
Une Ontologie de Protocoles pour la Coordination de Systèmes Distribués

Wassim Bouaziz

Laboratoire IRIT

UT1, 2 rue du Doyen Gabriel Marty, 31042 Toulouse Cédex, France

Wassim.Bouaziz@univ-tlse1.fr

RÉSUMÉ. Ce papier propose une ontologie de protocoles de haut niveau pour coordonner des systèmes distribués. Ces protocoles sont issus des Systèmes Multi-Agents (SMA) et permettent de structurer et supporter des interactions riches: recherche et négociation de services, sous-traitance de services. Ces protocoles peuvent être utilisés pour la coordination d'applications relevant de domaines aussi variés que le Workflow Inter-Organisationnel, les conversations entre web-services, les e-institutions, etc. Nous définissons dans un premier temps les concepts communs à ces protocoles, leurs liens et les axiomes exprimant des contraintes portant sur ces concepts. Puis nous expliquons comment cette ontologie peut être intégrée et utilisée au sein d'un Système de Gestion de Protocoles jouant le rôle de médiateur entre un ensemble de systèmes distribués et coopératifs. L'utilisation d'une telle ontologie a plusieurs avantages : donner de la sémantique à ces protocoles, les spécifier comme des objets à part entière, partageables, et plus facilement maintenables. Cette ontologie constitue un premier pas vers l'automatisation de la coordination dans les systèmes distribués et coopératifs.

ABSTRACT This paper proposes an ontology of high-level protocols in order to coordinate distributed systems. These protocols result from Multi-Agent System and enable rich interactions (negotiation, auctions, subcontracting protocols) to be structured and supported. The representation of protocols through an ontology enables the coordination of applications as various as Inter-Organisationnel Workflow, conversations between web-services, E-institutes, and, so on, to be supported. The common concepts to these protocols, their links and the axioms expressing their constraints define this ontology. Then, we also explain how this ontology can be integrated and used in a Protocol Management System which plays the role of mediator between a set of distributed and cooperative systems. This ontology has several advantages : to provide semantics to these protocols, to specify them as independent object, visible, public, and more easily maintainable. This ontology constitutes a first step towards the automation of coordination in distributed system.

MOTS-CLÉS : Systèmes Distribués, Protocole, Ontologie, Partage de connaissances.

KEYWORDS: Distributed Systems, Protocol, Ontology, knowledge representation.

1. Introduction

Dans les systèmes distribués, l'interaction est un mécanisme important pour supporter le partage des ressources et coordonner les activités entre les différents composants de ces systèmes. Le concept de protocole d'interaction est un moyen efficace pour structurer et organiser les échanges entre ces composants. Au delà des protocoles de bas niveau qui assurent la communication, l'intérêt se porte aujourd'hui sur des protocoles de haut niveau, issus des Systèmes Multi-agents (SMA), ne se limitant pas à des échanges de type client-serveur mais exprimant des interactions complexes de type négociation, enchère, délégation et sous-traitance. Par ailleurs, dans ce cadre des SMA, les protocoles se basent sur des langages à base de performatifs du type FIPA-ACL (propose, call for proposol, inform, subscribe, ...) qui enrichissent les échanges (Ferber, 1995).

Comme exemple d'applications dans lesquelles l'utilisation de protocoles est nécessaire on peut citer le Workflow Inter-Organisationnel (WIO) (Andonoff *et al.*, 2007), le commerce électronique ou la conversation entre services Web (Paurobally et Jennings, 2005). Détaillons le premier exemple d'application. Le WIO permet de gérer et d'automatiser des processus qui traversent les organisations. Dans ce contexte, par nature distribué et hétérogène, plusieurs organisations partenaires mettent en commun leurs Workflow et les font interagir selon certains *protocoles* pour rendre un nouveau service constituant une valeur ajoutée pour chaque partenaire. Ces protocoles correspondent à des protocoles de recherche de partenaires (Matchmaker) capables de rechercher un service workflow, à des protocoles de négociation (par exemple Heuristique, Argumentation, ...) autour de la qualité de services offerts par un workflow ou encore à des protocoles d'élaboration et d'exécution de contrat.

Dans ce type d'applications, les protocoles sont donc identifiables et récurrents. Il est alors utile de les isoler afin de bien les étudier, les modéliser et les implémenter comme des entités à part entière pour permettre ainsi leur partage, leur recherche et leur invocation. Pour mener à bien ce travail, il faut dans un premier temps définir une représentation des protocoles qui concourent à cet objectif.

Notre travail repose naturellement sur l'exploitation du Web comme technologie supportant la distribution, et sur *le Web Sémantique* (Charlet *et al.*, 2003) en particulier comme moyen de produire des représentations sémantiquement riches, partageables et accessibles. Plus précisément les ontologies, constituent une solution complète pour décrire d'une façon consensuelle les connaissances d'un domaine d'application, pour partager ces connaissances (Struder *et al.*, 1997) et pour résoudre des questions d'interopérabilité sémantique. Les ontologies vise à décrire de façon consensuelle l'ensemble des informations pertinentes d'un domaine d'application.

