

ChiliProject - Feature # 290: Add bundler

Status:	Closed	Priority:	Normal
Author:	Eric Davis	Category:	Libraries
Created:	2011-03-18	Assignee:	Holger Just
Updated:	2011-05-13	Due date:	
Remote issue URL:			
Affected version:			
Description:	Use bundler to replace @config.gems@ and some bundled gems.		
	https://github.com/finnlabs/chiliproject-gemfile		

Associated revisions

2011-05-10 08:40 am - Gregor Schmidt

[#290] using bundler

- * following setup instructions from <http://gembundler.com/rails23.html>
- * adding generated files to .gitignore and .hgignore
- * removing config.gem calls
- * adding Gemfile that should work with all supported dbs and interpreters
- * removing work-around for rubygems 1.5 deprecation. this seems to be fixed since now bundler handles the gem loading

2011-05-10 08:40 am - Gregor Schmidt

[#290] ignoring .rvmrc files

2011-05-10 08:40 am - Gregor Schmidt

[#290] Removing all vendored gems

2011-05-10 08:40 am - Gregor Schmidt

[#290] loading mocha after shoulda to make sure, that mocha's patches can be properly applied

2011-05-10 08:40 am - Gregor Schmidt

[#290] making openid opt-out by adding it to a group

2011-05-10 08:40 am - Gregor Schmidt

[#290] Adding rmagick, making it opt-out

using version < 2.0 since later versions do not work w/o backports on Debian Lenny

2011-05-10 08:40 am - Gregor Schmidt

[#290] Remove test gems from dev enviroment.

running single test without RAILS_ENV=test prefix seams to work without it.

2011-05-10 08:40 am - Gregor Schmidt

[#290] using ~> instead of < for i18n dependency

it is more accurate in this case

2011-05-13 06:19 pm - Eric Davis

[#290] Only print the plugin Gemfile messages when \$DEBUG is set

History

2011-03-19 03:22 pm - Holger Just

@Gregor: Thanks for your pull request at <https://github.com/chiliproject/chiliproject/pull/29>.

A few things that occurred to me:

* @rmagic@ is missing.

* @ruby-openid@ is required unconditionally.

It might be a good idea to have optional gems in it own group each, so that people can decide which features they want. This could be supported by a installation rake task (or a simple script) which interactively asks the user which gems/groups she likes to install and sets up bundler properly. What you you think?

2011-03-19 03:26 pm - Gregor Schmidt

Pull request is at <https://github.com/chiliproject/chiliproject/pull/29>

I am planning to update the ci script accordingly and let it run on <http://ci.nach-vorne.eu>. Will also try to test it with multiple RubyGems versions. I mainly opened the pull request, so that others can test it as well.

In order to use it, you need to

```
<pre>
gem install bundler
bundle install
</pre>
```

This will install all supported db adapters. If you don't want to install one of them, you can call

```
<pre>
bundle install --without postgres mysql mysql2
</pre>
```

The @--without@ parameter is sticky, i.e. it will be stored in @./.bundle@. Subsequent calls to bundle install will reuse these parameters, w/o the need to call them explicitly.

Using redmine will most likely happen as usual, i.e. @rake@ and @script/*@ will most likely work as expected. If you are running into strange errors, prefix your commands with @bundle exec@.

```
<pre>
bundle exec rake db:migrate db:migrate:plugins
bundle exec script/server
</pre>
```

Please let me know if anything weird happens.

Gregor

2011-03-19 04:02 pm - Gregor Schmidt

Note to self: Need to evaluate @gem@ calls in

```
<pre>
vendor/plugins/open_id_authentication/init.rb
5:  gem 'ruby-openid', '>=2.1.4'
```

```
vendor/plugins/open_id_authentication/test/test_helper.rb
```

```
13:gem 'ruby-openid'
```

```
</pre>
```

2011-03-19 04:08 pm - Gregor Schmidt

Note to self: Need to evaluate, if we should automatically load .gemspecs in plugin dirs

2011-03-19 07:35 pm - Eric Davis

Holger Just wrote:

> This could be supported by a installation rake task (or a simple script) which interactively asks the user which gems/groups she likes to install and sets up bundler properly. What you you think?

I like that idea but we should talk separately about it. There was an app that did a "one click install" I've been wanting to copy.

Gregor and Holger:

Since I have no experience with bundler, I'd defer to your expertise here. As long as I can easily install, add more dependencies from a plugin, and can define local dependencies I'd be happy. (We might need to allow a local bundler file for local depends. Like how @config/additional_environment.rb@ is loaded.)

2011-03-19 09:11 pm - Gregor Schmidt

Holger Just wrote:

> * @rmagic@ is missing.

> * @ruby-openid@ is required unconditionally.

The pull request was updated accordingly.

> It might be a good idea to have optional gems in it own group each, so that people can decide which features they want. This could be supported by a installation rake task (or a simple script) which interactively asks the user which gems/groups she likes to install and sets up bundler properly. What you you think?

This 'script' needs to run without any dependencies. At that point, you just have Ruby, RubyGems and bundler installed. That's it. It's a pity, that bundler does not allow to deactivate optional groups by default, but I think it could also work without a separate script. Let's see. I hope well get some feedback while bundler sits in the unstable branch.

Gregor

2011-03-19 09:16 pm - Gregor Schmidt

Eric Davis wrote:

> Since I have no experience with bundler, I'd defer to your expertise here. As long as I can easily install, add more dependencies from a plugin, and can define local dependencies I'd be happy. (We might need to allow a local bundler file for local depends. Like how @config/additional_environment.rb@ is loaded.)

