CakePHP: What I like

Posted At : July 24, 2008 11:48 PM | Posted By : Jon Hartmann
Related Categories: ColdFusion, Frameworks

I've been doing a lot of thinking about frameworks lately, and about how much work it is to get a MVC/Model2 framework going in ColdFusion compared to in other languages/servers. This lead me to investigate the documentation for a number of non-ColdFusion server/frameworks that I've used, or are popular, and I've been trying to round out a list of what I like and why.

The first installment of this mini-series is on CakePHP. Please remember that while I may have used some of these frameworks, I'm not an expert, so I might be missing *the* thing that makes that framework cool. If so, let me know that you think the best parts of the framework are, and I'll see if I agree.

CakePHP

Directory Structure

Cake's directory stucture is very neat, keeping nearly all of your code tucked into an /apps folder, and organizing files inside it into very organized bins.

Convention Based ORM

Although most frameworks are coming with built in an ORM system, I think it is still important to note that ORMs do speed up development, and ORMs that work based on naming conventions speed that proccess up even more. Coding XML to describe your database is just tedius.

HtmlHelper Class

When I code in PHP, I hate jumping in and out of HTML in order to create dynamic links and forms. Unlike the tag structure of ColdFusion, PHP just isn't friendly to that type of content creation. The HtmlHelper class gives you methods that you can call to generate common HTML snippets without breaking out of PHP, and thats a very good thing.

Simple URL to Controller Mapping

Cake URLs generally take the form of /controller/action/id, which is very simple and easy to work with. This means that /pages/view/123 would call the view method of the pages controller and give it an id of 123. This is dead simple.

Simple Controller to View Mapping

Notice how I just said that lining up a URL with its handler is dead simple? Guess what, linking up views is convention based as well. If you are looking for the view to go along with /controller/action, you just need to look in /app/views/controller/action.ththml.

Join me later as I find out what I like about Ruby on Rails and Django!