JWT WAM and AgilPro IFW/AFW Alignment

Claus.Wittmann@e-mundo.de ; 19. Mai 2007

The migration of AgilPo IFW/AFW to JWT WAM is planned in July 2007. We are looking forward to find someone at the eMundo staff to do the JWT WAM support. Anyway, everybody at the JWT project team is invited to participate at the WAM development ;)

Process definition management:

- Repository Management:
  Nothing implemented.
  Thoughts about: It may be useful to manage more than one repository with one tool (but in practice, this isn't a very common feature, even in commercial tools). To that effect uploading model files in an independent database and testing the models in separate engines have to be provided.

- Package Management:
  Considering the generic and vendor independence goal ofAgilPro we did not use the versioning system inside the jBOSS jBPM BPEL Extension Engine (which would have fit quite well). This would be contradictory to the idea of a generic WAM. At this point it makes sense to find and integrate an open, standardized and generic solution (e.g. an Eclipse based solution).

Process execution management:

- Process Instantiation Management:
  There is only little implemented yet: actions for starting, aborting and resuming processes. It would be a good idea to use the same tool for execution and modeling of the process (in read-only mode or preview picture of the process). See pictures at http://sourceforge.net/projects/agilpro

- Process Monitor:
  Nothing implemented yet.
  What functionality should a process monitor system include?
  Management Cockpit?
  Detailed Information about every User and his 'open processes'?
  Escalated Processes?
  …

Process runtime resources mapping:

- User Management:
  Actual there is no User Management implemented neither within our Framework nor in jBOSS jBPM BPEL Extension. It would be necessary to implement different systems like LDAP, Active Directory (which is also LDAP based) and so on, to ease integration in company IT-infrastructures. Additionally I can see there a problem with integrating a hierarchically user management system:
  In a WAM system there will be many users with different positions (CEO, IT, users) and different rights and constraints. Also there may be temporary replacements, permanent representations and so on. For instance think about two small groups.

Every group has a leader and two members. In a flat hierarchical system your team leader will have "all rights" to manage processes, while your team members will only be allowed to execute and manage their own.

So your Team Leader will also have the right to begin a process and maybe sign it by himself (e.g. application to leave for him?)?

This has to be considered by selecting/implementing a User Management System. At this point I suggest to find and integrate an open, standardized and generic solution (e.g. an Eclipse based solution).

- Application Mapping:
  The Adapter framework (AFW[1]) of AgilPro already provides the possibility to integrate real applications which can be accessed by a generic API. The AFW-API offers generic interfaces for external system integration and implementation. Actual we can provide Open Source - Adapters for ActiveX (Word, Excel, Adobe Acrobat), JDBC, Google Maps, OpenOffice, Simple Mail and Browsers (SWT-Implementation)

User-oriented process management:

- Worklist Management :
  This is comparable to Process Instantiation Management. Furthermore our second Tool "AgilPro Simulator" (in future called "AgilPro Previewer") is already capable of executing deployed BPEL-Processes alike executing Agilpro-LiMo processes. (These BPEL-Processes are deployed in a jBOSS BPEL Engine → so the Simulator connects itself to this engine).
  Relating to 'Admin'-Users I want to reference to my previous statement 'User Management'.

- "Desktop Business Process" tooling:
  This is one of the capabilities of the AgilPro Previewer (alias Simulator).

---

[1] We built for AgilPro an adapter framework (AFW). This framework instantiates, holds and manages all adapters. A special simulator (AgilPro Simulator) reads the model file and builds a flow context of the model. The simulator uses our adapter framework and gets instances of specified adapter e.g. Firefox.

The adapter implementation is specified by a generic interface containing methods to set (and get) data objects to the adapter implementation.