Un nouveau langage de workflow pour les sciences expérimentales

Yuan Lin, Isabelle Mougenot, Thérèse Libourel Rouge

Un nouveau langage de workflow pour les sciences expérimentales

Yuan Lin, Isabelle Mougenot, Thérèse Libourel

LIRMM Laboratory (CNRS - UM2)
161 rue Ada
F-34392 Montpellier Cedex 5
France
Prenom.Nom@lirmm.fr

RÉSUMÉ. *L'article propose un méta-modèle destiné à la construction de workflow scientifique. Le méta-modèle proposé découle d'une analyse de plusieurs autres méta-modèles de workflow existants.*

ABSTRACT. *This paper proposes a meta-model for building a scientific workflow. This meta-model is derived from a study of several existing workflow meta-models.*

MOTS-CLÉS : *Langage workflow, méta-modèle*

KEYWORDS: *Workflow language, meta-model*

1. Introduction

Les applications environnementales connaissent un essor considérable mais elles demandent la mise en place d'infrastructures de mutualisation efficaces car les données impliquées sont souvent onéreuses et complexes à acquérir. Les données existent et sont le plus souvent pérennes, cependant leurs traitements peuvent évoluer au cours du temps. Dans des situations critiques (risques naturels ou anthropiques), les données pérennes doivent être croisées avec des données acquises en temps réel, ce qui revient à exécuter des chaînes de traitements prédéfinies et ceci sur de nouveaux lots de données.

Le concept de workflow initialement créé dans le but d'automatiser les flux de travail organisationnels constitue dans ce contexte un atout incontournable. L'idée d'enchaîner et contrôler les différentes tâches pour réaliser un traitement complexe est pertinente. De plus dans les infrastructures actuelles distribuées, gérant des ressources hétérogènes, bénéficier d'un environnement autorisant définition et exécution de chaîne de traitements constitue une des fonctionnalités essentielles recherchée à la fois par les scientifiques et au-delà le grand public.

La réflexion menée dans notre équipe a abouti à la proposition d'une plateforme de mutualisation et localisation de ressources (données et traitements) guidée par les métadonnées (**?**).

Nous souhaitons donc proposer, dans ce contexte, un environnement dans lequel tout utilisateur pourrait définir des chaînes de traitements, les sauvegarder afin de les exécuter avec des éléments données et traitements localisés à la demande. Nous dénommons cet environnement *WorkFlow scientifique*, car nous souhaitons à terme contrôler l'exécution de l'enchaînement des traitements sur les flots de données correspondants et ceci afin de permettre la validation de résultats expérimentaux.

Le travail présenté est donc dans sa première phase qui consiste à analyser diverses propositions existantes afin de proposer un formalisme adéquat. Les spécificités qui ont retenues notre attention relèvent :

– d'une part du domaine expérimental visé (Biologie, Information Géographique)

– d'autre part de la dimension médiation : les données tout comme les traitements peuvent être distribués pour la communauté visée.

La synthèse des comparaisons menées nous amène à proposer un méta-modèle de workflow scientifique [1], qui sera illustré par un exemple simple. Les perspectives seront abordées en conclusion.

1. particulièrement dédié aux sciences environnementales

2. État de l'art

Le concept de workflow est apparu durant les années 70-80 (**?**). Pour notre part, nous nous sommes intéressés aux propositions présentant :

– Un niveau méta-modèle pour la description et la réalisation de chaîne de traitement. En effet, l'aspect généricité conféré par la méta modélisation est essentiel, à nos yeux.

– La prise en compte de l'aspect expérimental. Les données et traitements scientifiques ont des particularités qui doivent transparaitre au niveau du formalisme.

2.1. *Les propositions relèvant de l'OMG (Object Management Group)*

Les divers propositions restent très généralistes. Elles présentent soit sous forme de méta-modèles, soit sous forme de standard de notation, les éléments constitutifs, nécessaire à la définition du workflow.

