



Eclipse ALF Checkpoint Review:

Moving from Validation to Implementation Phase

Ali Kheiolomoom (ALF Lead) and the ALF team

April 2006

Agenda

- Objective of the Checkpoint Review
- Criteria satisfied to exit the Validation/Incubation Phase
 - Working codebase
 - In the open
 - Learned the ropes
 - Active communities
 - Well documented technical architecture
 - Other indications of project health
- Recommendation to enter Implementation Phase

Objective of this Checkpoint Review

To establish that:

The Eclipse Application Lifecycle Framework (ALF) project *has met the criteria* to:

- Exit the Validation/Incubation phase
- Enter the Implementation Phase

A Working and Demonstrable Codebase

- Code contributions to Eclipse
 - ALF Event Manager contributed to Eclipse following the IP Review Process
 - ALF Eclipse Plug-Ins submitted to Eclipse
 - Undergoing Eclipse IP review
- Moving forward, all new development will be done from the start “within the Eclipse CVS”
- Code demonstrated to community at EclipseCon
 - POC demo given at demo session
 - POC demos at podiums

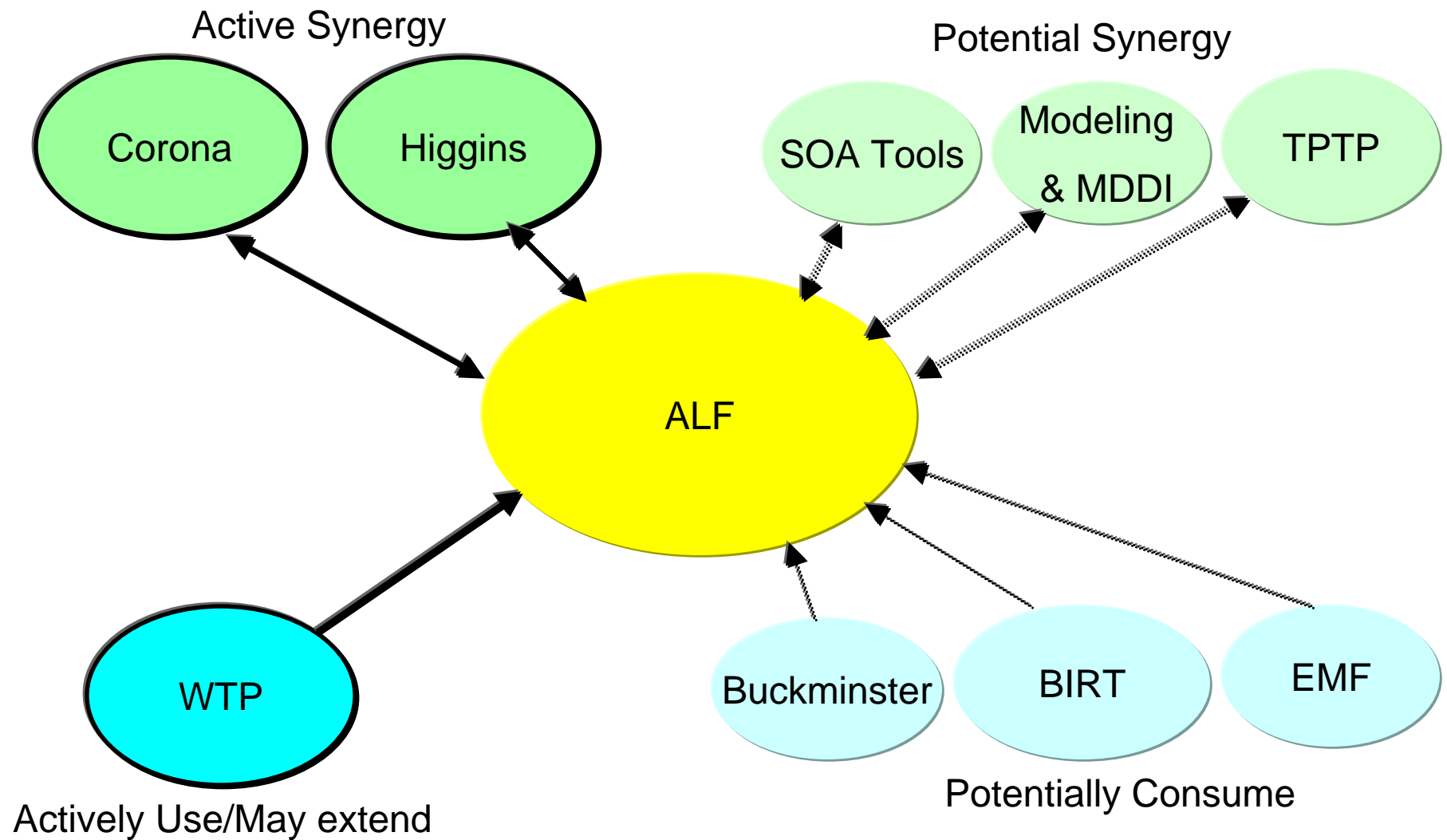
Project Operating *in the open*

- Newsgroups and mailing lists are actively used by the community
 - Weekly project conference calls
 - Minutes of meetings are posted so a newcomer can find out how the project is developing
- Extensive documentation and schedules on ALF website
 - Use cases, requirements, architecture and best practices documents
 - Schedules are published, open to review, and updated
 - Work items for Implementation phase are posted and scheduling is being refined

Project team members have learned the ropes

- The project operates in an open and transparent manner
 - Discussions and decisions
 - Conducted in conferences calls
 - Minutes posted to the newsgroups
- Bugs and enhancements are entered into Bugzilla.
- All new development will be done “in the Eclipse CVS”
- Embraced the frequent milestone approach to quality
 - See Implementation schedule on ALF website
- Other practices are known, but are still putting these in place
 - Packaged downloads, nightly builds, automated testing

Interactions with other projects



Active Communities

- Building momentum and community
 - Diverse Committer List
 - 4 Committers from Serena
 - 1 Committer from Segue Software (Borland)
 - 1 Committer (non-Serena) by September
 - 16 contributing companies
 - Framework users, tool users, multi-organization committer/contributor/developer community
 - Cover the major communities of developers, extenders, and users
 - Initial focus on “extend and use”
 - List is rapidly growing

Active Communities, cont.

- Additional channels for framework extension (besides extension points and plug-ins)
 - Vocabularies
 - Companies other than Serena are actively driving the various domain-specific vocabularies for ALF
 - Business Processes (Service Flows)

Technical Architecture

- An expanding set of documents
 - Use Cases
 - Requirements
 - Architecture – Events, Protocols
 - Best practices
 - Coming: “How to” guides
- Collaborating with the Corona project
 - Established a joint strategy
 - Identified and documented technical touch points
- Collaborating with the Higgins project
 - Active involvement and potential code contribution to Higgins
- Identified joint synergies with STP, TPTP, and BIRT
 - Contacts and discussions in progress
 - Code contribution to BIRT in progress

Other Indications of Health and Embracing the Eclipse way and community

- ALF Vocabulary definition effort gaining momentum
- Active participation and community interest at EclipseCon
 - Hosted 4 podiums
 - ALF demo session
 - ALF poster session
 - Joint ALF/Corona Poster session
- Presenting 2 sessions at EclipseWorld
- ALF Proof Of Concept (POC) now includes open source
 - CVS, Subversion
 - Extensibility and plugability of the ALF framework
 - Flash demo
 - Bugzilla (on Linux)

We recommend the Eclipse community approve:

- The Eclipse ALF project exits the Validation/Incubation Phase and enters the Implementation Phase

Q & A