



THESIS / THÈSE

MASTER EN SCIENCES INFORMATIQUES

Etude Préalable à un Nouveau Type de Broker pour Application SaaS

Marchal, Gilles

Award date:
2014

Awarding institution:
Universite de Namur

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.


UNIVERSITÉ DE NAMUR
Faculté d'informatique
Année académique 2013-2014

Étude Préalable à un Nouveau Type de Broker
pour Applications SaaS

Gilles MARCHAL



Maître de stage : Mme Yuhong YAN, CSE, Concordia University (Montréal)

Promoteur :  (Signature pour approbation du dépôt - REE art. 40)
Mr Philippe THIRAN, Université de Namur

Mémoire présenté en vue de l'obtention du grade de
Master en Sciences Informatiques.

Résumé et mots-clés

Résumé

Dans le contexte du Cloud Computing, le rôle des “Cloud Service Brokers” prend une importance de plus en plus considérable.

Le but de ce travail est de procéder à une étude préalable à la réalisation d’un SaaS Broker en vue d’aider les utilisateurs de ce dernier à choisir le meilleur service en adéquation avec leurs besoins fonctionnels et non-fonctionnels. Ceci se concrétise par une description d’un modèle d’architecture haut-niveau pour un Broker de ce type réalisant l’Arbitrage de services et proposant une plate-forme d’évaluation des fournisseurs.

Ainsi, pour comprendre les différents concepts utilisés, nous établissons dans un premier temps un cadre théorique servant de référence. Ensuite, un État de l’Art présente une large vision du contenu de la littérature sur le domaine des Cloud Service Brokers. En outre, avant d’aborder la description de l’architecture du Broker proposé, l’ajout d’une présentation d’un modèle de qualité permet de combler le manque, constaté lors de ce travail, de méthodes disponibles pour l’évaluation et la comparaison de la qualité des services et de leurs fournisseurs.

Mots-clés : Cloud Computing, Cloud Services Broker, Brokerage, Software-as-a-Service, SaaS Broker, Arbitrage, Sélection de services, SMI, Fédération Cloud, État de l’Art

Abstract

In the Cloud Computing context, “Cloud Service Brokers” are getting more and more important. This work aims to proceed to a preliminary study on a SaaS Broker in way to help its users to choose the right service that best suit with its fonctionnal and non-fonctionnal business requirements.

This leads to the description of a high-level architectural model for an Arbitrage Broker proposing a platform for the providers evaluation. For a better understanding of the related concepts, we first introduce a theoretical chapter used as reference. Then, a State of the Art gives a large overview of the literature for the Cloud Service Brokers domain. Besides our described Broker, we propose a quality model which allows the evaluation and the comparison of services quality along with their providers, filling the lack in this topic.

Keywords : Cloud Computing, Cloud Services Broker, Software-as-a-Service, SaaS Broker, Brokerage, Arbitrage, Service Selection, SMI, Cloud Federation, State of the Art

Avant-Propos

Ce Mémoire est le résultat d'un travail de recherche que nous avons commencé lors de notre stage à l'Université Concordia de Montréal et que nous avons continué à l'Université de Namur. Ce stage a été une expérience gratifiante tant d'un point de vue académique que personnel. Quant au Mémoire, il nous a permis d'enrichir nos connaissances sur le domaine du Cloud Computing ainsi que de nous amener à investiguer sur un nouveau système informatique qui est aussi un Business Model.

Nous voudrions remercier les personnes qui ont, de près ou de loin, permis l'aboutissement de ce travail.

Tout d'abord, nous tenons à remercier Mr. Ph. Thiran, le promoteur de ce Mémoire, pour sa guidance et ses conseils méthodologiques qu'il nous a donné tout au long de l'élaboration de ce travail enrichissant ; de même que pour nous avoir donné l'opportunité de nous rendre en stage à Montréal, et ainsi découvrir une autre culture académique et un autre mode de vie dans un cadre chaleureux.

