FrOSCamp 2010 in Zürich

by Karsten Dambekalns on September 24th, 2010

A while ago Lukas Smith from Liip AG (http://www.liip.ch/) in Switzerland invited me to FrOSCamp (http://wiki.froscamp.org/) in Zürich. He was gathering a group of people to talk about potentially using the JCR (http://en.wikipedia.org/wiki/Content_repository_API_for_Java) specification for the upcoming Symfony CMF (http://cmf.symfony-project.org/). Added bonus: drive Jackalope (https://fosswiki.liip.ch/display/jackalope/Home) and CouchDB for Doctrine (http://github.com/doctrine/couchdb-odm) further in two hackfests.

Among the attendees were not only Symfony developers but also Benjamin Eberlei (Doctrine Project (http://www.doctrine-project.org/)), Nils Nadermann (phpBB (http://www.phpbb.com/)), David Nuescheler (Day Software (http://www.day.com/)) and a lone FLOW3 (http://flow3.typo3.org/) developer - me.

On the first day David Nuescheler (the JCR specification lead) gave an introduction to JCR (http://www.slideshare.net/uncled/introducing-jsr283) and talked about the plans for JSR-333 (http://www.jcp.org/en/jsr/summary?id=333) (the next version of the specification). Since the basics were known to me, I was more curious about what will be the scope of JSR-333.

News from JCR land

One criticism about JSR-283 is it's modularity. That is clearly an advantage for implementors of the specification, but users can easily get lost in a jungle of maybe-available-maybe-not features. That and other things should get better with JSR-333, as David aims for more input from actual JCR users (so feel free to apply for an expert group membership).

Other aims are to make JCR more friendly towards scripting languages (with e.g. their implicit ways of using iterators) and to more clearly formulate the inherent client-server architecture found in the JCR specification.

Furthermore a mid- to long-term goal for Jackrabbit 3 will be further separation of "client" and "server" parts, so that eventually the "server" part (i.e. everything below the transient space in JCR) can be molded into an Apache HTTPD module, written in C, fast and easily available to PHP and the like.

We briefly touched CMIS (http://www.oasis-open.org/committees/cmis/) along the way, but far more interesting seems to be JSOP (http://www.slideshare.net/uncled/jsop) (proposed by David to the IETF (http://lists.w3.org/Archives/Public/w3c-dist-auth/2010JulSep/0008.html)), what is said to be "Essentially: WebDAV + friends but simple" as it's an approach driven by practitioners...

What I had to tell

Back in 2006 we were looking for a way to store content in the to-be-developed version 5 of TYPO3. We thought it would be nice to store content in a content repository and the JCR specification (back then still in the form of JSR-170) seemed nice. David Nuescheler presented it to us in more detail during the TYPO3 Developer Days in Dietikon and we were hooked. The spec contained versioning, workspaces, the best from RDBMS and filesystems and more.

The only problem was that we would have to develop it from scratch in PHP, as a dependency on Java was (and still is) a no-go for TYPO3. To have a working solution until we were done with the native implementation we decided to use a PHP/Java bridge to connect to Jackrabbit, the reference implementation of JCR. That soon turned into a nightmare, so we dropped that idea again - today there are better ways for such an approach, as Jackalope shows.

Even if those problems do not influence a project, there are still some problems caused by the Java roots of the JCR specification. Among them are the handling of different signatures for the same methods, type mismatches and naming problems (no, you cannot have a method name *clone* or *and* in a PHP class). The possibility of long-running sessions in a JCR can be cumbersome to implement in an inherently stateless environment as PHP. Some of the cool stuff in JCR (multi-valued properties) is not so cool if you are used to PHP (associative arrays) and you discover those do not match too well.

There is more, and in the end we came to the conclusion that the content repository in TYPO3 Phoenix will not be following the JCR specification. We gained a lot from that, but the drawbacks are to many to outweigh any benefit we might have from being "standards compliant" in that sector.

Whether the Symfony CMF will be based on JCR or not is still not decided, although some have their doubts (http://www.devorigin.fr/articles/to-jcr-or-not-choosing-the-right-persistence-solution-for-the-symfony-cmf) - and those are definitely not to be underestimated.

What I brought back from Zürich

After two days in Zürich I brought home:

- New insights into Doctrine2 (thanks Benjamin!) and it seems to be possible to use it with FLOW3 for object persistence - that way we would gain access to a lot of know how and well-tested code :)
- More knowlegde about CouchDB, it's use as well as it's internals, and now can at least read Erlang (thanks Nils!)
- A possibility to work on a nice universal installer for PHP applications, finally (thanks Kore!)

Top that off with discussions about politics, Git, exception hierarchies, culture and more and you get two days of awesomeness!

Thanks to everyone who was there and to Liip for taking care of the travel expenses!

Comments

Anonymous | 28.06.2011

what about cmis? i'm anxiously looking for a php cms that will implement it.

Iphal | 03.10.2013

One of the most critized point about the JCR is of cuorse the J letter at the beginning. What would be perhaps interesting could be the extraction of the section 3 of the spec (Repository Model) and to try to transform it into a language agnostic RESTful Model. Then you would of cuorse have to host and standardize it under anyother umbrella than the JCP. Such a JCR 3.0 revision would then only be the details of the JEE instance.Now which organisation could host such a generic model (OASIS; IETF;)? Which legal issues with the current JCP process which should has some IP rights on all these resources?I am not part of the JCP/JSR170/283 expert group, but I am sure this topic was already discussed last year. Especially with the start and raise of CMIS. But now with the apparition of PHP forks , yes I agree, it would perhaps make sense to host/maintain & discuss the model outside of a pure Java/JCP lobby.Now all these standardization effort requires lots of time and energy. So after the JCR and CMIS who is ready to invest more money into this?Stephane

Leave a reply

Email * (won't be displayed)

Submit

Karsten Dambekalns

Creative Code Engineer

 (https://twitter.com/kdambekalns) (https://google.com/+KarstenDambekalns)
(https://github.com/kdambekalns) (http://www.linkedin.com/in/kdambekalns)
(mailto:karsten@dambekalns.de)

Contact / Impressum (/contact.html)