

Kickoff in Stuttgart July 2010

Meeting minutes

Joram?

Action items

- Create wiki page in which companies share their motivation to participate in Activiti

Attendees

- Tom Baeyens
- Joram Barrez
- David Syer
- Micha Kiener
- Agim Emruli
- Christian Stettler
- Stefan Niederhausen
- Bernd Ruecker
- Jacob Freund
- Esteban Robles
- Christian Mulder

Goals

- Define the roadmap as concrete as possible
- Jumpstart collaboration process and practices

Meeting location

<http://www.reinform.de/reinform/anfahrt/loeffel.php>

Hotel most of us booked

<http://www.steigenberger.com/en/Stuttgart>

Wednesday June 30

Dinner

We'll go for dinner and this is a perfect occasion for other people to join informally to meet up.

We meet up in [Hotel Steigenberger](#) between 19:00 and 19:30. At 19:30 we leave for the restaurant.

Please put your name here or send me (t o m a t alfresco d-o-t com) a mail if you want to join so that we can book a restaurant.

- Tom Baeyens
- Joram Barrez
- David Syer

- Agim Emruli
- Christian Stettler

Thursday July 1st

Meeting schedule

| Time | Topic |
|---------------|--|
| 9:00 - 10:00 | Coffee |
| 10:00 - 10:30 | Round of introductions. Every company gets 6 minutes :-) |
| 10:30 - 12:00 | <p>Validate architecture and look for aspects of the architecture that need further discussion/decisions.</p> <ul style="list-style-type: none"> • Where Activiti came from. What made jBPM successful? <ul style="list-style-type: none"> • We're sensitive on practical api • Collaboration aspect: expect renamings if you add api • What do we keep and what do we want to improve? • PVM architecture overview and core execution algorithm • PVM persistence <ul style="list-style-type: none"> • Can we decide on myBatis? |
| 12:00 - 13:30 | Lunch |
| 14:00 - 14:30 | Timeline overview |
| 13:30 - 18:00 | <p>Scope discussions. List all the targeted features and indicate if they are</p> <ul style="list-style-type: none"> • 5.0 : Minimal scope • 5.x : Optional • Nice ideas |
| 18:00 - ... | Teambuilding ;-) |

Dinner

We'll probably meet up around 18:30 somewhere in the center of Stuttgart. Later will list the exact time and date. List your name if you want to join.

Friday July 2nd

Meeting schedule

| Time | Topic |
|---------------|--|
| 9:00 - 11:00 | Concrete roadmap |
| 11:00 - 12:00 | Collaboration practices <ul style="list-style-type: none">• Infrastructure: (IRC, mailing lists, forums, jira, roadmap, glossary pvm/bpmn,...)• Run update and check before commit• Decision process• Sprint layout |
| 12:00 - 13:30 | Lunch |
| 13:30 - 15:00 | Delayed topics |
| 16:00 - 17:00 | Roundup + group hug |

Other potential topics

PVM

- how much separate libraries would we target? e.g. pvm, bpmn, task, identity, persistent-api
- how would the API look like? now `processEngine.getXxxxService()`
- how would the docs be structured?
- how would the links between task completion and process continuation be implemented?
- how to offer the test helper method implementations?

API

Configuration, initialization of process engines

Currently there is the strategy of setup building jars containing configurations. There is the programmatic creation of a process engine, there is the ProcessEngines registry and the servlet for automatic initialization.

We should establish a common strategy on how we deal with configuration files and initialization.

Following aspects should be considered and as much as possible unified:

Registry pattern

- auto scanning and init and destroy
- basic process engine creation
- support for multiple engines
- ant process engine initialization, caching and destruction. also, how does an ant task reference a process engine

Test APIs

Should the test utilities and APIs be split off into a separate project? Would mean only unit tests in activiti-engine (but that's the way it is now anyway).

DB persistence

Current architecture

Alternatives

- JSON serialization into DB
- Cloud bigtable persistence

Is the current architecture OK for 5.0?

BPMN compatibility

Biggest question around BPMN is the suitability for developers. jPDL was much more readable. Will BPMN shortcut suffice? Will just have to take the pain of a verbose language? Or do we need to go for usability, compactness and readability without compromises.

How do we make it easy for users to work with all these namespaces (if we decide to put custom conveniences in namespaces)

Is there a clear scope we can target? Simple or descriptive conformance? Maybe Bernd could prepare this topic as he's closest to the spec. But most important is not the knowledge of the actual conformance, but arguments that give us guidance on what kind of conformance we target and how strict we'll be. The alternative is that we define our own subset of what we think is useful.

Database upgrades. Automatic QA for upgrade.

Cloud support

The biggest question here is: How will we align the roadmap of researching the cloud solution with the roadmap towards 5.0GA in november?

Query APIs

An important aspect of cloud is the query API. We currently plan to build out the query API similar to jBPM 4. For example

```
taskService.createTaskQuery( )  
    .processInstanceId(processInstanceId)  
    .activityName("task1")  
    .orderAsc(TaskQuery.PROPERTY_NAME)
```

But that might be hard to implement on cloud persistence. Maybe there are solutions. MapReduce?

Delegatable tasks

- Process instance migration
- History / reporting
- Email notifications / email form processor

- Business calendar
- Rules integration
- web service invocation
- Functional activity types / connectors: SQL, SAP/Abab, file, esb, xml transformation
- Identity pluggability + LDAP implementation
- Nomenclature

Define PVM event (api)/history event/event (listeners) and map it to BPMN

Other topics

- Discuss decision process
- Common 'terminology list'
- OSGi compatibility
- Process class loading
- To what extent can we have a dependency on Spring for core Activiti functionality?
- Naming of Job (name clash with spring Batch)
- Choreography: could it be a way to handle the multiplicity mismatches?
- Work out the cloud example: Elect the next european president, or process the voting for the next eurovision song contest.