Ruby on Rails 4.2 Release Notes

January 13, 2015

Highlights in Rails 4.2:

- Active Job
- Asynchronous mails
- Adequate Record
- Web Console
- Foreign key support

These release notes cover only the major changes. To learn about other features, bug fixes, and changes, please refer to the changelogs or check out the list of commits in the main Rails repository on GitHub.

1 Upgrading to Rails 4.2

If you're upgrading an existing application, it's a great idea to have good test coverage before going in. You should also first upgrade to Rails 4.1 in case you haven't and make sure your application still runs as expected before attempting to upgrade to Rails 4.2. A list of things to watch out for when upgrading is available in the guide Upgrading Ruby on Rails.

2 Major Features

2.1 Active Job

Active Job is a new framework in Rails 4.2. It is a common interface on top of queuing systems like Resque, Delayed Job, Sidekiq, and more.

Jobs written with the Active Job API run on any of the supported queues thanks to their respective adapters. Active Job comes pre-configured with an inline runner that executes jobs right away.

Jobs often need to take Active Record objects as arguments. Active Job passes object references as URIs (uniform resource identifiers) instead of marshaling the object itself. The new Global ID library builds URIs and looks up the objects they reference. Passing Active Record objects as job arguments just works by using Global ID internally.

For example, if `trashable` is an Active Record object, then this job runs just fine with no serialization involved:

```
class TrashableCleanupJob < ActiveJob::Base
  def perform(trashable, depth)
    trashable.cleanup(depth)
  end
end
```

See the Active Job Basics guide for more information.

2.2   Asynchronous Mails

Building on top of Active Job, Action Mailer now comes with a `deliver_later` method that sends emails via the queue, so it doesn't block the controller or model if the queue is asynchronous (the default inline queue blocks).

Sending emails right away is still possible with `deliver_now`.

2.3   Adequate Record

Adequate Record is a set of performance improvements in Active Record that makes common `find` and `find_by` calls and some association queries up to 2x faster.

It works by caching common SQL queries as prepared statements and reusing them on similar calls, skipping most of the query-generation work on subsequent calls. For more details, please refer to Aaron Patterson's blog post.

Active Record will automatically take advantage of this feature on supported operations without any user involvement or code changes. Here are some examples of supported operations:

```
Post.find(1)  # First call generates and cache the prepared statement
Post.find(2)  # Subsequent calls reuse the cached prepared statement

Post.find_by_title('first post')
Post.find_by_title('second post')

post.comments
post.comments(true)
```

It's important to highlight that, as the examples above suggest, the prepared statements do not cache the values passed in the method calls; rather, they have placeholders for them.

Caching is not used in the following scenarios:

- The model has a default scope
- The model uses single table inheritance
- `find` with a list of ids, e.g.:

```
# not cached
Post.find(1, 2, 3)
Post.find([1,2])
```

- `find_by` with SQL fragments:

```
Post.find_by('published_at < ?', 2.weeks.ago)
```

2.4   Web Console

New applications generated with Rails 4.2 now come with the Web Console gem by default. Web Console adds an interactive Ruby console on every error page and provides a `console` view and controller helpers.

The interactive console on error pages lets you execute code in the context of the place where the exception originated. The `console` helper, if called anywhere in a view or controller, launches an interactive console with the final context, once rendering has completed.

2.5   Foreign Key Support

The migration DSL now supports adding and removing foreign keys. They are dumped to `schema.rb` as well. At this time, only the `mysql`, `mysql2` and `postgresql` adapters support foreign keys.

```
# add a foreign key to 'articles.author_id' referencing 'authors.id'
add_foreign_key :articles, :authors

# add a foreign key to 'articles.author_id' referencing 'users.lng_id'
add_foreign_key :articles, :users, column: :author_id, primary_key: "lng_id"

# remove the foreign key on 'accounts.branch_id'
remove_foreign_key :accounts, :branches

# remove the foreign key on 'accounts.owner_id'
remove_foreign_key :accounts, column: :owner_id
```

See the API documentation on add_foreign_key and remove_foreign_key for a full description.

3   Incompatibilities

Previously deprecated functionality has been removed. Please refer to the individual components for new deprecations in this release.

The following changes may require immediate action upon upgrade.

3.1   `render` with a String Argument

Previously, calling `render "foo/bar"` in a controller action was equivalent to `render file: "foo/bar"`. In Rails 4.2, this has been changed to mean `render template: "foo/bar"` instead. If you need to render a file, please change your code to use the explicit form (`render file: "foo/bar"`) instead.

