Installing Ruby on WampServer

Before we get into this... these instructions look a lot worse than what they are-- I tried to be very thorough. As of 2 hours ago, I'm currently running WAMP 1.6.4 + RoR on Windows XP Pro (SP2).

Things you will need:

1. Wampserver (Prepackaged Apache 2, PHP, phpMyAdmin, MySQL) - http://www.wampserver.com/en/index.php
2. Ruby "One-Click" Installer for Windows - http://rubyforge.org/frs/?group_id=167
3. Ruby Gems for Windows (Similar to PEAR) - http://rubyforge.org/frs/?group_id=126
4. Ruby For Apache - http://rubyforge.org/projects/rubyforapache/

Step By Step Process:

First and foremost, I have used many servers for my development environment, but none have compared to WAMP. It allows you to easily flip Apache modules / PHP extensions on and off, not to mention switch between PHP 4 and PHP 5, all with an unintrusive GUI. There are quite a few plug-ins such as Zend Optimizer, Webalizer, Perl, etc. The best part is that WAMP is FREE. (although donations are welcome, which I myself have been rallying my organization to contribute to the project).

Keep in mind that for my entire example, wamp is setup on my D:\wamp drive / path. As you are following these instructions, substitute your drive / path. I will remind you periodically to substitute your drive / path. It is imparitive you pay attention, and follow these instructions verbatim. Do not skip ahead! (unless told to do so)

Step 1 - Install Wampserver (#1 of "Things you will need")

- If you already have WAMP installed, continue to Step 2

Step 2 - Install Ruby (#2 of "Things you will need")

- When the application starts, it will ask you if you want to install SciTE and FreeRIDE; These are not required
- I installed ruby alongside my php directories under D:\wamp, so the end path will be D:\wamp\ruby
- Click "Install"

Step 3 - Install Ruby Gems (#3 of "Things you will need")

- Extract the contents of your Ruby Gems to D:\wamp\ruby\rubygems
- Open a DOS prompt window, and enter the following commands:

```
> D:
> cd D:\wamp\ruby\rubygems
> setup
```

- Note: keep in mind your wamp drive / path
- If all went well, you should see the following (or similar):

```
Successfully built RubyGem
Name: sources
Version: 0.0.1
File: sources-0.0.1.gem
```

Step 4 - Install Rails

- Open a DOS prompt window, and enter the following commands:

```
> D:
> cd D:\wamp\ruby\rubygems
> gem install rails --include-dependencies
```

- Be patient... gems is downloading rails from rubyforge.org
- Again, if all went well, you should have seen commands executing, with a final note indicating

```
Installing RDoc documentation for activerecord-1.14.4...
Installing RDoc documentation for actionpack-1.12.5...
Installing RDoc documentation for actionmailer-1.2.5...
Installing RDoc documentation for actionwebservice-1.1.6...
```

- Keep in mind that there will be more statements than what I included above.

Step 5 - Install Ruby For Apache (#4 of "Things you will need")

- I installed Ruby for Apache in my D:\wamp\ruby\RubyForApache subdirectory
- Locate your Apache server directory, in my case, its D:\wamp\Apache2 (keep in mind that the application for whatever reason tries to append "RubyForApache" to the path. Simply make the path "D:\wamp\Apache2" (or where your root Apache directory is located))
- Next, locate your root ruby directory, which is D:\wamp\ruby for me. Again, remove the "RubyForApache" from the "D:wamp\ruby" path (see above).

- Finally, for the Installation Options, uncheck all options but "mod_fastcgi"
- Note: The application may alert you that it cannot open c:\xppro\system32\msvcp71.dll. Just rename it and retry. As I'm writing this documentation, I didn't have an issue, but had noticed it on a previous install.

