

Les processus métiers : concepts, modèles et systèmes

©Les processus métiers: concepts, modèles et systèmes

Organisation du cours

- Concepts et notations
- Modélisation des processus
- **Systemes de gestion de processus**
- Processus transactionnels
- Découverte de processus
- Intelligence des processus
- Processus interentreprises
- Processus sur le Web
- Validation des modèles de processus
- Dimension managériales des processus
- Conclusion

©Les processus métiers: concepts, modèles et systèmes

Chapitre 4 : Systèmes de gestion de processus

©Les processus métiers: concepts, modèles et systèmes

Contenu

- Concepts généraux
 - Modèle de référence de la WfMC
 - Rôle des interfaces
- Systèmes intégrés de gestion de processus
 - Bonita, IBM MQ Workflow
- Bibliothèques de processus
 - PVM, WWF

©Les processus métiers: concepts, modèles et systèmes

Introduction (1)

- Objectif d'un système de gestion de processus (SGP) :
 - Décrire la logique d'un processus et en contrôler le respect à l'exécution
 - Assurer l'interface entre ces fonctionnalités, les utilisateurs et les programmes d'application
- Concrètement
 - Des outils variés avec des architectures variées
 - Mais un ensemble standardisé de composants fonctionnels (modèle de la WfMC)
 - Si on retrouve ces fonctionnalités totalement ou en partie dans les outils, leurs mises en œuvre sont assez diverses

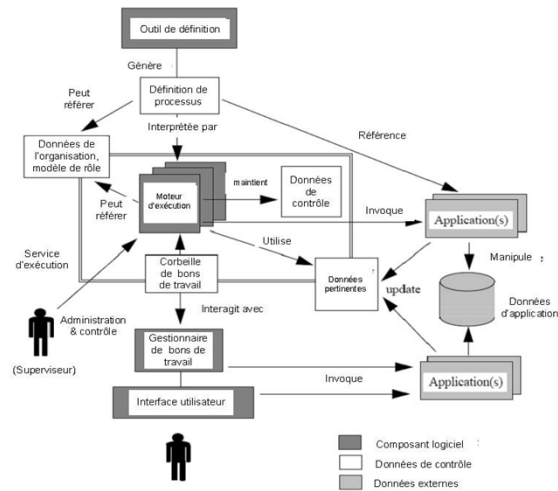
©Les processus métiers: concepts, modèles et systèmes

Introduction (2)

- Deux grandes approches :
 - Les suites intégrées de gestion de processus
 - Solution complète et générale pour une large gamme d'applications distribuées
 - Bonita, IBM MQ Workflow, W4 ...
 - Les bibliothèques de processus
 - Un ensemble de composants basiques pour une définition à la carte de SGP légers souvent embarqués dans des applications plus larges
 - Process Virtual Machine, Window Workflow Foundation

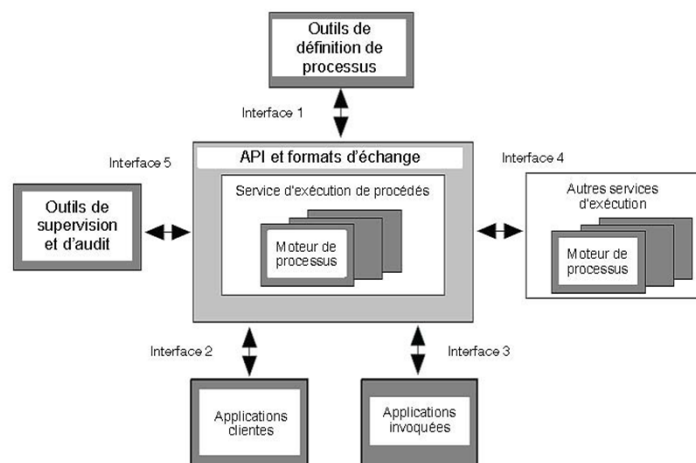
©Les processus métiers: concepts, modèles et systèmes

Les composants d'un SGWf (WfMC)



©Les processus métiers: concepts, modèles et systèmes

Le modèle de référence de la WfMC



©Les processus métiers: concepts, modèles et systèmes

Le service d'exécution des processus

- Création et terminaison des cas
- Navigation au sein d'un cas en interprétant le modèle de processus en fonction du contexte d'exécution
- Transfert des données entre les conteneurs d'entrée et de sortie des activités
- Affectation des bons de travail dans les corbeilles des ressources (en fonction des rôles, ...)
- Liaison des activités aux applications externes comme défini dans le modèle informationnel
- Capture des données historiques nécessaires à la supervision et à l'audit des processus

©Les processus métiers: concepts, modèles et systèmes

Le service de définition des processus

- La modélisation des processus en utilisant des notations (chapitres 2 et 3)
- La modélisation des rôles, des ressources et les liaisons statiques entre les activités et les rôles, les activités et les applications externes
- L'analyse et la simulation des processus (structuration, vivacité, blocage ...)

