

Le moteur de workflow JBPM

Claude Duvallet

Université du Havre
UFR Sciences et Techniques
25 rue Philippe Lebon - BP 540
76058 LE HAVRE CEDEX
Claude.Duvallet@gmail.com
<http://litis.univ-lehavre.fr/~duvallet/>

Le moteur de Workflow JBPM

- 1 Introduction à la notion de Workflow
- 2 Présentation de jBPM
- 3 Le langage JDPL
- 4 Outils

Généralisation de la notion de workflow (1/2)

- ▶ Par workflow on entend beaucoup de choses...
 - Cinématique des interactions utilisateurs.
 - La gestion des données.
- ▶ Toutes les applications ont un workflow :
 - Mais il est en général fixé, ou peu modifiable.
- ▶ Quelques exemples :
 - Un wizard,
 - Un formulaire suivi d'une validation.
- ▶ Certaines applications métiers ont une nécessité particulière de devoir s'adapter aux besoins d'une entreprise.
- ▶ Par exemple, tout gestionnaire de contenu (GED, CMS)
 - Rarement le même processus de validation.
 - Un nombre de validation différents, par des acteurs différents (un relecteur, suivi d'une validation par un autre journaliste, puis une autre validation par un directeur de programme).

Généralisation de la notion de workflow (2/3)

- ▶ Mais aussi toutes les applications liées à la sécurité :
 - PKI (Gestion du cycle de vie des certificats).
 - Provisionning d'identité.
- ▶ Quelques constats :
 - Les développeurs sont rarement spécialisés métier.
 - Les consultants métiers ont rarement des connaissances techniques suffisantes pour manipuler du code.
- ▶ Les idées :
 - Création de moteur de workflow permettant un découpage entre les parties techniques et les parties métiers.

Généralisation de la notion de workflow (3/3)

- ▶ Certaines personnes ont réfléchi à des moteurs de workflow :
 - WfMC (Workflow Management Coalition).
- ▶ Les idées :
 - Gérer, archiver les instances de workflow.
 - Gérer les données manipulées au sein d'une instance de workflow.
 - Orchestrer les différents modules que composent une instance de workflow (briques métiers).
- ▶ En pratique cela donne :
 - Une vision métier (représentation sous forme de graphique) d'un workflow, généralement réalisé par un consultant métier.
 - Une « glue » technique permettant la mise en oeuvre du workflow métier.

Les WebFlow

- ▶ Les webflow (ou pageflow) sont un cas particulier des workflow dans le cas d'une utilisation d'un site web.
- ▶ Décris l'enchaînement des pages.
- ▶ Pas de persistance systématique.
- ▶ Une brique métier (au sens workflow) peut être composé d'un webflow lorsqu'il y a interaction avec l'utilisateur via une interface web.

Exemples d'utilisation (1/2)

- ▶ L'équipe sécurité travaille sur le workflow pour l'application de PKI EJBCA, pourquoi ?
- ▶ Aucun client n'a le même besoin métier.
- ▶ Dans le cas d'une demande de certificat, il peut exister de très nombreux points d'entrée différents, et des cinématiques totalement différentes en fonction du point d'entrée.
- ▶ Les workflow métiers peuvent évoluer au sein d'un client, d'où l'idée de faire une application très modulaire.

Exemples d'utilisation (2/2)

- ▶ Pour bien comprendre, prenons un exemple :
 - La DRH - à la signature d'un contrat - remplit un formulaire web (outil de gestion des identités interne à la société).
 - Cela provoque la demande de deux certificats pour cet utilisateur (signature et chiffrement).
 - À son arrivée, l'employé génère une carte à puce avec son certificat de signature.
 - Plus tard dans la journée, obtention de son certificat de chiffrement.
- ▶ Et un autre totalement différent :
 - Une entreprise souhaite équiper tous ses routeurs de certificats pour authentification ipsec.
 - Les routeurs génèrent une demande de certificat (via le protocole SCEP) auprès de la PKI.
 - Un administrateur doit valider les demandes.

