULL" even after owner was saved. #8713 [Bryan Helmkamp]
- Refactor association create and build so before & after callbacks behave consistently. #8854 [lifofifo, mortent]
- Quote table names. Defaults to column quoting. #4593 [Justin Lynn, gwcoffey, eadz, Dmitry V. Sabanin, Jeremy Kemper]
- Alias association #build to #new so it behaves predictably. #8787 [lifofifo]
- Add notes to documentation regarding attr_readonly behavior with counter caches and polymorphic associations. Closes #9835 [saimonmoore, rick]
- Observers can observe model names as symbols properly now. Closes #9869 [queso]
- find_and_(initialize|create)_by methods can now properly initialize protected attributes [Tobias Luetke]
- belongs_to infers the foreign key from the association name instead of from the class name. [Jeremy Kemper]
- PostgreSQL: support multiline default values. #7533 [Carl Lerche, aguynamedryan, Rein Henrichs, Tarmo Tanav]
- MySQL: fix change_column on not-null columns that don't accept dfeault values of ''. #6663 [Jonathan Viney, Tarmo Tanav]
- validates_uniqueness_of behaves well with single-table inheritance. #3833 [Gabriel Gironda, rramdas, François Beausoleil, Josh Peek, Tarmo Tanav]
- Raise ProtectedAttributeAssignmentError in development and test environments when mass-assigning to an attr_protected attribute. #9802 [Henrik N]
- Speedup database date/time parsing. [Jeremy Kemper, Tarmo Tanav]

- Fix calling .clear on a has_many :dependent=>:delete_all association. [tarmo]

- Allow change_column to set NOT NULL in the PostgreSQL adapter [tarmo]

- Fix that ActiveRecord would create attribute methods and override custom attribute getters if the method is also defined in Kernel. methods. [Rick]

- Don't call attr_readonly on polymorphic belongs_to associations, in case it matches the name of some other non-ActiveRecord class/module. [Rick]

- Try loading activerecord-<adaptername>-adapter gem before trying a plain require so you can use custom gems for the bundled adapters. Also stops gems from requiring an adapter from an old Active Record gem. [Jeremy Kemper, Derrick Spell]

2.0.0 [Preview Release] (September 29th, 2007)

[Includes duplicates of changes from 1.14.2 – 1.15.3]

- Add attr_readonly to specify columns that are skipped during a normal ActiveRecord #save operation. Closes #6896 [dcmanges]

```ruby
class Comment < ActiveRecord::Base
  # Automatically sets Article#comments_count as readonly.
  belongs_to :article, :counter_cache => :comments_count
end

class Article < ActiveRecord::Base
  attr_readonly :approved_comments_count
end
```

- Make size for has_many :through use counter cache if it exists. Closes #9734 [xaviershay]

- Remove DB2 adapter since IBM chooses to maintain their own adapter instead. [Jeremy Kemper]

- Extract Oracle, SQLServer, and Sybase adapters into gems. [Jeremy Kemper]
- Added fixture caching that'll speed up a normal fixture-powered test suite between 50% and 100% #9682 [frederick.cheung@gmail.com]
- Correctly quote id list for limited eager loading. #7482 [tmacedo]
- Fixed that using version-targetted migrates would fail on loggers other than the default one #7430 [valeksenko]
- Fixed rename_column for SQLite when using symbols for the column names #8616 [drodriguez]
- Added the possibility of using symbols in addition to concrete classes with ActiveRecord::Observer#observe #3998 [robbyrussell/tarmo]
- Added ActiveRecord::Base#to_json/from_json [DHH/chuyeow]
- Added ActiveRecord::Base#from_xml [DHH]. Example:

```
xml = "<person><name>David</name></person>"
Person.new.from_xml(xml).name # => "David"
```

- Define dynamic finders as real methods after first usage. [bscofield]
- Deprecation: remove deprecated threaded_connections methods. Use allow_concurrency instead. [Jeremy Kemper]
- Associations macros accept extension blocks alongside modules. #9346 [Josh Peek]
- Speed up and simplify query caching. [Jeremy Kemper]
- connection.select_rows 'sql' returns an array (rows) of arrays (field values). #2329 [Michael Schuerig]
- Eager loading respects explicit :joins. #9496 [dasil003]
- Extract Firebird, FrontBase, and OpenBase adapters into gems.

#9508, #9509, #9510 [Jeremy Kemper]

- RubyGem database adapters: expects a gem named activerecord-<database>-adapter with active_record/connection_adapters/<database>_adapter.rb in its load path. [Jeremy Kemper]
- Fixed that altering join tables in migrations would fail w/ sqlite3 #7453 [TimoMihaljov/brandon]
- Fix association writer with :dependent => :nullify. #7314 [Jonathan Viney]
- OpenBase: update for new lib and latest Rails. Support migrations. #8748 [dcsesq]
- Moved acts_as_tree into a plugin of the same name on the official Rails svn #9514 [lifofifo]
- Moved acts_as_nested_set into a plugin of the same name on the official Rails svn #9516 [josh]
- Moved acts_as_list into a plugin of the same name on the official Rails svn [josh]
- Explicitly require active_record/query_cache before using it. [Jeremy Kemper]
- Fix bug where unserializing an attribute attempts to modify a frozen @attributes hash for a deleted record. [Rick, marclove]
- Performance: absorb instantiate and initialize_with_callbacks into the Base methods. [Jeremy Kemper]
- Fixed that eager loading queries and with_scope should respect the :group option [DHH]
- Improve performance and functionality of the postgresql adapter. Closes #8049 [roderickvd] For more information see: http://dev.rubyonrails.org/ticket/8049
- Don't clobber includes passed to has_many.count [danger]

- Make sure has_many uses :include when counting [danger]
- Change the implementation of ActiveRecord's attribute reader and writer methods [nzkoz] Generate Reader and Writer methods which cache attribute values in hashes. This is to avoid repeatedly parsing the same date or integer columns. Change exception raised when users use find with :select then try to access a skipped column. Plugins could override missing_attribute() to lazily load the columns. Move method definition to the class, instead of the instance. Always generate the readers, writers and predicate methods.
- Perform a deep #dup on query cache results so that modifying activerecord attributes does not modify the cached attributes. [Rick]
- Ensure that has_many :through associations use a count query instead of loading the target when #size is called. Closes #8800 [lifo]
- Added :unless clause to validations #8003 [monki]. Example:

```
def using_open_id?
  !identity_url.blank?
end
validates_presence_of :identity_url, :if => using_open_id?
validates_presence_of :username, :unless => using_open_id?
validates_presence_of :password, :unless => using_open_id?
```

- Fix #count on a has_many :through association so that it recognizes the :uniq option. Closes #8801 [lifofifo]
- Fix and properly document/test count(column_name) usage. Closes #8999 [lifofifo]
- Remove deprecated count(conditions=nil, joins=nil) usage. Closes #8993 [lifofifo]
- Change belongs_to so that the foreign_key assumption is taken from the association name, not the class name. Closes #8992