2.1.1. *Le méta-modèle UML limité au diagramme d'activité*

UML (**?**) est considéré comme un langage de modélisation universel. Dans le formalisme, les éléments dédiés à la représentation des diagrammes d'activité sont ceux qui sont le plus en adéquation avec la description de processus.

En terme d'élément de description, un processus est représenté par une *activité* (Activity), qui est considérée comme un ensemble de nœuds activités (ActivityNodes), reliés par les arcs (ActivityEdge) comme le montre la figure **??** tirée de (**?**).

Les nœuds activité peuvent être catégorisés : *objet* et *contrôle*. Une *action* peut accéder aux *objets* par les liens objets, et les liens *contrôles* sont utilisés pour décrire des exécutions parallèle, optionnelle, etc. cf. figure **??** tirée de (**?**).

En fait, le diagramme d'activité décrit un processus qui doit être exécuté, mais la notion de rôle potentiellement apte à réaliser cette exécution n'est pas présente.

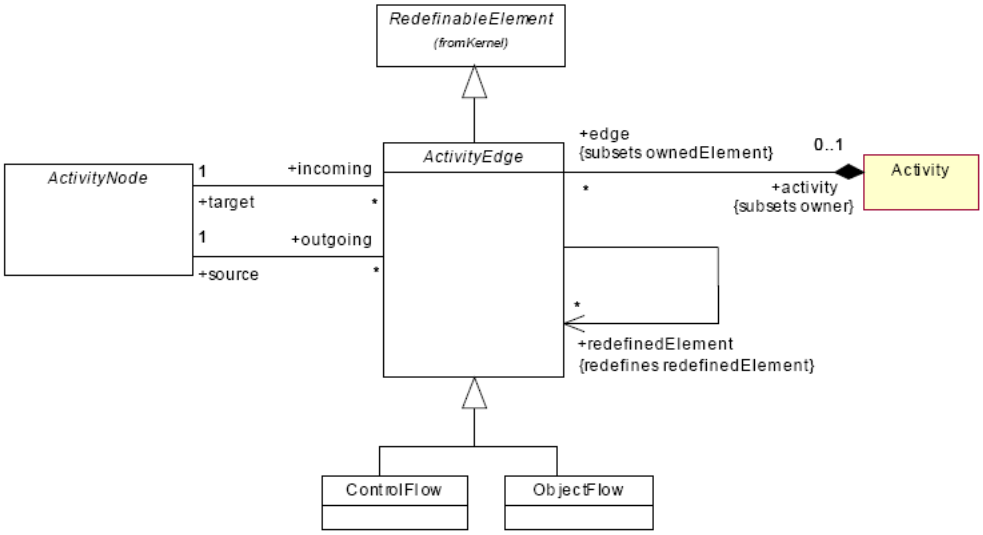

Figure 1. *Méta-modèle UML, diagramme d'activité : flows*

2.1.2. *SPEM*

SPEM (Software & Systems Process Engineering Metamodel) (**?**) est, selon les points de vue, un méta-modèle conforme au MOF ou un profil UML, qui se focalise sur la notion de développement de projet (cf. figures **??** et **??** tirées de (**?**)).

Le processus de développement recouvre un ensemble d'*activités*. Pour chaque activité, les *ProcessPerformers* sont définis pour représenter les humains ou les machines qui s'occupent de cette activité, corrélativement, les *rôles* sont définis pour bien préciser les responsabilités et les capacités requises par cette activité. L'enchaînement des activités découle des *WorkProducts* nécessaires en amont pour cette activité et des *WorkProducts* générés par celle-ci (cf. figures **??** et **??**).

2.1.3. *BPMN*

BPMN (Business Process Model Notation) est une norme de notation pour la modélisation de processus, édifiée dans le cadre de BPMI (Business Process Management Initiative). Son objectif est de fournir un cadre graphique permettant de décrire un processus d'une manière commune à tous les utilisateures et ce, indépendamment de l'outil utilisé.

Pour modéliser un processus, trois éléments principaux sont utilisés : *Tâche*, qui représente une action ; *Branchement*, qui représente la condition de routage entre les flux en entrées et les flux en sorties. *Évènement*, qui représente un état particulier dans le processus (début, intermédiaire, fin).