3.2   `respond_with` / Class-Level `respond_to`

`respond_with` and the corresponding class-level `respond_to` have been moved to the responders gem. Add `gem 'responders', '~> 2.0'` to your Gemfile to use it:

```
# app/controllers/users_controller.rb

class UsersController < ApplicationController
```

```
  respond_to :html, :json

  def show
    @user = User.find(params[:id])
    respond_with @user
  end
end
```

Instance-level `respond_to` is unaffected:

```
# app/controllers/users_controller.rb

class UsersController < ApplicationController
  def show
    @user = User.find(params[:id])
    respond_to do |format|
      format.html
      format.json { render json: @user }
    end
  end
end
```

3.3 Default Host for `rails server`

Due to a change in Rack, `rails server` now listens on `localhost` instead of `0.0.0.0` by default. This should have minimal impact on the standard development workflow as both http://127.0.0.1:3000 and `http://localhost:3000` will continue to work as before on your own machine.

However, with this change you will no longer be able to access the Rails server from a different machine, for example if your development environment is in a virtual machine and you would like to access it from the host machine. In such cases, please start the server with `rails server -b 0.0.0.0` to restore the old behavior.

If you do this, be sure to configure your firewall properly such that only trusted machines on your network can access your development server.

3.4 HTML Sanitizer

The HTML sanitizer has been replaced with a new, more robust, implementation built upon Loofah and Nokogiri. The new sanitizer is more secure and its sanitization is more powerful and flexible.

Due to the new algorithm, the sanitized output may be different for certain pathological inputs.

If you have a particular need for the exact output of the old sanitizer, you can add the rails-deprecated_sanitizer gem to the `Gemfile`, to have the old behavior. The gem does not issue deprecation warnings because it is opt-in.

`rails-deprecated_sanitizer` will be supported for Rails 4.2 only; it will not be maintained for Rails 5.0.

See this blog post for more details on the changes in the new sanitizer.

3.5 `assert_select`

`assert_select` is now based on Nokogiri. As a result, some previously-valid selectors are now unsupported. If your application is using any of these spellings, you will need to update them:

- Values in attribute selectors may need to be quoted if they contain non-alphanumeric characters.

  ```
  # before
  a[href=/]
  a[href$=/]

  # now
  a[href="/"]
  a[href$="/"]
  ```

- DOMs built from HTML source containing invalid HTML with improperly nested elements may differ. For example:

  ```
  # content: <div><i><p></i></div>

  # before:
  assert_select('div > i')  # => true
  assert_select('div > p')  # => false
  assert_select('i > p')    # => true

  # now:
  assert_select('div > i')  # => true
  assert_select('div > p')  # => true
  assert_select('i > p')    # => false
  ```

- If the data selected contains entities, the value selected for comparison used to be raw (e.g. `AT&T`), and now is evaluated (e.g. `AT&T`).

  ```
  # content: <p>AT&amp;T</p>

  # before:
  assert_select('p', 'AT&amp;T')  # => true
  assert_select('p', 'AT&T')      # => false

  # now:
  assert_select('p', 'AT&T')      # => true
  assert_select('p', 'AT&amp;T')  # => false
  ```

4 Railties

Please refer to the Changelog for detailed changes.

4.1 Removals

- The `--skip-action-view` option has been removed from the app generator. (Pull Request)

- The `rails application` command has been removed without replacement. (Pull Request)

4.2 Deprecations

- Deprecated missing `config.log_level` for production environments. (Pull Request)

- Deprecated `rake test:all` in favor of `rake test` as it now run all tests in the `test` folder. (Pull Request)

- Deprecated `rake test:all:db` in favor of `rake test:db`. (Pull Request)

- Deprecated `Rails::Rack::LogTailer` without replacement. (Commit)

4.3 Notable changes

- Introduced `web-console` in the default application Gemfile. (Pull Request)

- Added a `required` option to the model generator for associations. (Pull Request)

- Introduced the `x` namespace for defining custom configuration options:

```
# config/environments/production.rb
config.x.payment_processing.schedule = :daily
config.x.payment_processing.retries  = 3
config.x.super_debugger              = true
```

These options are then available through the configuration object:

```
Rails.configuration.x.payment_processing.schedule # => :daily
Rails.configuration.x.payment_processing.retries  # => 3
Rails.configuration.x.super_debugger              # => true
```

(Commit)

- Introduced `Rails::Application.config_for` to load a configuration for the current environment.

```
# config/exception_notification.yml:
production:
  url: http://127.0.0.1:8080
  namespace: my_app_production
development:
  url: http://localhost:3001
  namespace: my_app_development
```

```
# config/production.rb
Rails.application.configure do
  config.middleware.use ExceptionNotifier, config_for(:exception_notification)
end
```

