



Announcer You're listening to the Sun Microsystems Podcast Network.

Announcer Welcome to another edition of [Innovating@Sun](#), with your host, Hal Stern. Today's topic: Today's topic, Architecture for Humanity. And now, here's Hal Stern.

Hal Stern:

Hello and welcome to another episode of Innovating@Sun. I'm your host, Hal Stern, Vice President of Global Systems Engineering at Sun Microsystems. And today, I'm joined by Jim Manico from CodeMagi, and Scott Mattoon, who's the Chief Architect of the Western U.S. within Global Systems Engineering at Sun Microsystems, to talk about our work with Architecture for Humanity, and in particular, the Open Architecture Network, a project that we've been involved with since the TED Conference last year, working with Cameron Sinclair, winner of last year's TED Prize, and fulfilling his wish of being able to go provide an online resource for architects and builders and anyone else involved in driving architecture out into affected areas, whether they're affected by global displacement, by natural disaster, or anywhere else that there's an economic disruption. So, welcome to the show, guys.

Both:

Thanks, Hal.

Hal:

So, tell us essentially how this idea of the Open Architecture Network works.

Scott:

Well, Cameron Sinclair and Kate Stohr, the founders of Open Architecture Network and Architecture for Humanity, had a vision to enable pro bono work from architects around the world who had an interest in supplying their designs in a sort of open source of creative commons fashion to the communities in need. And these communities tend to be the ones that are affected by natural disasters and are displaced from their homes, or refugee situations. And so, they provide the transitional shelter and housing for people in those situations, and what they want to do is create community online where these designs that are site-appropriate and situation-appropriate can be shared, adapted and you know, widely distributed and made accessible as much as possible to those communities in need.

Jim:

Yeah, in short, their mission statement is to lift the living standards for everyone in the world. I've never worked on a project with so audacious of a mission statement.

Hal:

And clearly, one of the challenges here is taking the notions of architecture - and this is architecture in a building buildings sense - and applying it to areas where local materials, availability of materials, climate conditions, impose all kinds of creative - I would say in some places, very, very diverse - requirements on the designs being

considered.

Scott:

Absolutely. And in fact, there's so much learning happening out in the field with respect to the materials and the construction, and it varies by, you know, climate and environment and so forth, yet that data's not being shared for the use in other situations that are similar. The challenge that they're trying to address, really, is making that information and that knowledge available, rather than having to, you know, relearn it every time in the situation and find out that, you know, this choice of material may have been the wrong one in this given condition.

Jim:

Yeah, as opposed to other Web 2.0-type technologies that do similar things, like collecting information of this sort in an essential way, it also has very extensive collaboration - Web 2.0-type collaboration features - where architects looking at diagrams of one particular type of building design could all make comments in exact X, Y locations on the image, and using heavy Java Script, they could mouse over different comments and see within the context of one image, over the web, over standardized browsers, and be able to collaborate in that deep of a way on specific projects. I don't think anything else out there is doing that right now in the architectural world at all.

Hal:

And that kind of collaboration allows those experts with knowledge of local materials to go take a design and special case it or extrude it into something that's going to be most applicable for an area where they have the appropriate expertise. They might be looking at a house design and figuring out how you go and build it out of bamboo, or looking at something that's been designed for a tropical climate and figuring out if it's gonna hand the windshear and the stress of surviving a typhoon. So, tell us a little bit about what goes on behind the web server. What does this look like from a back-end perspective? It sounds very Web 2.0, in terms of everything being write-able and things being shared. What's really going on behind the domain name?

Scott:

Well, the physical infrastructure is actually a pair of Sun Fire X2200 M2 servers, a late generation 2.0 processor, four-core servers. They're built out with 16 gigabytes of memory and each has - each is attached to a shared 3511 storage array with six terabytes of data. A lot of storage space, because this is a media-intensive project that will allow for the ingest of many, many photos, CAD designs and renderings that tend to be very, you know, dense data. So, you know, the projection is to have at some point in the future thousands of designs up there, each with, you know, dozens and dozens - if not hundreds - of photos, CAD designs and renderings.

Jim:

And some of these CAD designs are very, very large files. They can even upload sound files, for that matter, all in the context of a specific design in a specific project, as well. So, this content - it's fairly rich content, for certain.

Hal:

So, how are we actually managing all that content? What's the software to do the content management system part of this website?