©Les processus métiers: concepts, modèles et systèmes

La gestion des applications clientes

- La gestion des sessions
- la gestion des corbeilles de bons de travail des participants en fonction des rôles
- La possibilité de trier et d'organiser les bons de travail
- L'instanciation, le démarrage, la suspension d'un processus, d'une activité
- L'information sur l'état du processus et des activités

©Les processus métiers: concepts, modèles et systèmes

La gestion des applications invoquées

- La gestion des connexions, déconnexions aux applications
- La gestion des activités invoquant une application
 - Démarrage/suspension/redémarrage/abandon d'une activité, gestion des signaux, notification de terminaison
 - La gestion des données en entrée et en sortie des applications

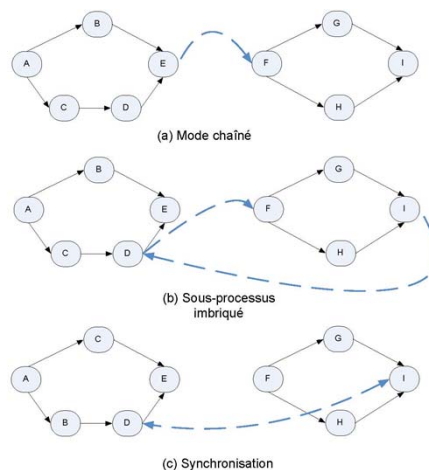
©Les processus métiers: concepts, modèles et systèmes

L'interopérabilité des processus (1)

- Mode d'interopérabilité
 - (a) mode chaîné
 - (b) mode hiérarchique
 - (c) mode synchronisation parallèle
 - (d) mode pair-à-pair
- Des notations pour l'interopérabilité (WfXML, XPDL)
- Supporte les processus inter-organisationnels (chapitre 8)

©Les processus métiers: concepts, modèles et systèmes

L'interopérabilité des processus (2)



©Les processus métiers: concepts, modèles et systèmes

La supervision des processus

- La gestion (évaluation, ajustement) des rôles, des utilisateurs
- La reconfiguration pour améliorer des paramètres (performance ...)
- L'inspection d'un processus en cours d'exécution
- La modification « à la main » d'une instance de processus
- La journalisation des informations d'audit
- Des heuristiques d'audit (taux moyen d'attente, d'exécution, taux d'utilisation des ressources ...)

©Les processus métiers: concepts, modèles et systèmes

Les utilisateurs d'un SGP

- Les concepteurs et analystes de processus
 - Modélisation a priori et re-ingénierie du modèle de processus, surveillance de l'exécution, analyse des traces d'exécution
- L'administrateur de processus
 - Responsable du fonctionnement d'un processus à l'exécution; est informé de tout dysfonctionnement et réagit en fonction
- L'administrateur opérationnel
 - Responsable du bon fonctionnement du matériel et du logiciel
- Les utilisateurs finaux
 - On peut distinguer : les clients (simple utilisateurs, par exemple Web) , les techniciens métiers qui peuvent gérer des activités complexes, le service support qui fait le lien entre les deux.

©Les processus métiers: concepts, modèles et systèmes

Les systèmes intégrés de gestion de processus

©Les processus métiers: concepts, modèles et systèmes

Avertissement : ces transparents ont été
réalisés sur la base de la version de Bonita
disponible en 2009

Depuis, le système a évolué, mais
l'architecture présentée est encore
représentative

©Les processus métiers: concepts, modèles et systèmes

Objectifs

- Proposer l'ensemble des fonctionnalités
- Proposer une large gamme d'outils pour l'intégration au système d'information de l'organisation hôte (interface BD ...)
- Assurer un haut niveau de sûreté et de fiabilité pour garantir la continuité de l'exploitation
- Exemples : Bonita, IBM MQ Workflow

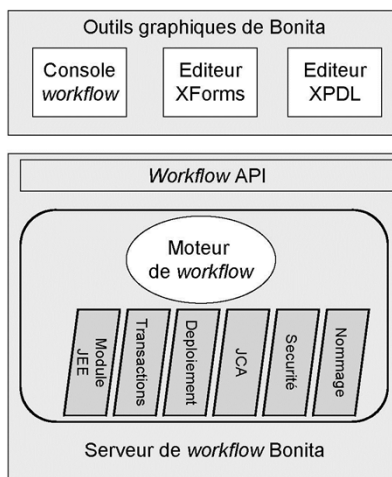
©Les processus métiers: concepts, modèles et systèmes

Bonita

- Communauté Bonita : <http://wiki.bonita.ow2.org/>
- Première version LGPL en 2004
- Nova Bonita depuis 2008
 - Contributeur à, et utilisateur de la PVM ([Process Virtual Machine](#))
- Aujourd'hui Bonita, un des leaders du BPM en France : <http://fr.bonitasoft.com/>

©Les processus métiers: concepts, modèles et systèmes

Architecture



©Les processus métiers: concepts, modèles et systèmes

Architecture

- Architecture *multitiers Java Enterprise Edition*
- Toutes les interactions avec les utilisateurs se font à travers la « *console workflow* »
- Le *moteur de processus* gère la description et l'exécution des processus, ainsi que l'interface avec les programmes d'exécution

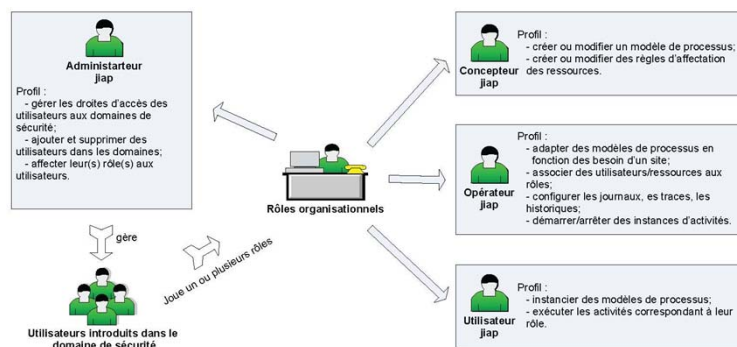
©Les processus métiers: concepts, modèles et systèmes