[hasmanyjosh]

```
# OLD
belongs_to :visitor, :class_name => 'User'
# => inferred foreign_key is user_id
# NEW
belongs_to :visitor, :class_name => 'User'
# => inferred foreign_key is visitor_id
```

- Remove spurious tests from deprecated_associations_test, most of these aren't deprecated, and are duplicated in associations_test. Closes #8987 [lifofifo]
- Make create! on a has_many :through association return the association object. Not the collection. Closes #8786 [lifofifo]
- Move from select * to select tablename.* to avoid clobbering IDs. Closes #8889 [dasil003]
- Don't call unsupported methods on associated objects when using :include, :method with to_xml #7307, [manfred, jwilger]
- Define collection singular ids method for has_many :through associations. #8763 [lifofifo]
- Array attribute conditions work with proxied association collections. #8318 [kamal, theamazingrando]
- Fix polymorphic has_one associations declared in an abstract class. #8638 [lifofifo, daxhuiberts]
- Fixed validates_associated should not stop on the first error #4276 [mrj/manfred/josh]
- Rollback if commit raises an exception. #8642 [kik, Jeremy Kemper]
- Update tests' use of fixtures for the new collections api. #8726 [kamal]
- Save associated records only if the association is already loaded. #8713 [blaine]

- MySQL: fix show_variable. #8448 [matt, Jeremy Kemper]
- Fixtures: correctly delete and insert fixtures in a single transaction. #8553 [Michael Schuerig]
- Fixtures: people(:technomancy, :josh) returns both fixtures. #7880 [technomancy, Josh Peek]
- Calculations support non-numeric foreign keys. #8154 [kamal]
- with_scope is protected. #8524 [Josh Peek]
- Quickref for association methods. #7723 [marclove, Mindsweeper]
- Calculations: return nil average instead of 0 when there are no rows to average. #8298 [davidw]
- acts_as_nested_set: direct_children is sorted correctly. #4761 [Josh Peek, rails@33lco.net]
- Raise an exception if both attr_protected and attr_accessible are declared. #8507 [stellsmi]
- SQLite, MySQL, PostgreSQL, Oracle: quote column names in column migration SQL statements. #8466 [marclove, lorenjohnson]
- Allow nil serialized attributes with a set class constraint. #7293 [sandofsky]
- Oracle: support binary fixtures. #7987 [Michael Schoen]
- Fixtures: pull fixture insertion into the database adapters. #7987 [Michael Schoen]
- Announce migration versions as they're performed. [Jeremy Kemper]
- find gracefully copes with blank :conditions. #7599 [Dan Manges, johnnyb]
- validates_numericality_of takes :greater_than, :greater_than_or_equal_to, :equal_to, :less_than, :less_than_or_equal_to, :odd, and :even options. #3952 [Bob Silva, Dan Kubb, Josh Peek]

- MySQL: create_database takes :charset and :collation options. Charset defaults to utf8. #8448 [matt]
- Find with a list of ids supports limit/offset. #8437 [hrudududu]
- Optimistic locking: revert the lock version when an update fails. #7840 [plang]
- Migrations: add_column supports custom column types. #7742 [jsgarvin, Theory]
- Load database adapters on demand. Eliminates config.connection_adapters and RAILS_CONNECTION_ADAPTERS. Add your lib directory to the $LOAD_PATH and put your custom adapter in lib/active_record/connection_adapters/adaptername_adapter.rb. This way you can provide custom adapters as plugins or gems without modifying Rails. [Jeremy Kemper]
- Ensure that associations with :dependent => :delete_all respect :conditions option. Closes #8034 [danger, joshpeek, Rick]
- belongs_to assignment creates a new proxy rather than modifying its target in-place. #8412 [mmangino@elevatedrails.com]
- Fix column type detection while loading fixtures. Closes #7987 [roderickvd]
- Document deep eager includes. #6267 [Josh Susser, Dan Manges]
- Document warning that associations names shouldn't be reserved words. #4378 [murphy@cYcnus.de, Josh Susser]
- Sanitize Base#inspect. #8392, #8623 [Nik Wakelin, jnoon]
- Replace the transaction {|transaction|..} semantics with a new Exception ActiveRecord::Rollback.   [Koz]
- Oracle: extract column length for CHAR also. #7866 [ymendel]
- Document :allow_nil option for validates_acceptance_of since it defaults to true. [tzaharia]

- Update documentation for :dependent declaration so that it explicitly uses the non-deprecated API. [danger]

- Add documentation caveat about when to use count_by_sql. [fearoffish]

- Enhance documentation for increment_counter and decrement_counter. [fearoffish]

- Provide brief introduction to what optimistic locking is. [fearoffish]

- Add documentation for :encoding option to mysql adapter. [marclove]

- Added short-hand declaration style to migrations (inspiration from Sexy Migrations, http://errtheblog.com/post/2381) [DHH]. Example:

  ```
  create_table "products" do |t|
    t.column "shop_id",    :integer
    t.column "creator_id", :integer
    t.column "name",       :string,   :default => "Untitled"
    t.column "value",      :string,   :default => "Untitled"
    t.column "created_at", :datetime
    t.column "updated_at", :datetime
  end
  # can now be written as:
  create_table :products do |t|
    t.integer :shop_id, :creator_id
    t.string  :name, :value, :default => "Untitled"
    t.timestamps
  end
  ```

- Use association name for the wrapper element when using .to_xml. Previous behavior lead to non-deterministic situations with STI and polymorphic associations. [Koz, jstrachan]

- Improve performance of calling .create on has_many :through associations. [evan]

- Improved cloning performance by relying less on exception raising #8159 [Blaine]

- Added ActiveRecord::Base.inspect to return a column-view like #<Post id:integer, title:string, body:text> [DHH]
- Added yielding of Builder instance for ActiveRecord::Base#to_xml calls [DHH]
- Small additions and fixes for ActiveRecord documentation. Closes #7342 [jeremymcanally]
- Add helpful debugging info to the ActiveRecord::StatementInvalid exception in ActiveRecord::ConnectionAdapters::SqliteAdapter#table_structure. Closes #7925. [court3nay]
- SQLite: binary escaping works with $KCODE='u'. #7862 [tsuka]
- Base#to_xml supports serialized attributes. #7502 [jonathan]
- Base.update_all :order and :limit options. Useful for MySQL updates that must be ordered to avoid violating unique constraints. [Jeremy Kemper]
- Remove deprecated object transactions. People relying on this functionality should install the object_transactions plugin at http://code.bitsweat.net/svn/object_transactions. Closes #5637 [Koz, Jeremy Kemper]
- PostgreSQL: remove DateTime → Time downcast. Warning: do not enable translate_results for the C bindings if you have timestamps outside Time's domain. [Jeremy Kemper]
- find_or_create_by_* takes a hash so you can create with more attributes than are in the method name. For example, Person.find_or_create_by_name(:name => 'Henry', :comments => 'Hi new user!') is equivalent to Person.find_by_name('Henry') || Person.create(:name => 'Henry', :comments => 'Hi new user!'). #7368 [Josh Susser]
- Make sure with_scope takes both :select and :joins into account when setting :readonly. Allows you to save records you retrieve using method_missing on a has_many :through associations. [Koz]

