

Enrichissement sémantique d'une architecture orientée Web service

A.Boubekeur ¹, M. Malki ², A.Chouarfia ³

¹Département d'Informatique Université de Tiaret, Algérie Boubekeur@univ-tiaret.dz

²Laboratoire EEDIS, Département d'Informatique, Université de Sidi Bel-Abbes, Algérie

Malki_m@univ-sba.dz

³Département d'Informatique Université USTO-Oran, Algérie Chouarfia@univ-usto.dz

Résumé. L'orientation service est un paradigme qui permet l'unification dans les approches d'intégration des systèmes d'information. Il inclut l'intégration des données de part la description sémantique des données manipulées par les services. Cette intégration « implicite » des données est plus souple vu le nombre limité des concepts utilisés par les services. Ceci nous a donc conduit à proposer une architecture visant d'une part, à réduire la complexité dans le développement et l'intégration des ontologies de domaine. Et d'autre part à assurer la découverte et l'invocation automatique des services. Cela en intégrant des ontologies sans apporter de changements majeurs dans le mode de fonctionnement des web services (http, soap...). Ce travail offre donc une architecture qui représente une étape vers la satisfaction de cette automatisation à travers les web services sémantiques sans avoir à redéfinir l'ensemble du système d'information.

Mots clefs. Web service sémantique, goal, médiateur, WSMO, WSDL, SOA, intégration

1 Introduction

De nos jours les informations échangées sur le web deviennent de plus en plus importantes. La recherche d'information est alors imprécise et lente. Le langage commun XML renforcé par les méta-données s'est montré insuffisant. Pour satisfaire le nombre énorme d'utilisateurs, il est nécessaire d'introduire la notion de sémantique des données. Ce qui a donné naissance au projet du web sémantique appelé aussi « web 2.0 ». Le web 2.0 utilise les outils de l'Intelligence Artificielle (IA) notamment ceux de l'Ingénierie de la Connaissance (IC), parmi lesquels on retiendra les ontologies.

Par ailleurs, des recherches ont permis l'ajout d'un niveau fonctionnel au web. Les services web satisfont l'interopérabilité à ce niveau. Les travaux montrent que cette dernière est insuffisante. Il faut donc une annotation sémantique explicite pour parfaire l'interopérabilité. Une fois de plus les ontologies seront utilisées.

L'intégration sémantique des applications est un domaine de recherche très actif. Il peut concerner aussi bien l'intégration par les données que celle par les traitements. L'intégration sémantique par les données nécessite le développement d'ontologie volumineuse, lourde à concevoir. Du fait qu'au niveau des données le nombre de concepts à prendre en compte est énorme. Une analyse plus fine permet de déduire que l'orientation service peut constituer un élément fédérateur pour l'ensemble des deux approches. En effet, l'orientation service est un paradigme qui permet l'unification dans les approches d'intégration des systèmes d'information. Il inclut l'intégration des données de part la description sémantique des données manipulées par les services. Cette intégration « implicite » des données est plus souple vu le nombre limité des concepts utilisés par les services. Ceci nous a donc conduit à nous focaliser principalement sur les services web sémantiques, qui constituent l'un des moyens les plus efficaces pour mettre en œuvre l'intégration sémantique des systèmes.

Nous proposons donc une architecture pour l'enrichissement Sémantique de l'Architecture SOA. Elle vise d'une part, à réduire la complexité dans le développement et l'intégration des ontologies de domaine. Et d'autre part à assurer la découverte et l'invocation automatique des services. Cela en intégrant des ontologies sans apporter de changements majeurs dans le mode de fonctionnement des web services (http, soap...). Notre objectif est de passer d'un système interopérable via les classiques web services vers un système interopérable via les services web sémantiques sans avoir à redéfinir l'ensemble du système d'information.

La section 2 de ce papier présente l'état de l'art des différentes approches visant à satisfaire l'intégration des systèmes d'information. L'objectif est une étude comparative de ces approches. La section 3 présente notre architecture ou apparaît la spécification et la description sémantique du cas AVV¹. Et enfin La dernière section fait une comparaison de notre travail avec des travaux existants et des suggestions de conclusion pour le futur.

2 Etat de l'art

L'intégration des applications d'entreprise peut intervenir au niveau de plusieurs couches et principalement: au niveau des données et des traitements². Cela demeure également valable dans le cadre des approches d'intégration sémantique, et en particulier dans le cas des approches d'intégration basées sur les ontologies [8].

