

Y.o.m.b.a.r

Rails from a .NET perspective

Live Search:

Taming FastCGI+Apache2 on Windows

Posted by dema on August 24, 2005

I've had my share of wrestling with Apache2 and FastCGI to get my Rails (version 0.13.1) apps running on my Windows XP box (not to mention my attempts with [Lighttpd](#)). Long hours trying to figure out configuration trickery for things that were not primarily designed to run on Bill's OS.

But I have good news, I finally managed to get it working well enough these days. And to spare you, who are in the same boat, the same trouble, here are some useful tips.

First, let's cover the basics:

- Install [Apache2, latest stable version](#)
- Install [MySQL 4.1 Windows Essentials, latest stable version](#) (I will only cover MySQL as a db backend here)
- Install the latest [Ruby Installer for Windows](#)
- Install the latest [Ruby For Apache](#)

Now, make sure you have mod_fastcgi.so in your Apache2\modules folder and that you have mysql.so and fcgi.so in your ruby\lib\ruby\site_ruby\1.8\i386-msvcrt folder.

Start a normal IRB session and type **require 'fcgi'** and **require 'mysql'**. You should get '=> true' as a response from both.

Make sure you have these directives in your Apache2 httpd.conf file:

```
LoadModule fastcgi_module modules/mod_fastcgi.so

<IfModule mod_fastcgi.c>
  AddHandler fastcgi-script .fcgi
</IfModule>
```

Now, a typical Virtual Host entry that should point to your Rails application would look like this:

```
<VirtualHost *:3000>
  DocumentRoot c:/dev/myrailsapp/public
  ErrorLog c:/dev/myrailsapp/log/server.log

  <Directory c:/dev/myrailsapp/public/>
    Options ExecCGI FollowSymLinks
    AllowOverride all
    Allow from all
    Order allow,deny
  </Directory>
</VirtualHost>
```

Remember to add a 'Listen 3000' directive as well up in the httpd.conf file if you're going to use that port.

Now, in your dispatch.fcgi file that resides in your Rails app public\ folder, make sure the shebang line is correct:

```
#!c:/ruby/bin/ruby
```

Finally, in your .htaccess file that resides in the same folder as the dispatch.fcgi file, change the rewrite rule that points to the 'dispatch.cgi' file and replace it with 'dispatch.fcgi'.

So far, pretty standard stuff. Now, let's get a little fancier. To make Apache2 more conservative when running on Windows, go back to the httpd.conf file and add these lines:

```
EnableSendfile Off
EnableMMAP Off
Win32DisableAcceptEx
```

Also, in the same file, you should dumb down the FastCGI powers (we are talking about a development machine, right?), and so, you need to limit the number of processes it is allowed to start, otherwise, it will just take too long when starting your app, and it will consume too much memory as well, and that might cause you trouble.

For that, you need to add the following directive to keep FastCGI on a minimum resource level usage:

```
FastCgiConfig -maxClassProcesses 1 -maxProcesses 1 -minProcesses 1 -processSlack 1
```

If you wish (or need), you might increase those numbers a little bit, but I don't recommend going over 5 processes.

And then to the trickiest part. If you're using native Ruby bindings such as fcgi.so or mysql.so or iconv.so or RMagick.so, those bindings usually rely on native DLLs that reside in folders such as Windows\System32 and others. And that is normally a source of problems and error messages such as 'uninitialized constant Mysql/Iconv/...'.

The problem is that FastCGI server processes doesn't get any (yes, none, zero) of the environment variables of your system, being them either at machine-level or user-level. And that includes RAILS_ENV, RUBYOPT and more importantly the PATH variable. And without the PATH variable all native extensions that rely on DLLs outside the ruby\bin directory will fail. I discovered this after very long hours as you may imagine.

And don't bother trying to use SetEnv or PassEnv directives. They won't work at all with FastCGI.