- Allow a polymorphic :source for has_many :through associations. Closes #7143 [protocool]
- Consistent public/protected/private visibility for chained methods. #7813 [Dan Manges]
- Oracle: fix quoted primary keys and datetime overflow. #7798 [Michael Schoen]
- Consistently quote primary key column names. #7763 [toolmantim]
- Fixtures: fix YAML ordered map support. #2665 [Manuel Holtgrewe, nfbuckley]
- DateTimes assume the default timezone. #7764 [Geoff Buesing]
- Sybase: hide timestamp columns since they're inherently read-only. #7716 [Mike Joyce]
- Oracle: overflow Time to DateTime. #7718 [Michael Schoen]
- PostgreSQL: don't use async_exec and async_query with postgres-pr. #7727, #7762 [flowdelic, toolmantim]
- Fix has_many :through << with custom foreign keys. #6466, #7153 [naffis, Rich Collins]
- Test DateTime native type in migrations, including an edge case with dates during calendar reform. #7649, #7724 [fedot, Geoff Buesing]
- SQLServer: correctly schema-dump tables with no indexes or descending indexes. #7333, #7703 [Jakob S, Tom Ward]
- SQLServer: recognize real column type as Ruby float. #7057 [sethladd, Tom Ward]
- Added fixtures :all as a way of loading all fixtures in the fixture directory at once #7214 [manfred]
- Added database connection as a yield parameter to ActiveRecord::Base.transaction so you can manually rollback

[DHH]. Example:

```
transaction do |transaction|
  david.withdrawal(100)
  mary.deposit(100)
  transaction.rollback! # rolls back the transaction
                        # that was otherwise going to
                        # be successful
end
```

- Made increment_counter/decrement_counter play nicely with optimistic locking, and added a more general update_counters method [Jamis Buck]
- Reworked David's query cache to be available as Model.cache {...}. For the duration of the block no select query should be run more then once. Any inserts/deletes/executes will flush the whole cache however [Tobias Luetke]

  ```
  Task.cache { Task.find(1); Task.find(1) } #=> 1 query
  ```

- When dealing with SQLite3, use the table_info pragma helper, so that the bindings can do some translation for when sqlite3 breaks incompatibly between point releases. [Jamis Buck]
- Oracle: fix lob and text default handling. #7344 [gfriedrich, Michael Schoen]
- SQLServer: don't choke on strings containing 'null'. #7083 [Jakob S]
- MySQL: blob and text columns may not have defaults in 5.x. Update fixtures schema for strict mode. #6695 [Dan Kubb]
- update_all can take a Hash argument. sanitize_sql splits into two methods for conditions and assignment since NULL values and delimiters are handled differently. #6583, #7365 [sandofsky, Assaf]
- MySQL: SET SQL_AUTO_IS_NULL=0 so 'where id is null' doesn't

select the last inserted id. #6778 [Jonathan Viney, timc]

- Use Date#to_s(:db) for quoted dates. #7411 [Michael Schoen]
- Don't create instance writer methods for class attributes. Closes #7401 [Rick]
- Docs: validations examples. #7343 [zackchandler]
- Add missing tests ensuring callbacks work with class inheritance. Closes #7339 [sandofsky]
- Fixtures use the table name and connection from set_fixture_class. #7330 [Anthony Eden]
- Remove useless code in #attribute_present? since 0 != blank?. Closes #7249 [Josh Susser]
- Fix minor doc typos. Closes #7157 [Josh Susser]
- Fix incorrect usage of #classify when creating the eager loading join statement. Closes #7044 [Josh Susser]
- SQLServer: quote table name in indexes query. #2928 [keithm@infused.org]
- Subclasses of an abstract class work with single-table inheritance. #5704, #7284 [BertG, nick+rails@ag.arizona.edu]
- Make sure sqlite3 driver closes open connections on disconnect [Rob Rasmussen]
- [DOC] clear up some ambiguity with the way has_and_belongs_to_many creates the default join table name. #7072 [jeremymcanally]
- change_column accepts :default => nil. Skip column options for primary keys. #6956, #7048 [Dan Manges, Jeremy Kemper]
- MySQL, PostgreSQL: change_column_default quotes the default value and doesn't lose column type information. #3987, #6664 [Jonathan Viney, manfred, altano@bigfoot.com]
- Oracle: create_table takes a :sequence_name option to override the

'tablename_seq' default. #7000 [Michael Schoen]

- MySQL: retain SSL settings on reconnect. #6976 [randyv2]
- Apply scoping during initialize instead of create. Fixes setting of foreign key when using find_or_initialize_by with scoping. [Cody Fauser]
- SQLServer: handle [quoted] table names. #6635 [rrich]
- acts_as_nested_set works with single-table inheritance. #6030 [Josh Susser]
- PostgreSQL, Oracle: correctly perform eager finds with :limit and :order. #4668, #7021 [eventualbuddha, Michael Schoen]
- Pass a range in :conditions to use the SQL BETWEEN operator. #6974 [Dan Manges]

```
Student.find(:all, :conditions => { :grade => 9..12 })
```

- Fix the Oracle adapter for serialized attributes stored in CLOBs. Closes #6825 [mschoen, tdfowler]
- [DOCS] Apply more documentation for ActiveRecord Reflection. Closes #4055 [Robby Russell]
- [DOCS] Document :allow_nil option of #validate_uniqueness_of. Closes #3143 [Caio Chassot]
- Bring the sybase adapter up to scratch for 1.2 release. [jsheets]
- Rollback new_record? and id when an exception is raised in a save callback. #6910 [Ben Curren, outerim]
- Pushing a record on an association collection doesn't unnecessarily load all the associated records. [Obie Fernandez, Jeremy Kemper]
- Oracle: fix connection reset failure. #6846 [leonlleslie]
- Subclass instantiation doesn't try to explicitly require the corre-

sponding subclass. #6840 [leei, Jeremy Kemper]

- fix faulty inheritance tests and that eager loading grabs the wrong inheritance column when the class of your association is an STI subclass. Closes #6859 [protocool]
- Consolidated different create and create! versions to call through to the base class with scope. This fixes inconsistencies, especially related to protected attribtues. Closes #5847 [Alexander Dymo, Tobias Luetke]
- find supports :lock with :include. Check whether your database allows SELECT ... FOR UPDATE with outer joins before using. #6764 [vitaly, Jeremy Kemper]
- Add AssociationCollection#create! to be consistent with AssociationCollection#create when dealing with a foreign key that is a protected attribute [Cody Fauser]
- Added counter optimization for AssociationCollection#any? so person.friends.any? won't actually load the full association if we have the count in a cheaper form [DHH]
- Change fixture_path to a class inheritable accessor allowing test cases to have their own custom set of fixtures. #6672 [zdennis]
- Quote ActiveSupport::Multibyte::Chars. #6653 [Julian Tarkhanov]
- Simplify query_attribute by typecasting the attribute value and checking whether it's nil, false, zero or blank. #6659 [Jonathan Viney]
- validates_numericality_of uses \A \Z to ensure the entire string matches rather than ^ $ which may match one valid line of a multiline string. #5716 [Andreas Schwarz]
- Run validations in the order they were declared. #6657 [obrie]
- MySQL: detect when a NOT NULL column without a default value is misreported as default ''. Can't detect for string, text, and binary columns since '' is a legitimate default. #6156 [simon@redhillcon-

sulting.com.au, obrie, Jonathan Viney, Jeremy Kemper]