Quelle que soit la couche retenue, il est possible de caractériser les approches d'intégration à travers trois activités principales, qui sont les plus pertinentes, et qui sont comme suit [7]:

- ✓ Description: est l'activité qui permet de décrire sémantiquement les aspects du système d'information;
- ✓ Découverte: est l'activité qui permet de rechercher les informations et les fonctionnalités qui peuvent être fournies par le système;

¹ Agence de voyage virtuelle : Il s'agit d'un cas d'utilisation proposé par le W3C.

² Certains auteurs parlent aussi de l'intégration au niveau des processus. A notre point de vue ce niveau est inclus au niveau traitement.

- ✓ Composition: est l'activité qui permet de comparer et de combiner des informations, des fonctionnalités et des processus.

L'intégration sémantique des applications au niveau des données consiste à définir une ou plusieurs ontologies permettant de décrire la sémantique du domaine. Ces ontologies sont utilisées dans des mécanismes de conciliation et d'unification de la sémantique afin de résoudre les conflits sémantiques liés aux données échangées et/ou manipulées par les applications. De telles approches sont souvent appelées par la communauté des bases de données approches de médiation de contexte [2]. La résolution de conflits se base généralement sur des mécanismes de classification et de comparaison d'entités d'ontologies.

En ce qui concerne l'intégration sémantique des applications au niveau des traitements, il existe principalement les travaux basés sur les services Web sémantiques [10], [9], [3]. La notion de service Web sémantique a pour objectif de combler certaines lacunes des Services Web traditionnels liées à la prise en compte de l'aspect sémantique. Les initiatives les plus pertinentes autour du concept de Service Web sémantique sont principalement : OWL-S [12], WSMF [5], [13], WSMO [15] et METEOR-S [11], [4].

WSMO constitue une approche basée sur des fondements conceptuels issus de WSMF pour la réalisation des services sémantiques. Elle est beaucoup plus orientée utilisateur dans le sens où elle permet d'offrir un langage très proche des utilisateurs. De plus, l'une des caractéristiques les plus importantes est le fait qu'elle soit compatible avec les standards web services (WSDL [14] en particulier). En effet bien qu'elle utilise son propre langage, la notion de « Grounding » permet de définir des mappings avec des entrées WSDL [14] ce qui suppose une intégration sans grande peine avec un système SOA existant.

WSMF est un cadre générique pour la réalisation des services sémantiques. Il constitue donc un Framework qui permet d'offrir beaucoup plus un cadre méthodologique qu'une infrastructure opérationnelle.

OWL-S constitue une ontologie générique de services permettant principalement de décrire les services web dans un cadre général. Elle est aussi compatible avec les standards web services à travers le « ServiceGrounding » qui permettent de définir des mappings entre OWL-S et WSDL. Son défaut majeur est l'absence d'environnement d'exécution ce qui fait que la médiation des services reste encore à implémenter.

METEOR-S constitue une approche basée sur l'extension de standards web services tels que WSDL et UDDI [10]. Le principe d'enrichissement sémantique est basé principalement sur l'utilisation des fichiers WSDL-S qui sont le résultat de l'annotation sémantique de fichiers WSDL. Bien que cette approche repose sur des standards, il n'en demeure pas moins qu'elle manque d'abstraction dans le sens où elle constitue une approche plus proche du niveau d'implémentation que du niveau conceptuel.

Nous pouvons retenir qu'OWL-S est une approche générique, basée sur le standard OWL [8]. WSMF constitue beaucoup plus un cadre méthodologique. WSMO est une approche orientée utilisateurs, basée sur le langage (Web Service Modeling Language) WSML [15] combinant plusieurs logiques. METEOR-S est une approche pragmatique mais qui manque d'abstraction. Il s'avère qu'aucune de ces approches n'est complètement mature pour prendre en charge efficacement le processus de

sémantification³. Toutes ces approches souffrent plus ou moins du manque de méthodologies, du manque d'architectures flexibles pouvant décrire efficacement la sémantique liée aux applications, et aussi du manque de maturité ou parfois même de l'absence de mécanismes de description et de médiation efficaces. Cependant, au vu de tous les éléments présentés dans cette section, l'approche WSMO est celle qui présente le plus de maturité et d'abstraction. C'est la raison fondamentale qui nous amène à nous baser sur cette approche (WSMO) comme fondement de base pour le travail mené dans le cadre de la « sémantification » d'un système SOA. Dans la prochaine section nous exposons l'architecture ESASOA qui est la solution que nous proposons pour satisfaire la prise en compte de la sémantique dans un système SOA.