De nombreux travaux actuels se sont intéressés aux protocoles (Benatallah *et al.*, 2006), (Hanachi et Sibertin-Blanc, 2004), (Paurobally et Jennings, 2005), (Amit *et al.*, 2004), (Mazouzi *et al.*, 2002) et (Tamma *et al.*, 2002) mais très peu d'entre eux ont envisagé la définition *d'une ontologie pour la représentation de ces protocoles*. Deux travaux méritent d'être mis en avant : (Tamma *et al.*, 2002) et (Amit *et al.*,

2004). (Tamma *et al.*,2002) proposent une ontologie limitée à une classe de protocoles (les protocoles de négociation) qui est spécifique au commerce électronique. (Amit *et al.*, 2004) proposent une ontologie de protocoles et un langage OWL-P pour les spécifier. Cependant, cette ontologie contraint le concepteur à décrire les protocoles nécessairement avec le concept d'engagement. Par ailleurs, OWL-P ne permet pas de décrire les structures de contrôle complexes (parallélisme, séquençement, etc.) des protocoles.

Notre travail comporte trois contributions:

- La définition d'une ontologie de protocoles permettant des interactions complexes, indépendantes du domaine d'application, et intégrant aussi bien les aspects statiques des protocoles que leur structure de contrôle (comportement).

- L'intégration de cette ontologie dans le système de gestion de protocoles. Ce système joue le rôle de médiateur pour coordonner des systèmes distribués en utilisant une pile de protocoles.

- Le mode d'exploitation de cette ontologie à la fois pour la conception et l'exécution d'un protocole.

Les avantages de notre approche sont :

- la réutilisation, la visibilité et la simplification de la conception puisque les protocoles de coordination sont vus comme des entités à part entière;

- le partage et l'accessibilité des protocoles de coordination dû à une représentation sous forme d'ontologie;

- la définition d'une sémantique pour ces protocoles de coordination;

- la possibilité donnée aux participants impliqués dans une conversation régie par un protocole, de raisonner ou de négocier sur les termes mêmes de ce protocole.

- la définition d'un modèle formel manipulable par une machine.

Ce papier est organisé comme suit. La section 2 présente quelques travaux sur l'utilisation des ontologies pour la représentation des protocoles d'interaction. Dans la section 3, nous détaillons l'ensemble des concepts et liens de cette ontologie de protocoles. La section 4 donne l'opérationnalisation de cette ontologie. Enfin, nous montrons comment utiliser l'ontologie pour la représentation des protocoles de coordination avant de conclure par un rapide aperçu des perspectives de ce travail.

2. Principaux travaux autour des ontologies de protocoles d'interaction

Nous présentons dans cette section les principaux travaux sur les ontologies pour les protocoles d'interaction. L'objectif est de positionner notre travail et de mettre en valeur nos contributions.

(Tamma *et al.*, 2002) s'intéressent à l'automatisation de la négociation entre agents interagissant dans le cadre du commerce électronique. Ils proposent de décrire les protocoles de négociation à l'aide d'une ontologie partagée par les agents. Cette ontologie est une description explicite et déclarative de ces protocoles. Elle reprend la taxonomie des règles de négociation proposées par Bartolini *et al.* (Bartolini *et al.*, 2002). Elle a l'avantage d'intégrer la description du domaine de

négociation (partie statique) et celle du processus de négociation (partie dynamique ou structure de contrôle). Le processus y est décrit à travers le langage PSL (Gruninger, 2003). Leur travail a cependant l'inconvénient d'être limité aux protocoles de négociation, alors que nous essayons ici d'être plus large et de concevoir une ontologie plus générale intégrant un ensemble de protocoles de coordination.

L'idée sous jacente à notre approche est de représenter le comportement d'un protocole à la manière de la spécification du comportement des processus d'entreprises. La différence est que ce comportement prend en compte aussi les messages échangés au même niveau que la représentation de structures de contrôle complexes du protocole.

Desai et al. (Desai *et al.*, 2005) proposent une ontologie générique pour la description des protocoles pour les processus d'entreprise. Dans leur description, ils utilisent des règles décrites dans le langage SWRL (Horrocks *et al.*, 2004). Leur ontologie regroupe des concepts importants tel que les messages, les rôles, les règles, et la notion d'engagement (commitment). Cette ontologie permet aussi la composition de protocoles. L'inconvénient de cette proposition est la faiblesse du pouvoir d'expression des protocoles en terme de structure de contrôle car elle ne permet pas de spécifier des protocoles complexes (notamment concurrents, itératifs, ... (Hanachi et al., 2004)) mais seulement des échanges de messages assez simples.

Toivonen et Helin (Toivonen et Helin, 2003) proposent une représentation en DAML des protocoles d'interaction. Cette ontologie partagée est téléchargeable par les agents qui doivent l'intégrer dans leur code pour pouvoir jouer un rôle dans le protocole. Comme (Tamma *et al.*, 2005) cette ontologie s'intéresse à la représentation de processus d'interaction. Alors que notre approche considère les protocoles comme des entités logicielles à part entière et indépendantes avec qui les participants interagissent, dans (Toivonen et Helin, 2003) les agents doivent encapsuler la description des protocoles et modifier leur structure interne pour jouer le protocole.