Implantation

- Développement Java Enterprise Edition (J2EE) :
 - *Entity beans* pour la sauvegarde des informations
 - Java Authentication and Authorization services pour la gestion des droits d'accès
 - Java Transaction Service pour la gestion des transactions (longues)
 - Java Messaging pour la gestion des alertes
 - Java Mail pour l'envoi des emails

©Les processus métiers: concepts, modèles et systèmes

Gestion des utilisateurs 4 profils principaux



©Les processus métiers: concepts, modèles et systèmes

Profil utilisateur

- Démarrage d'un processus
- Traitement des activités
- Visualisation des processus et des activités dans lesquels il est impliqué
- Lister les informations sur les activités terminées

©Les processus métiers: concepts, modèles et systèmes

Edition d'une condition

mozilla Firefox

Marque-pages Outils ?

http://localhost:9000/jiapAdmin/Welcome.do

BULL Workflow Console de gestion des

bsoa

bsoa	voyages
client_dette	SBC false

Submit

©Les processus métiers: concepts, modèles et systèmes

Etat d'une instance de processus

Nom de l'activité	Process name	Version	Etat
SBC	voyages_instance19	1.0	TERMINATED
RH_split_RV	voyages_instance19	1.0	TERMINATED
RV	voyages_instance19	1.0	READY
RH	voyages_instance19	1.0	READY
RH_join_RV	voyages_instance19	1.0	INITIAL

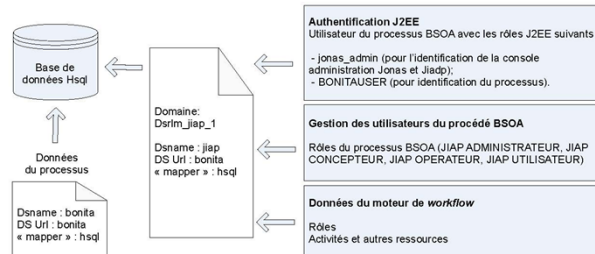
©Les processus métiers: concepts, modèles et systèmes

Profil administrateur

- Orienté système
- Gère la base de données des utilisateurs
- Définit le profil des utilisateurs

©Les processus métiers: concepts, modèles et systèmes

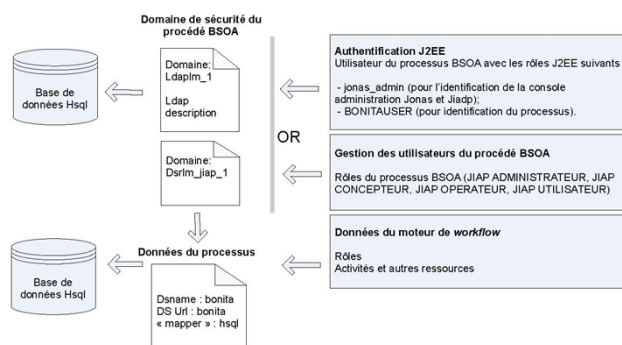
Gestion des rôles et des utilisateurs



Par défaut, Bonita fournit un référentiel de gestion des utilisateurs s'appuyant sur une base de données ...

©Les processus métiers: concepts, modèles et systèmes

Gestion des rôles et des utilisateurs



... mais ce référentiel peut être remplacé ou interconnecté à un référentiel d'entreprise existant, typiquement LDAP

©Les processus métiers: concepts, modèles et systèmes

Profil opérateur

- Peut intervenir sur le déroulement des instances de processus
- Gestion des erreurs en utilisant les traces et les historiques

©Les processus métiers: concepts, modèles et systèmes

Historique d'un processus

The screenshot shows a web browser window displaying the 'Workflow Console de gestion des processus'. The interface includes a navigation menu on the left and a main content area. The 'Historique' section is highlighted, showing details for a process instance named 'voyages_instance19'. Below the instance details is a table of activities with columns for 'Activités', 'Exécutant', and 'Date de démarrage'.

Activités	Exécutant	Date de démarrage
SBC	Client	Wed Feb 05 14:10:57 CET 2008
rh_split_rv	bsoa	Wed Feb 05 14:10:57 CET 2008
RV	Admin	Wed Feb 05 14:16:19 CET 2008
RH	Admin	Wed Feb 05 14:16:24 CET 2008
RH_JOIN_RV	bsoa	Wed Feb 05 14:16:24 CET 2008
FAI	Client	Wed Feb 05 14:16:28 CET 2008
cor_split	bsoa	Wed Feb 05 14:16:28 CET 2008
INT	bsoa	
EDF	Admin	Wed Feb 05 14:16:31 CET 2008
EDD	bsoa	
cor_join	bsoa	Wed Feb 05 14:16:32 CET 2008
RR	Admin	Wed Feb 05 14:16:37 CET 2008
BivoltaEnd	bsoa	Wed Feb 05 14:16:37 CET 2008