Nous voudrions également remercier notre maître de stage, Mme Y. Yan, et les étudiants de son laboratoire, Min, Jing, Ruijia, Kushal et Yifei, pour leur accueil chaleureux, pour leur sympathie et pour leur aide précieuse dans nos activités à l'Université Concordia ;

Ensuite, nos différents enseignants, professeurs et assistants de l'Université de Namur qui auront contribué à notre formation, ainsi qu'à nous forger un esprit critique et inculquer une culture multidisciplinaire ;

Et enfin, notre famille et nos amis qui nous ont toujours soutenus tout au long de nos études.

Table des matières

Résumé et Mots-clés	3
Avant-Propos	5
Table des matières	9
Glossaire	11
Introduction	15
1 Cadre Théorique	19
1.1 Qu'est-ce que le paradigme du Cloud Computing?	19
1.2 Clarification des concepts du paradigme du Cloud Computing	23
1.2.1 Les services du Cloud Computing	23
1.2.2 Les acteurs du Cloud Computing	25
1.3 Cloud Computing – Que retenir ?	26
1.4 Cloud Service Brokers et Cloud Brokerage	29
1.4.1 Pourquoi utiliser des Brokers ?	29
1.4.2 Revue des caractéristiques des Cloud Service Brokers	32
1.4.3 Proposition d'une définition pour le concept de Cloud Service Broker	34
1.5 Qualité de Service et Service Level Agreement	39
1.6 Interopérabilité et Architectures Inter-Cloud	41
1.6.1 Interopérabilité entre Clouds – Cloud Interoperability	41
1.6.2 Inter-Cloud : Multi-Cloud & Cloud Federation	42
2 État de l'art des Cloud Service Brokers	47
2.1 Méthodologie	47
2.2 Principaux Cloud Service Brokers	48
2.2.1 Cloud Service Broker for SLA-based SaaS, E. Badidi	50
2.2.2 A Services Brokerage and Business Model for SaaS, Moore & Mahmoud	52
2.2.3 OPTIMIS, EU-FP7	54
2.2.4 SLA@SOI, EU-FP7	57
2.2.5 RESERVOIR, EU-FP7	58
2.2.6 mOSAIC, EU-FP7	60
2.2.7 STRATOS, Canada	62
2.2.8 InterCloud, Buyya et al.	64
2.2.9 Contrail, EU FP7	65

2.2.10	Federated Cloud Management : an Architecture for Integrating IaaS Cloud Systems, Marosi et al.	66
2.2.11	Cloud service delivery across multiple cloud platforms, Houdi et al.	66
2.2.12	A brokerage-based approach for cloud service selection – CSP-index, Sundareswaran et al.	67
2.2.13	Cloud service recommendation and selection for enterprises, Yan et al.	68
2.2.14	CloudCmp, Li et al.	70
2.2.15	SMICloud, Garg et al.	73
2.2.16	CloudAdvisor : A Recommendation-as-a-Service Platform for Configuration and Pricing, Jung et al.	76
2.2.17	Cloud Service Selection Based on the Aggregation of User Feedback and Quantitative Performance Assessment, Qu et al.	76
2.2.18	Cloud Model for Service Selection, Wang et al.	78
2.3	Comparaison des Cloud Service Brokers présentés	79
2.4	Quelques Brokers commerciaux	83
2.5	Possibilités de recherche	84
3	Vers un nouveau SaaS Broker	85
3.1	Étude préalable pour un SaaS Broker	85
3.1.1	Pourquoi un nouveau SaaS Broker ?	85
3.1.2	Quels sont nos objectifs dans l'étude d'un nouveau Broker ?	86
3.2	Étude des propriétés clés dans la sélection de SaaS	87
3.3	Étude d'un Modèle de Qualité pour SaaS	91
3.3.1	Concepts	91
3.3.2	Paramètres du Modèle de Qualité	92
3.4	Objectifs et Fonctionnalités du SaaS Broker	97
3.4.1	Quels sont les objectifs du SaaS Broker ?	97
3.4.2	Quelles sont les fonctionnalités du SaaS Broker ?	98
3.5	Modèle d'Architecture du SaaS Broker	100
3.5.1	Hypothèses sur le système	100
3.5.2	Acteurs du système	101
3.5.3	Composants haut-niveau du Broker	104
3.5.4	Illustrations du modèle	105
4	Discussion	109
4.1	Avantages et différenciation de notre Broker	109
4.2	Étendue de la recherche	109
4.3	Perspectives de développements ultérieurs	110
4.3.1	Comment implémenter l'architecture du Broker ?	110
4.3.2	Comment évaluer le modèle de Broker proposé ?	110
	Conclusion	113
A	Revue des définitions pour le paradigme du Cloud Computing	115
B	Revue des définitions pour le concept de Cloud Service Broker	119
C	CSMIC Service Measurement Index Framework	121