En résumé, aucun des travaux présentés ne fournit une description générique des protocoles de coordination. De plus ces travaux ne permettent pas de représenter des comportements de protocoles complexes. Par ailleurs, aucun de ces travaux ne propose une classification des protocoles et des solutions pour la sélection dynamique de protocoles.

3. Une Ontologie pour la description des protocoles de coordination

3.1. Spécification de l'ontologie de protocoles : concepts et terminologie

3.1.1. Notion d'ontologie

La définition la plus commune présente une ontologie comme étant « *une spécification explicite et formelle d'une conceptualisation partagée* » (Gruber,

1993). En d'autres termes, une ontologie est une description formelle d'un domaine du monde réel. C'est une conceptualisation dans le sens où elle fournit un vocabulaire formalisé de concepts, de leurs relations, et des axiomes qui leur sont associés. L'ontologie fournit ainsi une base de compréhension commune à plusieurs communautés ou systèmes. Dans ce qui suit, la notion d'ontologie sera adaptée au cadre de la conception des protocoles de coordination dans les systèmes distribués.

3.1.2. Ontologie des protocoles de coordination

L'ontologie que nous proposons est considérée comme un modèle de référence pour la description des protocoles de coordination. Elle fournit ainsi une représentation déclarative et explicite de ces protocoles.

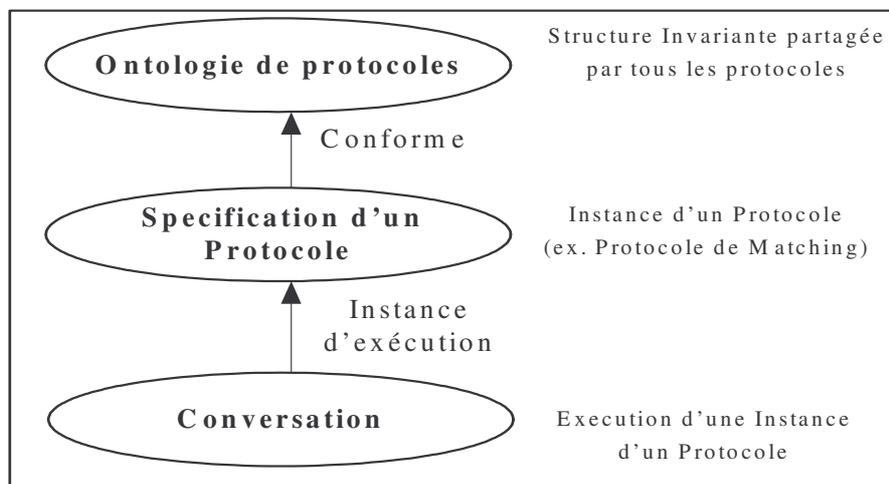


Figure 1. Trois niveaux de description de protocoles

La figure 1 illustre les trois niveaux de description d'un protocole. Le premier niveau, le plus abstrait, correspond à l'ontologie des protocoles définissant la structure invariante partagée par tous les protocoles. Le deuxième niveau représente la spécification d'un protocole par l'instanciation de cette ontologie (par exemple le protocole de Matchmaking). Le dernier niveau, la conversation, le plus concret, correspond à l'exécution ou l'occurrence d'un protocole (par exemple la recherche par un client A via un matchmaker M d'un partenaire B capable de rendre un service donné).

La construction d'une ontologie formelle se révèle être un travail délicat indépendamment du domaine d'application. Le recensement des constituants de l'ontologie (les concepts, les relations et leur significations) est généralement difficile à résoudre la représentation formelle de ces connaissances à l'aide des langages d'échange. Pour cette raison, il est habituel de procéder par étapes, en essayant de répondre à ces questions : (i) comment acquérir les connaissances (par

représente la fonction ou l'objectif de chaque protocole (négocier, rechercher, contracter, ...).

Un protocole se compose d'un ensemble d'éléments qui le définissent : les règles, les rôles, les activités, les contraintes et les messages. Détaillons chacun de ces éléments :

– *Règle* : Ce concept définit les normes imposées par le protocole lors d'une conversation. Ces normes réglementent le comportement des participants qui jouent des rôles dans un protocole. Par exemple, dans un protocole d'enchère un participant ne peut pas proposer un prix inférieur au prix courant ou bien un participant ne peut pas effectuer une proposition une fois le temps d'attente écoulé.

– *Rôle* : c'est le comportement que peut prendre un participant jouant ce rôle. Par exemple, le protocole de Matchmaking comporte deux rôles : un demandeur et un fournisseur de services. La définition des rôles inclut la spécification d'un ensemble de contraintes « de casting » permettant de préciser quel participant a le droit de jouer tel rôle.

– *Message* : Il correspond à l'interaction élémentaire permise par un protocole et que les participants d'une conversation peuvent échanger. Un message comporte un certain nombre de paramètres qui sont définis à travers des propriétés telles que expéditeur, receveur et contenu. Le message doit posséder une sémantique pour être interprété sans ambiguïté par tous les participants. Cette sémantique peut être décrite soit à travers une logique de description soit en utilisant des langages standards tel que le langage FIPA-ACL ou bien OWL pour la description du contenu du message.