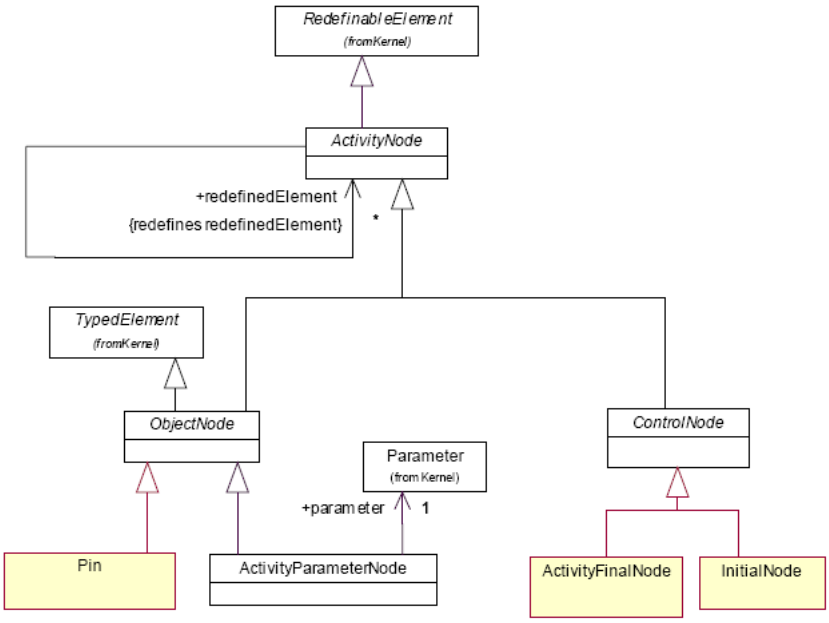

Figure 2. *Méta-modèle UML, diagramme d'activité : Nodes*

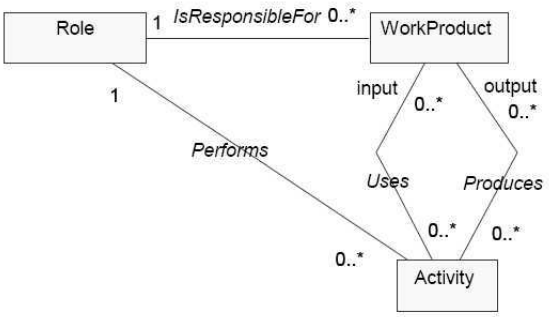

Figure 3. *SPEM 1.1 : Les concepts du méta-modèle*

Hormis les trois éléments essentiels, les *Artifacts*, comme les données, les annotations, etc, sont reliées au modèle pour porter plus d'informations. De plus un élément organisationnel *swimlane* est proposé dans le but de regrouper les différents activités selon leur fonctionnalité.