D Exemples d'interfaces	127
Liste des figures	130
Liste des tableaux	131
Bibliographie	137

Glossaire

- **Agrégation** : Combine et intègre plusieurs services de différents fournisseurs, dans le but de les rassembler en un nouveau service à mettre à disposition de l'utilisateur.
- **Analytic Hierarchy Process (AHP)** : Technique structurée, basée sur les mathématiques et la psychologie, pour organiser et analyser des décisions complexes. Cette méthode de résolution de problèmes multi-critères (MCDM) décompose d'abord la problématique de décision en une hiérarchie de sous-problèmes. Le solveur évalue ensuite l'importance relative de ces divers éléments par des paires de comparaisons. L'AHP convertit les résultats en valeurs numériques (des poids ou des priorités), qui sont employées afin de calculer un score pour chaque alternative. Un index de cohérence mesure à quel point le décideur a été cohérent dans ses réponses.
- **Application Programming Interface (API)** : Composant logiciel rassemblant un ensemble normalisé de classes, de méthodes ou de fonctions qui sert de façade par laquelle il offre des services à d'autres logiciels. Les opérations, inputs, outputs ainsi que leur type sont indépendants des implémentations. L'API est offerte par une bibliothèque logicielle ou un service web, le plus souvent accompagné d'une description qui spécifie comment des programmes "consommateurs" peuvent se servir des fonctionnalités du programme "fournisseur".
- **Arbitrage** : Offre un service de credit-scoring des différents fournisseurs et permet ainsi la sélection du fournisseur le mieux côté.
- **Benchmark** : Point de référence servant à effectuer une mesure et permettant d'évaluer les performances d'un système dans le but de le comparer à d'autres.
- **Brokerage** : Opérations, traitements et services offerts par un Cloud Services Brokers. Les principaux rôles de Brokerage sont l'intermédiation, l'agrégation et l'arbitrage.
- **Charge de travail – Workload** : Quantité de travail réalisée par une entité en une période de temps donnée. Cette charge de travail donne une estimation de l'efficience et de la performance de cette entité. Capacité à gérer et effectuer une tâche.
- **Cloud Broker** : Broker de services entre des Cloud Providers et des SaaS Providers. Ces derniers déploient et exécutent leurs applications sur ces Clouds.
- **Cloud Computing** : Paradigme et modèle de distribution de ressources informatiques sous forme de services qui comprend les Infrastructures Cloud et les Software-as-a-Service. Le Cloud Computing rassemble les fournisseurs d'Infrastructures Cloud (Cloud Providers) et les fournisseurs d'applications logicielles (SaaS Providers). Les utilisateurs des SaaS (SaaS Users) ou les utilisateurs finaux (End-Users) accèdent à ces applications SaaS par l'Internet.