– *Type de Message* : Ce concept catégorise l'ensemble des messages contenu dans un protocole (information, proposition, promesse, ...). La description de la sémantique de l'action peut être décrite en logique de description. Avant l'envoi d'un message, un certain nombre de contraintes doivent être vérifiées. Ces contraintes limitent le comportement des participants au sein d'un protocole.

– *Activité d'interaction* : Ce concept décrit les opérations effectuées par un protocole. Une activité comprend des messages et des conditions pour son déclenchement et permet l'exécution d'un ensemble d'actions. L'action représente l'opération déclenchée par le protocole suite à la vérification des conditions. Les structures de contrôle permettent la composition des activités d'interaction qui structurent le comportement de protocole. Un exemple d'action au niveau du protocole de recherche (Matchmaking) est l'exécution de la fonction de Matching.

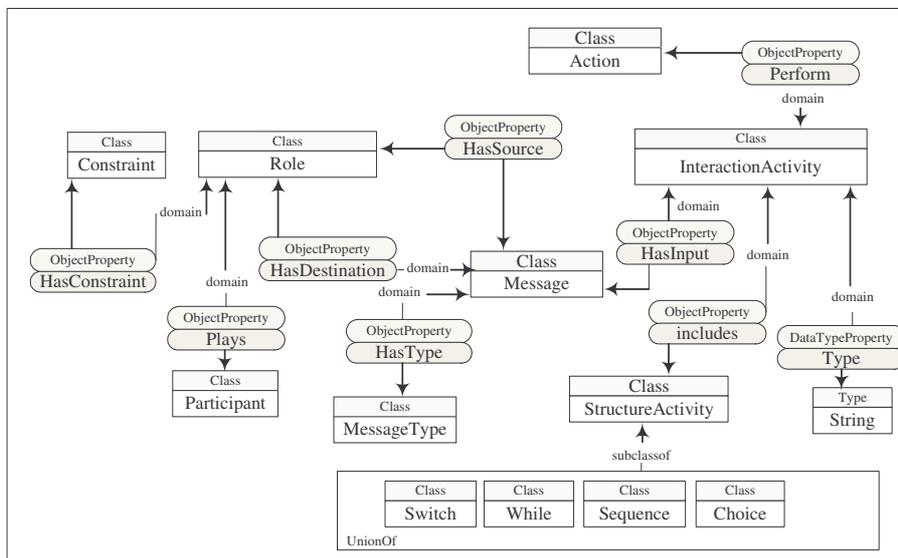


Figure 3. *Ontologie de protocoles : aspect dynamique*

La figure 3 illustre la combinaison des concepts d'activité d'interaction et de message. Ces activités définissent les séquences d'interaction permises par un protocole. La définition du comportement du protocole inclut à la fois les activités ou les opérations et le séquençage des messages ainsi que les rôles impliqués. Chaque interactions dans un protocole comprend des participants qui jouent des rôles dans le protocole.

3.2. Représentation formelle de l'ontologie

On distingue en représentation des connaissances deux types de langages: les langages d'échange et les langages d'opérationnalisation. Les langages du web proposés par le W3C sont des langages d'échange (RDF, OWL, ...). Ils servent à représenter formellement des connaissances et à les échanger. Les langages d'opérationnalisation eux disposent de services inférentiels capables de raisonner sur les connaissances représentées.

La description des ontologies dans des langages d'échange comme OWL fournit des éléments pour améliorer la communication entre les personnes, entre les personnes et les systèmes, et entre les systèmes. L'intégration d'une ontologie à un système permet donc de déclarer formellement un certain nombre de connaissances. utilisées pour caractériser les informations gérées par le système et de se baser sur ces caractérisations et la formalisation de leur signification pour automatiser des

tâches de traitement de l'information (Gandon. 2006). Nous avons choisi le langage OWL (W3C¹, 2004a) pour cette finalité.

OWL est un langage de représentation des connaissances d'un domaine défini pour être utilisé dans le cadre du Web sémantique. Il permet de définir des concepts et des relations à partir d'un ensemble de connecteurs spécifiques au langage et dispose d'une syntaxe XML. Ce dernier repose sur la logique des descriptions pour décrire les hiérarchies, les axiomes et les contraintes. OWL permet de représenter les classes (owl : class), les liens sémantiques (owl: objectProperty), les types de données (owl : dataTypeProperty) et aussi les cardinalités. Il supporte aussi les hiérarchies de spécialisation, l'équivalence entre les concepts (owl : equivalentClass), l'intersection et l'union entre classes ou concepts (owl : unionof, owl : intersectionof).

Pour cette représentation formelle de l'ontologie, nous avons utilisé le « plugin » OWL de Protégé-2000.

