

## Evaluation des Performances d'une Architecture pour la Découverte et la Composition des Web Services

N. BERNINE<sup>a</sup>, H. NACER<sup>b</sup>, K. ADEL<sup>c</sup> et D. AÏSSANI<sup>d</sup>

Unité de recherche LaMOS (Laboratoires de Modélisation et d'Optimisation des Systèmes)  
Université de Bejaïa, Bejaïa 06000, Algérie  
Tél. (213) 34 21 51 88,

<sup>a</sup> email : [nassima.bernine@gmail.com](mailto:nassima.bernine@gmail.com)

<sup>b</sup> email : [sino\\_nacer@yahoo.fr](mailto:sino_nacer@yahoo.fr)

<sup>c</sup> email : [ka\\_adel@yahoo.fr](mailto:ka_adel@yahoo.fr)

<sup>d</sup> email : [lamos\\_bejaia@hotmail.com](mailto:lamos_bejaia@hotmail.com)

**Résumé** Les Web services sont des applications accessibles sur Internet réalisant chacune une tâche spécifique. Pour fournir une solution à une tâche complexe, on peut regrouper des Web services pour n'en former qu'un seul ; on parle alors de composition de Web services. L'objectif des Web services est de faciliter l'accès aux applications entre entreprises et ainsi de simplifier les échanges de données. Plusieurs algorithmes de compositions ont été proposés tels que : Les graphes orientés, les réseaux de Petri ou les automates d'état finis, basée sur le concept de similarité sémantique.

L'évaluation de performances est un procédé qui consiste à déterminer les caractéristiques des systèmes obtenus pour ces entrées/sorties dans le cadre de l'accomplissement de la tâche qui leur a été assignée.

Dans ce travail, on a évalué les performances d'un système de composition avec le graphe, en utilisant un simulateur implémenté sous Matlab.

**Mots clés** : Web service, Composition des Web services, Evaluation des performances, Graphe.

### 16.1 Introduction

Actuellement, les Web services sont très utilisés. L'émergence de ces services a permis aux applications Web d'être vues comme un ensemble de services bien structurés et correctement décrits, plutôt qu'un ensemble d'objets et de méthodes. En effet, cela facilite les échanges entre les applications d'une même organisation, mais permet aussi l'ouverture vers les applications des autres organisations d'une façon distribuée ; ce qui permet d'offrir des avantages primordiaux en termes d'interopérabilité et de maintenance de telles applications.

### 16.2 Notions de base

Un Web service est dit composé ou composite lorsque son exécution implique des interactions avec d'autres Web services afin de faire appel à leurs fonctionnalités. La

composition de Web services spécifie quels services ont besoin d'être invoqués, dans quel ordre et comment gérer les conditions d'exception [3, 5].

La découverte des Web services consiste à trouver les mises en correspondance adéquates entre les éléments de besoin des utilisateurs et les éléments des Web services existants. Ces besoins peuvent être couverts par un Web service simple, ou un Web service complexe issu de la tâche de la composition [6].

Pour assurer la qualité d'un système, on fait appel à l'évaluation de performances. L'évaluation des performances fait l'objet de discussions sur l'espace informatique. Elle détermine la qualité d'un système à partir des résultats obtenus pour les entrées/sorties de ce système dans le cadre de l'accomplissement de la tâche qui leur a été assignée.

L'évaluation de performances s'intéresse au calcul des paramètres (indices) de performances d'un système. Ces derniers sont représentés sous forme de valeurs quantitatives, comme le débit, le temps d'attente, le temps de réponse, le nombre moyen d'une entité donnée, le taux d'utilisation, [1, 2, 4, 7].

L'évaluation de performances est basée sur la simulation ou des méthodes analytiques.

Ce papier est structuré de la manière suivante :

- Dans la première section, nous présentons le système de composition avec le graphe.
- Dans la deuxième section, nous présentons la modélisation du système de composition avec le graphe.
- Dans la troisième section, nous présentons le calcul des performances du système proposé.