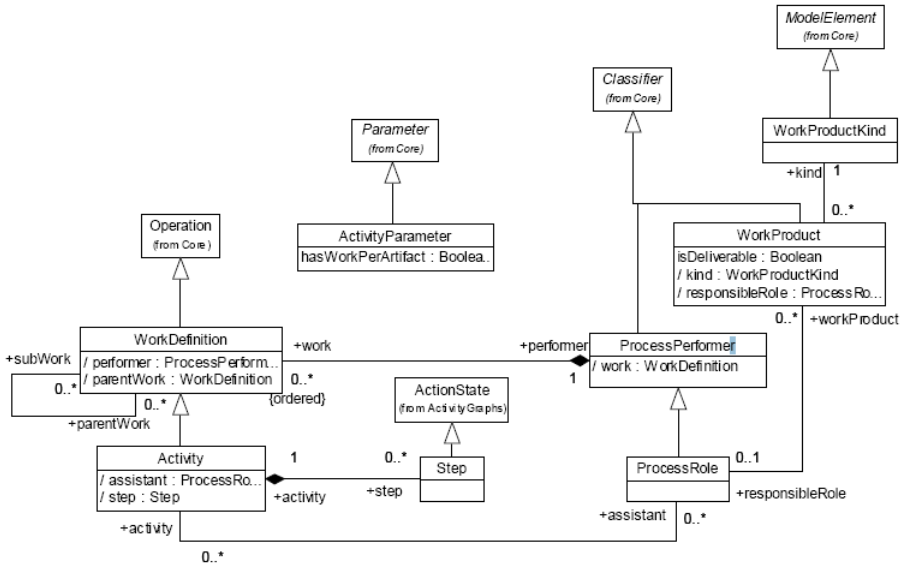

Figure 4. *SPEM 1.1 : Process Structure package*

2.2. *Les propositions dédiées aux workflow scientifiques*

Les travaux retenus l'ont été pour leurs caractères spécifiques liés aux domaines cibles.

2.2.1. *KEPLER*

KEPLER est un système particulier de workflow scientifique guidé par l'ontologie du domaine concerné (**?**). Il fournit un environnement de développement dédié à la conception d'ontologie avec une partie navigateur / visualiseur sur celle-ci. En ce qui concerne les traitements, KEPLER adopte une métaphore organisation humaine.

Un ou plusieurs responsables *Director* planifient les tâches pour les acteurs de l'organisation et ceci en s'appuyant sur l'ontologie. Le workflow est représenté par un graphe liant les acteurs et les connexions entre acteurs via des ports.

Le plan d'exécution d'une chaîne de traitement est donc créé par un *Director* du système, (plusieurs sortes de Director de spécialité différente peuvent être utilisées).

Une des originalités complémentaire de KEPLER réside dans le fait que l'exécution du plan est déclenchée par l'intermédiaire d'objet *Receiver* déposé dans un des ports du graphe.

2.2.2. *Méta-modèle WDO-It !*

WDO-IT ! est un outil de calcul scientifique qui repose sur les concepts classiques de données et d'actions (**?**). Comme KEPLER, il se veut aussi un outil d'aide à la représentation des connaissances (WDOs workflow-driven ontologies).

La racine du modèle *WDOConcept* est une spécialisation du concept OWL :Thing, et deux sous-modèles *WFSequenceElement* et *Data* sont construits pour représenter les composantes de workflow et les données reliées (cf. figure **??** tirée de (**?**)).

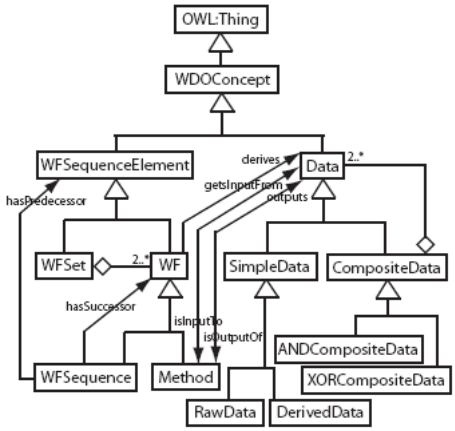

Figure 5. *Méta-modèle WDO-It !*

Le plan d'ordonnancement du workflow est décrit :

– pour les actions séquentielles par *WFSequence*

– pour les actions parallèles par *WFSet* ainsi que par des connecteurs *AND* et *XOR* sur les flux de données.

La construction de la chaîne de traitements est déterminée, en chainage arrière, à partir du résultat ou but désiré en s'appuyant sur l'ontologie de traitements possibles.

2.2.3. *CIMFlow*

CIMFlow est aussi un WFMS (Workflow Management Systems) (**?**). Le modèle sous-jacent est aussi classique, le workflow étant un graphe reliant des nœuds activités par des arcs. Les activités décrites sont spécialisées selon leur nature (manuelle, automatisée simple ou complexe), les arcs sont aussi spécialisés en arc flux de données et arc de contrôle. L'originalité réside dans la vision systémique des activités qui intervient par la prise en compte des divers modèles proposés par CIMFlow : *Processus Modèle*, *Organization Modèle*, *Ressource Modèle* et les *données* reliées à workflow (cf. figure **??**). Une activité comporte donc une description qui détaille ses caracté-

ristiques, ses entrées, sorties, et précise quelle partie de l'organisation est concernée, quelle ressource est impliquée (cf. figure **??**).