3 Présentation de notre architecture

L'architecture que nous proposons est conçue pour opérer sur les descriptions sémantiques selon le modèle conceptuel Web Services Modeling Ontology (WSMO). Ces éléments sont décrits en utilisant le Web Service Modeling Language (WSML). La conception de l'architecture est guidée par quelques principes. Ces derniers reflètent les exigences fondamentales d'un environnement distribué, orienté service et centré sur l'utilisateur, suivant les indications de la figure (fig.1.).

Le principe orienté service représente une approche distincte pour l'analyse, la conception, et l'implémentation. Elle introduit des principes particuliers qui gouvernent les aspects de communication, de l'architecture et de traitement logique. Cela inclut la réutilisation du service, le faible couplage et l'abstraction. Nous distinguons, par ailleurs des services selon deux points de vue. En premier, nous distinguons deux types de services selon les fonctionnalités qu'ils fournissent dans l'architecture, à savoir les services métiers et services du middleware. Les services métiers sont fournis par plusieurs fournisseurs de services. Ces services donnent accès au S.I de leurs fournisseurs. Par ailleurs, les services du middleware sont les orchestrateurs principaux de l'intégration et l'interopération des services métiers.

Aussi, nous distinguons deux types de services du point de vue de leur abstraction dans l'architecture, à savoir les Web Services, et les services. Le Web Service est un service général qui peut prendre plusieurs formes quand il est instancié (par exemple réservation d'un vol) alors que le service est une instance réelle du Web Service qui est consommée par un utilisateur et fournit une valeur concrète à l'utilisateur (par exemple réservation d'un vol d'Alger à Oran). Nous utilisons cette distinction au niveau des services métiers dans l'architecture.

Le principe de sémantique en général est considéré comme une description riche et formelle de l'information et des modèles de comportements permettant l'automatisation de certaines tâches au moyen de raisonnement logique. Combiné avec le principe de l'orientation service, la sémantique permet la définition de modèles de service riches et formels et des ontologies qui permettent l'automatisation

³ Nous empruntons les anglicismes 'sémantifier' et 'sémantification' pour désigner respectivement l'action et le processus d'enrichissement sémantique des services.

totale ou partielle des tâches telles que la découverte du service, médiation, invocation, etc.

Le principe de distribution est le processus d'intégration de plusieurs entités informatiques collaborant pour l'exécution d'une tâche computationnelle de manière transparente et cohérente, afin qu'elles puissent paraître comme un système unique et centralisé. Le principe de distribution est appliqué au système du middleware de l'architecture. Il peut être aussi appliqué aux services métiers qui permettent l'exécution d'un processus à travers la composition de plusieurs services métiers distribués sur le réseau.

Le principe "orienté utilisateur" met l'utilisateur au centre de l'architecture. Il fait référence aux concepts comme la personnalisation des services métiers, faciliter l'usage et la distribution du service, accomplir avec efficacité les objectifs des utilisateurs, etc. L'architecture établit l'infrastructure qui au moyen de ses fonctionnalités encourage le principe de l'orientation vers l'utilisateur dans la fourniture du service.