- **Cloud Service Broker (CSB)** : Business Model et rôle d'intermédiaire IT entre des Service Providers (fournisseurs) et des Services Users (utilisateurs).
- **Cloud Service Measurement Initiative Consortium (CSMIC)** : Consortium initié par l'Université Carnegie-Mellon Silicon Valley (Pittsburgh) pour satisfaire le besoin de mesures, globalement admises à l'échelle industrielle, pour calculer les avantages et les risques des services du Cloud Computing. Le résultat principal des travaux du consortium est le SMI Framework.
- **Constraint Satisfaction Problem (CSP)** : Un problème de satisfaction par contrainte est défini parce qu'il existe un ensemble de variables et que chacune d'elle possède un nombre fini de valeurs possibles appelées son domaine. Un CSP est alors défini par une quantité de contraintes limitant les valeurs que les variables peuvent prendre.
- **Customer Relationship Management (CRM)** : Progiciel regroupant un ensemble d'outils et de techniques au niveau de l'organisation, l'automatisation et la synchronisation des ventes, du marketing, du service à la clientèle et du support technique. Il vise à proposer des solutions permettant les interactions entre l'entreprise et ses clients actuels ou futurs facilitant les relations avec sa clientèle par automatisation des différentes composantes de celle-ci.
- **Elasticity – Elasticité** : Capacité d'un système à s'adapter à la charge de travail en augmentant ou diminuant l'approvisionnement en ressources de manière automatique et la plus efficiente possible. Capacité d'un fournisseur à ajuster la consommation de ses ressources de façon rapide et automatique pour satisfaire la demande des clients.
- **Enterprise Resource Planning (ERP)** : Progiciel de Gestion Intégrée qui donne une vue sur l'ensemble des processus/applications de gestion et de business d'une société. Il est défini comme étant l'interconnexion et l'intégration de toutes les applications de l'entreprise dans un système informatique centralisé.
- **Fédération Cloud** : Coopération entre un ensemble de fournisseurs qui interconnectent volontairement leurs ressources afin de permettre la combinaison des données et des infrastructures entre eux.
- **Feedback** – Évaluation par un SaaS User d'un service délivré par un fournisseur et sur sa confiance envers celui-ci.
- **Fuzzy Simple Additive Weighted Method** : Méthode utilisée pour la résolution de problèmes comprenant des paramètres accompagnés d'un poids de préférence.
- **Gartner, Inc.** : Société américaine de conseil et de recherche dans le domaine des technologies de l'information. Gartner fournit ses études aux leaders mondiaux afin de prendre des initiatives déterminantes et "informées".
- **Human Resource Management (HRM)** : Système pour maximiser la performance des employés en service ou des objectifs stratégiques de leur employeur.
- **Inter-Cloud** : L'accès à un Inter-Cloud permet de facilement diversifier et de répartir les services Clouds et SaaS parmi différentes offres de multiples fournisseurs.
- **Intermédiation** : Opérations et services offerts par le Broker, pour le compte des utilisateurs, qui améliorent la qualité et les fonctionnalités des services de base tels que proposés par les fournisseurs, et fournissent des fonctionnalités supplémentaires à valeur ajoutée.