©Les processus métiers: concepts, modèles et systèmes

Le moteur d'exécution

- Environnement Java EE
- Données des processus gérées par le *container* EJB du serveur d'applications
- Stockage des données par le moteur de transformation Objet/Relationnel
- Notification des clients, des outils ... avec JMS
- Interprète des descriptions de processus en XPDL
- Les activités, les transitions s'exécutent comme des transactions JTS
- Connectivité vers des systèmes d'information par Web services ou connecteurs *Java Connector Architecture*
- Possibilité d'associer des actions aux événements du cycle de vie des activités (*Hooks*)

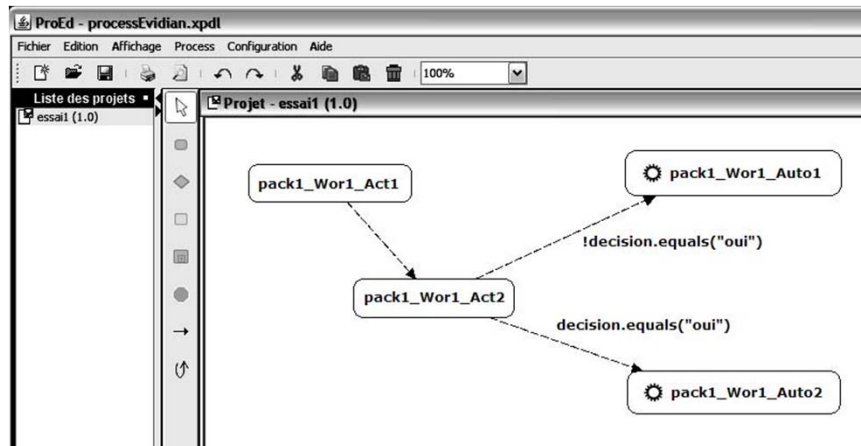
©Les processus métiers: concepts, modèles et systèmes

L'outil de définition de processus

- Outil graphique ProEd qui génère en sortie des description XPDL
 - Définition des activités, des transitions, des participants, des *hooks*, des sous-processus, des blocs d'activités
 - *Plugin* Eclipse
 - Formulaire de définition des activités

©Les processus métiers: concepts, modèles et systèmes

Editeur ProEd

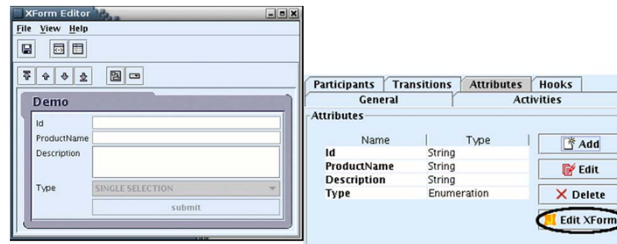


©Les processus métiers: concepts, modèles et systèmes

Formulaire de définition d'activité

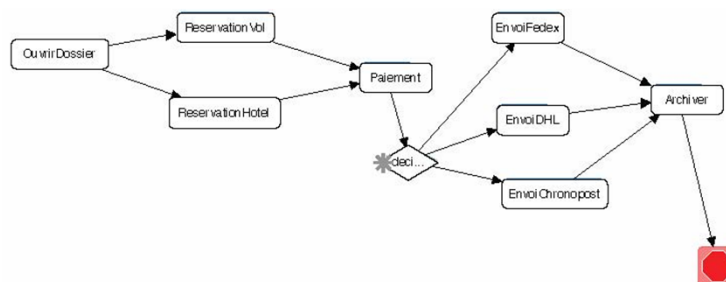
©Les processus métiers: concepts, modèles et systèmes

Formulaire d'une activité



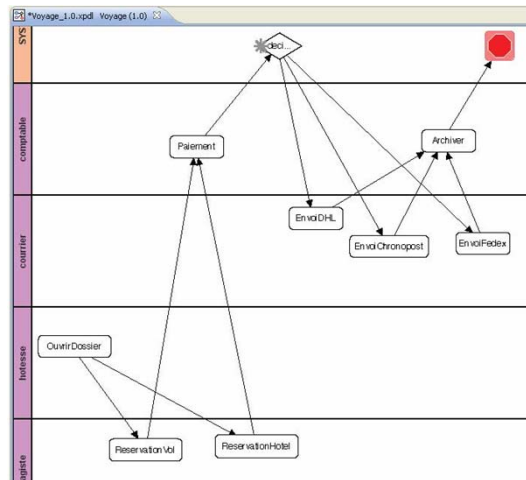
©Les processus métiers: concepts, modèles et systèmes