(Pull Request)

- Introduced a `--skip-turbolinks` option in the app generator to not generate turbolinks integration. (Commit)

- Introduced a `bin/setup` script as a convention for automated setup code when bootstrapping an application. (Pull Request)

- Changed the default value for `config.assets.digest` to `true` in development. (Pull Request)

- Introduced an API to register new extensions for `rake notes`. (Pull Request)

- Introduced an `after_bundle` callback for use in Rails templates. (Pull Request)

- Introduced `Rails.gem_version` as a convenience method to return `Gem::Version.new(Rails.version)`. (Pull Request)

5   Action Pack

Please refer to the Changelog for detailed changes.

5.1   Removals

- `respond_with` and the class-level `respond_to` have been removed from Rails and moved to the `responders` gem (version 2.0). Add `gem 'responders', '~> 2.0'` to your `Gemfile` to continue using these features. (Pull Request, More Details)

- Removed deprecated `AbstractController::Helpers::ClassMethods::MissingHelperError` in favor of `AbstractController::Helpers::MissingHelperError`. (Commit)

5.2   Deprecations

- Deprecated the `only_path` option on `*_path` helpers. (Commit)

- Deprecated `assert_tag`, `assert_no_tag`, `find_tag` and `find_all_tag` in favor of `assert_select`. (Commit)

- Deprecated support for setting the `:to` option of a router to a symbol or a string that does not contain a "#" character:

```
get '/posts', to: MyRackApp    => (No change necessary)
get '/posts', to: 'post#index' => (No change necessary)
get '/posts', to: 'posts'      => get '/posts', controller: :posts
get '/posts', to: :index       => get '/posts', action: :index
```

(Commit)

- Deprecated support for string keys in URL helpers:

```
# bad
root_path('controller' => 'posts', 'action' => 'index')

# good
root_path(controller: 'posts', action: 'index')
```

(Pull Request)

5.3   Notable changes

- The `*_filter` family of methods have been removed from the documentation. Their usage is discouraged in favor of the `*_action` family of methods:

```
after_filter           => after_action
append_after_filter    => append_after_action
append_around_filter   => append_around_action
append_before_filter   => append_before_action
around_filter          => around_action
before_filter          => before_action
prepend_after_filter   => prepend_after_action
prepend_around_filter  => prepend_around_action
prepend_before_filter  => prepend_before_action
skip_after_filter      => skip_after_action
skip_around_filter     => skip_around_action
skip_before_filter     => skip_before_action
skip_filter            => skip_action_callback
```

If your application currently depends on these methods, you should use the replacement `*_action` methods instead. These methods will be deprecated in the future and will eventually be removed from Rails.

(Commit 1, 2)

- `render nothing: true` or rendering a `nil` body no longer add a single space padding to the response body. (Pull Request)

- Rails now automatically includes the template's digest in ETags. (Pull Request)

- Segments that are passed into URL helpers are now automatically escaped. (Commit)

- Introduced the `always_permitted_parameters` option to configure which parameters are permitted globally. The default value of this configuration is `['controller', 'action']`. (Pull Request)

- Added the HTTP method `MKCALENDAR` from RFC 4791. (Pull Request)

- *_fragment.action_controller notifications now include the controller and action name in the payload. (Pull Request)

- Improved the Routing Error page with fuzzy matching for route search. (Pull Request)

- Added an option to disable logging of CSRF failures. (Pull Request)

- When the Rails server is set to serve static assets, gzip assets will now be served if the client supports it and a pre-generated gzip file (`.gz`) is on disk. By default the asset pipeline generates `.gz` files for all compressible assets. Serving gzip files minimizes data transfer and speeds up asset requests. Always use a CDN if you are serving assets from your Rails server in production. (Pull Request)

- When calling the `process` helpers in an integration test the path needs to have a leading slash. Previously you could omit it but that was a byproduct of the implementation and not an intentional feature, e.g.:

```
test "list all posts" do
  get "/posts"
  assert_response :success
end
```

6   Action View

Please refer to the Changelog for detailed changes.