Jim:

Well, it's an open source system Drupal; in fact, we're using Drupal 4.7. You can see this at [spells] Drupal.org, all the details about this open source CMS system. It's a fairly popular system; it's all based on PHP. Again, it's all open source; we're running on top of a MySQL database engine, and it gives us a rather large number of CMS features for free out of the box. And in addition, on top of that, we've built a large number - over a dozen - custom

modules in the Drupal framework in PHP, very specific to Open Architecture Network. [overlap] That's a kind of high-level view of what we're doing - there's a lot of devil in the details, of course.

Scott:

Right, and to be specific about the underlying software infrastructure, as Jim said, it's a MySQL database, PHP Pearl engine driving the Drupal CMS system. And we use Sun's Cool Stack of the AMP software. This stack of software's been tuned to run on the x64 platform; there's also a SPARC version available that's been tuned to run on our CoolThread server, as well. We've got the 64-bit version of MySQL, for example, and all of the components are running in Solaris 10 containers and we're running these processes in least privilege modes, so that, you know, we're feeling confident that while there may be some exposure from a sort of PHP scripting kind of vulnerability standpoint, if there were any kind of a breach into a process's environment, at least they'd be contained in this really secure environment.

Jim:

And on the software side, Drupal has had its share of security vulnerabilities, but their level of patching is almost instant upon these vulnerabilities being found, if they're critical. And there's a security team at Drupal, as well as several security email lists to subscribe to, and they'll provide exact Gifs on the code base so we can mitigate these risks almost immediately upon them being discovered. So, I'm really impressed with Drupal's - how they're considering security. And on top of that, that handling of security issues, there's a lot of security built into the framework, and I think this is a great step in terms of application development in general, where instead of having a low level language that we're writing our own security functionality in, they provide a framework where a lot of the common security problems like CSRF is mitigated by form keys, and XSS protection, in a lot of ways, is built in to form validation. So, we're doing a lot of things in the security world really right, from my perspective and the application development perspective.

Hal:

I would say that notion of security is not just what's happening in the infrastructure, it's also in the ability to go track a project, particularly when you're working with government or non-governmental agencies or there's a funder involved who wants to see what's going on. So, have you had to add features to do project tracking or project reporting or other sorts of reporting out, not just the basic content management, but also looking at the history of a project.

Scott:

Yes, so there's functionality in the application to get an overview of the project and then, you know, look back historically at the progress of the project. You can see, you know, photos at the early stage where it was just, you know, an empty site location with maybe chalk marks as to where foundations are gonna be poured and so forth, and you can follow the progress of the project all the way through to completion and get the corresponding designs and renderings for each stage of the project, as well. And what's really nice about this is, as Jim said, it's a collaborative environment, so the designers, the community participants and villagers, for example, all have access and are participating in a discussion about, you know, what's appropriate with the design, what maybe needs to be changed, how can they best meet the needs of the local community so that, you know, ultimately, they're maximizing the opportunity to have every stakeholder's concern addressed in the design before the first bit of concrete gets poured, and then they can make adjustments, as appropriate, along the way, as the thing begins to take shape.

Jim:

And every tiny modification to a project is extensively logged - that's a fairly built-in feature at the Drupal level.

So, if compliance ever came up or if you just want an historical record, every tiny step is logged and saved within the core of the system. And on top of that, there's also a pretty extensive calendaring system, as well, for scheduling specific events and tasking out specific items to members of the project. There's, you know, graphical Web 2.0-type calendars they can navigate through, create events, create tasks and actually - more than just collaborate, but actually start to add a level of organization to a specific project and try to, you know, move the group towards very specific goals to move the project along. So, it's more than just like an open-ended, Web 2.0, generic comment system - it's a project management tool, information centralization management tool, a lot of metadata added to an image. It's not just uploading a few images to the site, it's adding a great deal of very specific metadata about the importance of that particular image, in the context of the project, in the context of trying to solve very specific goals. So, I am very impressed with the design of the system.

Hal:

And what we're getting to now is, as we talk about things being Web 2.0 or Web 3.0 - essentially writeable data on the web - is this notion of collaboration, that we're gonna have bits and pieces of ideas floating from a variety of experts, and it's the incremental effort, the incremental addition, of expertise that really makes this scalable, and I would say, long term sustainable. So, I guess, Jim and Scott, a question I'd like to ask you: What's the coolest thing about this? I mean, there's clearly - you know, we're talking about open source technology, an optimized stack, very, very low cost of entry and the total software price for this was [unintelligible] about zero dollars, in terms of your cost of software products used. When you look at it from a systems perspective, what's the coolest thing to you?