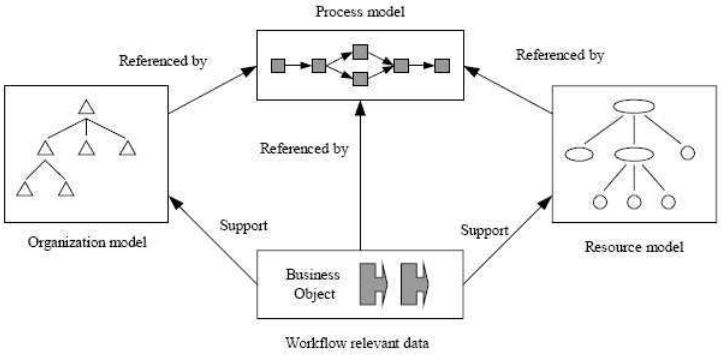

Figure 6. *Le Framework de CIMFlow*

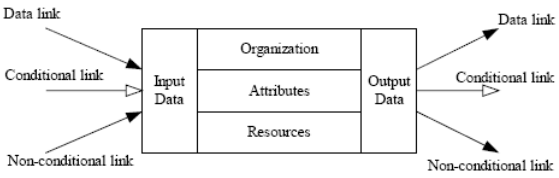

Figure 7. *La structure d'une activité dans CIMFlow*

2.3. *Synthèse*

Nous avons analysé les diverses propositions selon des critères relatifs d'une part à l'expressivité des méta-modèles dans la représentation des aspects statiques, d'autre part dans la manière dont les aspects dynamiques sont envisagés. Nous présentons les résultats sous la forme d'un tableau récapitulatif.

Nom Modèle	Aspect Statique		Aspect dynamique
	Concept	Ordonnancement	
Diagramme d'activité (UML)	Noeud, Arc, etc.	Les activités sont reliées par les arcs de contrôle.	Transparait dans les diagrammes de cas d'utilisation et de séquence associés.
SPEM	Rôle, WorkProduit, Activité, etc.	L'ordre est représenté par les étapes d'une activité.	? ?
BPMN	Tâche, Évènement, Branchement, Arc de connexion, etc.	Les éléments de ce méta-modèle sont reliés par les différents arcs de connexion.	BPEL peut être un candidat.
Kepler	Directeur, Acteur, Port, Channel, etc.	Les acteurs sont reliés par les channels (avec ports à deux côtés).	Les directeurs organisent les rôles des acteurs pour planifier un processus. Les différents directeurs pré-définis s'occupent des différents genres d'exécution.
WDO-It !	Donnée, (WF)élément, etc.	WFSéquence	On peut construire dynamiquement un processus à partir des résultats escomptés et de l'ontologie.
CIMFlow	Noeud, Arc, etc.	Les noeuds sont reliés par les arcs.	? ?

Figure 8. *Tableau récapitulatif*

Nom Modèle	Aspect Statique			Aspect dynamique
	Concept	Ordonnancement	Standard	
Diagramme d'activité (UML)	Noeud, Arc, etc.	Les activités sont reliées par les arcs de contrôle.	UML	Transparait dans les d grammes de cas d'ut sation et de séquence sociés.
SPEM	Rôle, WorkProduit, Activité, etc.	L'ordre est représenté par les étapes d'une activité.	UML	? ?
BPMN	Tâche, Évènement, Branchement, Arc de connexion, etc.	Les éléments de ce méta-modèle sont reliés par les différents arcs de connexion.	? ?	BPEL peut être un can dat.
Kepler	Directeur, Acteur, Port, Channel, etc.	Les acteurs sont reliés par les channels (avec ports à deux côtés).	? ?	Les directeurs organis les rôles des acteurs p planifier un process Les différents directe pré-définis s'occup des différents gen d'exécution.
WDO-It !	Donnée, (WF)élément, etc.	WFSéquence	MDO, MBW	On peut construire dy miquement un proces à partir des résultats comptés et de l'onto gie.
CIMFlow	Noeud, Arc, etc.	Les noeuds sont reliés par les arcs.	? ?	? ?