Les aspects pratiques de la gestion de Workflow

- ▶ Dans un premier temps, il est question de définir un langage permettant de représenter le workflow, malheureusement il en existe un certains nombres :
 - BPM (Business Process Management)
 - XPDL (XML Process Definition Language)
 - BPEL (Business Process Execution Language)
- ▶ La définition de ces langages permettent d'utiliser indépendamment différents moteurs de workflow.
- ▶ Permet également la création d'interface graphique permettant la manipulation des workflow.

Présentation de JBPM

- ▶ Projet racheté par JBoss.
- ▶ Licence apache.
- ▶ L'un (sinon le) des projets de workflow le mieux documenté.
- ▶ Très léger : 2 jar indépendants, ne nécessite pas l'utilisation d'un serveur d'application (ex : JBoss).
- ▶ Le seul à supporter la plupart des langues (XPDL, BPEL, etc.).
- ▶ Enrichi XPDL pour donner JDPL (JBPM Process Definition Language).

Le moteur de workflow jBPM (1/3)

- ▶ JBPM est un moteur de workflow et de gestion de processus BPEL, adapté aux individus, aux applications et aux services.
- ▶ Le produit est parfait pour être intégrée dans un environnement Java lightweight.
- ▶ Le kernel workflow est très léger et peut être extensible pour supporter des langages de définition de processus très variées.
- ▶ Actuellement les langages suivants sont supportés :
 - JPDL : pour le workflow humain,
 - PageFlow : pour la définition des enchaînements des formulaires,
 - BPEL : pour l'orchestration des services web.
- ▶ Les outils d'administration et monitoring workflow sont limités dans la version actuelle.

Le moteur de workflow jBPM (2/3)

- ▶ JBoss jBPM est un moteur de workflow et de gestion de processus métiers.
- ▶ Il permet le contrôle des processus qui coordonnent les personnes, les applications et les services.
- ▶ Avec son architecture modulaire, JBoss jBPM combine le développement des applications workflow avec un moteur de processus flexible et extensible.
- ▶ L'éditeur de processus JBoss jBPM représente graphiquement les étapes d'un processus métier et facilite le lien entre les analystes et les développeurs.
- ▶ Puisque JBoss jBPM est présenté sous la forme d'une simple bibliothèque Java, il peut être utilisé
 - de façon indépendante dans un programme Java
 - ou avec un serveur d'applications comme JBoss Application Server, pour des applications d'entreprise hautement extensibles.

Le moteur de workflow jBPM (3/3)

- ▶ JBoss jBPM facilite les modifications des processus métiers complexes, ce qui améliore significativement la souplesse de l'entreprise.
- ▶ Le projet est hébergé par la communauté JBoss et fait partie du stack middleware JEMS proposé par JBoss Group.
- ▶ RedHat est le leader du projet avec une équipe de développement partagé entre la Belgique et les États Unis.
- ▶ En plus de la communauté animée par JBoss Group, le projet est actif avec des utilisateurs et validateurs en Europe et États-Unis.

Les fonctionnalités de jBPM (1/2)

JBPM est un moteur de processus flexible et extensible :

- ▶ Conçu pour fonctionner seul ou pour fonctionner parfaitement au sein de toute application Java ou J2EE.
- ▶ Modèle de programmation orienté processus qui associe le meilleur de Java aux avantages des techniques de programmation déclaratives.
- ▶ Support des langages de processus JPDL, PageFlow et BPEL.
- ▶ Définition graphique des processus via le Graphical Process Designer (GPD).
- ▶ Le langage JPDL proposé pour la définition des processus workflow humain est très intuitif mais on regrette que des standards workflow humain type XPDL ne soit pas supportés.

Les fonctionnalités de jBPM (2/2)

- ▶ L'interface graphique de définition des processus sous forme de plugin Eclipse permet de masquer l'utilisation du langage JPDL.
- ▶ L'intégration avec le framework JBoss Seam permet d'interfacier facilement le workflow avec une application web.
- ▶ Les outils d'administration et monitoring workflow sont très limités dans la version actuelle.
- ▶ Néanmoins, elles offrent la possibilité d'exécuter graphiquement un processus workflow.
- ▶ De futures versions mettront en œuvre la technologie innovante de 'Process Virtual Machine' développée en commun par JBoss, OW2 et Bull.