- Simplify association proxy implementation by factoring construct_scope out of method_missing. #6643 [martin]
- Oracle: automatically detect the primary key. #6594 [vesaria, Michael Schoen]
- Oracle: to increase performance, prefetch 100 rows and enable similar cursor sharing. Both are configurable in database.yml. #6607 [philbogle@gmail.com, ray.fortna@jobster.com, Michael Schoen]
- Don't inspect unloaded associations. #2905 [lmarlow]
- SQLite: use AUTOINCREMENT primary key in >= 3.1.0. #6588, #6616 [careo, lukfugl]
- Cache inheritance_column. #6592 [Stefan Kaes]
- Firebird: decimal/numeric support. #6408 [macrnic]
- make add_order a tad faster. #6567 [Stefan Kaes]
- Find with :include respects scoped :order. #5850
- Support nil and Array in :conditions => { attr => value } hashes. #6548 [Assaf, Jeremy Kemper]
- find(:all, :conditions => { :topic_id => [1, 2, 3], :last_read => nil }
- Consistently use LOWER for uniqueness validations (rather than mixing with UPPER) so the database can always use a functional index on the lowercased column. #6495 [Si]
- SQLite: fix calculations workaround, remove count(distinct) query rewrite, cleanup test connection scripts. [Jeremy Kemper]
- SQLite: count(distinct) queries supported in >= 3.2.6. #6544 [Bob Silva]
- Dynamically generate reader methods for serialized attributes. #6362 [Stefan Kaes]

- Deprecation: object transactions warning. [Jeremy Kemper]
- has_one :dependent => :nullify ignores nil associates. #4848, #6528 [bellis@deepthought.org, janovetz, Jeremy Kemper]
- Oracle: resolve test failures, use prefetched primary key for inserts, check for null defaults, fix limited id selection for eager loading. Factor out some common methods from all adapters. #6515 [Michael Schoen]
- Make add_column use the options hash with the Sqlite Adapter. Closes #6464 [obrie]
- Document other options available to migration's add_column. #6419 [grg]
- MySQL: all_hashes compatibility with old MysqlRes class. #6429, #6601 [Jeremy Kemper]
- Fix has_many :through to add the appropriate conditions when going through an association using STI. Closes #5783. [Jonathan Viney]
- fix select_limited_ids_list issues in postgresql, retain current behavior in other adapters [Rick]
- Restore eager condition interpolation, document it's differences [Rick]
- Don't rollback in teardown unless a transaction was started. Don't start a transaction in create_fixtures if a transaction is started. #6282 [Jacob Fugal, Jeremy Kemper]
- Add #delete support to has_many :through associations. Closes #6049 [Martin Landers]
- Reverted old select_limited_ids_list postgresql fix that caused issues in mysql. Closes #5851 [Rick]
- Removes the ability for eager loaded conditions to be interpolated, since there is no model instance to use as a context for interpolation. #5553 [turnip@turnipspatch.com]

- Added timeout option to SQLite3 configurations to deal more gracefully with SQLite3::BusyException, now the connection can instead retry for x seconds to see if the db clears up before throwing that exception #6126 [wreese@gmail.com]

- Added update_attributes! which uses save! to raise an exception if a validation error prevents saving #6192 [jonathan]

- Deprecated add_on_boundary_breaking (use validates_length_of instead) #6292 [BobSilva]

- The has_many create method works with polymorphic associations. #6361 [Dan Peterson]

- MySQL: introduce Mysql::Result#all_hashes to support further optimization. #5581 [Stefan Kaes]

- save! shouldn't validate twice. #6324 [maiha, Bob Silva]

- Association collections have an _ids reader method to match the existing writer for collection_select convenience (e.g. employee.task_ids). The writer method skips blank ids so you can safely do @employee.task_ids = params[:tasks] without checking every time for an empty list or blank values. #1887, #5780 [Michael Schuerig]

- Add an attribute reader method for ActiveRecord::Base.observers [Rick Olson]

- Deprecation: count class method should be called with an options hash rather than two args for conditions and joins. #6287 [Bob Silva]

- has_one associations with a nil target may be safely marshaled. #6279 [norbauer, Jeremy Kemper]

- Duplicate the hash provided to AR::Base#to_xml to prevent unexpected side effects [Koz]

- Add a :namespace option to AR::Base#to_xml [Koz]

- Deprecation tests. Remove warnings for dynamic finders and for the foo_count method if it's also an attribute. [Jeremy Kemper]

- Mock Time.now for more accurate Touch mixin tests. #6213 [Dan Peterson]
- Improve yaml fixtures error reporting. #6205 [Bruce Williams]
- Rename AR::Base#quote so people can use that name in their models. #3628 [Koz]
- Add deprecation warning for inferred foreign key. #6029 [Josh Susser]
- Fixed the Ruby/MySQL adapter we ship with Active Record to work with the new authentication handshake that was introduced in MySQL 4.1, along with the other protocol changes made at that time #5723 [jimw@mysql.com]
- Deprecation: use :dependent => :delete_all rather than :exclusively_dependent => true. #6024 [Josh Susser]
- Document validates_presences_of behavior with booleans: you probably want validates_inclusion_of :attr, :in => [true, false]. #2253 [Bob Silva]
- Optimistic locking: gracefully handle nil versions, treat as zero. #5908 [Tom Ward]
- to_xml: the :methods option works on arrays of records. #5845 [Josh Starcher]
- Deprecation: update docs. #5998 [jakob@mentalized.net, Kevin Clark]
- Add some XmlSerialization tests for ActiveRecord [Rick Olson]
- has_many :through conditions are sanitized by the associating class. #5971 [martin.emde@gmail.com]
- Tighten rescue clauses. #5985 [james@grayproductions.net]
- Fix spurious newlines and spaces in AR::Base#to_xml output [Jamis Buck]
- has_one supports the :dependent => :delete option which skips the

typical callback chain and deletes the associated object directly from the database. #5927 [Chris Mear, Jonathan Viney]