Le processus de gestion de voyage avec ProEd



©Les processus métiers: concepts, modèles et systèmes

Introduction de Swimlanes



©Les processus métiers: concepts, modèles et systèmes

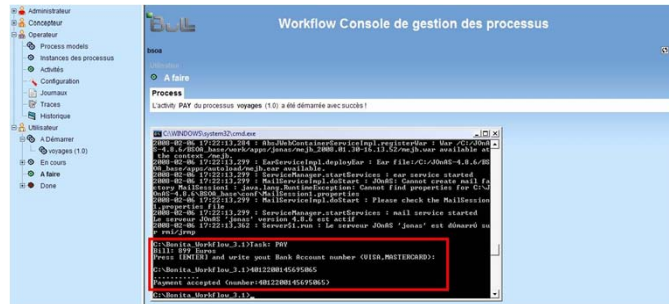
Exemple de Hook dans un processus XPD

```

Eclipse SDK
Navigate Search Project Run ProfEd Window Help
...voyages.xpdl voyages (1.0)
<!-- Performer:Client/Performer -->
<StartNode>
</StartNode>
<TransitionRestrictions>
</TransitionRestrictions>
<Join Type="AND" />
</TransitionRestrictions>
<ExtendedAttributes>
<ExtendedAttribute Name="XOffsetParticipantView" Value="27" />
<ExtendedAttribute Name="YOffsetParticipantView" Value="302" />
<ExtendedAttribute Name="XOffset" Value="662" />
<ExtendedAttribute Name="YOffset" Value="130" />
<ExtendedAttribute Name="CellHeight" Value="45" />
<ExtendedAttribute Name="hook" Value="hero.hook.payment">
  <HookEventName>afterStart</HookEventName>
  <HookScript>byte[] buffer=new byte[16];
  System.out.println("Task: " + n.getName());
  System.out.println("Bill: 899 Euros");
  System.out.println("Press [ENTER] and write your Bank Account number (VISA, MASTERCARD): ");
  System.in.read(buffer, 0, 16);
  System.out.println(".....");
  String card = new String(buffer).trim();
  System.out.println("Payment accepted (number:" + card + ")");</HookScript>
</ExtendedAttribute>
</ExtendedAttributes>
</Activity>
<Activity Id="RH_join_RV" Name="RH_join_RV">
  
```

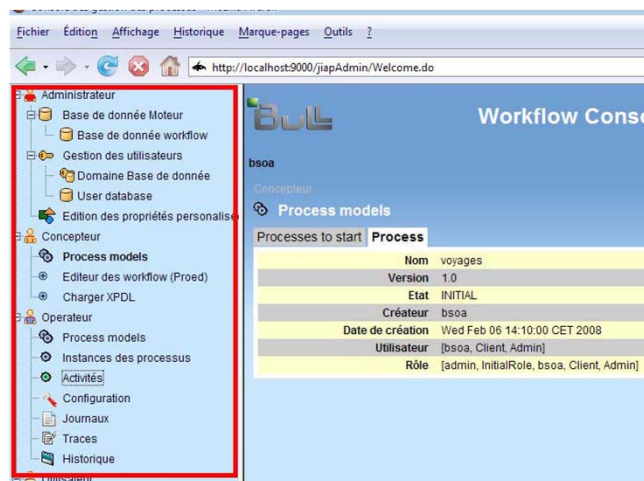
©Les processus métiers: concepts, modèles et systèmes

Exécution d'un *Hook*



©Les processus métiers: concepts, modèles et systèmes

Administration de Bonita



©Les processus métiers: concepts, modèles et systèmes

Gestion des transactions

- Gestion de processus de longue durée
 - Chaque opération système d'écriture et de mise à jour de l'état du processus lance une transaction Java EE
 - La gestion des opérations *Commit* et *Rollback* est déléguée au container EJB
 - Il est possible de choisir la politique de gestion transactionnelle des *hooks*
 - Néanmoins, il s'agit de transactions « classiques », pas de transactions avancées comme proposées WS-Coordination/Transaction (Chapitre 5)

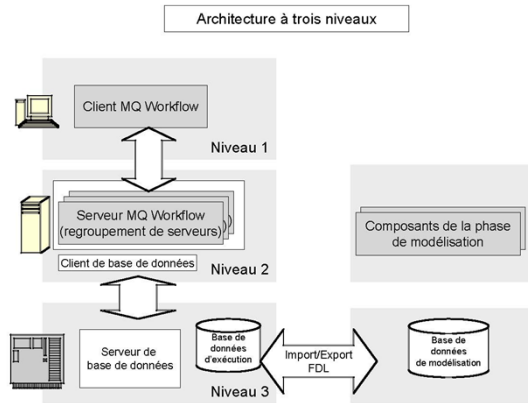
©Les processus métiers: concepts, modèles et systèmes

IBM MQ Workflow

- Système intégré complet
- Architecture à 3 niveaux
 - Le workflow comme un intergiciel entre le serveur de base de données et les applications clients
- Intégration par échange de message,
 - Plus grande fiabilité, passage à l'échelle
 - Bénéficie du contexte IBM

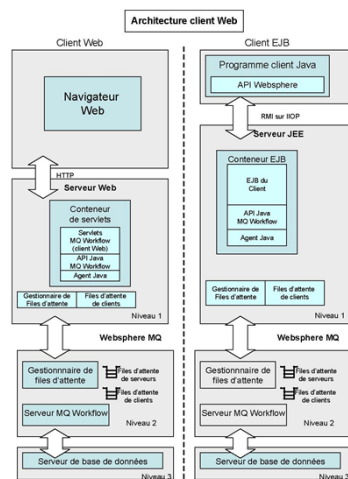
©Les processus métiers: concepts, modèles et systèmes

Architecture



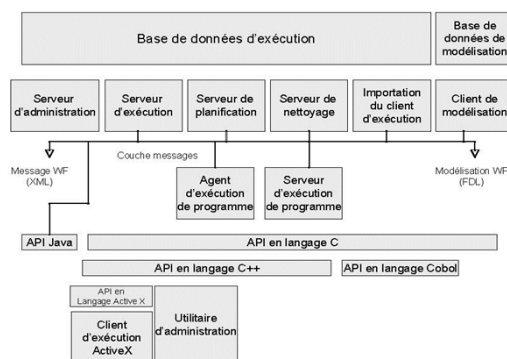
©Les processus métiers: concepts, modèles et systèmes