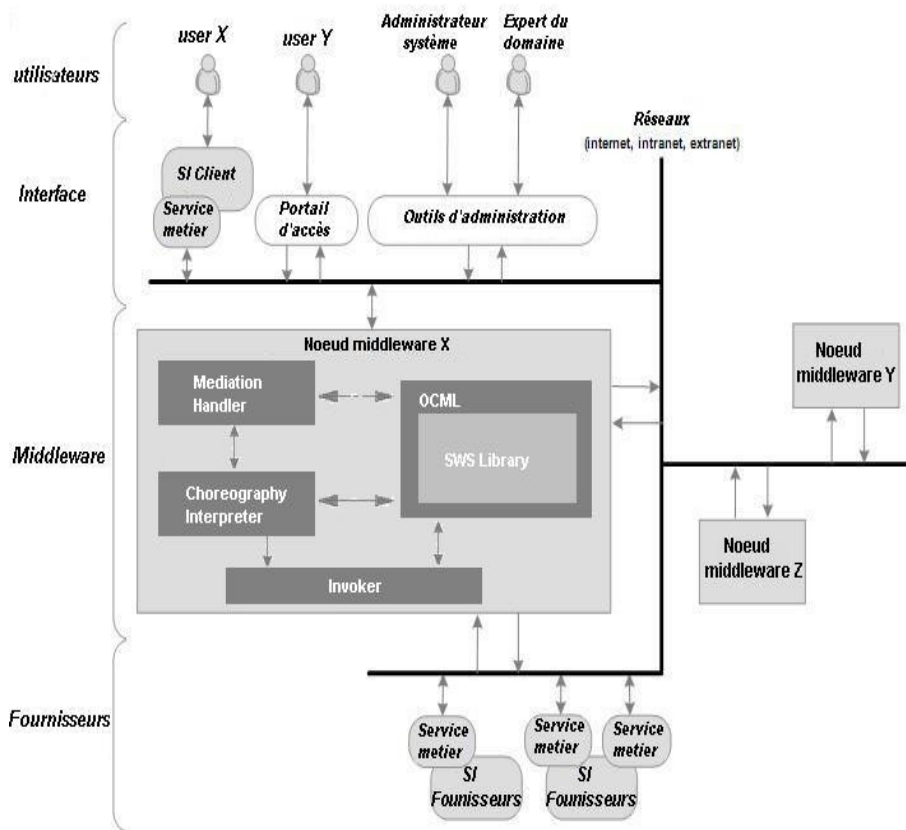


Fig. 1. L'architecture proposée

Le middleware (IRS III), le noyau de l'architecture, se compose de :

- ✓ SWS Library : Au centre du serveur IRS-III, nous avons une librairie SWS où sont entreposées les descriptions sémantiques des services web qui sont faites en OCML⁴. Les ontologies du domaine sont aussi disponibles dans la Librairie.
- ✓ Choreography Interpreter : Ce composant interprète le grounding et les transitions dans la description de la chorégraphie quand ils sont demandés par le composant mediation handler.
- ✓ Mediation Handler : Les activités d'IRS-III comme la sélection ou l'invocation sont chacun supportée par un composant de médiation spécifique dans le composant médiation handler. Ces activités peuvent impliquer l'exécution de la médiation de service. Il facilite, lors de l'exécution, la médiation entre les interfaces des différents services impliqués dans le processus en cours. Il est utilisé lors des processus de découverte pour évaluer si la conversation entre le service et son utilisateur est possible. Durant cette analyse il découvre les éventuels conflits à résoudre par exemple le retrait d'un message s'il s'avère non nécessaire.
- ✓ Invoker : Le composant invoker du serveur communique avec la plate-forme du fournisseur, en envoyant les entrées du client et en rapportant le résultat au client.

Notre proposition est une solution aux problèmes de l'intégration sémantique en ajoutant une couche abstraite supplémentaire dans l'architecture classique des web services. C'est une architecture ouverte et extensible via l'ajout de nouveaux composants. Elle ne révolutionne pas l'actuelle technologie des web services. Notre architecture ne fait qu'apporter un enrichissement sémantique à la description des services et des données qu'ils utilisent à travers l'intégration d'ontologies (fig.2.).

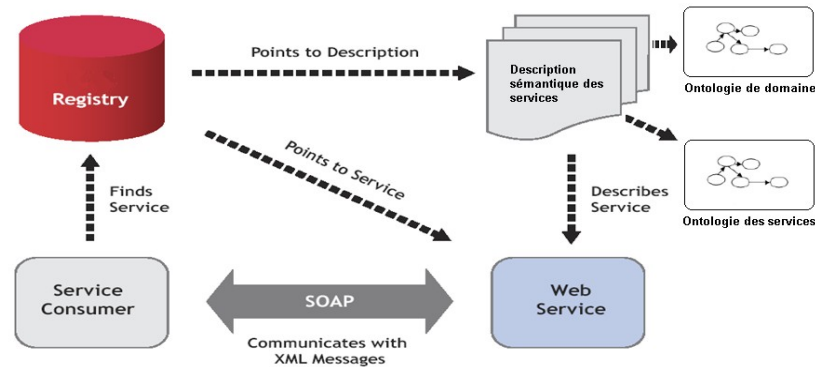


Fig. 2. Fonctionnement des WS dans notre architecture

⁴ Le langage OCML est une combinaison des frames et de la logique de premier ordre et permet de représenter les concepts, la taxonomie des concepts, les relations, les fonctions, les axiomes et les instances.