- Nested subclasses are not prefixed with the parent class' table_name since they should always use the base class' table_name. #5911 [Jonathan Viney]
- SQLServer: work around bug where some unambiguous date formats are not correctly identified if the session language is set to german. #5894 [Tom Ward, kruth@bfpi]
- SQLServer: fix eager association test. #5901 [Tom Ward]
- Clashing type columns due to a sloppy join shouldn't wreck single-table inheritance. #5838 [Kevin Clark]
- Fixtures: correct escaping of \n and \r. #5859 [evgeny.zislis@gmail.com]
- Migrations: gracefully handle missing migration files. #5857 [eli.gordon@gmail.com]
- MySQL: update test schema for MySQL 5 strict mode. #5861 [Tom Ward]
- to_xml: correct naming of included associations. #5831 [josh.starcher@gmail.com]
- Pushing a record onto a has_many :through sets the association's foreign key to the associate's primary key and adds it to the correct association. #5815, #5829 [josh@hasmanythrough.com]
- Add records to has_many :through using <<, push, and concat by creating the association record. Raise if base or associate are new records since both ids are required to create the association. #build raises since you can't associate an unsaved record. #create! takes an attributes hash and creates the associated record and its association in a transaction. [Jeremy Kemper]

```
# Create a tagging to associate the post and tag.
post.tags << Tag.find_by_name('old')
post.tags.create! :name => 'general'
```

```
# Would have been:
post.taggings.create!(:tag => Tag.find_by_name('finally')
transaction do
  post.taggings.create!(:tag => Tag.create!(:name =>
'general'))
end
```

- Cache nil results for :included has_one associations also. #5787 [Michael Schoen]
- Fixed a bug which would cause .save to fail after trying to access a empty has_one association on a unsaved record. [Tobias Luetke]
- Nested classes are given table names prefixed by the singular form of the parent's table name. [Jeremy Kemper]
- Example: Invoice::Lineitem is given table name invoice_lineitems
- Migrations: uniquely name multicolumn indexes so you don't have to. [Jeremy Kemper]

```
# people_active_last_name_index, people_active_deactivated_
at_index
add_index    :people, [:active, :last_name]
add_index    :people, [:active, :deactivated_at]
remove_index :people, [:active, :last_name]
remove_index :people, [:active, :deactivated_at]
# WARNING: backward-incompatibility.
# Multicolumn indexes created before this
# revision were named using the first column name only.
# Now they're uniquely named using all indexed columns.
#
# To remove an old multicolumn index:
remove_index :table_name, :first_column
```

- Fix for deep includes on the same association. [richcollins@gmail.com]
- Tweak fixtures so they don't try to use a non-ActiveRecord class. [Kevin Clark]

- Remove ActiveRecord::Base.reset since Dispatcher doesn't use it anymore. [Rick Olson]
- Document find's :from option. Closes #5762. [andrew@redlinesoftware.com]
- PostgreSQL: autodetected sequences work correctly with multiple schemas. Rely on the schema search_path instead of explicitly qualifying the sequence name with its schema. #5280 [guy.naor@famundo.com]
- Replace Reloadable with Reloadable::Deprecated. [Nicholas Seckar]
- Cache nil results for has_one associations so multiple calls don't call the database. Closes #5757. [Michael A. Schoen]
- Add documentation for how to disable timestamps on a per model basis. Closes #5684. [matt@mattmargolis.net Marcel Molina Jr.]
- Don't save has_one associations unnecessarily. #5735 [Jonathan Viney]
- Refactor ActiveRecord::Base.reset_subclasses to #reset, and add global observer resetting. [Rick Olson]
- Formally deprecate the deprecated finders. [Koz]
- Formally deprecate rich associations. [Koz]
- Fixed that default timezones for new / initialize should uphold utc setting #5709 [daniluk@yahoo.com]
- Fix announcement of very long migration names. #5722 [blake@near-time.com]
- The exists? class method should treat a string argument as an id rather than as conditions. #5698 [jeremy@planetargon.com]
- Fixed to_xml with :include misbehaviors when invoked on array of model instances #5690 [alexkwolfe@gmail.com]
- Added support for conditions on Base.exists? #5689 [josh@josh-

peek.com]. Examples:

```
assert (Topic.exists?(:author_name => "David"))
assert (Topic.exists?(:author_name => "Mary", :approved =>
true))
assert (Topic.exists?(["parent_id = ?", 1]))
```

- Schema dumper quotes date :default values. [Dave Thomas]
- Calculate sum with SQL, not Enumerable on HasManyThrough Associations. [Dan Peterson]
- Factor the attribute#{suffix} methods out of method_missing for easier extension. [Jeremy Kemper]
- Patch sql injection vulnerability when using integer or float columns. [Jamis Buck]
- Allow #count through a has_many association to accept :include. [Dan Peterson]
- create_table rdoc: suggest :id => false for habtm join tables. [Zed Shaw]
- PostgreSQL: return array fields as strings. #4664 [Robby Russell]
- SQLServer: added tests to ensure all database statements are closed, refactored identity_insert management code to use blocks, removed update/delete rowcount code out of execute and into update/delete, changed insert to go through execute method, removed unused quoting methods, disabled pessimistic locking tests as feature is currently unsupported, fixed RakeFile to load sqlserver specific tests whether running in ado or odbc mode, fixed support for recently added decimal types, added support for limits on integer types. #5670 [Tom Ward]
- SQLServer: fix db:schema:dump case-sensitivity. #4684 [Will Rogers]
- Oracle: BigDecimal support. #5667 [schoenm@earthlink.net]

- Numeric and decimal columns map to BigDecimal instead of Float. Those with scale 0 map to Integer. #5454 [robbat2@gentoo.org, work@ashleymoran.me.uk]
- Firebird migrations support. #5337 [Ken Kunz <kennethkunz@gmail.com>]
- PostgreSQL: create/drop as postgres user. #4790 [mail@matthew-painter.co.uk, mlaster@metavillage.com]
- Update callbacks documentation. #3970 [Robby Russell robby@planetargon.com]
- PostgreSQL: correctly quote the ' in pk_and_sequence_for. #5462 [tietew@tietew.net]
- PostgreSQL: correctly quote microseconds in timestamps. #5641 [rick@rickbradley.com]
- Clearer has_one/belongs_to model names (account has_one :user). #5632 [matt@mattmargolis.net]
- Oracle: use nonblocking queries if allow_concurrency is set, fix pessimistic locking, don't guess date vs. time by default (set OracleAdapter.emulate_dates = true for the old behavior), adapter cleanup. #5635 [schoenm@earthlink.net]
- Fixed a few Oracle issues: Allows Oracle's odd date handling to still work consistently within #to_xml, Passes test that hardcode insert statement by dropping the :id column, Updated RUNNING_UNIT_TESTS with Oracle instructions, Corrects method signature for #exec #5294 [schoenm@earthlink.net]
- Added :group to available options for finds done on associations #5516 [mike@michaeldewey.org]
- Minor tweak to improve performance of ActiveRecord::Base#to_param.
- Observers also watch subclasses created after they are declared. #5535 [daniels@pronto.com.au]

- Removed deprecated timestamps_gmt class methods. [Jeremy Kemper]

- rake build_mysql_database grants permissions to rails@localhost. #5501 [brianegge@yahoo.com]

- PostgreSQL: support microsecond time resolution. #5492 [alex@msgpad.com]

- Add AssociationCollection#sum since the method_missing invocation has been shadowed by Enumerable#sum.