Intégration par queues de messages



©Les processus métiers: concepts, modèles et systèmes

API IBM MQ Workflow



©Les processus métiers: concepts, modèles et systèmes

Conclusion sur les systèmes intégrés

- Des fonctionnalités de plus en plus standardisées mais avec des mises en œuvre différentes
- Bonita
 - Une belle réussite du monde du logiciel libre
 - Une contribution à la PVM
 - Création en 2009 de la société BonitaSoft
- IBM MQ Workflow
 - Le standard du monde IBM
 - Bénéficie du contexte IBM (intégration fiable par queues de messages)

©Les processus métiers: concepts, modèles et systèmes

Les bibliothèques de processus

©Les processus métiers: concepts, modèles et systèmes

Objectifs

- Un ensemble de composants basiques
 - pour une définition à la carte de processus légers
 - généralement à embarquer dans des applications plus larges
- Deux approches :
 - La PVM du monde Java
 - La WWF du monde Microsoft

©Les processus métiers: concepts, modèles et systèmes

La *Process Virtual Machine*

- Noyau de SGP du monde Java
- Fondé sur JBPM (Java Business Process Management) et Bonita
- La PVM propose un noyau de modèle de processus basique mais extensible

©Les processus métiers: concepts, modèles et systèmes

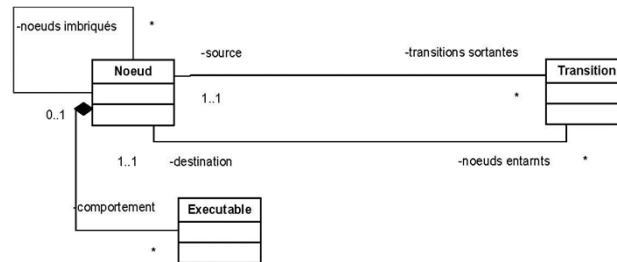
Principes

- Tous les modèles ont en commun
 - D'être représentés comme un graphe pour faciliter la communication
 - D'exprimer d'une façon ou d'une autre des flots d'exécution
 - D'être potentiellement de longue durée et d'inclure des états d'attente d'événements
- Mais varient en fonction de :
 - Du modèle de persistance choisi
 - Des types de transactions qui encapsulent les activités
 - Des services annexes utilisés
 - Des liaisons avec les applications de l'entreprise

©Les processus métiers: concepts, modèles et systèmes

Noyau de base

Modèle de graphes de processus, définition du comportement d'un nœud



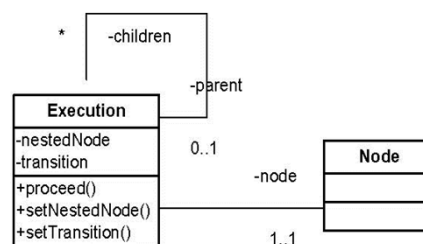
```

public interface Executable {
    void execute (Execution execution) throws
    Exception;
}

```

©Les processus métiers: concepts, modèles et systèmes

Modèle d'exécution de PVM



- A une exécution est associée un pointeur qui pointe sur le noeud courant
- L'exécution attend un événement externe qui déclenche la méthode *proceed* du noeud courant qui permet de décider quelle transition franchir

©Les processus métiers: concepts, modèles et systèmes

Définition d'un nœud de type *Tâche humaine*

```
public class TaskNode implements Executable {
    String taskName;
    public void execute (Execution execution) {
        //affecter une personne responsable de cette tâche
        User assignedUser = calculateUser(taskName, execution);
        //créer la tâche
        Task task = new Task(taskName, assignedUsr, execution);
        //ajouter la tâche au répertoire de tâche
        TaskRepository taskRepository =
            execution.getContext().getTaskRepository();
        taskRepository.addTask(task);
    }
}
```

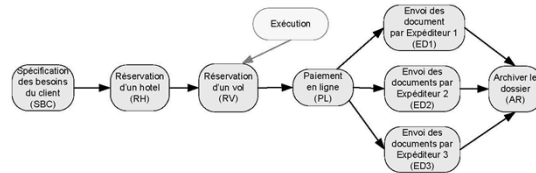
©Les processus métiers: concepts, modèles et systèmes

Définition d'un nœud de type *Courrier*

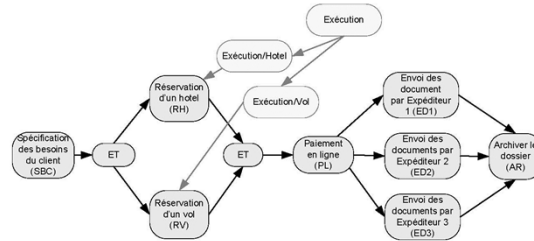
```
public class EmailNode implements Executable {
    String recipient;
    String subject;
    String text;
    public void execute (Execution execution) {
        // envoyer le mail
        sendEmail(recipient,subject,text,execution);
        // propager l'exécution
        execution.proceed();
    }
}
```

©Les processus métiers: concepts, modèles et systèmes

Extensions (1)



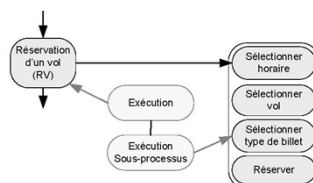
a. Graphe de processus et objet « exécution »



