

Brian Carroll
April 6, 2006

Minutes from ALF Architecture Conference Call on March 30, 2006

Participants on the call

Bob Brady (Segue)
Daniel Gross (University of Toronto)
Tim Buss (Serena)
Brian Carroll (Serena)

Announcements and Agenda

Announcements:

- Call for discussion on the frequency we should hold the ALF Architecture calls
- Notes from the EclipseCon conference
- Defining services to make ALF easier to use, and become useful quickly, and can be called from BEPL
- ALF Security paper discussions – still soliciting feedback
- Interactions with the Eclipse Higgins project on the ALF SSO security implementation

Discussions

Frequency of ALF Architecture calls

Tim suggested we combine the ALF Architecture call with the Weekly project meeting. Bob Brady seconded that idea. Daniel Gross indicated he may not be able to participate if the calls are on Thursday.

ALF Bugs

Bob expressed an interest in working on the ALF Event Manager bug in bugzilla regarding the EventManager only handling event types enumerated in the XML Schema. Brian thought that idea was great and applauded it in general. Unfortunately just that morning he had had an email exchange with Manny on that bug and Manny was already working on that particular bug.

A discussion followed on fixing bugs as part of the path to becoming a committer. Brian took an action to encourage all bugs be entered into Bugzilla to provide an opportunity for interested participants to have a selection of bugs to fix.

Notes from EclipseCon

Attendees at EclipseCon briefly reported on some activities and projects of relevance to ALF:

Corona Project

There were several meetings with the Corona team. A number of “touch points” for integration were identified:

- Corona is building a Semantic store based on RDF and OWL that ALF could use to store cross-tool relationships
- Bridge: Corona focuses at the eclipse platform level (to distribute OGSF events at the desktop level). Although event granularity is finer than ALF, significant events could be passed on to ALF via a bridge.

SOA Tools Platform (STP) Project

There we some discussion with the SOA Tool Platform (STP), which has a focus on the Enterprise Service Bus (ESB). STP is a well attended project. Some possible “touch points with STP were identified:

- ALF may run on an enterprise service bus rather than on HTTP only.
- ALF could use an ESB for executing service flows. (For some ESBs, BPEL is translated into a proprietary execution language or to native Java (BPEL2J)
- Implementing the ALF Event Manager as SCA modules.

Buckminster project

A project called Buckminster seemed interesting and potentially applicable to ALF. Buckminster addresses the relationships between plug-ins and jars, making it easier to download plug-ins that require jars, since the rich component dependency mechanism of plug-ins does not extend to jars files

Mylar project

The Mylar projects has Java editing combined with a Bugzilla plugging. IT works real-time - bugs pop up in your view. Didn't look like lot of overlap with ALF however, as there were no business flows; there may be more overlap with Corona.

Bob noted that representatives from large software firms, have checked out the ALF project, looked through documents, and had lot of praise for ALF in terms of the requirements and architecture documents.

Discussion on How to build community of users.

Some suggestions were:

- Work on the ALF download process
- Increase project transparency
- Ensure the source code is including the pluggins up on CVS
- Allow users to get ALF and make it useful quickly.

Discussion on which services would be useful to have available for authors of ALF ServiceFlows

Some observations were:

- Login service
- Generalized notification service. (The Arch team had talked about an email notification service in the context of a default BEPL compensation mechanism.)
- Data transformation modules
- Source control services. Tim noted that the subject of source control manager services is fairly popular
- File transfer – Tim explained: so transfers don't go through the BEPL engine. And perhaps to share a common location for checkout (and checkin). Bob seconded the suggestion, noting that test manager doesn't need source control, but did need to upload test results (which could be large files).
- A generator of ALF events based on timers. For example to drive processes launches on a time-based mechanism.
- Increasing number of open source tools available that have web service interfaces (bugzilla).

ALF Security

At Eclipse there were discussions about doing the implementation within the Higgins project. This would increase the likelihood that the design and implementation would be scrutinized by security professionals.

Bob noted that Higgins has published excellent piece of code. And it uses the Eclipse Communication Framework (ECF) and is it relevant to ALF. There was a brief discussion of ECF and the fact that Corona was leveraging ECF for collaboration.

There was also a discussion the fit between the ALF SSO and Higgins. Brian thought they in that sense it complementary. Higgins has had great interest from its mailing list to figure out the data model.

Interestingly, (another observation from EclipseCon) the Eclipse Healthcare project also has a need for security and had read the ALF Security architecture paper.

Brian has been working on the Technology and Design for ALF SSO. It contains a list candidate of technology that may be applicable to ALF SSO. Brian noted that it is a work in progress, and as soon as it is in a basic coherent shape, it would be good to get public scrutiny. That document is a follow-on to the ALF Architecture fro SSO paper. It defines what the components ALF will need to build, and possible sources of these components – e.g, the Liberty Alliance, Shibboleth, Yale CAS, Open AML, etc. - and then evaluate the appropriateness of each technology for ALF. (The technologies were identified in the meeting minutes from this team a few weeks ago.)

Once that technology is defined, Brian intends to translate that into an implementation plan identifying the resources and skills needed to realize ALF SSO.

[More observations from EclipseCon.] Security is becoming an issue in Eclipse, and for RCP apps it becomes important. For example an RCP application in a doctor's office, where various people log on and off throughout the day. Interestingly, there is no log-on to Eclipse and no use is made of Java Security. As interest in RCP is increasing there will be increasing demand for enterprise class features, such as security at the OSGi level. We would like to have ALF SSO aligned with that work.

[Many thanks to Daniel Gross for taking notes which were helpful in preparing these minutes.]