- Added find_or_initialize_by_X which works like find_or_create_by_X but doesn't save the newly instantiated record. [Sam Stephenson]

- Row locking. Provide a locking clause with the :lock finder option or true for the default "FOR UPDATE". Use the #lock! method to obtain a row lock on a single record (reloads the record with :lock => true). [Shugo Maeda]

```
# Obtain an exclusive lock on person 1
# so we can safely increment visits.
Person.transaction do
  # select * from people where id=1 for update
  person = Person.find(1, :lock => true)
  person.visits += 1
  person.save!
end
```

- PostgreSQL: introduce allow_concurrency option which determines whether to use blocking or asynchronous #execute. Adapters with blocking #execute will deadlock Ruby threads. The default value is ActiveRecord::Base.allow_concurrency. [Jeremy Kemper]

- Use a per-thread (rather than global) transaction mutex so you may execute concurrent transactions on separate connections. [Jeremy Kemper]

- Change AR::Base#to_param to return a String instead of a Fixnum. Closes #5320. [Nicholas Seckar]

- Use explicit delegation instead of method aliasing for AR::Base.to_param → AR::Base.id. #5299 (skaes@web.de)
- Refactored ActiveRecord::Base.to_xml to become a delegate for XmlSerializer, which restores sanity to the mega method. This refactoring also reinstates the opinions that type="string" is redundant and ugly and nil-differentiation is not a concern of serialization [DHH]
- Added simple hash conditions to find that'll just convert hash to an AND-based condition string #5143 [hcatlin@gmail.com]. Example:

```
Person.find(:all, :conditions => { :last_name => "Catlin",
:status => 1 }, :limit => 2)
# is the same as:
Person.find(:all, :conditions => [ "last_name = ? and status
= ?", "Catlin", 1 ], :limit => 2)
# This makes it easier to pass in the options from a form
# or otherwise outside.
```

- Fixed issues with BLOB limits, charsets, and booleans for Firebird #5194, #5191, #5189 [kennethkunz@gmail.com]
- Fixed usage of :limit and with_scope when the association in scope is a 1:m #5208 [alex@purefiction.net]
- Fixed migration trouble with SQLite when NOT NULL is used in the new definition #5215 [greg@lapcominc.com]
- Fixed problems with eager loading and counting on SQL Server #5212 [kajism@yahoo.com]
- Fixed that count distinct should use the selected column even when using :include #5251 [anna@wota.jp]
- Fixed that :includes merged from with_scope won't cause the same association to be loaded more than once if repetition occurs in the clauses #5253 [alex@purefiction.net]
- Allow models to override to_xml. #4989 [Blair Zajac <blair@orcaware.com>]

- PostgreSQL: don't ignore port when host is nil since it's often used to label the domain socket. #5247 [shimbo@is.naist.jp]
- Records and arrays of records are bound as quoted ids. [Jeremy Kemper]

```
Foo.find(:all, :conditions => ['bar_id IN (?)', bars])
Foo.find(:first, :conditions => ['bar_id = ?', bar])
```

- Fixed that Base.find :all, :conditions => ["id IN (?)", collection] would fail if collection was empty [DHH]
- Add a list of regexes assert_queries skips in the ActiveRecord test suite. [Rick]
- Fix the has_and_belongs_to_many #create doesn't populate the join for new records. Closes #3692 [josh@hasmanythrough.com]
- Provide Association Extensions access to the instance that the association is being accessed from. Closes #4433 [josh@hasmanythrough.com]
- Update OpenBase adaterp's maintainer's email address. Closes #5176. [Derrick Spell]
- Add a quick note about :select and eagerly included associations. [Rick]
- Add docs for the :as option in has_one associations. Closes #5144 [cdcarter@gmail.com]
- Fixed that has_many collections shouldn't load the entire association to do build or create [DHH]
- Added :allow_nil option for aggregations #5091 [ian.w.white@gmail.com]
- Fix Oracle boolean support and tests. Closes #5139. [schoenm@earthlink.net]
- create! no longer blows up when no attributes are passed and a

:create scope is in effect (e.g. foo.bars.create! failed whereas foo.bars.create!({}) didn't.) [Jeremy Kemper]

- Call Inflector#demodulize on the class name when eagerly including an STI model. Closes #5077 [info@loobmedia.com]
- Preserve MySQL boolean column defaults when changing a column in a migration. Closes #5015. [pdcawley@bofh.org.uk]
- PostgreSQL: migrations support :limit with :integer columns by mapping limit < 4 to smallint, > 4 to bigint, and anything else to integer. #2900 [keegan@thebasement.org]
- Dates and times interpret empty strings as nil rather than 2000-01-01. #4830 [kajism@yahoo.com]
- Allow :uniq => true with has_many :through associations. [Jeremy Kemper]
- Ensure that StringIO is always available for the Schema dumper. [Marcel Molina Jr.]
- Allow AR::Base#to_xml to include methods too. Closes #4921. [johan@textdrive.com]
- Replace superfluous name_to_class_name variant with camelize. [Marcel Molina Jr.]
- Replace alias method chaining with Module#alias_method_chain. [Marcel Molina Jr.]
- Replace Ruby's deprecated append_features in favor of included. [Marcel Molina Jr.]
- Remove duplicate fixture entry in comments.yml. Closes #4923. [Blair Zajac <blair@orcaware.com>]
- Update FrontBase adapter to check binding version. Closes #4920. [mlaster@metavillage.com]
- New Frontbase connections don't start in auto-commit mode. Closes #4922. [mlaster@metavillage.com]

- When grouping, use the appropriate option key. [Marcel Molina Jr.]
- Only modify the sequence name in the FrontBase adapter if the FrontBase adapter is actually being used. [Marcel Molina Jr.]
- Add support for FrontBase (http://www.frontbase.com/) with a new adapter thanks to the hard work of one Mike Laster. Closes #4093. [mlaster@metavillage.com]
- Add warning about the proper way to validate the presence of a foreign key. Closes #4147. [Francois Beausoleil <francois.beausoleil@gmail.com>]
- Fix syntax error in documentation. Closes #4679. [mislav@nippur.irb.hr]
- Add Oracle support for CLOB inserts. Closes #4748. [schoenm@earthlink.net sandra.metz@duke.edu]
- Various fixes for sqlserver_adapter (odbc statement finishing, ado schema dumper, drop index). Closes #4831. [kajism@yahoo.com]
- Add support for :order option to with_scope. Closes #3887. [eric.daspet@survol.net]
- Prettify output of schema_dumper by making things line up. Closes #4241 [Caio Chassot <caio@v2studio.com>]
- Make build_postgresql_databases task make databases owned by the postgres user. Closes #4790. [mlaster@metavillage.com]
- Sybase Adapter type conversion cleanup. Closes #4736. [dev@metacasa.net]
- Fix bug where calculations with long alias names return null. [Rick]
- Raise error when trying to add to a has_many :through association. Use the Join Model instead. [Rick]

```
@post.tags << @tag # BAD
@post.taggings.create(:tag => @tag) # GOOD
```