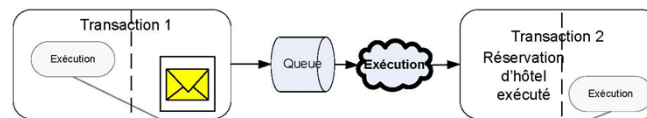
b. Flots d'exécution concurrents

©Les processus métiers: concepts, modèles et systèmes

Extensions (2)



c. Processus composé



d. Continuation asynchrone

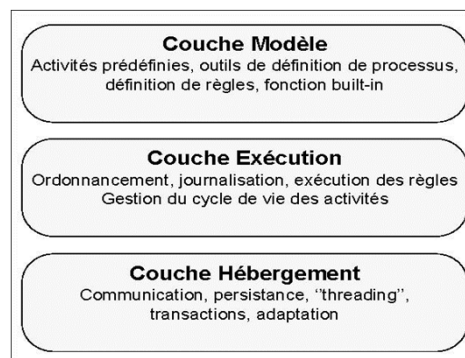
©Les processus métiers: concepts, modèles et systèmes

Window Workflow Foundation

- Objectif semblable à PVM
- Mais conçu pour le monde .NET
- Bénéficie du contexte et permet de générer aussi bien :
 - Des processus classiques
 - Des flots de page d'un site Web
 - Des processus de circulation de documents

©Les processus métiers: concepts, modèles et systèmes

Architecture : 3 couches principales



©Les processus métiers: concepts, modèles et systèmes

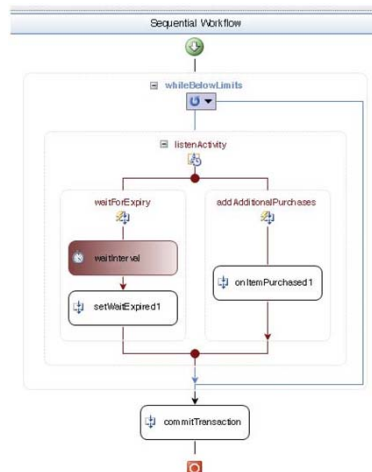
Définition d'une activité en étendant une activité existante

```
public class CreateActivity : Activity {
    public string Assignee { ... }
    public DateTime DueDate { ... }

    protected override ActivityExecutionStatus Execute(
        ActivityExecutionContext context) {
        // assign task to assignee...
        return ActivityExecutionStatus.Closed;
    }
}
```

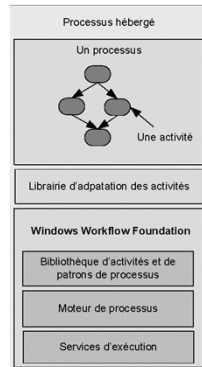
©Les processus métiers: concepts, modèles et systèmes

Outil de modélisation de processus (intégré à Visual Studio)



©Les processus métiers: concepts, modèles et systèmes

Hébergement d'un processus



Chaque application de la famille .NET 2007 intègre WWF en son coeur

©Les processus métiers: concepts, modèles et systèmes

Hébergement du moteur d'exécution

```
WorkflowRuntime runtime = new WorkflowRuntime();
runtime.AddService(...)
WorkflowInstance instance = runtime.CreateWorkflow(...);
instance.Start();
Guid id = instance.InstanceId;
```

Instanciation du moteur de processus, ajout de services,
création d'une instance de processus,
démarrage de l'instance

©Les processus métiers: concepts, modèles et systèmes

Les bibliothèques de processus

Conclusion

- Permettent de répondre à des demandes plutôt simples sans gros investissement ni gros déploiement
- Un effort de programmation est néanmoins encore nécessaire, mais beaucoup moins important qu'en partant de rien
- Et apporte la garantie d'une forme d'interopérabilité

©Les processus métiers: concepts, modèles et systèmes

Conclusion

- Une architecture fonctionnelle de référence (WfMC)
- Des mises en œuvre logicielles variées, libres ou propriétaires
- Des solutions plus ou moins lourdes pour répondre à une large gamme de besoins
- Des outils encore à destination des informaticiens, encore difficiles à mettre directement dans les mains des « hommes du métier »

©Les processus métiers: concepts, modèles et systèmes

Références

- [BON 08] BONITA, Système de gestion de workflow Bonita, wiki.bonita.objectweb.org/, 2008.
- [EB 08] EB, Entity Beans, java.sun.com/products/ejb/, 2008.
- [EJB 08] EJB, Enterprise Java Beans, java.sun.com/j2ee/tutorial/1_3-fcs/doc/EJBConcepts4.html/, 2008.
- [HIB 08] HIBERNATE, Persistance relationnelle pour Java, www.hibernate.org/, 2008.
- [JAA 08] JAAS, Java Authentication and Authorization Service, java.sun.com/javase/, 2008.
- [JBP 08] JBPM, Java Business Process Management, www.jboss.com/products/jbpm/, 2008.
- [JCA 08] JCA, Java Connector Architecture, java.sun.com/j2ee/connector/, 2008.
- [JDB 08] JDBC, Java Database Connectivity, java.sun.com/javase/technologies/database/, 2008.
- [JM 08] JM, Java Mail, java.sun.com/products/javamail/, 2008.
- [JMS 08] JMS, Java Messaging Service, java.sun.com/products/jms/, 2008.
- [JTS 08] JTS, Java Transaction Service, java.sun.com/products/jts/, 2008.