Currently, you may add dependencies from within plugins by simply putting a Gemfile into your plugin directory. We are using this technique at Finnlabs for some time now and, although not perfectly the way bundler was intended to be used, it works quite ok. As I noted above, I plan to investigate the use of @gemspec@s from within the plugins as well. Although they are i bit more difficult to write, the should provide more flexibility when it comes to partially conflicting dependencies.

I hope, that this will be covered by some How-Tos, once it is released (ref #292).

2011-03-19 09:33 pm - Gregor Schmidt

Note to self: Consider removing the test gems from the development environment.

Adding them, like it is done today, allows us to write:

```
<pre>
ruby test/units/user_test.rb
</pre>
```

without prefixing it with `@RAILS_ENV=test@`. On the other hand it results in nasty test output when shutting down the dev server and probably other things as well:

```
<pre>
$ script/server
```

[...]

```
^C[2011-03-19 22:31:38] INFO going to shutdown ...
[2011-03-19 22:31:38] INFO WEBrick::HTTPServer#start done.
```

```
Exiting
Loaded suite script/server
Started
```

```
Finished in 0.000307 seconds.
```

```
0 tests, 0 assertions, 0 failures, 0 errors
```

```
</pre>
```

2011-03-22 12:00 pm - Balazs Nagy

Test gems are available in development mode for a reason: it allows you to access generators in rails3, and to access test frameworks (which will switch to test environment before loading Rails environment). We can say it's a convenience feature.

2011-03-22 04:24 pm - Rodrigo Rosenfeld Rosas

Eric Davis wrote:

```
> ...
>
> I like that idea but we should talk separately about it. There was an app that did a "one click install" I've been wanting to copy.
```

Hi Eric,

I've worked on a Chef recipe to install Gitorious, which is much more complicate than installing Redmine/ChiliProject. Maybe we could think in maintaining some Chef cookbook for installing ChiliProject, instead of a one-click install. What do you think?

2011-03-22 04:30 pm - Holger Just

Rodrigo Rosenfeld Rosas wrote:

```
> Maybe we could think in maintaining some Chef cookbook for installing ChiliProject, instead of a one-click install. What do you think?
```

Most of the people here are more in Puppet's camp. But I'm with you, as I thought about that too :) Could you please head over to the forums for discussing this to not become OT in this issue.

However, recipes for the different popular configuration management systems are nice and all, but do not replace a well thought out basic deployment strategy. Both sides could help each other though :)

2011-04-06 09:56 pm - David Kowis

<http://gembundler.com/rails23.html>

This might be of use, if it hasn't been encountered yet.

2011-05-06 10:01 pm - Eric Davis

Gregor and Holger:

What else do we need for this? I'm thinking it might be good to add what we have now and change it as we need to support other environments.

2011-05-07 07:53 am - Holger Just

I vote for adding bundler rather sooner than later into unstable and fix any remaining issues there.

The only thing we still need to decide on is the status of Ruby 1.8.6. I know Gregor spent some time to find gem versions which work with 1.8.6 through 1.9.2, which was sometimes a PITA.

After reading again through the "README":<https://github.com/finnlabs/chiliproject-gemfile/blob/master/README>, it might be a good idea to have a common Gemfile which handles the most common / Supported cases (i.e. Ruby >= 1.8.7, recent libraries) and ship some Gemfiles for other systems (Debian Lenny, Ruby 1.8.6 users, ...), or at least document the required changes somewhere.

We could also introduce something like `@./configure@` which generates our Gemfile :)

2011-05-08 05:20 pm - Felix Schäfer

Holger Just wrote:

> I vote for adding bundler rather sooner than later into unstable and fix any remaining issues there.

Agreed.

> The only thing we still need to decide on is the status of Ruby 1.8.6. I know Gregor spent some time to find gem versions which work with 1.8.6 through 1.9.2, which was sometimes a PITA.

>

> After reading again through the "README":<https://github.com/finnlabs/chiliproject-gemfile/blob/master/README>, it might be a good idea to have a common Gemfile which handles the most common / Supported cases (i.e. Ruby >= 1.8.7, recent libraries) and ship some Gemfiles for other systems (Debian Lenny, Ruby 1.8.6 users, ...), or at least document the required changes somewhere.

Yay for documentation for "edge-cases", IIRC we already decided 1.8.6 would be supported "if possible" in 2.0, "if possible" meaning "coming with docs about the necessary changes if not", which IMHO fits this case.

> We could also introduce something like `@./configure@` which generates our Gemfile :)

Are you out of your mind? O_o

2011-05-08 09:21 pm - Gregor Schmidt

I would be happy if bundler was added to unstable. Most work needs to go into documentation, for developers, plugin authors and administrators. I'm not good at that part, that's why it didn't happen yet.

From the code side there the following topics are still open:

- * Test the branch with Rails 2.3.11 and maybe rebase the pull request to ease merging
- * Evaluate the usage of `.gemspec` files for specifying dependency of plugins
- * Evaluate adding something like `additional_environment.rb` to easily add local dependency that neither belong to CP itself nor to any plugin

And finally, I need to update `finnlabs/chiliproject-ci` once, the pull request was accepted.

I'll try to attack these issues within the next week.

2011-05-10 06:46 am - Gregor Schmidt

The branch and therefore the pull request is now rebased on current unstable. I will continue integration testing today.

2011-05-13 04:21 pm - Eric Davis

- *Status changed from Open to Closed*

Merged into unstable. Thanks for your work on this Gregor.

Next we will need to update the installation, development, and release documentation and then see about those edge cases in the Gemfile. But we can handle those in separate issues and the forum.