ValidateThis! - Object Oriented Validations Demo Now Online

Posted At : October 10, 2008 8:50 AM | Posted By : Bob Silverberg
Related Categories: OO Design, cfUniForm, ColdFusion, ValidateThis

I have put together a demo of my validations framework for ColdFusion that I have been **working on** for the past several weeks. It's available **here**. The demo shows how the framework can take a simple xml file, which describes the business rules for an object, and generate client-side and server-side validations on the fly. It is a proof-of-concept, running in an environment identical to that which would be used in a real application. It is a simple, one-page example, but is utilizing **Coldspring** and **Transfer** behind the scenes.

From the demo page one can view the xml file that is being used to drive all of the validations. The schema for this file is still a work in progress, but it provides some insight into how one would implement the framework. Note that the schema itself is somewhat irrelevant, as the framework is designed to accommodate any schema, as long as it contains the expected metadata. Also, as a result of a conversation I had with **Matt "I hate xml" Quackenbush**, I have added the ability to define validations using ColdFusion code inside either your business object or another object injected into your business object. This is not something that I'd probably use, but I was led to believe that there are those out there who would appreciate not having to write an xml file to define their validations.

I've used Matt's **cfUniForm library** to generate my demo form, and I've done some integration between my framework and his library. I found that I was able to accomplish the following via this integration:

- The isRequired attribute on the field tag becomes unnecessary.
- The validation attribute on the field tag becomes unnecessary.
- Many new client-side validation types are now supported.
- All server side validation errors generated by the framework are placed by the cfUniForm library.

Note that I did this integration because I am a big fan of cfUniForm and plan to use it for all of my forms, however cfUniForm is by no means a requirement for using the validation framework. You can write or generate your forms any way you choose and the framework will work with them.

I'll be writing more about the framework in the future, including a description of the xml file and the metadata that the framework uses. I also hope to have some sample code up on **RIAForge** in the near future. For now, if you're interested, check out the **demo page**.

A Final Word: Unfortunately the demo is running on a shared server, and I've found that the combination of Coldspring and Transfer on this server do not perform particularly well. So it may run a bit slow. Please note that this has nothing to do with the framework itself - it performs very well on my development machine.