A ColdFusion Developer Examines Ruby on Rails: A Night in Austin

Posted At : July 29, 2011 1:27 PM | Posted By : Admin
Related Categories: Ruby on Rails

On July 27, I had the distinct pleasure of meeting with other ColdFusion developers at the Austin CFUG. I wanted to share what I had learned in the process of learning Ruby on Rails. You see, I was tricked into learning Rails. John, a very good friend (and another ColdFusion developer) had learned Rails and tried to interest me in it as well. I would have none of it. I didn't need it and I didn't like it. Further, I didn't like those smug Rails fanboys. I had been working in ColdFusion for almost fifteen years and I had run into very few situations where I needed something else. And I always had Java that I could use if the need arose. So who needed Ruby on Rails? Not me. I thought.

So after receiving several rebuffs to his invitation to mentor me, he quit pestering me. But he did ask if I could just *help* on a JavaScript issue he was having on one of his applications. By sheer luck, apparently, it was a Rails application. Well, it seemed the least I could do.

It turned out that before we could fix the JavaScript issue, John needed to fix the data being sent back from his Rails application via Ajax. John opened up the strange Ruby file. I was surprised by how readable the code was. And succinct. One simple line of code to convert an array of objects into JSON and return it. And look there! That was the problem. I didn't know Ruby at all, but it was obvious that John had simply misspelled the name of his object array. "Fix that and I bet it will work," I told him. And it did. As for the JavaScript, I was surprised that John, as smart a person as I know, would be having trouble with something so simple as a callback.

Of course, John wasn't having problems with JavaScript--nor with Rails. But he was betting that looking at Ruby code and being introduced to Rails might pique my interest. And yes, it was *somewhat* interesting. And could you really do all that work in just those few lines of code? I supposed it couldn't hurt to learn a bit more.

That "bit more" had me volunteering to help John finish one application and tackling, with him, a much larger, more complicated one. By the end of that second application, I was hooked. I had joined the ranks of card-carrying Railies. I was a dyed-in-the-wool Rails fanboy. (Thankfully, I managed to avoid handing out tracts to passers-by, extolling the virtues of Rails.) Over time, my fanboyism mellowed into a more proper appreciation for Ruby and Rails. But what I didn't count on was how the learning process gave me a renewed esteem for ColdFusion.

ColdFusion developers love ColdFusion for how it removes much of the drudgery of web development. Need to query a database? Or send an email? Or any number of other things that ColdFusion allows you to do with a single tag? While it seems that some languages sprang from the mind of a headmaster of a boarding school for wayward souls ("And don't even *think* about using variables without first declaring them!"), ColdFusion devotes itself to making developers lives *easier*.

And now, here was Ruby on Rails--also intent on making things better for developers by relieving so much of the drudgery of web development by disregarding the rules of *proper* web development. ColdFusion developers have long had to fight against the perception that "real"

development involved pain and frustration. Now, with Rails, ColdFusion gets a powerful ally and friend.

And that's what I wanted to share with my fellow ColdFusion developers in Austin. Yes, the language *looks* very different from ColdFusion--but they share a much deeper affinity--and that makes learning Ruby on Rails a natural for ColdFusion developers.

Over the next several months, I'm going to be blogging and writing more about adding Rails to your skill set--and doing so from a ColdFusion point of view. If you're anxious to get started, though, here's a link (with thanks to Engine Yard for compiling it) of books, blogs, podcasts, screencasts, etc--all devoted to helping you learn Ruby on Rails: http://www.engineyard.com/blog/2010/resources-for-getting-started-with-ruby-on-rails/.

In the meantime, if you have any specific questions--or would just like to discuss Rails from a CF POV, feel free to email me at hal@halhelms.com.