The only solution is going back to the FastCgiConfig directive and adding all your environment variables using the '-initial-env' option. So, my complete FastCgiConfig directive now looks like this:

```
FastCgiConfig -maxClassProcesses 1 -maxProcesses 1 -minProcesses 1 -processSlack 1 \
-initial-env PATH="c:/ruby/bin;c:/windows/system32;c:/windows;c:/mysql/bin;c:/Program~1/Common~1/GTK/2.0/bin" \
-initial-env RUBYOPT=rubygems \
-initial-env MAGICK_CONFIGURE_PATH=c:/ruby/lib/ruby/gems/1.8/gems/RMagick-win32-1.7.0-mswin32/config
```

Important: always use "" on paths and always use the 8.3 short names when the path contain spaces.

With (all) this, I am able to use FastCGI very reliably and with excellent performance and also make all native bindings that I use (Iconv, Mysql, RMagick, and others) to work as well.

Originally, I tried all this to see if the UploadProgress helpers would work on Windows, since it requires FastCGI, but even then, it didn't. I know I am supposed to allow more than 1 FastCGI process to make UploadProgress work and I did try that, but it didn't work anyhow. So, if you manage to get it working, please, let me know how you did it! ;-)

An important tip to diagnose problems is that, if you get the infamous "Application Error – Rails application failed to start properly" message, there is usually something going wrong on in your 'environment.rb' file (or the files it is requiring). Rails won't be able to present you any useful debugging information unless all of the 'environment.rb' gets executed right. So, it's pretty hard to track the cause of problem in this situation, but normally, this is caused by some custom 'require' that you added to the file. The standard 'environment.rb' that gets generated from Rails is safe.

Remember to always restart Apache2 when you change the httpd.conf file (duh! But I did forget sometimes! ;-)).

I hope you can manage to get this combo working on your WinXP box with all this info. Let me know if you have any trouble, and I'll try to help.

Posted in [Rails](#) | [14 comments](#) | [no trackbacks](#)

Comments

[Leave a response](#)

1. **Sascha Ebach** said about 1 hour later:

Amazing, I also ran into those uninitialized constant Mysql ... errors. It would be great if Rails would actually work natively on WinXP. I gave up hope completely, but I also knew I wouldn't have to jump through all the hoops you did. Now all we need is having all this in the Ruby installer so that windows user can have a seamless experience, at least with the standard tools.

2. **Larry Kelly** said 1 day later:

Do you think this setup is stable enough for production?

3. **dema** said 1 day later:

It might be, specially if the load is somewhat below 100 requests/second, which would be the case if you're thinking about using it for a internal (intranet) application. For a public, internet application, I just can't see a good reason not to use a Linux or FreeBSD box (preferably with Lighttpd), since it's not only more reliable, but much cheaper as well.

4. **Jochen Schulz** said 50 days later:

Thanks a lot for this information. You saved my day.

5. **Daniel Zilberman, danfrompiter@gmail.com** said 56 days later:

Hello, First of all - thank you much for putting up this page - it almosty helped me to get my Ruby on Rails/FastCGI/Apache setup running, but there are problems... I have followed your page all the way - installed MySQL DB, Apache 2 server, Ruby one-click installer from the page: <http://electricvisions.com/rails.html> and updated httpd.conf file to load FastCGI module:
 LoadModule fastcgi_module modules/mod_fastcgi.so #comment out loading of Windows DLL #LoadModule fastcgi_module
 # "C:/windows/system32/mod_fastcgi-2.4.2-AP20.dll" <IfModule mod_fastcgi.c> AddHandler fastcgi-script .fcgi </IfModule>
 also I'm setting FastCGI environment at the end of httpd.conf file: FastCgiConfig -maxClassProcesses 1 -maxProcesses 1 -minProcesses 1 -processSlack 1 and have my virtual host entry configured properly: VirtualHost *:80> ServerAdmin danfrompiter@gmail.com ServerName hungryforce_dz DocumentRoot "D:/Apache Group/Apache2/htdocs/HungryForce/public" <IfModule mod_ssl.c> SSLEngine off </IfModule> <Directory "D:/Apache Group/Apache2/htdocs/HungryForce/public/"> Options ExecCGI FollowSymLinks AllowOverride all Allow from all Order allow,deny AddHandler cgi-script .cgi AddHandler fastcgi-script .fcgi </Directory> </VirtualHost> and I have modified the .htaccess file to have directive: RewriteRule ^(.*)\$ dispatch.fcgi [QSA,L] Still when I start Apache and try to hit my virtusl host http://hungryforce_dz I get this error in Apache log: [Tue Oct 18 20:37:51 2005] [warn] FastCGI: (dynamic) server "D:/Apache Group/Apache2/htdocs/HungryForce/public/dispatch.fcgi" has failed to remain running for 30 seconds given 3 attempts, its restart interval has been backed off to 600 seconds many many times. Can you please advice me what may be wrong in my setup - it is very important for me to figure that out. many thanks, Dan Zilberman

6. **Douglas Ausech** said 63 days later:

Hi Dema! Have you got your rails application running on both Apache2 and WEBrick in the same box? In my case WEBrick stopped working just after I install Ruby for Apache. It seems that there is a conflict within the configuration of Active Record's MySql adapter. Thank you.

7. **Dema** said 64 days later:

Yep, I got them both running. Are you using the right MySql driver? Take a look at my article on that.

8. **Max** said 65 days later:

Yipieeh! It works (for some reason only after restarting Apache several times....). The ENVs were indeed the missing pieces. Thank you very much.

9. **Paul** said 71 days later:

Thanks for the help, this post really worked.

10. **ATM** said 71 days later:

Hi: Thanks for all of the help with this. I struggled for a long time before eliminating the quotes you have around the PATH statement - that seemed to do the trick for me. So I'm currently qualifying FastCgiConfig with -initial-env PATH=c:/ruby/bin;c:/windows/system32;c:/windows;C:/mysql/bin You might want to check this out? Again - thanks for the help!

11. **Arnd Jan** said 73 days later:

require 'mysql' returns the following error: LoadError: 126: The specified module could not be found. - F:/Ruby/lib/ruby/site_ruby/1.8/i386-msvcrt/mysql.so although I have verified that mysql.so is in this location... Any suggestions please...

12. **Arnd Jan** said 73 days later:

One other thing: in httpd.conf, you have to enable the mod_rewrite module, by uncommenting "LoadModule rewrite_module modules/mod_rewrite.so".

13. **david** said 75 days later:

A few notes from my experience: ---- 1. MySQL 'include': If you've got apache/mysql installed as part of a package (eg xampp or triad), the MySQL .dll might not be in win/system32. You'll need to copy it there for the ruby 'include mysql' to return true. (It seems like since you feed the path to fastCGI in the apache conf, you might not need to do this... but I did anyway just in case.) - -----2. In the "shebang" line in dispatch.fcgi, I had to include the .exe extention. I'm embarrassed that it took me FOREVER to figure this out: #!c:/ruby/bin/ruby.exe ----- Thanks for your help! 'luck

14. **Mike Blake** said 88 days later:

Thanks for all the help! I'm finally up on FastCGI. One interesting note, after connecting to MySQL and everything I the got this error in my app: [Sun Nov 20 16:35:50 2005] [error] [client 67.34.132.234] FastCGI: comm with server "C:/AppTrain/public/dispatch.fcgi" aborted: error parsing headers: malformed header ""in create"", referer: <http://www.apptrain.com/form/new> Turn out I had to remove all "p" statments from my controller as they aren't printing out to the log, but right to the response. Strange, right?

Trackbacks

Use the following link to trackback from your own site:

<http://dema.ruby.com.br/articles/trackback/24>

[RSS feed for this post](#) [trackback uri](#)

Your name

[\(leave url/email »\)](#)