3.1 Approche de conception de l'ontologie de description de services

L'analyse des solutions d'intégration sémantique actuelles, et en particulier des approches basées sur les services sémantiques, a montré un certain nombre de limites. Elles concernent principalement les architectures des ontologies actuelles, le manque de méthodologie pour définir les ontologies et les services sémantiques et enfin le manque de maturité des mécanismes de description et de médiation de services.

Le problème de développement est résolu en utilisant une approche de développement modulaire. En effet cette approche permet d'avoir une ontologie complète à travers l'intégration de plusieurs modules. Cette intégration se fait en établissant des relations entre les différents concepts. L'approche modulaire impose certaines contraintes au développeur d'ontologie notamment le fait de se référer à une ontologie unique dans leur conception: ontologie consensuelle [1].

Une telle approche de conception nécessite tout d'abord la recherche d'ontologies formalisant les concepts génériques du domaine. Nous les appelons les ontologies de niveau 1.

Nous importons les ontologies sources dans une ontologie nouvellement créée que nous appelons « ontologie d'application ». Cette dernière contient l'ensemble des concepts permettant de définir les données utilisées dans le système. Elle permet la souplesse de l'ajout ou du retrait d'un service⁵, assurant ainsi la flexibilité de l'architecture. Les ontologies sources et l'ontologie d'application constituent les ontologies de niveau 2. Le niveau 3 se compose d'une unique ontologie important l'ontologie d'application et l'ontologie générique de description de service⁶. La figure (fig. 3.) donne les étapes de développement et la figure (fig. 4.) donne le schéma général de modélisation en WSMO de notre cas d'utilisation issu d'une telle conceptualisation.

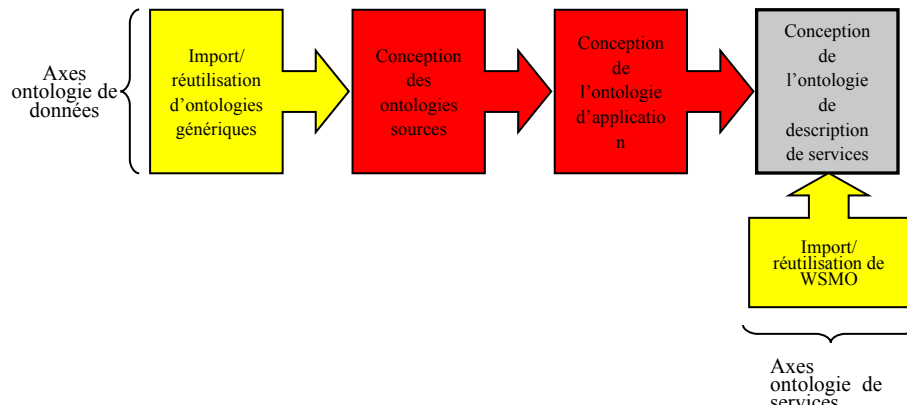


Fig. 3. Etape de conception de l'ontologie de services

⁵ A travers l'ajout ou le retrait de l'ontologie en question.

⁶ WSMO dans notre cas.

Une définition répandue de l'ontologie est donnée par [6] où une ontologie est définie comme « spécification explicite et formelle d'une conceptualisation partagée ». Dans WSMO, les ontologies fournissent une sémantique compréhensible par la machine de l'information employée par tous les acteurs impliqués dans l'utilisation des web services. Permettant, ainsi l'échange et l'interopérabilité des composants. Les types de base d'une ontologie sont les concepts, les relations, les fonctions, les instances, et les axiomes.

Nous définissons cinq ontologies à savoir `algerianCitiesKb`, `nigerianCitiesKb`, `malianCitiesKb`, `AfricanAirplaneTravelApplication`, `AfricanTravelServicesDescriptions`. Elles sont reliées aux trois ontologies `commonConcepts`, `countryOntology` et `simpleTime` qui sont des ontologies déjà existantes.