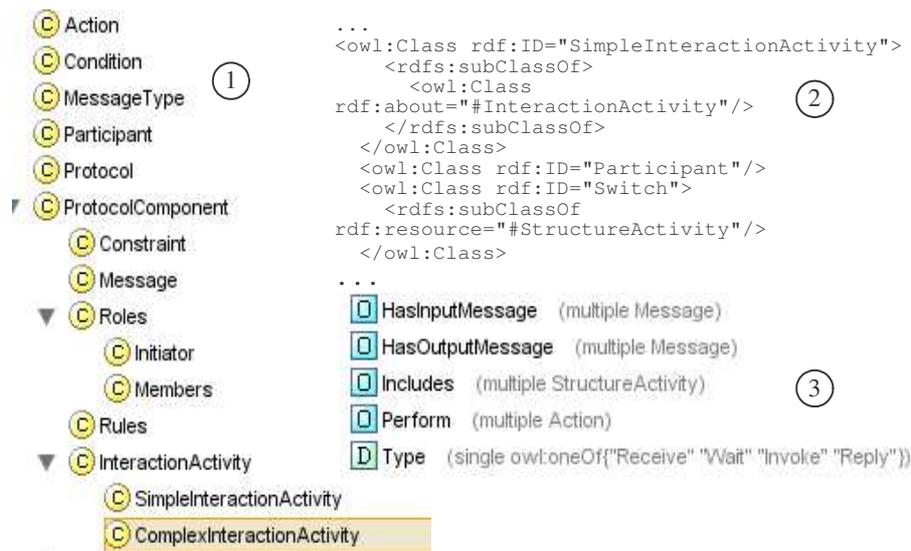


Figure 4. Hiérarchie de classes de l'ontologie sous protégé-2000

La figure 4 montre une partie de la hiérarchie de classes de l'ontologie obtenue à l'aide de l'outil protégé-2000 (1). Elle présente également une sous partie en OWL du concept d'activité d'interaction, ainsi que l'ensemble des propriétés (Attributs) et les liens avec les autres concepts (Action, Structure d'Activité etc.).

Pour définir entièrement notre ontologie de protocoles, il est également nécessaire de définir les contraintes et les axiomes qui portent sur les éléments de

¹ World Wide Web Consortium

l'ontologie (classes, propriétés et rapports). Nous donnons ci-dessous d'une brève vue d'ensemble des axiomes que nous avons défini pour cette ontologie exprimée en OWL.

```
<owl:Class rdf:about="#ComplexInteractionActivity">
  <owl:equivalentClass>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#InteractionActivity"/>
        <owl:Restriction>
          <owl:onProperty>
            <owl:ObjectProperty rdf:about="#Includes"/>
          </owl:onProperty>
          <owl:someValuesFrom rdf:resource="#StructureActivity"/>
        </owl:Restriction>
      </owl:intersectionOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>
```

Figure 5. Exemple d' Axiome exprimé en OWL

Sur la figure 5, nous présentons un exemple d'axiome qui illustre ce qu'est une activité d'interaction complexe définie en OWL. Cet axiome définit les conditions nécessaires et suffisantes pour définir ce qu'est une activité complexe. Il faut qu'elle contienne au moins une structure de contrôle. De la même manière, nous définissons une activité d'interaction simple qui est le complément d'une activité d'interaction complexe.

4. Usage de l'ontologie

L'introduction d'une ontologie dans un système d'information vise à réduire, voire éliminer, la confusion conceptuelle et terminologique et à étendre vers une compréhension partagée pour améliorer la communication.

Dans cette section, nous montrons l'usage de notre ontologie tant au niveau de la représentation des protocoles qu'au niveau de leur exécution. L'exploitation de l'ontologie est conforme à notre approche de spécification de protocoles décrite en section 3. Cette ontologie est intégrée à un Système de Gestion de Protocole (SGP) (Andonoff *et al.*, 2007) qui joue le rôle de médiateur pour la coordination d'applications impliquées dans un type de système distribué : le workflow Inter-Organisationnel. La figure 6 présente les composants de cette architecture abstraite de ce système. Ce système comprend deux blocks chacun d'eux fournit un service particulier. Le premier block permet la représentation et la sélection des protocoles de coordination alors que le deuxième permet leur exécution. Une description de l'ensemble des composants de cette architecture est détaillée dans (Andonoff *et al.*, 2007). Nous intéressons dans ce travail à la spécification et l'usage des composants du premier block, plus particulièrement l'ontologie de protocoles et l'ontologie de domaine. Comme le montre la figure 6, ce premier block comprend 3 agents

permettant respectivement la spécification de protocoles, la sélection dynamique de protocoles, et l'instanciation de protocoles en vue de leur future exécution. Ces agents utilisent deux sources de connaissances, notre ontologie de protocoles et des ontologies de domaine.

Les connaissances ontologiques étant séparées, elles peuvent être réutilisées et constituent la base d'une interopérabilité entre différents systèmes.