3. Notre proposition

Notre objectif, à long terme, est, nous l'avions précisé en introduction, d'offrir un environnement complet de description de chaînes de traitement puis d'exécution des celles-ci.

A partir de la synthèse précédente, nous proposons, dans un premier temps, un langage de description de workflow, ce langage est défini par un méta-modèle qui est naturellement inspiré des méta-modèles existants analysés précédemment.

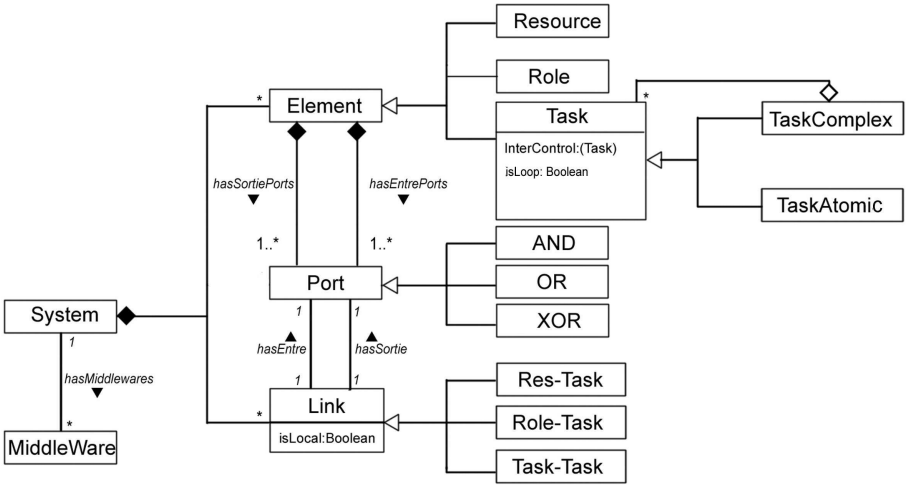

Figure 9. *Notre méta-modèle*

3.1. *Explication du modèle*

Le méta-modèle se place selon le point de vue environnement logiciel de workflow. Il est donc vu, au niveau le plus abstrait, comme une composition d'éléments et de liens entre éléments. La connexion entre élément et liens est assurée par le concept de port.

Les éléments se diversifient en :

– *Tâches* (task) pré-définies à utiliser ou réutiliser,

– *Rôles* existants (qui interviendront durant la phase d'exécution),

– *Ressources* disponibles à mobiliser.

La notion de tâche (task) correspond aux notions d'Activité, Processus (etc.) généralement invoquées dans les autres méta-modèles de workflow. Nous déclinons cette notion avec le patron composite : une tâche peut être complexe ou atomique, avec la possibilité de réutiliser une tâche complexe agrégée comme tâche atomique.

Les éléments sont reliés par des *liens* unidirectionnels. Nous distingons :

– Les liens entre les tâches qui nous permettront de représenter l'ordonnancement des tâches dans un processus.

– Les liens entre tâche et rôle. Ils permettent de préciser quel rôle peut intervenir sur la tâche.

– Les liens entre tâche et ressource précisent si la ressource est utilisée ou produite par la tâche.

Remarque : il n'y a pas de lien direct entre rôle et ressource. Dans la plupart des cas, les liens entre rôle et ressource sont déductibles du lien rôle-tâche et du lien tâche-ressource.

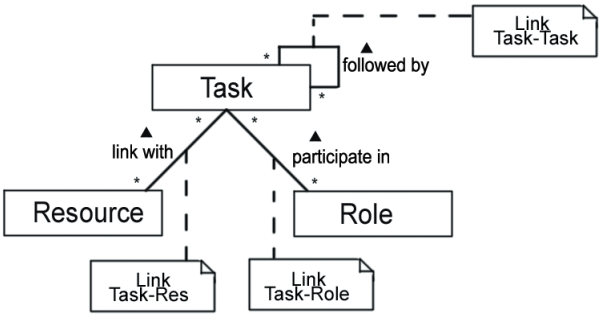

Figure 10. *Un modèle exprimant la sémantique des liens*

Les liens relient les éléments par l'intermédiaire des ports qui leur sont rattachés.

