

Added by [Nicolas Modrzyk](#), last edited by [Nicolas Modrzyk](#) on Sep 15, 2008

What are the standard sets of actions provided by Tempo?

Tempo provides out of the box most action as defined in the BPEL4People specifications. The WSDL exposed by Tempo are closely following those proposed by [WS-HumanTask](#).

How to add another framework action to Tempo?

Say, "Forward" (reassign a task to another user or role) or "Delegate" (when a potential task owner is on vacation, he/she can name a delegate to process tasks on his/her behalf).

Delegate and Reassign are fully implemented in Tempo. To implement a wholly new action, you would modify the Task Management Process, ideally graphically using Intalio|Designer, and map the new action in the process.

Task history

We do not provide task history right now, but we are using OpenJPA for persistence of tasks, which means we could just enable the versioning of objects in the different tempo tables, and propose a filter to retrieve sensible information ; OR we could obviously propose a history framework, since what users are interested, or have to implement to be legally compliant is a standard "who did what, when on which object", which is more or less a traceable log on the RBAC interfaces.

Task deadline

E.g. Automatic rejection when a task deadline is exceeded

Automatic look up and escalate to a task owner's manager when a task deadline is exceeded (how strong is our "role resolution" functionality? Does Tempo support run-time role resolution?)

Tempo does support tasks deadline. By default, if the deadline of a task is reached it is in put in a FAULT state, and then requires administrative intervention. To change this behavior, you would open the Task Manager Process in designer and make the few minor changes according to your plans and ideas. In Intalio|Designer you would actually call a "role resolution service" that would reassign the task to the proper user in your organization, depending on the company policy. And of course, Tempo supports run-time role resolution.

Does Tempo support chained escalation?

Let's take the [IBM nomenclature](#) to define "chained escalation".

In a chained escalation, the escalations are executed sequentially, or one after the other. In such a case, the first escalation must fully complete before the next one is initiated.

This is fully supported, and you can find a [sample process](#) on the Intalio website, in the samples section.

Attachment handling

Fully supported by the Task Attachment Services. Storage by default is WDS (a restful storage service), we also have support for direct storage to alfresco, and could support pretty much anything that store documents with credentials, since Tempo has a very generic interface for this that can accustom to most needs.

Can Tempo assign tasks by organization units or user groups?

We have a very clean RBAC implementation that can be layered on top of any directory service. It's easy to use roles to describe groups. A group is simply a role without any specific privileges.

Does tempo support user exclusion (i.e. the 4-eye principle specified in BPEL4People)?

Yes, we do, through the use of XPath.

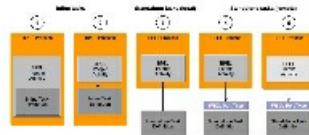
Task properties (e.g. priority)

Priority is implemented out of the box in Tempo. We are working on a Task Descriptor to implement Generic Metadata on task ; Right now, it's a few changes in TMS to add a new property to a task.

Matrix of supported BPEL4People features and scenarios.

BPEL4People	Tempo
supports role based interaction of people	x
provides means of assigning users to generic human roles	x
takes care to delegate ownership of a task to a person only	x
four eyes scenario	x
nomination scenario	x
escalation scenario	x
chained execution scenario	x

In the BPEL4People specifications, five different interaction models between Tasks and Processes are presented. Which one does Tempo follows ?



Tempo follows interaction model #5 (using BPEL invoke activity). The main difference between models 4 and 5 is the use of the BPEL4People activity versus the standard BPEL <invoke>, so it's mostly a semantical modeling difference. If you're using a higher-level semantic model (e.g. BPMN) then you don't really see the difference other than when looking at the execution details.

Where is the status of the BPEL4People specification ?

<http://www.osoa.org/jira/browse/BP>

Comments

Site powered by a free **Open Source Project / Non-profit License** ([more](#)) of [Confluence - the Enterprise wiki](#).
[Learn more](#) or [evaluate Confluence for your organisation](#).

Powered by [Atlassian Confluence](#), the [Enterprise Wiki](#). (Version: 1.4.1 Build:#212 Jun 02, 2005) - [Bug/feature request](#) - [Contact Administrators](#)