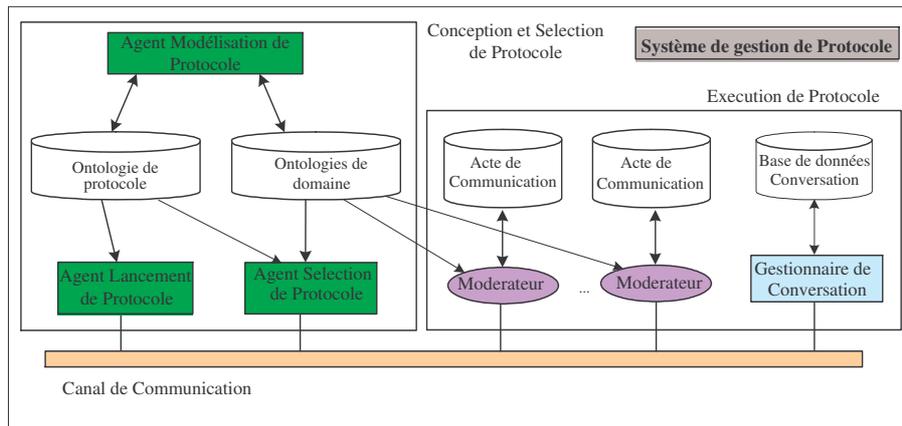


Figure 6. Hiérarchie de classes de l'ontologie sous protégé-2000

La figure 7 montre comment les agents de sélection de protocoles et de lancement de protocoles utilisent les ontologies OWL à l'aide de la librairie Jena (Carroll *et al.*, 2004). Jena est une API qui offre une abstraction pour les ontologies OWL et qui aide les agents à les manipuler et à les activer à l'aide d'un raisonneur. Cette API, développée en Java, constitue un cadre applicatif pour le développement des applications pour le Web sémantique.

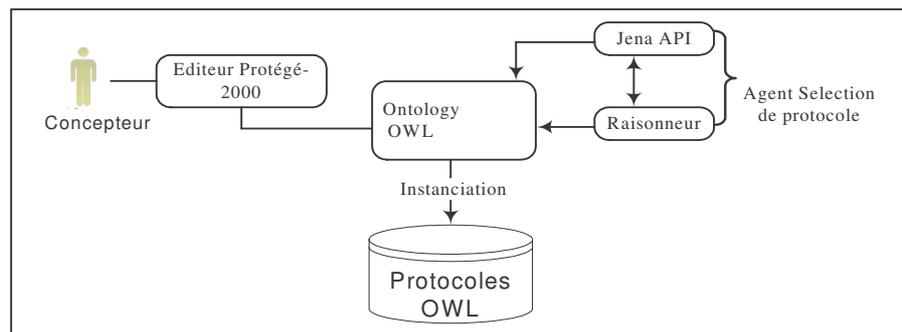


Figure 7. Architecture logicielle pour l'utilisation de l'ontologie

L'interface utilisateur pour manipuler l'ontologie est celle de l'outil Protégé-2000. Cette interface sert pour la spécification du schéma de l'ontologie et pour l'instanciation des protocoles.

La figure 7 donne l'architecture logicielle utilisée pour sélectionner et instancier les protocoles de coordination. Une fois instanciés ces protocoles seront stockés dans un répertoire constituant une bibliothèque de protocoles en OWL. L'ontologie de protocoles est stockée dans deux fichiers OWL séparés : le schéma et l'instance. Le schéma comprend la définition des concepts des relations entre concepts, les contraintes et les axiomes.

Les protocoles sont définis d'une manière déclarative et leurs instances sont stockés dans un fichier d'instances accessible à l'ensemble des agents du système de gestion de protocoles. Lors de la création d'une nouvelle conversation, l'agent sélection de protocole doit rechercher le protocole approprié à cette conversation à partir du fichier des instances. Une fois le protocole est sélectionné, l'agent doit créer une instance de la nouvelle conversation conforme au protocole en question. Au moment de l'exécution, ces deux fichiers sont traités par un Raisonneur (par exemple Racer ou Pellet). Ce dernier supporte l'ensemble des requêtes pour la sélection dynamique des protocoles de coordination. Ces requêtes peuvent être lancées pour interroger le fichier des instances pour la sélection et le choix dynamique des protocoles. Au niveau de l'exécution, la recherche et la sélection d'un protocole pour la coordination au niveau du SGP peuvent être effectuée en prenant en compte l'ensemble des propriétés ou caractéristiques du protocole à mettre en œuvre. Par conséquent, la recherche d'un protocole est non limitée au critère *nom* du protocole mais peut prendre en compte toutes les propriétés permettant sa description.

Dans la suite, nous montrerons à travers un exemple l'instanciation d'un protocole à partir de l'ontologie. Le protocole considéré est celui du Matching. Une instance du protocole de Matching est définie comme l'ensemble d'instance des concepts permettant sa description. La figure 8 illustre une définition partielle du protocole de Matchmaking.

La colonne 1 de la figure 8 donne des éléments partiels de spécification d'un schéma de protocoles : elle montre en fait comment sont définies les connaissances pour la description des protocoles par exemple, la description de la classe protocole (`<owl:Class rdf:ID="Protocol">`), l'ensemble des propriétés comme (`owl:DatatypeProperty rdf:about="#MinNumRole`). A l'opposé, la colonne 2 montre comment est instancié ce schéma de protocoles lors de la définition du protocole de matchmaking. La classe protocole est utilisée pour définir le protocole `ProtocolMatchmaking` (`<OWL:Protocol rdf:ID="ProtocolMatchmaking">`) définit le protocole. Ce protocole est de type "FindingPartner" (`<OWL:HasType rdf:resource="#FindingPartner" />`), et comprend une séquence d'activités d'interaction définissant l'ensemble des opérations effectuées par le protocole. Ces opérations seront déclenchées lors de la réception des messages. Cette activité d'interaction se définit par `<OWL:ComplexInteractionActivity rdf:ID = "RequestReception">` ayant comme input le message `<OWL:HasInputMessage`

rdf:resource="#Inform"/> et comme action à exécuter <OWL:Perform
 rdf:resource="#VerificationOffer"/>.