## **16.3 Proposition : Evaluation des performances dans la composition des Web services par un graphe :**

### **16.3.1 Architecture de la composition par un graphe :**

Les efforts de recherche et de développement récents autour des Web services ont conduit à un certain nombre de spécifications qui définissent aujourd'hui le fonctionnement des Web services. Le fonctionnement d'un Web service repose sur un modèle en couches dont les trois couches principales sont représentées dans la figure (Fig.16.1). La couche découverte a pour objectif de rechercher et de localiser un Web service. Ceci est possible grâce au protocole standard de découverte UDDI (Universal Description Discovery and Integration).

La deuxième couche - Description - permet de décrire les interfaces des Web services (c.-à-d. les paramètres des fonctions, les types de données, etc.). L'architecture du système proposé est illustrée par la figure suivante :

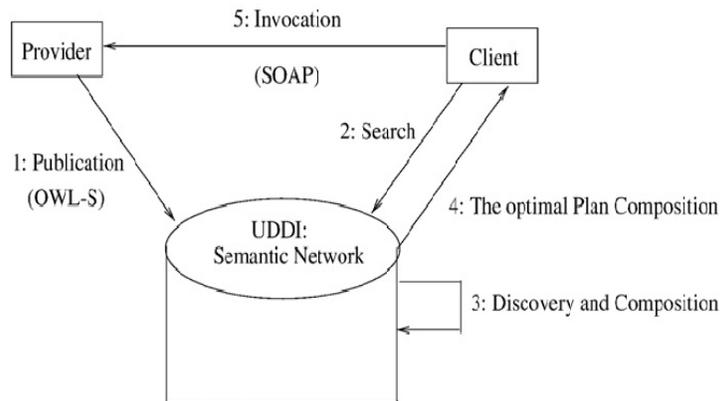


Figure 16.1. Architecture des Web services.

## 16.4 Modélisation du système

### 16.4.1 La modélisation :

Ce système est modélisé par un modèle markovien, car on a les deux quantités stochastiques principales "les temps des inter-arrivées" et "la durée de service" sont des variables aléatoires indépendantes, exponentiellement distribuées. La propriété sans mémoire de la loi exponentielle facilite l'analyse de ce modèle. Le système est caractérisé par cinq types de services :

1. **Translation** : Après l'affectation des Web services au système, le service qui s'ensuit est la translation de descriptions des Web services de WSDL aux OWL-S. Ce service s'effectue groupe par groupe.
2. **Calcul du graphe** : Consiste au calcul du graphe des Web services existants. Ce calcul se fait groupe par groupe.
3. **Extraction** : Après l'affectation des requêtes au système, le service qui suit est l'extraction des entrées et sorties de ces requêtes. L'extraction des requêtes s'effectue une par une (traitement par client).
4. **Composition** : Consiste à chercher tous les chemins possibles pour satisfaire les requêtes. La composition se fait pour une requête après une.

5. **Sélection** : Consiste à choisir le meilleur plan qui répond à la requête, Le service des requêtes dans ce cas s'effectue également pour une requête après une.

La discipline de service est FIFO (First In First Out) pour tous les serveurs.

La méthode que l'on développera dans cette étude est basée sur les réseaux de files d'attente. En effet, le système de composition avec le graphe est un système que l'on peut représenter par un ensemble de demandeurs (qui représentent les requêtes et les Web services) d'un service. Les ressources (le traducteur de WSDL aux OWL-S, calcul du graphe, extracteur des entrées et sorties des requêtes, composeur, sélection) permettant de répondre à cette demande étant en nombre limité, les demandeurs vont devoir attendre la libération d'une ressource ce qui va provoquer des attentes devant ces ressources. Par conséquent, il apparaît naturel de modéliser notre système par un réseau de files d'attente (voir la figure suivante) :

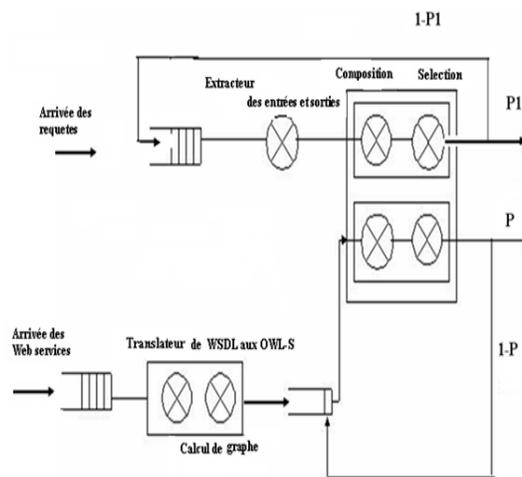


Figure 16.2. Modèle de composition des Web services par un graphe.