6.1   Deprecations

- Deprecated `AbstractController::Base.parent_prefixes`. Override `AbstractController::Base.local_prefixes` when you want to change where to find views. (Pull Request)

- Deprecated `ActionView::Digestor#digest(name, format, finder, options = {})`. Arguments should be passed as a hash instead. (Pull Request)

6.2   Notable changes

- `render "foo/bar"` now expands to `render template:  "foo/bar"` instead of `render file:  "foo/bar"`. (Pull Request)

- The form helpers no longer generate a <div> element with inline CSS around the hidden fields. (Pull Request)

- Introduced a `#{partial_name}_iteration` special local variable for use with partials that are rendered with a collection. It provides access to the current state of the iteration via the `index`, `size`, `first?` and `last?` methods. (Pull Request)

- Placeholder I18n follows the same convention as `label` I18n. (Pull Request)

7 Action Mailer

Please refer to the Changelog for detailed changes.

7.1 Deprecations

- Deprecated `*_path` helpers in mailers. Always use `*_url` helpers instead. (Pull Request)

- Deprecated `deliver` / `deliver!` in favor of `deliver_now` / `deliver_now!`. (Pull Request)

7.2 Notable changes

- `link_to` and `url_for` generate absolute URLs by default in templates, it is no longer needed to pass `only _path:  false`. (Commit)

- Introduced `deliver_later` which enqueues a job on the application's queue to deliver emails asynchronously. (Pull Request)

- Added the `show_previews` configuration option for enabling mailer previews outside of the development environment. (Pull Request)

8 Active Record

Please refer to the Changelog for detailed changes.

8.1 Removals

- Removed `cache_attributes` and friends. All attributes are cached. (Pull Request)

- Removed deprecated method `ActiveRecord::Base.quoted_locking_column`. (Pull Request)

- Removed deprecated `ActiveRecord::Migrator.proper_table_name`. Use the `proper_table_name` instance method on `ActiveRecord::Migration` instead. (Pull Request)

- Removed unused `:timestamp` type. Transparently alias it to `:datetime` in all cases. Fixes inconsistencies when column types are sent outside of Active Record, such as for XML serialization. (Pull Request)

8.2 Deprecations

- Deprecated swallowing of errors inside `after_commit` and `after_rollback`. (Pull Request)

- Deprecated broken support for automatic detection of counter caches on `has_many :through` associations. You should instead manually specify the counter cache on the `has_many` and `belongs_to` associations for the through records. (Pull Request)

- Deprecated passing Active Record objects to `.find` or `.exists?`. Call `id` on the objects first. (Commit 1, 2)

- Deprecated half-baked support for PostgreSQL range values with excluding beginnings. We currently map PostgreSQL ranges to Ruby ranges. This conversion is not fully possible because Ruby ranges do not support excluded beginnings.

  The current solution of incrementing the beginning is not correct and is now deprecated. For subtypes where we don't know how to increment (e.g. `succ` is not defined) it will raise an `ArgumentError` for ranges with excluding beginnings. (Commit)

- Deprecated calling `DatabaseTasks.load_schema` without a connection. Use `DatabaseTasks.load_schema_current` instead. (Commit)

- Deprecated `sanitize_sql_hash_for_conditions` without replacement. Using a `Relation` for performing queries and updates is the preferred API. (Commit)

- Deprecated `add_timestamps` and `t.timestamps` without passing the `:null` option. The default of `null: true` will change in Rails 5 to `null:  false`. (Pull Request)

- Deprecated `Reflection#source_macro` without replacement as it is no longer needed in Active Record. (Pull Request)

- Deprecated `serialized_attributes` without replacement. (Pull Request)

- Deprecated returning `nil` from `column_for_attribute` when no column exists. It will return a null object in Rails 5.0. (Pull Request)

- Deprecated using `.joins`, `.preload` and `.eager_load` with associations that depend on the instance state (i.e. those defined with a scope that takes an argument) without replacement. (Commit)

8.3   Notable changes

- `SchemaDumper` uses `force:  :cascade` on `create_table`. This makes it possible to reload a schema when foreign keys are in place.

- Added a `:required` option to singular associations, which defines a presence validation on the association. (Pull Request)

- `ActiveRecord::Dirty` now detects in-place changes to mutable values. Serialized attributes on Active Record models are no longer saved when unchanged. This also works with other types such as string columns and json columns on PostgreSQL. (Pull Requests 1, 2, 3)

- Introduced the `db:purge` Rake task to empty the database for the current environment. (Commit)

- Introduced `ActiveRecord::Base#validate!` that raises `ActiveRecord::RecordInvalid` if the record is invalid. (Pull Request)

- Introduced `validate` as an alias for `valid?`. (Pull Request)

- `touch` now accepts multiple attributes to be touched at once. (Pull Request)

- The PostgreSQL adapter now supports the `jsonb` datatype in PostgreSQL 9.4+. (Pull Request)