- **ITIL** : Formellement connu sous le nom de “Information Technology Infrastructure Library” (Bibliothèque d’Infrastructure de Technologie de l’Information), il constitue l’approche la plus largement admise dans la gestion de services informatiques à travers le monde. ITIL fournit un ensemble cohésif des pratiques, tiré des secteurs publics et privés à un niveau international.
- **Key Performance Indicators (KPIs)** : Indicateurs Clés de Performance ou “métriques” utilisés en vue d’appréhender les facteurs cruciaux au succès d’une organisation.
- **Multiple-Criteria Decision-Making (MCDM)** : Prise de décision impliquant de multiples critères. Il s’agit d’une sous-discipline de la recherche opérationnelle.
- **Mixed Integer Programming (MIP)** – “Problème de programmation en nombres entiers” ou “problème d’optimisation linéaire” : problème/programme d’optimisation mathématique ou de faisabilité dans lequel une partie ou toutes les variables sont limitées à des nombres entiers. La fonction que l’on veut minimiser/optimiser et ses contraintes peuvent donc être décrites par des fonctions linéaires. Il s’agit de problèmes NP-complets.
- **National Institute of Standards and Technology (NIST)** : Agence Fédérale pour les Technologies aux États-Unis. Elle travaille de pair avec l’industrie afin de développer et d’appliquer des technologies, métriques et standards.
- **OCCI** – L’Open Cloud Computing Interface (OCCI) est un protocole et une API. Il a été lancé pour créer une API de gestion à distance de services du Cloud Computing, tenant compte du développement d’outils interopérables pour des tâches comprenant le déploiement, l’adaptation (scalability) et la surveillance.
- **Open Grid Forum (OGF)** : Communauté ouverte globale consacrée à conduire l’évolution et l’adoption rapide de systèmes avancés distribués (e.g. le Cloud Computing).
- **Organisation** : Ensemble de moyens structurés ayant des frontières identifiables fonctionnant en continu, en vue d’atteindre un ensemble d’objectifs partagés par ses membres (salariés, dirigeants, actionnaires).
- **Paramètre Qualitatif** : Propriétés subjectives concernant la qualité d’expérience d’utilisation d’un service et de la satisfaction de l’utilisateur. L’évaluation de ces paramètres est collectée à l’aide de feedbacks et d’évaluation sur les services.
- **Paramètre Quantitatif** : Propriétés techniques de qualité de service (QoS) mesurées et suivies par des méthodes ou des outils de monitoring.
- **Pay-per-use / Pay-as-you-go** : Se dit d’un service du Cloud Computing dont la facturation est établie en fonction de son utilisation réelle (e.g. chaque fois que le logiciel est utilisé) et/ou de son utilisation pour une durée déterminée.
- **Quality of Service (QoS)** – Qualité de Service : Performance générale d’un réseau ou d’une application perçue par les utilisateurs. Elle permet de mesurer quantitativement le niveau de performance réalisé. Elle est calculée à l’aide de métriques et permet de garantir aux utilisateurs un niveau minimum de performance.
- **Resources Acquisition Decision (RAD)** : Résolution de problèmes concernant la prise de décision concernant l’allocation des ressources informatiques. Il est lié à un problème d’optimisation multi-critères (MCDM).

- **SaaS Broker** : Broker de services entre les SaaS Providers et les SaaS Users. Ils peuvent notamment assister le SaaS User dans la sélection, l’intégration, l’interopérabilité des services et réalisent une gestion automatisée des SLAs.
- **Scalability – Scalabilité** : Capacité d’un service à s’adapter à un changement d’ordre de grandeur de la demande (montée en charge de travail), en particulier sa capacité à maintenir ses fonctionnalités et ses performances en cas de forte demande. La scalability peut faire référence à la capacité d’un système à accroître sa capacité de calcul sous une charge accrue quand des ressources (généralement matérielles) sont ajoutées. Capacité d’augmenter ou de diminuer une quantité de service offerte pour respecter les besoins d’un client.
- **Service Level Agreement (SLA)** : Document formel et légal qui est négocié entre fournisseur et utilisateur qui décrit les conditions du service presté et le niveau de performance des opérations (QoS) qui sera offert et maintenu.
- **Service Measurement Index (SMI)** : Index de mesures des services du Cloud Computing développé par le CSMIC. Cet index est basé sur un ensemble de caractéristiques critiques (de business et techniques), d’attributs associés, et de mesures (KPIs) indépendantes à partir des quelles il fournit aux entreprises une méthode normalisée afin de mesurer et de comparer un service du Cloud Computing.
- **Service-Oriented Architecture (SOA)** – Architecture Orientée Service : Modèle d’architecture logicielle d’intermédiation qui est un modèle d’interaction applicative mettant en œuvre des composants logiciels avec une “forte cohérence interne” (par l’utilisation d’un format d’échange pivot, le plus souvent XML) et des “couplages externes lâches” (par l’utilisation d’une couche d’interface interopérable, le plus souvent un service web). Elle est indépendante de tout fournisseur, produit ou technologie.
- **Service-Oriented Infrastructure (SOI)** – Architecture Orientée Infrastructure : Modèle d’architecture qui fournit un système pour décrire une infrastructure des Technologies de l’Information en tant que service.
- **Single Sign-On (SSO)** : Propriété et mécanisme de contrôle d’accès qui permet à un utilisateur d’accéder à plusieurs ressources et applications informatiques, associées mais indépendantes et auxquelles il est autorisé d’accéder, en ne procédant qu’à une seule authentification.
- **Total Cost of Ownership (TCO)** : Coût global de possession : évaluation financière prévue pour aider les organisations à déterminer les frais directs ou indirects d’un produit ou d’un système. En informatique, le TCO tente de mesurer l’impact financier induit lors du déploiement d’un produit IT et au cours de son cycle de vie.

