

Re-Imagining BlogCFC

Posted At : June 13, 2014 1:09 PM | Posted By : Cutter

Related Categories: BlogCFC, Team CF Advance, Development

Team CF Advance is dedicated to providing and maintaining Open Source software built in CFML. We welcome anyone to get involved, whether it's writing code, creating tests, or assisting with technical documentation, any contribution is a good one. There's already a **decent list of ongoing projects** that we are working on or maintaining.

On that track, how about giving us a little feedback. Let us look at a re-imagining of **BlogCFC**. BlogCFC was the first Open Source CFML based blogging platform, and still shares a wide install base. In that knowledge, it is important to consider not only maintaining full feature support, but also maintaining support for existing linking structures, data integrity in data migration, as well as support for multiple platforms.

There are several other ColdFusion based platforms for blogging. **Mura CMS** provides a way of creating blog pages. Mura is not truly a blogging platform, per se, but does make it possible. **Mango Blog** is also a fairly pervasive blogging platform, providing some additional level of extensibility as well. **ContentBox** is another platform, based largely on the popular **ColdBox** framework. Like Mura, ContentBox is really a CMS, so not really designed to be a blog.

Knowing that these options exist, one question then becomes "Why do ColdFusion developers adopt **WordPress** or **Ghost** as their blogging platform?" I believe this comes down to three core reasons. 1) ColdFusion hosting can be expensive, 2) the complexity of these other ColdFusion based applications, or the learning curve involved in their customization and maintenance, is more than some of these developers want to undertake, or 3) the developer was just trying to learn something different.

Issue #3 doesn't factor into this discussion. The decision of any developer to expand their knowledge base and learn new languages is a good thing. Doesn't necessarily look good to have a largely ColdFusion topic specific blog hosted on PHP, but we aren't writing this to solely serve developers. We want to build an application that anyone can deploy for their use, whether it's a development blog, a political blog, or just something to journal your trip to Istanbul.

So, you then have two main issues you possibly need to address: 1) Hosting, and 2) simple install, usage, customization, and maintenance.

Issue #1 will still be hard to address (Team CF Advance is already in talks with a hosting provider who *may* offer low cost Blog hosting with everything preinstalled), but

it's worth noting for future discussions. Issue #2 is more easily addressed.

Proposals For Discussion

1. The current BlogCFC is actually two applications, one desktop and one mobile. I propose that we look at a responsive site design for a single maintainable codebase that is then viewable from multiple screen resolutions and devices. I propose adopting **Bootstrap** as the client-side CSS framework, to assist in that task. (One contributor **has already done this** with the current version of BlogCFC, so we may get a branch for this one soon...)
2. There is currently no set style guide for skinning sites in BlogCFC. First, I think it would be extremely beneficial to build a template converter system, that could take existing themes for WordPress, and possibly Ghost, and convert them to BlogCFC themes. There is already a huge number of available WordPress themes out there, and the Ghost theme base is expanding daily as well. If we can leverage that, and make it dead simple for a user to implement, then we've covered a major hurdle to adoption. Ghost themes are the easier side of the coin, as their themes are already in Bootstrap, and already responsive designs. Second, a base template with well thought out HTML architecture and a solid CSS naming convention, that is then thoroughly documented, should make it easy for a user to upload a stylesheet that would completely customize their blog. At some point, we could even build a module for theme customization directly in the admin. And, with all of it, we heavily review page architecture to completely maximize SEO visibility.
3. I think BlogCFC would thoroughly benefit from a server-side framework. I propose using FW/1 with DI/1. FW/1 (and DI/1) is extremely powerful and light weight, operates as a very natural extension of ColdFusion's built in application framework, and it's subsystem architecture lends well to extension/addition of new modules. I'm not positive, but we may also be able to leverage it's routing capabilities when considering legacy link support.
4. I think we definitely need an "override" service architecture, where all instantiated objects are (at inception) blank objects that extend our base objects. This is used for individual developer customization, in a directory that never gets updated (other than the addition of new objects). Our tests should always pass, in development. Any individual overrides are the responsibility of that developer, and them running our tests should allow them to identify holes in their changes with any upgrade.
5. I think we should employ REST services for Ajax interaction, and SOLR for site indexing and search. I do not propose the use of ORM. I think it is very important that our SQL is handwritten, well optimized, and that our indexes are carefully chosen for high availability/performance.
6. I propose using AngularJS as a client-side JS interaction framework for the

"admin" side of the blog, for data-binding and interaction. I think it best to avoid it on the front-end, in favor of generated code bits and more favorable SEO. Bootstrap should be sufficient for any "site" front-end functionality.

7. I think we put considerable effort into the user stories centered around UI/UX, especially in terms of the admin area.
8. I think that we generate output, where possible. Blog posts are, ultimately, static bits of page content.
9. I think that we build in a simple plugin architecture, allowing others to contribute to furthering the platform by creating new add-ons that just be "dropped in".

As with all projects overseen by Team CF Advance, we will be including tests for BlogCFC, written in **TestBox**. This will allow us to build out the project iteratively, testing each step along the way, to ensure that one patch doesn't break other parts of the system.

Again, none of this is in stone. Just some random thoughts to get the ball rolling. This is just to really start the discussion, so please share your feedback. We already have a working group within Team CF Advance, who is contributing to this discussion. Ultimately these projects are community driven to fulfill a need, so knowing what you want is critical to the process. This also doesn't deter us from any ongoing patching efforts. If anyone has begun writing any patches, please let the team know so we aren't duplicating effort anywhere (or just join The Team).