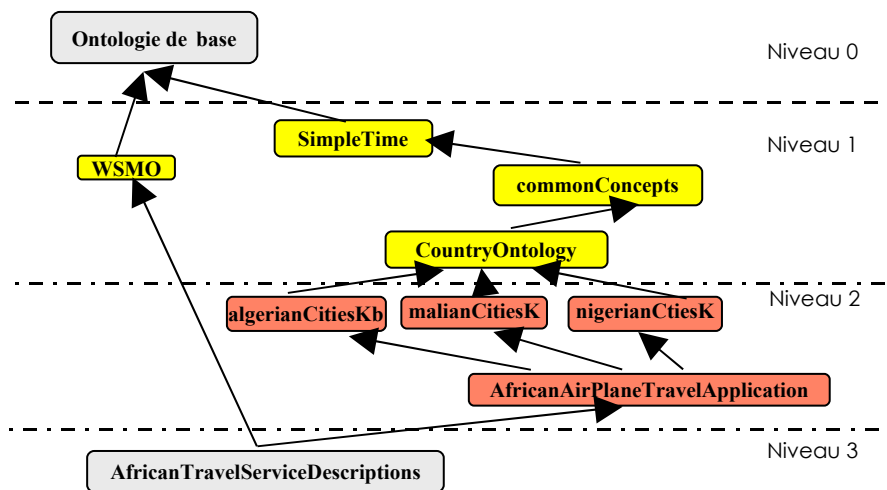


Fig. 4. Modélisation en WSMO du cas d'utilisation

3.2 Architecture de base de l'AVV

Dans le cas d'utilisation, l'AVV est le point central d'interaction entre le Client et les autres Web services. Concernant les exigences technologiques, il est évident à partir des descriptions du scénario de l'utilisation que les Web Services offerts par les fournisseurs de service doivent être accompagnés d'informations descriptives suffisantes pour supporter l'utilisation automatique des Web Services et que l'AVV doit fournir tous les mécanismes pour manier les Web services sémantiques.

L'architecture de base d'une telle AVV comme entité centrale pour l'exécution des Web Services sémantiques est montrée dans la figure (fig. 5.). Les fonctionnalités essentielles des AVVs orientés web services sémantique en considérant les exigences de la technologie des web services sémantiques sont :

- ✓ Fournir une interface utilisateur pour l'interaction avec le client (pour un utilisateur humain ou une machine).
- ✓ Découvrir les Web Services convenable pour une demande de l'utilisateur.
- ✓ Invoquer les services externes.
- ✓ Fournir un environnement d'exécution de Web Service Sémantique.
- ✓ Traiter des ressources hétérogènes, donc tenir compte de la médiation appropriée.
- ✓ Fournir des interfaces pour la coopération avec les fournisseurs de services.

L'AVV est une application B2C orientée SWS qui fournit un service à un utilisateur finale qui suit un modèle client-serveur. Afin de supporter un fonctionnement cohérent de l'AVV et assurer la compatibilité des descriptions de Web Services aux services physiques, une plateforme générique pour les web services sémantiques est nécessaire. Cela est fourni par ESASOA à travers WSMO. La technologie web service sémantique peut être appliquée pour faciliter l'utilisation du système (trouver l'AVV et accomplir son but (goal)) et pour l'AVV qui bénéficie des descriptions sémantiques des services offerts par ses associés (fournisseurs de services).