#### 16.4.2 Identification des événements influant sur l'état du système :

##### 1. Événement arrivée

On a deux types d'arrivée :

- **Arrivées des requêtes** : Une arrivée est générée selon le processus stochastique qui caractérise le phénomène des arrivées dans le système. L'arrivée ainsi obtenue est affectée au premier serveur, pour effectuer l'extraction des paramètres (entrés/sorties) de la requête. Ensuite au deuxième serveur pour la composition et

enfin la sélection. Le temps des services est alors généré selon la loi exponentielle. Dans le cas contraire, cette arrivée est placée dans la file d'attente.

- **Arrivés des Web services :** Une arrivée est générée selon le processus stochastique qui caractérise le phénomène des arrivées dans le système. L'arrivée ainsi obtenue est affectée par groupe au premier serveur à chaque temps de latence, pour effectuer la translation de WSDL aux OWL-S ensuite au deuxième serveur pour le calcul du graphe. Le temps des services est alors généré selon la loi exponentielle. Dans le cas contraire, cette arrivée est placée dans la file d'attente.

## 2. Événements fin de service

On distingue cinq cas (voir la figure1) :

- À la fin d'extraction, la requête sortante entre directement dans le second serveur sans attente.
- À la fin de la composition, la requête sortant rentre directement dans le second serveur sans attente.
- À la fin de la sélection, on aura la fin de service de la requête, donc les trois serveurs sont libérés. Si la requête n'est pas satisfaite, elle passera à la file d'attente de requêtes pour reprendre le service. La première requête se trouvant dans la file commence son service.
- A la fin de la translation, les Web services sortants entrent directement dans le second serveur sans attente.
- A la fin du calcul du graphe, on aura la fin de service des Web services, donc les deux serveurs sont libérés. On peut recevoir un autre groupe de Web services et le résultat de ce service sera en attente d'être utilisé lors de service des requêtes.

## 16.5 Calcul des performances

Chaque occurrence d'un événement correspond à un changement de l'état du système. De ce fait, un certain nombre de mesures sont effectuées dans le calcul des performances du système.

Les résultats de la simulation établis après avoir saisi les différents paramètres :

- Le temps maximum de simulation est :  $10000\mu s$ .
- L'intervalle de temps moyen séparant deux arrivées des requêtes consécutives  $1/\lambda_1$  est :  $1\mu s$ ,  $1\text{milli.s}$ ,  $1s$ ,  $1\text{minute}$ ,  $1\text{heure}$ .
- L'intervalle de temps moyen séparant deux arrivées des Web services consécutives  $1/\lambda_2$  est :  $1\mu s$ ,  $1\text{milli.s}$ ,  $1s$ ,  $1\text{minute}$ ,  $1\text{heure}$ .
- La durée moyenne du serveur de l'extraction  $\mu_1$  est :  $1\mu s$ ,  $1000\mu s$ ,  $10000\mu s$ ,  $100000\mu s$ ,  $1s$ .
- La durée moyenne du serveur de la composition  $\mu_2$  est :  $1\mu s$ ,  $1000\mu s$ ,  $10000\mu s$ ,  $100000\mu s$ ,  $1s$ .
- La durée moyenne du serveur de la sélection  $\mu_3$  est :  $1\mu s$ ,  $1000\mu s$ ,  $10000\mu s$ ,  $100000\mu s$ ,  $1s$ .
- La durée moyenne du serveur de la translation  $\mu_4$  est :  $1\mu s$ ,  $1000\mu s$ ,  $10000\mu s$ ,  $100000\mu s$ ,  $1s$ .
- La durée moyenne du serveur de calcul du graphe  $\mu_5$  est :  $1\mu s$ ,  $1000\mu s$ ,  $10000\mu s$ ,  $100000\mu s$ ,  $1s$ .