<pre> <owl:Class rdf:ID="Protocol"> <owl:DatatypeProperty rdf:about="#MinNumRole"> <rdfs:range rdf:resource="#integer"/> <rdfs:domain rdf:resource="#Protocol"/> </owl:DatatypeProperty> <owl:FunctionalProperty rdf:ID="HasType"> <rdfs:range rdf:resource="#integer"/> <rdfs:domain rdf:resource="#Protocol"/> </owl:FunctionalProperty"/> </owl:Class> <owl:Class rdf:ID="ComplexInteractionActivity"> <rdfs:subClassOf> <owl:Class rdf:about="#InteractionActivity"/> </rdfs:subClassOf> </owl:Class> <owl:ObjectProperty rdf:ID="HasInputMessage" /> <owl:ObjectProperty rdf:about="#Perform"> <rdfs:domain rdf:resource="#InteractionActivity"/> <rdfs:range rdf:resource="#Action"/> </owl:ObjectProperty> <owl:ObjectProperty rdf:about="#Includes"> <rdfs:range rdf:resource="#StructureActivity"/> <rdfs:domain rdf:resource="#InteractionActivity"/> </owl:ObjectProperty> </pre>	<pre> <OWL:Protocol rdf:ID="ProtocolMatching"> <OWL:MinNumRole rdf:datatype="&xsd;integer"> 2 </OWL:MinNumRole> <OWL:HasType rdf:resource="#FindingPartner" /> <OWL:sequence rdf:ID="sequence"> <OWL:ComplexInteractionActivity rdf:ID="RequestReception"> <OWL:HasInputMessage rdf:resource="#Inform"/> <OWL:AnswerDelay rdf:datatype="&xsd:int"> 5 </OWL:AnswerDelay> <OWL:Type rdf:datatype="&xsd:string"> Receive </OWL:Type> <OWL:Includes rdf:resource="#sequence"/> <OWL:HasInputMessage rdf:resource="#Inform"/> <OWL:Perform rdf:resource="#VerificationOffer"/> <OWL:HasOutputMessage rdf:resource="#Agree"/> </OWL:ComplexInteractionActivity> </OWL:sequence > </pre>
--	--

Figure 8. *Instanciation du protocole de Matchmaking*

5. Conclusion

Les protocoles d'interaction sont des mécanismes utiles pour faire interagir et coordonner des entités distribuées. Ces protocoles sont efficaces et fiables seulement s'ils possèdent une sémantique claire.

Dans ce papier nous avons défini une ontologie de protocoles permettant la représentation d'interactions complexes, indépendantes du domaine d'application et intégrant aussi bien les aspects statiques des protocoles que leur structure de contrôle (comportement). Cette ontologie représente la structure invariante de tous les protocoles et constitue un cadre clair et cohérent pour l'ingénierie des protocoles.

Par ailleurs, cette ontologie s'intègre dans un système de gestion de protocoles (SGP) comme schéma de référence pour aider à concevoir une bibliothèque de protocoles. Le SGP peut dès lors exploiter cette bibliothèque pour coordonner des systèmes distribués.

La suite de notre travail portera principalement sur la spécification d'un ensemble de protocoles concrets : négociation, sous-traitance, enchères, etc. Nous pensons également utiliser ces protocoles de coordination dans des contextes concrets comme le workflow inter-organisationnel pour assurer la coordination des différents processus qui y sont impliqués.

Remerciements

L'auteur tient à remercier Eric Andonoff et Chihab Hanachi pour leur relecture. Ce travail a été en partie financé par le projet ANR « **ISyCri** » (Interopérabilité des Systèmes en situation de Crise).

Bibliographie

- Amit K., Nirmal D., Ashok M., Leena W., Munindar P. Singh., «A Semantic Protocol-Based Approach for Developing Business Processes», In The Second International Conference on Service-Oriented Computing (ICSOC), New York City, USA, November 2004, p. 99-107.
- Andonoff E., Bouaziz B., Hanachi C., «Protocol Management Systems as a Middleware for Inter-Organizational Workflow Coordination», In The First IEEE International Conference on Research Challenge in Information Science (RCIS), Ouarzazate, Morocco, Université Hassan II, April 2007, p. 85-96.
- Bachimont B., «Engagement sémantique et engagement ontologique : conception et réalisation d'ontologies en ingénierie des connaissances». In J. Charlet et al. (eds), Ingénierie des Connaissances ; Évolutions récentes et nouveaux défis, Eyrolles, 2000, p. 305-323.
- Benattallah B., Casati F., Toumani F., «Representing, Analyzing and Managing Web Service Protocols». Data and Knowledge Engineering, Vol. 58, n° 3, September 2006, p. 327-357.