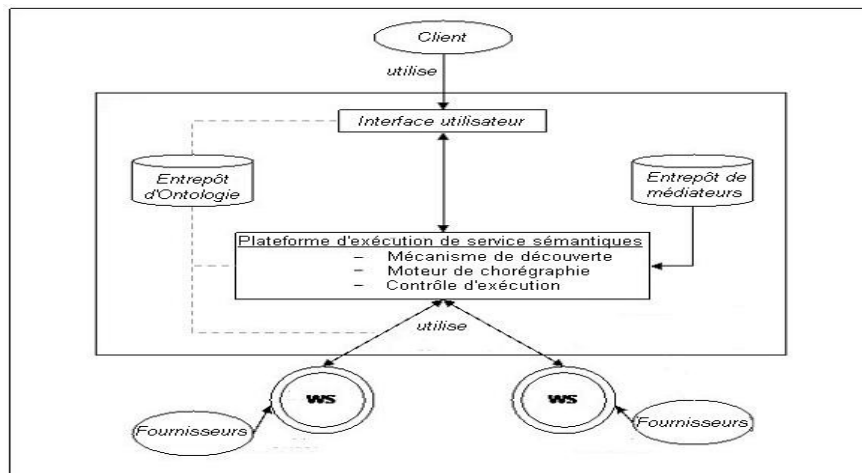


Fig. 5. Architecture de l'AVV

L'AVV de notre cas d'étude intègre les web services de quatre compagnies aériennes différentes. En bref, elle permet à un client d'effectuer des réservations de vol. L'AVV découvre et invoque les Web Services offerts par les compagnies aériennes. Pour faciliter ceci, il peut y avoir donc un contrat dit du "parapluie" entre les fournisseurs du service et l'AVV pour un usage régularisé et une allocation du Web Service. La figure (fig. 6.) donne une vue d'ensemble (adapté de W3C Voyage Agent Usage Cas Overview).

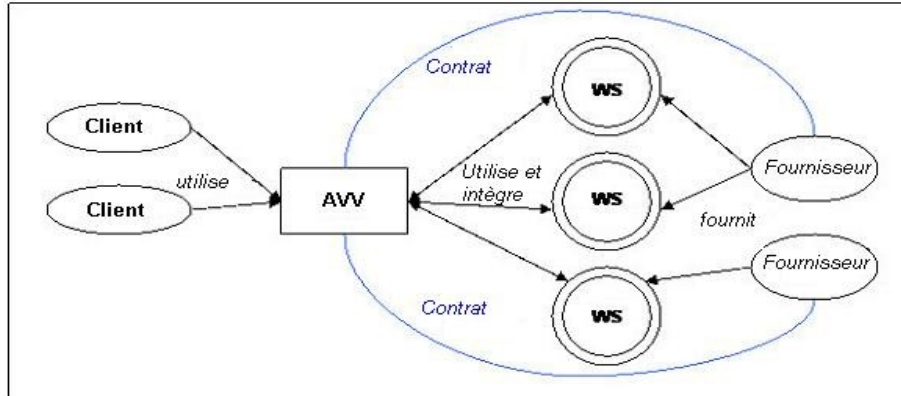


Fig. 6. Vue d'ensemble du cas d'utilisation : Agence virtuelle de voyage

4 Conclusion

Nous avons mis en œuvre une architecture flexible d'intégration sémantique des services. Elle est basée sur l'approche WSMO qui est, parmi les différentes approches se proposant de résoudre l'intégration sémantique des services, la plus mature dans le sens où elle propose une conceptualisation dont la mise en œuvre est possible avec les moyens actuelles.

Notre architecture repose sur quatre principes à savoir, elle est orientée services, prend en compte la sémantique (à travers une formalisation explicite), est distribuée et est orientée utilisateur. Le principe orienté service assure une flexibilité de l'architecture et une intégration facile de plusieurs systèmes SOA. Le principe de sémantique est dû à l'intégration d'ontologie dans le système. Le principe de distribution assure l'évolutivité de l'architecture. Le principe d'orientation utilisateur tient compte de l'utilisateur final à travers l'intégration d'une interface utilisateur.

Chacun de ces principes est fondamentale et permet la présentation d'une solution applicable à un cas d'utilisation réel. Ceci constitue un premier apport de notre architecture. Le second porte sur l'approche de développement des ontologies de domaines. Nous avons adopté le développement modulaire qui est une méthode peu exploitée dans les travaux actuels. En effet elle permet l'intégration des ontologies lors de la conception (design-time) réduisant ainsi au maximum, voir le ramenant à zéro, le nombre de mapping nécessaire pendant l'exécution (run-time).

L'application de notre approche porte sur la satisfaction du cas d'utilisation proposé par le W3C. Il s'agit d'une agence de voyage virtuelle prenant en charge tous les mécanismes nécessaires au processus de voyage. Elle offre la sémantique nécessaire pour que l'utilisateur puisse au bout de quelques cliques s'assurer un itinéraire tout en satisfaisant l'ensemble des contraintes collatérales (réservation de billet, d'hôtel, de voiture...). Ce cas illustre parfaitement une application possible des

SWS. Notre application intègre les services proposés par trois agences de voyage. Elle satisfait un premier niveau du cas d'utilisation.