Scott:

Well, I guess, you know, it's the enabling of this, another really killer example of social networking and how it all comes together in a very sort of organic process of, you know, pulling down the latest open source package for whatever, you know, given component we're talking about, and connecting it to a database and lighting it up. It was really very straightforward process; the thing that took the most time was to really get the very specific and pretty elaborate requirements from Architecture for Humanity implemented in the Open Architecture Network. But from my point of view, it's just the use of all the open source technologies and the communities behind them leveraging, engaging with those communities behind this - more specifically, the Drupal community, and you know, sort of tapping into their best practices and lessons learned on how to implement this stuff. And I think now that the beta site is coming online, we'll find that Architecture for Humanity and all of the participants in that community are going to engage very closely with the Drupal community and we'll see spurs and sub-projects sort of grow up around this that also involve the Drupal community. So, you know, we're seeing the Drupal community expand at the same time that the Open Architecture Networks community is coming online and growing, as well.

Jim:

Yeah, I agree with that fully, Scott. This is a complete, open piece of software - all the code that runs the entire Drupal system, as well as our custom modules, is sitting there right in front of each developer to explore into. And there were several times where I'd found subtle bugs within the core of Drupal which I posted at Drupal.org and watched get fixed within the context timeline of a very, very brief timeline project. So, the community is not just there, it's a very, very active developer community. That, for me - the best thing I can do to help Architecture for Humanity and the Open Architecture Network, was to write a lot of code, as much as I could, in a very short period of time at a high level of quality. And because of the Drupal community, because of having so many developers actively looking at the bugs that we posted and then trying to solve them right away, it made the ability to get this project done much more fast. I mean, I literally had thousands of programmers looking over my shoulder, able to lend a hand in specific cases. That's awfully handy.

Hal:

I think it's also that economy of scale that is one of the mechanisms that's going to make Cameron Sinclair successful in fulfilling his TED wish of making his architecture resources available around the world. It doesn't take a lot of effort by each individual when you have that many individuals contributing; and furthermore, when you start to take diverse viewpoints and diverse global viewpoints and put them together, that's where innovation's gonna come from. It's when someone looks at the problem a new way or applies a new solution to an old problem and suddenly breaks through a constraint that you used to be hampered by, whether it's cost or material or complexity. And you know, the net result really does have an impact. I highly recommend taking a look at Cameron's book, which is called *Design Like You Give a Damn*, which catalogs the work he and Architecture for Humanity have done around the world in building low-cost, economically- and environmentally- sustainable housing. So, any final thoughts Jim Scott?

Scott:

Well, I guess I would just like to, you know, sort of point out the importance and the potential for this project that Kate and Cameron had a vision for. And you think about the kinds of communities that are gonna be engaged in using the Open Architecture Network, Architecture for Humanity has a relationship with the United Nations High Commissioner for Refugees, and so very likely for any of the displaced people's projects that they have happening around the world, somebody from the UN HCR will be a member of these communities on the Open Architecture Network and participating in the design, learning from those designs, and then being able to parlay those into other similar situations around the world. So, there's gonna be a catalyst effect here that probably goes way beyond what anyone could even imagine to this point. So, you know, Metcalf's Law is in full force in this project, and I think that's, you know, sort of the really cool thing, from a physical building architect's point of view, and someone who's interesting in helping with these humanitarian crisis situations.

Jim:

You just can't take away the possibility of having several thousand programmers on your team for free, helping you build the core of your software. Those eyes - those free eyes is so valuable. These are also people who really care about the technology, as well, not people who are just being paid to do it. These are people who want to contribute in the community, as well. And because there are so many peers looking at the same code base, there's a - it's kind of hard to slip very poor code into the base. The peer review increases the quality. So, you know, Drupal is still maturing, but as time goes on, it just gets better and better and it's definitely something to keep a close eye on, as time goes on.

Hal:

So, whether it's computer systems architects improving the socialization of their software, or physical building architects improving the socialization of their designs, either way, architecture remains the key ingredient here, and the notion of open sourcing and sharing and making things available certainly drives the attraction and lowers the barrier for anybody to be able to go contribute and to benefit from the work that's been done. So, great stuff, and certainly exciting to see us working with Cameron Sinclair and Kate, in fulfilling Cameron's TED Prize wish. And I want to thank Jim Manico and Scott Matton for being our guests here on *Innovating@Sun*, and I'm your host, Hal Stern.

Announcer *You've been listening to *Innovating@Sun*. Join us next time for the latest in innovation from Sun Microsystems. Only on the Sun Microsystems Podcast Network.*