- Allow all calculations to take the :include option, not just COUNT (closes #4840) [Rick]
- Update inconsistent migrations documentation. #4683 [machomagna@gmail.com]
- Add ActiveRecord::Errors#to_xml [Jamis Buck]
- Properly quote index names in migrations (closes #4764) [John Long]
- Fix the HasManyAssociation#count method so it uses the new ActiveRecord::Base#count syntax, while maintaining backwards compatibility. [Rick]
- Ensure that Associations#include_eager_conditions? checks both scoped and explicit conditions [Rick]
- Associations#select_limited_ids_list adds the ORDER BY columns to the SELECT DISTINCT List for postgresql. [Rick]
- DRY up association collection reader method generation. [Marcel Molina Jr.]
- DRY up and tweak style of the validation error object. [Marcel Molina Jr.]
- Add :case_sensitive option to validates_uniqueness_of (closes #3090) [Rick]

```
class Account < ActiveRecord::Base
  validates_uniqueness_of :email,
                          :case_sensitive => false
end
```

- Allow multiple association extensions with :extend option (closes #4666) [Josh Susser]

```
class Account < ActiveRecord::Base
  has_many :people, :extend => [
    FindOrCreateByNameExtension, FindRecentExtension]
end
```

ActiveSupport

SVN

- Hash#symbolize_keys behaves well with integer keys. #9890 [PotatoSalad]
- Multibyte: String#slice supports regexp argument. #9646 [yob]
- object.duplicable? returns true if object.dup is safe. False for nil, true, false, symbols, and numbers; true otherwise. #9333 [sur]
- Time, Date and DateTime #advance accept :weeks option. #9866 [Geoff Buesing]
- Fix Time#years_ago and #years_since from leap days. #9865 [Geoff Buesing]
- Time and DateTime#advance accept :hours, :minutes, and :seconds options. #9825 [Geoff Buesing]
- Fix Date#years_ago and #years_since from leap days. #9864 [Geoff Buesing]
- Refactor Time and Date#months_since and #months_ago to use #advance. #9863 [Geoff Buesing]
- Rebundle Builder 2.1.2 but prefer a newer RubyGem if available. [Jeremy Kemper]
- Add Range#overlaps?(range), Range#include?(range), and Range#step without a block. [brandon]
- Correct BufferedLogger#level? checks. #9806 [wildchild, Johan Sorensen]
- String#to_xs uses Eric Wong's fast_xs extension, if available, for Builder speedup. http://bogomips.org/fast_xs/ [Jeremy Kemper]

- Introduce BasicObject as Builder::BlankSlate for Ruby 1.9 forward compatibility. [Jeremy Kemper]
- Unbundle Builder in favor of a gem dependency. [Jeremy Kemper]
- Disambiguate Time, Date, and DateTime#to_json formatting. #9750 [Geoff Buesing, Chu Yeow]
- Hash#to_json takes :only or :except options to specific or omit certain hash keys. Enumerable#to_json passes through its options to each element. #9751 [Chu Yeow]
- BufferedLogger#auto_flushing = N flushes the log every N messages. Buffers with an array instead of string. Disabling auto_flushing still flushes when the buffer hits a maximum size, as a failsafe against memory-gobbling. [Jeremy Kemper]
- Fixed Date#xmlschema for dates outside the range of what can be created with Time #9744 [Geoff Buesing]

2.0.0 [PREVIEW RELEASE] (SEPTEMBER 29TH, 2007)

- Fixed that La Paz was included in -25200 and -14400 offsets when it should only be in -14400 #9735 [bermi]
- Fixed JSON encoding to use quoted keys according to the JSON standard #8762 [choonkat/chuyeow]
- Alias Object#send to send! for Ruby 1.9 forward compatibility. [Jeremy Kemper]
- Backport Object#instance_variable_defined? for Ruby < 1.8.6. [Jeremy Kemper]
- BufferedLogger#add converts the message to a string. #9702, #9724 [eigentone, DrMark, tomafro]
- Added ActiveSupport::BufferedLogger as a duck-typing alternative (albeit with no formatter) to the Ruby Logger, which provides a very nice speed bump (inspired by Ezra's buffered logger) [DHH]

- Object#instance_exec produces fewer garbage methods. [Mauricio Fernandez]
- Decode json strings as Dates/Times if they're using a YAML-compatible format. Closes #9614 [Rick]
- Fixed cache_page to use the request url instead of the routing options when picking a save path #8614 [josh]
- Object.subclasses_of includes anonymous subclasses. [Jeremy Kemper]
- Fixed that pluralizing an empty string should return the same empty string, not "s" #7720 [josh]
- Added call to inspect on non-string classes for the logger #8533 [codahale]
- Deprecation: remove deprecated :mday option from Time, Date, and DateTime#change. [Jeremy Kemper]
- Fix JSON decoder with nested quotes and commas. #9579 [zdenis]
- Hash#to_xml doesn't double-unescape. #8806 [Ezran]
- Added Array#rand #9170 [norbert]. Examples:

```
[].rand       # => nil
['a'].rand    # => 'a'
[1,2,3].rand  # => 1 or 2 or 3
```

- Deprecation: removed Reloadable. [Jeremy Kemper]
- Make the utf-handler return the correct value for non-matching regular expressions. Closes #9049 [manfred]
- Add ljust, rjust and center to utf8-handler. Closes #9165 [manfred]
- Fix Time#advance bug when trying to advance a year from leap day. Closes #8655 [gbuesing]
- Add support for []= on ActiveSupport::Multibyte::Chars. Closes

#9142. [ewan, manfred]

- Added Array#extract_options! to encapsulate the pattern of getting an options hash out of a variable number of parameters #8759 [norbert].
- Let alias_attribute work with attributes with initial capital letters (legacy columns etc). Closes #8596 [mpalmer]
- Added Hash#except which is the inverse of Hash#slice—return the hash except the keys that are specified [DHH]
- Added support for pluralization with a different starting letter than the singular version (cow/kine) #4929 [norri_b/hasmanyjosh]
- Demote Hash#to_xml to use XmlSimple#xml_in_string so it can't read files or stdin. #8453 [candlerb, Jeremy Kemper]
- Backport clean_logger changes to support ruby 1.8.2 [mislav]
- Added proper handling of arrays #8537 [hasmanyjosh]

```
# Before:
Hash.from_xml '<images></images>'
# => {:images => nil}

Hash.from_xml '<images><image>foo.jpg</image></images>'
# => {:images => {:image => "foo.jpg"}}

Hash.from_xml '<images><image>foo.jpg</image><image>bar.
jpg</image></images>'
# => {:images => {:image => ["foo.jpg", "bar.jpg"]}}

# After:
Hash.from_xml '<images type="array"></images>'
# => {:images => []}

Hash.from_xml '<images type="array"><image>foo.jpg</image></
images>'
# => {:images => ["foo.jpg"]}

Hash.from_xml '<images type="array"><image>foo.jpg</
image><image>bar.jpg</image></images>'
# => {:images => ["foo.jpg", "bar.jpg"]}
```