Chaque élément a des ports d'entrées / sorties (le type E/S est lié au sens du lien correspondant). Les ports entrées / sorties peuvent être spécialisés en ports XOR, OR et AND qui préciseront le type d'exécution nécessaire (parallèle, optionnelle, etc.).

Dans un environnement distribué, l'exécution de processus fera appel à des ressources et tâches dispersées sur des plate-formes différentes. La notion de *Middleware* présente dans le méta-modèle est dédié à cet aspect.

3.2. *Les spécificités*

3.2.1. *Points de vue sur les éléments*

Tout élément peut être interpreté selon deux points de vue : boîte noire et boîte blanche.

1) Point de vue boîte noire : L'utilisateur choisit un élément défini initialement en ignorant sa réalisation, mais après avoir choisi sa fonctionnalité et ses paramètres (Ex : une tâche est choisie pour son nom et ses entrées / sorties). Il compose les boîtes noires par l'intermédiaire de leurs ports et des liens de composition.

Le point de vue boîte noire peut être étendu. Il autorise ainsi l'encapsulation d'une chaîne de traitements complexe réifiée en tant que tâche simple boîte noire.

2) Point de vue boîte blanche : La description des éléments est plus fine et détaille la réalisation de l'élément.

Une chaîne de traitements peut être considérée comme la boîte blanche d'une tâche boîte noire encapsulée.

3.2.2. *Les boucles*

La boucle joue un rôle indispensable dans la programmation, une boucle peut aussi être considérée comme l'exécution multiple d'une même tâche. Dans notre modèle, il n'y a pas d'élément spécifique pour représenter cette notion, mais par contre, une propriété "isBoucle" est ajoutée sur l'élément *Task*. Si cette propriété est égale à "Vrai", la tâche va être exécutée plusieur fois, sinon, elle n'est exécutée qu'une fois.

La tâche boucle par rapport à une tâche normale autorise en entrée / sortie des collections de données. Donc c'est la taille de collection qui détermine le nombre d'itérations.

4. Exemple illustratif

4.1. *Description du exemple*

Pour mieux comprendre le contexte, on réalise l'analyse simplifiée d'un exemple réel. En cas de risque naturel comme la rupture d'une digue sur la commune de Mauguio, on souhaite reconnaître les bâtiments vulnérables sur une carte de la zone.

Le scientifique, connait :

– les données dont il dispose et qui vont constituer les entrées de sa chaîne de traitements et connait le type d'information qu'il souhaite en terme de résultat.

1) Entrée : Une couche de données relatives au bâti de la zone concernée.

2) Entrée : Une couche de données relatives aux digues (linéaire) de la zone.

3) Résultat : une carte où figurent les bâtiments vulnérables dans la zone d'inondation suite à une rupture de digue (celle-ci sera indiquée par un expert sur le terrain).

– les méthodes et traitements appropriés :

1) Superposer des couches de données . Cette méthode prend un ensemble de lots de données en entrée (en vérifiant qu'ils obéissent à des contraintes de géolocalisation et codage). Comme sortie, elle rend un ensemble de données cohérent résultat de l'intégration de toutes les données initiales.

2) Positionner les coordonnées sur une couche. Diverses techniques peuvent être utilisées relevant du géocodage.

3) Construire une zone tampon (buffer) à partir d'une géolocalisation.

4) Illustrer une couche de données. Cette méthode ajoute une légende circsconstanciée à la couche de données sous-jacente.

4.2. *Illustration par workflow*

Au niveau modèle, les éléments instanciés conformes au méta-modèle que nous proposons seront représenter par la symbolique décrite figure **??**.