- Carroll J., Reynolds D., Dickinson I., Seaborne A., Dollin C., Wilkinson K., «Jena : Implementing the Semantic Web Recommendations», In The Thirteen Conference World Wide Web (WWW), New York,USA, 2004, p 74-83.
- Chandrasekaran B., Josephson J.R., Benjamins R., «The Ontology of Tasks and Methods», In The Eleven European Knowledge Acquisition Modeling and Management Workshop, (EKAW), Banff, Canada, April 1998.
- Charlet J., Laublet P., Reynaud C., «Web sémantique», Rapport final : action scientifique 32 CNRS/STIC.<http://doc.enssib.fr/IMG/pdf/ASWebSemantique2003.pdf>
- Corcho O., Fernandez-lopez M., gomez-perez A., «Methodologies, tools and languages for building ontologies. Where is their meeting point? ». Data & Knowledge Engineering, Vol.46, n° 1, 2003, p. 41-64.
- Cranefield S., Purvis M., «Ontologies for interaction protocols», In Proceedings of Workshop on Ontologies in Agent Systems (AAMAS), Tilburg, The Netherlands, 2003, CEUR Publications, KUB Tilburg University.
- Desai N., Ashok U., Mallya, Amit K. Chopra, Munindar P. Singh., «Interaction Protocols as Design Abstractions for Business Processes», IEEE Transactions on Software Engineering, Vol. 31, n° 12, December 2005, p. 1015-1027.
- Ferber J. «Les systèmes multi-agents : Vers une intelligence collective». InterEditions.1995.
- Gandon F., Ontologies informatiques. Mai 2006.
- Gruber T.R., «Toward principle for the design of ontologies used for knowledge sharing», Technical report KSL n° 93-04, 1993, Stanford University.
- Gruninger, M., «Ontology of the process specification language». In : Handbook of ontologies.2003.
- Hanachi C., Sibertin-Blanc C., «Protocol Moderators as active Middle-Agents in Multi-Agent Systems», Autonomous Agents and Multi-Agent Systems (AAMAS), Vol.8, n°3, Avril 2004, p. 131-164.
- Horrocks I., Patel-Schneider P. F., Boley H., Tabet S., Grosz B., Dean M., «SWRL: A semantic web rule language combining OWL and RuleML». May, 2004 (W3C Submission), <http://www.w3.org/Submission/2004/SUBM-SWRL-20040521/>.
- Horrocks I., Peter F., Schneider P., and Harmelen., «From SHIQ and RDF to OWL: The making of a web ontology language». Journal of Web Semantics, Vol. 1, n° 1, 2003,p.7-26.
- Mazouzi H., EL Fallah-serghrouchni., Haddad S., «Open protocol design for complex interactions in multi-agents systems», In The First International joint Conference on Autonomous Agents and Multi-agent Systems (AAMAS) ,Bologna, Italy, July 2002.
- Nirmit D., Ashok U. Mallya, Amit K., Singh MP., «OWL-P: A Methodology for Business Process Modeling and Enactment». Agent-Oriented Information Systems-III, Lecture Notes in Computer Science (LNCS), Springer Verlag, 2006, p. 79-94.

- Noy N.F., Fergerson R.W. et Musen M.A. «The knowledge model of Protégé-2000: Combining interoperability and flexibility». In The Twelve International Conference on Knowledge Engineering and Knowledge Management (EKAW), Juan-les- Pins, Springer, 2000, p. 17-32.
- OWL, Web Ontology Language, Feb. 2004, <http://www.w3.org/TR/2004REC-owl-features-20040210/>.
- OWL-S Services Coalition., «Ontology Web Language for Services». Version 1.0. Documentation available at: <http://xml.coverpages.org/ni2004-01-08-a.html>.
- Paurobally S., Jennings N. R., «Protocol engineering for web services conversations», Journal of Engineering Applications of Artificial Intelligence, Special Issue on Agent Oriented Software Engineering, Vol. 18, March 2005, p. 237-254.
- Struder R., Benjamin V., Fensel D., «Knowledge Engineering, Principle and Methods». Data and Knowledge Engineering, Vol. 25, p. 161-197.
- Tamma V., Phelps S., Dickinson I., Wooldridge M., «Ontologies for supporting negotiation in e-commerce», Engineering applications of artificial intelligence, Vol. 18, March 2005, p.223-236.
- Tamma V., Wooldridge M., and Dickinson I., «An ontology for automated negotiation», In Proceedings of the international workshop on ontologies in agent systems (OAS), Bologna, Italy,2002.
- Toivonen S., Helin H., «Representing interaction protocols in daml», In Ludger van Elst, Virginia Dignum, and Andreas Abecker, editors, Agent Mediated Knowledge Management, International Symposium, Vol. 2926,USA, 2003. Springer, p 310–321.
- Vitteau B., Huget M.-P., «Modularity in interaction protocols». in: F. Dignum (Ed.), Advances in Agent Communication, in Lecture Notes Artificial Intelligence., vol. 2922, Berlin, 2004, p. 291-309.