©Les processus métiers: concepts, modèles et systèmes

Références

- [LDA 08] LDAP, Lightweight Directory Access Protocol, www.sop.inria.fr/semir/personnel/Laurent.Mirtain/ldap-livre.html/, 2008.
- [NE 08] .NET, La plate-forme d'entreprise Microsoft, msdn.microsoft.com/fr-fr/netframework/default.aspx, 2008.
- [NOV 08] NOVABONITA, Bonita v4 basé sur la *Process Virtual Machine*, wiki.bonita.-objectweb.org/xwiki/bin/view/Main/Nova/, 2008.
- [PVM 08] PVM, Process Virtual Machine, docs.jboss.com/jbpm/pvm/article/, 2008.
- [WFM 08] WFMC, Modèle de référence, www.wfmc.org/reference-model.html, 2008.
- [WFX 08] WF-XML, Workflow XML, www.wfmc.org/wfmc-wf-xml.html, 2008.
- [WWF 08] WWF, Windows Workflow Foundation, www.workflow-foundation.com, 2008.
- [XAM 08] XAML, Extensible Application Markup Language, msdn.microsoft.com/en-us/library/ms752059.aspx, 2008.

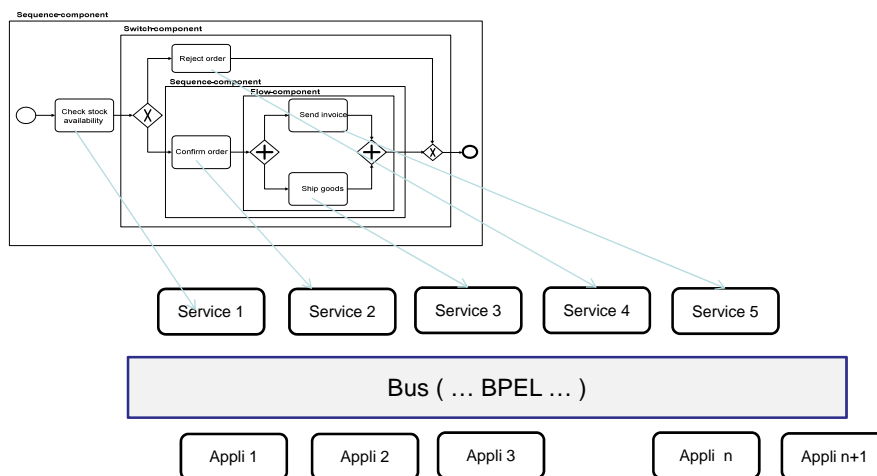
©Les processus métiers: concepts, modèles et systèmes

Exécution dans le Web ...

- Les activités comme des services Web
- La logique en BPEL (Business Process Execution Language)
 - Syntaxiquement un milieu entre BPMN et XPDL
- (la suite dans le chapitre « services Web »)

©Les processus métiers: concepts, modèles et systèmes

Les activités comme des services Web

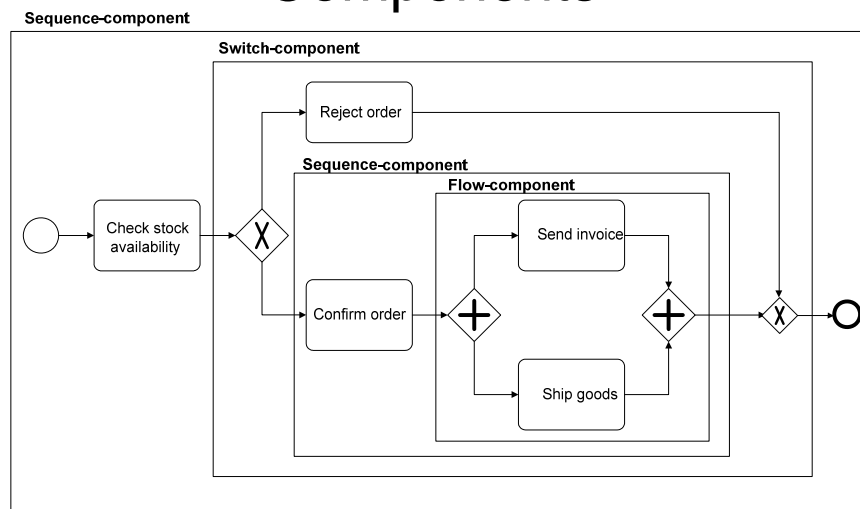


©Les processus métiers: concepts, modèles et systèmes

BPEL in a Nutshell

- **Basic activities:** `<assign>`, `<invoke>`, `<receive>`
- **Block-structured constructs:**
 - `<sequence>`, `<while>`, `<switch>`, `<flow>`
- **Graph-oriented flow construct:** `<link>`
 - Not “graphically” supported by most BPEL editors
 - Considered “less readable” than structured constructs
- **Event-action rules:** `<onEvent>`
 - Considered even less readable than control links!
- *Other constructs not relevant to this talk...*

Example: Only Structured Components



Generated BPEL code

```
<sequence>  
  <invoke “check stock availability” .../>  
  <switch (...) .../>  
    <case “reject”> <invoke “order rejection” .../> </case>  
    <otherwise>  
      <sequence>  
        <invoke “order acceptance” ... />  
        <flow> <invoke “invoicing” ... />  
          <invoke “ship goods” ... />  
        </flow>  
      </sequence> </otherwise> </switch> </sequence>
```