- The PostgreSQL and SQLite adapters no longer add a default limit of 255 characters on string columns. (Pull Request)

- Added support for the `citext` column type in the PostgreSQL adapter. (Pull Request)

- Added support for user-created range types in the PostgreSQL adapter. (Commit)

- `sqlite3:///some/path` now resolves to the absolute system path `/some/path`. For relative paths, use `sqlite3:some/path` instead. (Previously, `sqlite3:///some/path` resolved to the relative path `some/path`. This behavior was deprecated on Rails 4.1). (Pull Request)

- Added support for fractional seconds for MySQL 5.6 and above. (Pull Request 1, 2)

- Added `ActiveRecord::Base#pretty_print` to pretty print models. (Pull Request)

- `ActiveRecord::Base#reload` now behaves the same as `m = Model.find(m.id)`, meaning that it no longer retains the extra attributes from custom `SELECT`s. (Pull Request)

- `ActiveRecord::Base#reflections` now returns a hash with string keys instead of symbol keys. (Pull Request)

- The `references` method in migrations now supports a `type` option for specifying the type of the foreign key (e.g. `:uuid`). (Pull Request)

9   Active Model

Please refer to the Changelog for detailed changes.

9.1   Removals

- Removed deprecated `Validator#setup` without replacement. (Pull Request)

9.2   Deprecations

- Deprecated `reset_#{attribute}` in favor of `restore_#{attribute}`. (Pull Request)

- Deprecated `ActiveModel::Dirty#reset_changes` in favor of `clear_changes_information`. (Pull Request)

9.3   Notable changes

- Introduced `validate` as an alias for `valid?`. (Pull Request)

- Introduced the `restore_attributes` method in `ActiveModel::Dirty` to restore the changed (dirty) attributes to their previous values. (Pull Request 1, 2)

- `has_secure_password` no longer disallows blank passwords (i.e. passwords that contains only spaces) by default. (Pull Request)

- `has_secure_password` now verifies that the given password is less than 72 characters if validations are enabled. (Pull Request)

10   Active Support

Please refer to the Changelog for detailed changes.

10.1   Removals

- Removed deprecated `Numeric#ago`, `Numeric#until`, `Numeric#since`, `Numeric#from_now`. (Commit)

- Removed deprecated string based terminators for `ActiveSupport::Callbacks`. (Pull Request)

10.2   Deprecations

- Deprecated `Kernel#silence_stderr`, `Kernel#capture` and `Kernel#quietly` without replacement. (Pull Request)

- Deprecated `Class#superclass_delegating_accessor`, use `Class#class_attribute` instead. (Pull Request)

- Deprecated `ActiveSupport::SafeBuffer#prepend!` as `ActiveSupport::SafeBuffer#prepend` now performs the same function. (Pull Request)

10.3   Notable changes

- Introduced a new configuration option `active_support.test_order` for specifying the order test cases are executed. This option currently defaults to `:sorted` but will be changed to `:random` in Rails 5.0. (Commit)

- `Object#try` and `Object#try!` can now be used without an explicit receiver in the block. (Commit, Pull Request)

- The `travel_to` test helper now truncates the `usec` component to 0. (Commit)

- Introduced `Object#itself` as an identity function. (Commit 1, 2)

- `Object#with_options` can now be used without an explicit receiver in the block. (Pull Request)

- Introduced `String#truncate_words` to truncate a string by a number of words. (Pull Request)

- Added `Hash#transform_values` and `Hash#transform_values!` to simplify a common pattern where the values of a hash must change, but the keys are left the same. (Pull Request)

- The `humanize` inflector helper now strips any leading underscores. (Commit)

- Introduced `Concern#class_methods` as an alternative to `module ClassMethods`, as well as `Kernel#concern` to avoid the `module Foo; extend ActiveSupport::Concern; end` boilerplate. (Commit)

- New guide about constant autoloading and reloading.

11   Credits

See the full list of contributors to Rails for the many people who spent many hours making Rails the stable and robust framework it is today. Kudos to all of them.

12   Feedback

You're encouraged to help improve the quality of this guide.

Please contribute if you see any typos or factual errors. To get started, you can read our documentation contributions section.

You may also find incomplete content, or stuff that is not up to date. Please do add any missing documentation for master. Make sure to check Edge Guides first to verify if the issues are already fixed or not on the master branch. Check the Ruby on Rails Guides Guidelines for style and conventions.

If for whatever reason you spot something to fix but cannot patch it yourself, please open an issue.

And last but not least, any kind of discussion regarding Ruby on Rails documentation is very welcome in the rubyonrails-docs mailing list.