### 16.5.1 Énumération des résultats :

Les résultats de la simulation sont résumés dans le tableau suivant :

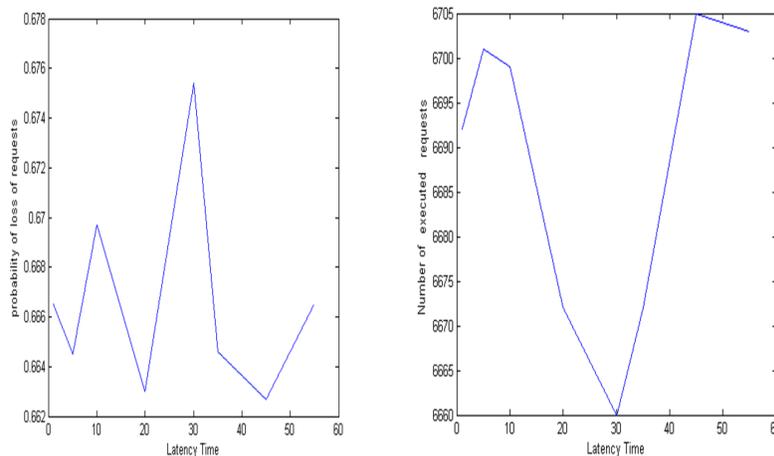
**Table 16.1.** Tableau des résultats d'évaluation de performances de composition des Web services avec la graphe.

## 16.6 Conclusion

Dans ce travail, nous avons évalué les performances d'un système de composition des Web services par un graphe.

Malgré les efforts de recherche et de développement autour de la problématique de la composition des services, elle reste une tâche hautement complexe et pose un certain nombre de défis. Sa complexité provient généralement des sources suivantes :

Paramètres de performances	Les mesures
• Taille du groupe de Web services	500000
• Temps de latence	30
• Débit des requêtes	0.3353
• Débit des Web services	0.6767
• Temps moyens de réponse des requêtes	8.7408xe7μs
• Temps moyen de services des requêtes	20834 μs
• Temps moyens de réponse des Web services	4.9668 xe5μs
• Temps moyen de services des Web services	698 μs
• Temps moyen d'exécution des requêtes	20834.2 μs
• Temps moyen d'exécution des Web services	698 μs
• Temps moyen d'extraction	9956.2 μs
• Temps moyen de composition	1008.2 μs
• Temps moyen de sélection	9869.8 μs
• Temps moyen de translation	311 μs
• Temps moyen de calcul du graphe	387 μs
• Nombre de requêtes satisfaites	1712 requêtes
• Nombre total des requêtes arrivées	10021 requêtes
• Nombre de requêtes exécutés	3471 requêtes



**Figure 16.3.** La probabilité de perte des requêtes et le nombre de requêtes exécutés en fonction du temps de latence.

- L'augmentation exponentielle du nombre des services web sur le web rend très difficile la recherche et la sélection des services web pouvant répondre à un besoin donné.
- Les services sont créés et mis à jour de façon hautement dynamique.

## Références

1. A. Aissani "Modèles stochastiques de la théorie de fiabilité", Office de publications Universitaires, Alger, 1992.
2. M. Ettl, G. E. Feigin "A Supply Network Model with Base- Stock Control and Service Requirements", Operations Research, vol. 48, n° 2, pp. 216-232, 2000.

3. H. Kadima. *"Les Web services"*, Edition Eyrolles, 2003.
4. A. Koubaa *"Introduction à l'évaluation de performances des systèmes informatiques et de communication"*, Rapport de recherche, University of Minnesota, [www.amazon.com/Performance Evaluation of Computer and Communication Systems/3540](http://www.amazon.com/Performance-Evaluation-of-Computer-and-Communication-Systems/3540), Janvier 2004.
5. B . Medjahed, A. Bouguettaya and A. K. Elmagarmid *"Composing Web services on the Semantic Web"*, The VLDB Journal, 12 (4), 2003.
6. H. Nacer, D. Aissani et N. Boudjlida *"Les Web services complexes"*, Edition europeenne, 2011.
7. C. E. Riddalls and S. Bennett *" The stability of supply chains"*, International Journal of Production Research, vol. 40, n° 2, pp. 459-475, 2002.