- Improve Time and Date test coverage. #8646 [Josh Peek]
- Add Date#since, ago, beginning_of_day, and end_of_day. Date + seconds works now. #8575 [Geoff Buesing]
- String#to_time overflows to DateTime. Add String#to_datetime. #8572 [Geoff Buesing]
- Date.yesterday and .tomorrow. #8571 [Geoff Buesing]
- Readable Date and DateTime#inspect. #8570 [Geoff Buesing]
- Move common DateTime calculations to Date. #8536 [Geoff Buesing]
- Added Date#change (like Time#change) [DHH]
- DateTime#to_time converts to Time unless out of range. #8512 [Geoff Buesing]
- Date#to_datetime, #to_s(:rfc822). #8512 [Geoff Buesing]
- Time durations use since instead of + for accuracy. #8513 [Geoff Buesing]
- escape <'s and >'s in JSON strings. #8371 [Rick]
- Inflections: MatrixTest → MatrixTests instead of MatricesTest. #8496 [jbwiv]
- Multibyte strings respond_to the String methods they proxy so they can be duck-typed. #6549 [Tuxie]
- Array#to_xml yields the builder just like Hash and ActiveRecord::Base. #8472 [seth]
- Date, Time, and DateTime support formatting blocks in addition to strftime strings. Introduce :long_ordinal format, e.g. "February 21st, 2005". #8191 [Coda Hale]
- Document Object#blank?. #6491 [Chris Mear]

- Date, Time, and DateTime#to_json. #8399 [wycats]
- Simplify API of assert_difference by passing in an expression that is evaluated before and after the passed in block. See documenation for examples of new API. [Marcel Molina Jr.]
- Added assert_difference and assert_no_difference to test/unit assertions [Tobias Luetke]
- Removed breakpointer and Binding.of_caller in favor of relying on ruby-debug by Kent Sibilev since the breakpointer has been broken since Ruby 1.8.4 and will not be coming back [DHH]
- Added parsing of file type in Hash.xml_in so you can easily do file uploads with base64 from an API [DHH]

```
# <person>
#   <name>David</name>
#   <avatar type="file" name="me.jpg"
#           content_type="image/jpg">R0lGODlhkA</avatar>
# </person>

# ...becomes:

attributes = { :person => { :name => "David", :avatar =>
"#<StringIO>" } }
attributes[:person][:avatar].content_type      # => "image/
jpg"
attributes[:person][:avatar].original_filename # => "me.jpg"
attributes[:person][:avatar].read # => binary data of the
file

# Which is duck-type compatible with the files
# that you get when doing multipart uploads through HTML.
```

- Improved multibyte performance by relying less on exception raising #8159 [Blaine]
- Use XSD-compatible type names for Hash#to_xml and make the converters extendable #8047 [Tim Pope]
- Added yielding of builder in Hash#to_xml [DHH]

- Hash#with_indifferent_access now also converts hashes kept in arrays to indifferent access (makes it easier to treat HTML and XML parameters the same) [DHH]
- Hash#to_xml supports YAML attributes. #7502 [jonathan]
- Refactor ActiveSupport::JSON to be less obtuse. Add support for JSON decoding by way of Syck with ActiveSupport::JSON.decode(json_string). Prevent hash keys that are JavaScript reserved words from being unquoted during encoding. [Sam Stephenson]
- alias_method_chain preserves the original method's visibility. #7854 [Jonathan Viney]
- Update Dependencies to ignore constants inherited from ancestors. Closes #6951. [Nicholas Seckar]
- Array#to_query preserves its ordering. #7756 [Greg Spurrier]
- Out-of-range Time calculations transparently overflow to DateTime. Introduce Time#to_datetime. #7706, #7715 [Geoff Buesing]
- DateTime calculations analogous to the Date and Time extensions. #7693 [Geoff Buesing]
- Give DateTime correct .to_s implementations, lets it play nice with ActiveRecord quoting. #7649 [Geoff Buesing]
- Add File.atomic_write, allows you to write large files in an atomic manner, preventing users from seeing half written files. [Koz]
- Allow users to provide custom formatters to Logger. [aeden]
- Hash#to_query CGI-escapes its keys. [Jeremy Kemper]
- Optimize Class Inheritable Attributes so that unnecessary hashes are not created. Closes #7472 [Bruce Perens]
- :db format for Date#to_s [Jeremy Kemper]

  ```
  Date.new(2007, 1, 27).to_s(:db) # => '2007-01-27'
  ```

- Added :instance_writer option to #mattr_writer/accessor, #cattr_writer/accessor, and #class_inheritable_writer to skip the creation of the instance writer. [Rick]
- Added Hash#to_query to turn a hash of values into a form-encoded query string [Nicholas Seckar]
- Increase test coverage for subclasses_of. Closes #7335. [Roman2K, Nicholas Seckar]
- Remove unused code from Duration#inspect. Closes #7180. [Rich Collins]
- Added test coverage for Inflector.inflections.clear. Closes #7179. [Rich Collins]
- ActiveSupport::Multibyte::Handlers::UTF8Handler should raise when a range and an integer are passed in (just like the native implementation). Closes #7176 [Rich Collins]
- A couple extra tests for #classify. Closes #7273. [Josh Susser]
- Better docs for Object extensions [zackchandler, Jamis Buck]
- Fix that Dates couldn't be subtracted from Dates after [5940]. [Sam Stephenson]
- Add Object#acts_like? and Time#acts_like_time? and Date#acts_like_date? to facilitate duck-typing. [Jamis Buck]
- Make 1.months and friends accurate by introducing a Duration class. #6835 [eventualbuddha]

Revisions

CHAPTER 6

DECEMBER 3, 2007

- Minor bugfixes.

OCTOBER 31, 2007

- Added section about Foxy Fixtures.

OCTOBER 19, 2007

- Correctly identified forward-slash on p16.
- Added section on asset servers and asset tags.
- Added debugger example, with sample session.
- Added appendix with changelogs from ActionMailer, ActionPack, ActiveRecord, and ActiveSupport.

OCTOBER 17, 2007

- First release

Rails2: New features for your applications

Every effort was made to provide accurate information in this document. However, neither yFactorial, LLC nor Topfunky Corporation shall have any liability for any errors in the code or descriptions presented in this book.

"Rails" and "Ruby on Rails" are trademarks of David Heinemeier Hansson.

This document is available for US$9 at PeepCode.com (http://peepcode.com). Group discounts and site licenses can also be purchased by sending email to peepcode@topfunky.com.

OTHER PEEPCODE PRODUCTS

- RSpec (http://peepcode.com/products/rspec-basics) – A three part series on the popular behavior-driven development framework.
- Rails from Scratch (http://peepcode.com) – Learn Rails!
- RESTful Rails (http://peepcode.com/products/restful-rails) – Teaches the concepts of application design with REST.
- Subscription pack of 10 (http://peepcode.com/products/subscription-pack-of-10) – Save money! Buy 10 PeepCode credits.
- Javascript with Prototype (http://peepcode.com/products/javascript-with-prototypejs) – Code confidently with Javascript!
- Rails Code Review PDF (http://peepcode.com/products/draft-rails-code-review-pdf) – Common mistakes in Rails applications, and how to fix them.