Caractéristiques de jBPM (1/2)

- ▶ Versions :
 - La version actuelle est la 4.4.
 - Depuis la version 4.x, on a une nouvelle architecture plus légère et facilement extensible.
 - Elle inclut le support complet du langage BPEL ainsi que l'éditeur graphique BPEL grâce à l'intégration du plugin Eclipse BPEL.
- ▶ Robustesse et performance :
 - JBPM est déployé en production dans des organisations appartenant à des domaines d'activités très différents tels que les finances ou la santé.
- ▶ Documentation :
 - La documentation en Anglais est complète.
 - Elle couvre l'installation du produit, les concepts de base, les exemples d'utilisation ainsi que la définition des processus et les concepts liés à la programmation.
 - Elle est disponible sous forme d'un document html.

Caractéristiques de jBPM (2/2)

- ▶ Facilité d'emploi :
 - JBPM est une librairie générique pour la gestion des différents langages des processus.
 - Cette généricité rajoute de la complexité lors de la première prise en main du produit.
 - Le modèle de base est basé sur le concept « Graph Oriented Programming » qui certes apporte de la flexibilité mais qui diffère des approches traditionnelles de workflow.
- ▶ Intégration :
 - JBPM peut être intégré dans des systèmes très légers type Java :
 - application Java stand-alone,
 - ou application web ainsi que dans systèmes plus complexes type J2EE.
 - L'utilisation de JBPM dans des solutions non-java nécessite le développement d'une couche de communication web services.

Le langage JDPL

JBPM Process Definition Language

- ▶ Un workflow est composé d'un ensemble de node :
 - Un (unique) startNode : étape à l'origine de la création de l'instance du workflow.
 - TaskNode : étape en attente d'une interaction avec l'utilisateur.
 - State : étape en attente d'une action extérieure (webservices, attente de réponse d'un composant externe à l'application).
 - Un ou plusieurs endNode : archivage de l'instance du workflow et libération des ressources.
- ▶ Les nodes task et state ont un état wait

D'autres types de nodes

- ▶ Fork : séparation du workflow en N branches devant se réunir via un node de type join
- ▶ DecisionNode : condition sur une variable de l'instance de workflow (utilisation d'un langage simple ou délégation à une classe java).
- ▶ Node : action interne à l'application mais n'ayant pas d'interaction ni avec l'utilisateur ni avec un élément externe (exemple : notification par courriel).
- ▶ On relie tous ces nodes par des transitions, représentant le changement d'un node à un autre au sein de l'instance.

Le comportement d'un node

- ▶ Chaque node possède trois événements déclencheurs :
 - `onNodeEnter` : déclenchée à l'entrée du node.
 - `onNodeLeave` : déclenchée à la sortie du node (n'a pas conscience du node suivant).
 - `onTransition` : déclenchée lors de la transition vers le node suivant.
- ▶ Pour information : un événement n'est pas lié qu'à une seule action, mais peut déclencher plusieurs actions.

La gestion des données dans un workflow

- ▶ Très important : un node n'a pas à connaître l'existence des autres nodes, il manipule un ensemble de données dans un contexte d'exécution.
- ▶ C'est-à-dire :
 - Il n'y a pas de variables directement passées d'un node à l'autre
 - Un node utilise l'instance du workflow pour manipuler ses données
 - Limitation possible de la visibilité des variables en utilisant le mécanisme des swimlanes

Les outils

- ▶ JBPM est livrée sous la forme d'un fichier ZIP téléchargeable.
- ▶ Ce fichier permet d'installer JBPM avec Tomcat ou JBoss au moyen de Ant.
- ▶ Le plugin Eclipse GPD est intégré au sein de l'archive.
- ▶ De nombreux exemples sont fournis avec l'archive.
- ▶ La documentation officielle est assez bien faite.