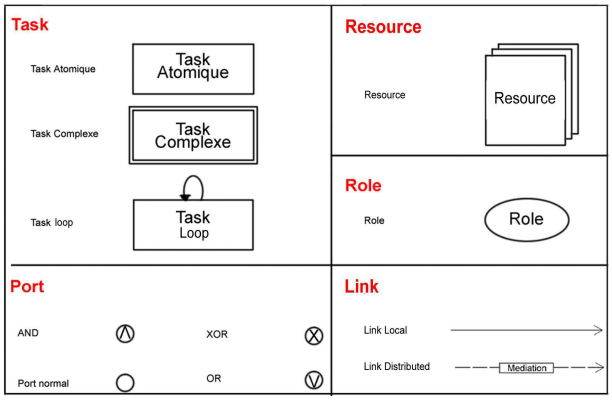

Figure 11. *Langage symbolique représentant les éléments de modèle*

L'analyse précédente aboutit á la représentation de la figure **??**.

Remarque : Le trait pointillé sur la figure **??** peut servir à encapsuler la chaîne de traitements sous-jacente pour définir une tâche réutilisable boîte noire : *afficher vulnérabilité*.

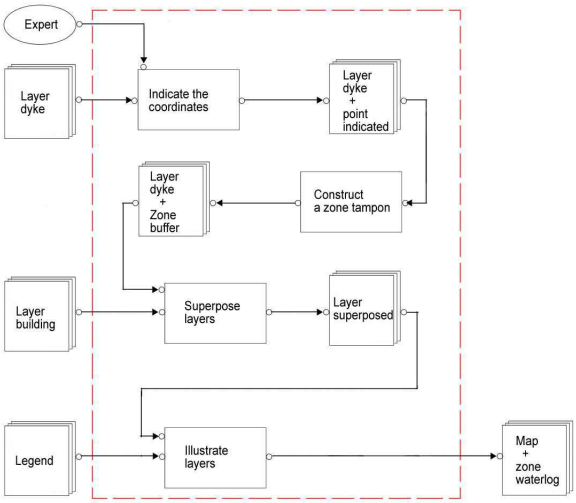

Figure 12. *Exemple illustré*

5. Conclusion

Le méta-modèle dont nous avons présenté l'ébauche a été réalisé après l'étude d'un panorama de travaux existants. Cependant, notre analyse a passé sous silence les travaux relatifs aux services web et à la chorégraphie de ses services. Nous avons aussi délaissé les langages de composants et d'assemblage de composants. Dans un état de l'art plus complet, nous souhaitons les intégrer afin d'améliorer la réflexion synthétique.

Les utilisateurs visés doivent disposer d'un langage simple et manipuler des concepts d'appropriation facile, c'est ce qui nous a amené á choisir une symbolique relativement simple.

Les perspectives que nous pouvons rapidement dégager sont :

– D'une part, à court terme, le besoin de compléter le méta-modèle et de le confronter à divers exemples afin d'assurer sa pertinence ;

– D'autre part, à plus long terme, développer l'aspect dynamique (exécution). au-delà du méta-modèle descriptif.

6. Bibliographie

Barde J., Libourel T., Maurel P., « A Metadata Service for Integrated Management of Knowledges Related to Coastal Areas », *Multimedia Tools Appl.*, vol. 25, n° 3, p. 419-429, 2005.

da Silva P. P., Salayandia L., Q.Gates A., « WDO-It !A Tool for Building Scientific Workflows from Ontologies », n.d.

Khoshafian S., Buckiewicz M., « Groupware & workflow », 1998.

Ludäscher B., Altintas I., Berkley C., Higgins D., Jaeger E., Jones M., Lee E. A., Tao J., Zhao Y., « Scientific workflow management and the Kepler system », *Concurrency and Computation : Practice and Experience*, vol. 18, n° 10, p. 1039-1065, 2006.

OMG, « Software Process Engineering Metamodel Specification version 1.1 », January 2005.

OMG, « UML 2.0 Superstructure Specification », n.d.

Zhang Z., Fan Y., « Implementation of WPDL Conforming Workflow Model », 2002.