Bug #4436
FileUtils verbose mode prints output to stderr

02/24/2011 04:19 AM - jrdioko (Johnathan Ritzi)

<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td></td>
</tr>
<tr>
<td>Target version:</td>
<td>3.0</td>
</tr>
<tr>
<td>Backport:</td>
<td>2.5: UNKNOWN, 2.6: UNKNOWN</td>
</tr>
</tbody>
</table>

Description

I'm using a plugin in a Ruby on Rails application that uses FileUtils's `cp_r` method to copy files into place (with verbose mode on). When deploying my application with Capistrano, I get several "error" lines during this part (http://groups.google.com/group/capistrano/browse_thread/thread/64e95a26af3035cd?pli=1).

It appears that FileUtils#cp_r prints all output in verbose mode to stderr, even if the output is not erroneous (http://stackoverflow.com/questions/3163585/getting-executed-command-from-ruby-fileutils). It seems that fu_output_message (and related code) should be changed so that output goes to stdout and errors go to stderr, as usual.

History

#1 - 07/15/2019 04:43 PM - jeremyevans0 (Jeremy Evans)
- Backport set to 2.5: UNKNOWN, 2.6: UNKNOWN
- ruby -v deleted (1.8.7)
- Description updated
- Subject changed from FileUtils#cp_r verbose mode prints output to stderr to FileUtils verbose mode prints output to stderr
- Project changed from Ruby 1.8 to Ruby master

This was set to the Ruby 1.8 tracker, but I think it is still an issue currently for all FileUtils methods. Changing the default behavior could possibly be done in Ruby 3, and potentially we could add an option for $stdout (or another I/O stream) for non-error messages in 2.7.

#2 - 07/31/2019 07:56 PM - jeremyevans0 (Jeremy Evans)
- Target version set to 3.0

After a more thorough review, fu_output_message is only called in verbose mode. You can get the behavior you want via:

```ruby
FileUtils.instance_variable_set(:@fileutils_output, $stdout)
```

before calling other methods on FileUtils. I think changing the default from $stderr to $stdout should be done in Ruby 3.

#3 - 03/09/2020 04:49 PM - jeremyevans0 (Jeremy Evans)
I've submitted a pull request upstream for this: https://github.com/ruby/fileutils/pull/49

#4 - 06/11/2020 03:22 PM - jeremyevans0 (Jeremy Evans)
- Status changed from Open to Closed