Step 6 - Setup Apache

- After the Ruby For Apache is installed, you will notice 'mod_fastcgi.so' in your D:\wamp\Apache2\modules folder
- Next, move to D:\wamp\Apache2\config directory and open 'httpd.conf'
- After the LoadModules section, append the following code:

```
        LoadModule fastcgi_module modules/mod_fastcgi.so

        <IfModule>
            FastCgiConfig -maxClassProcesses 10 -maxProcesses 10 -
minProcesses 1 -processSlack 1 \
               -initial-env
PATH="d:/wamp/ruby/bin;c:/windows/system32;c:/windows;d:/wamp/mysql/bin/"
\
               -initial-env RUBYOPT=rubygems
            AddHandler fastcgi-script .fcgi .fpl
        </IfModule>
```

- UPDATE: phpBB is parsing the above code incorrectly. The opening "IfModule" tag above needs to be changed to: "IfModule mod_fastcgi.c"
- Note: Substitute your wamp drive / path!
- Also, make for certain the following is uncommented:

```
    LoadModule rewrite_module modules/mod_rewrite.so
```

- Restart Apache via the Wampserver system tray GUI

Step 7 - Install your first Rails App

- If you have a "work" directory for Ruby projects, change to that directory. Open a DOS prompt window, and enter the following commands:

```
    > D:
    > cd D:\wamp\www\projects\RUBY
    > rails test
```

- This creates your first rails application in D:/wamp/www/projects/RUBY/test
- Note: your "projects\RUBY\" directory may differ. Simply substitute your path as needed.

Step 8 - Modify your "test" install's "public/.htaccess" file

- Navigate to your D:\wamp\www\projects\test\public\ folder and open the .htaccess file
- Look for:

```
# General Apache options
AddHandler fastcgi-script .fcgi
AddHandler cgi-script .cgi
Options +FollowSymLinks +ExecCGI
```

- And change to:

```
# General Apache options
# AddHandler fastcgi-script .fcgi
# AddHandler cgi-script .cgi
Options +FollowSymLinks +ExecCGI
```

- We do this because we've already defined the handlers in step 6.
- Next, look for:

```
RewriteRule ^(.*)$ dispatch.cgi [QSA,L]
```

- And change to (change .cgi extension to .fcgi):

```
RewriteRule ^(.*)$ dispatch.fcgi [QSA,L]
```

- Save and close the file

Step 9 - You are almost done. One last change to the "public/dispatch.fcgi" file

- Navigate to your D:\wamp\www\projects\test\public\ folder and open the dispatch.fcgi file
- Change the #! (shebang - line #1) at the top of the file to reflect the path to the Ruby interpreter (where its located on your system). I had to change mine to:

```
#!D:/wamp/ruby/bin/ruby
```

- Note: Your path may differ. If you don't remember the ruby intepreter's path, refer to Step 2.
- Save and close the file

In Conclusion

Finally, you should be ready to see some action. Make for certain that you restart your wampserver. If not all of the services fire up (mainly Apache), you might have skipped a step, or "fat-fingered" something. Go back now and check your work.

If the servers did start fine, open a browser (preferably something that is not IE ;)), and point it to http://127.0.0.1/projects/RUBY/test/public/ Again, change "projects/RUBY/" to the appropriate path, but DONT REMOVE "public" from the URL! The directory "public/" is where rails is initialized. Depending on server configuration (production/live), this is usually done with .htaccess files, http.conf virtual hosts, etc. Don't worry about this so much.

Once you direct your browser to the aforementioned URL, you should see the "Welcome Aboard" index page-- don't celebrate yet. Directly beneath the header, there is a link that says "About your application's environment," click it. If you see a yellow box animate open, you are good to go. If you see a red box, you missed a step.

Assuming you are seeing the yellow box, append something to the URL like "http://127.0.0.1/projects/RUBY/test/public/weblog"; you WILL get a Routing Error. If you see this, pat yourself on the back for a job well done because rails is installed. Your next step is to look up articles on www.rubyonrails.com or www.google.com, for the next steps. Just keep in mind that RoR comes with a plethora of scripts that automate creation of new controllers and models, which relates to the "scripts/generate" references you will see.

I apologize now for typos / mispellings... I'm currently at work and don't have a lot of time for finesse :)

Last edited by csanford on Fri Oct 20, 2006 18:56; edited 1 time in total