Les SWS sont à la base de recherches actives. Cela dit bon nombre de ces recherches sont au stade conceptuel. Nous avons recensé quelques travaux que nous regroupons avec notre architecture dans le tableau comparatif dans table 1.

Dans cette première version de notre architecture ne prend pas en compte la composition des services. La complexité du problème d'intégration sémantique des services nécessite une approche incrémentale dans la conception de toute architecture. Nous nous sommes limités aux seuls problèmes de découverte et d'invocation des services. Différentes perspectives sont alors à envisager. Les travaux futurs porteront sur les problèmes ajoutant un niveau de complexité à l'architecture telle que la composition des services sémantiques et l'amélioration des interfaces utilisateurs à travers l'amélioration du mécanisme de requête.

Table 1. Tableau comparatif de travaux de recherches

Travaux Critères	Notre architecture	PoliMi - Cefriel ⁷	DERI AT & DERI IE ⁸	FSU Jena ⁹
Approche utilisée	WSMO	WSMO	WSMO	-
Ontologie	Multi-ontologique (Développement modulaire) (Langage OCML)	-	Multi- ontologique (Langage WSML)	Multi-ontologique (Langage DSD)
Middleware Orientée service	Oui	Oui	Oui	Oui
Orientée Utilisateur	Oui	Non	Non	Non
Distribuée	Oui	Non	Oui	Non
Middleware	IRS III	SWE-ET	WSMX	DIANE

⁷ <http://www.cefriel.it/>

⁸ <http://www.deri.at/>

⁹ <http://hnsp.inf-bb.uni-jena.de/DIANE>

References

1. Keita A. K., C. R. (2006). Un outil d'aide à la construction d'ontologies pré-consensuelles : le projet townology. INFORSID.
2. Benslimane S.M., Amar Bensaber D., Malki M. (2004) "Vers une médiation de contexte à base d'ontologies multi-représentation. " Colloque sur l'Optimisation et les Systèmes d'Information (COSI), pp 101-112, Tizi Ouzou, Algérie, 07 au 09 juin 2004 .
3. Bussler C. (2004), "Semantic Web Services". ICWE'04, Munich, 2004.
4. Cardoso J., et Sheth A. (2004) , "Semantic Web services and web processes composition" Springer-Verlag, New York Inc, 2004.
5. Fensel D., et Bussler C. (2002). V. Fensel, and Bussler, "The Web Service Modeling Framework WSMF". In Electronic Commerce Research and Applications, 1(2), Elsevier, Scinces B. V., 2002.
6. Gruber, T. (1993). "A Translation Approach to Portable Ontology Specifications". Knowledge Acquisition, Vol. 5 (No. 2) pp. 199-220.
7. Harriche K., Hatchabi M., Maiga A., Boubekeur A. (encadrement), (2008) "Vers une approche d'intégration sémantique via les web services". Mémoire d'Ingéniorat. Université Ibn Khaldoun de Tiaret, juin 2008.
8. Izza S., Vincent L., and Burlat P., (2006) "Dealing with Semantic Application Integration Withn Large and Dynamic Enterprises". International Journal of Cooperative Information systems, Vol. 15, No. 4, pp. 507-533, [OnLine] <http://www.worldscinet.com/ijcis/15/1504/S02188430061504.html>.
9. Kellert P., et Toumani F., (2003) "Les web services sémantiques". Research Report LIMOS/RR 03-1511.
10. McIlraith S., Son T., et Zeng H. (2001), "Semantic Web Services". IEEE Intelligent Systems Vol. 16 (No. 2), pp. 46-53.
11. METEOR-S, (2005) "METEOR-S project deliverables". [On Line] <http://lsdis.cs.uga.edu/Projects/METEOR-S/>.
12. OWL-S Coalition, (2004) "OWL-S: Semantic Markup for Web Services". OWL-S Coalition, [On Line] <http://www.daml.org/services/>.
13. Semantic Web Services Framework (SWSF) Overview. (2005). Récupéré sur W3C Member Submission: <http://www.w3.org/Submission/SWSF/>
14. WSDL-S. (2005). Web Service Semantics - WSDL-S, Version 1.0. Récupéré sur W3C Member Submission: <http://www.w3.org/Submission/WSDL-S/>
15. WSMO, (2005) "Web Service Modeling Ontology Project Deliverables". WSMO Project.