Introduction

Depuis quelques années, la dynamique de l'économie et de l'environnement IT entraîne une compétitivité accrue entre les entreprises. Cette compétitivité conduit le management de ces entreprises à acquérir des infrastructures et des applications informatiques toujours plus performantes, notamment des solutions logicielles, en adéquation avec leurs besoins stratégiques afin d'améliorer leurs activités business et de réduire leur Total Cost of Ownership (coût global de possession).

Ainsi, l'offre de services des Infrastructures Clouds (infrastructures, ressources informatiques fondamentales) et des applications logicielles Software-as-a-Service (SaaS) ne cesse de croître avec l'émergence du Cloud Computing. Ce terme "Cloud Computing", qui est souvent usité dans le langage courant pour signifier une manière de délivrer des services informatiques par l'Internet, apparaît comme un paradigme clé dans le domaine technologique.

Plus précisément, il s'agit d'un "style d'informatique dans lequel des capacités IT adaptables sont délivrées en tant que services à des clients externes utilisant les technologies de l'Internet par lesquelles les services sont suivis à l'aide de métriques définies et convenues avec chacun des utilisateurs afin de permettre des modèles de déploiement multiples" (Gartner). Certains auteurs (e.a. R. Buyya) ajoutent que le Cloud Computing vise à fournir des services "à la demande" en tant que commodité ("Utility", comme l'eau, l'électricité et le gaz) suivant le principe d'une tarification à l'usage ("pay-per-use").

Cet environnement, qui rassemble donc des consommateurs-utilisateurs de services d'une part et des fournisseurs de services d'autre part, devient populaire dans le monde industriel, et de plus en plus d'entreprises rejoignent ce système pour adopter le Cloud Computing. En effet, nous observons une croissance toujours plus forte d'années en années du nombre d'entreprises qui se tournent vers les services du Cloud Computing, et une multiplication du nombre de fournisseurs proposant leurs offres qui se différencient toutes les unes par rapport aux autres tant par leurs fonctionnalités que par la qualité de service assurée.

Face à cette multitude de fournisseurs et les nombreuses offres de services disponibles, la démarche des utilisateurs dans le choix d'un service est un processus qui pourrait se révéler fastidieux, car l'utilisateur a tendance à vouloir obtenir le service qui correspond le mieux à sa stratégie IT, c'est-à-dire, le service avec les fonctionnalités désirées, les niveaux de qualité de service attendus et certaines propriétés de conformité ou répondant aux exigences légales. Ce sont ces mêmes conditions de prestation de services qui sont connues dans l'utilisation des Service Level Agreements. L'acquisition de services du Cloud Computing n'est également pas sans problème au niveau de leur intégration et interopérabilité avec les infrastructures et les applications IT déjà en place chez l'utilisateur. Bien souvent, l'intégration requiert une adaptation due au manque de standardisation. Ce même manque de standardisation

pousse les utilisateurs à se cantonner chez un fournisseur unique alors que les offres d'autres fournisseurs pourraient mieux leur convenir.

Dans ce contexte, nous pouvons y voir l'intérêt d'une plate-forme ou d'un environnement proposant des mécanismes d'intégration et d'interopérabilité entre les différents systèmes de multiples acteurs afin d'interagir entre eux. Une telle plate-forme peut être installée par le fournisseur lui-même ou introduite par un third party jouant le rôle d'intermédiaire entre des utilisateurs et des fournisseurs de services. Cet agent, ou système intermédiaire qui peut être automatisé, est appelé Broker. Il réalise, pour les utilisateurs, diverses opérations d'intermédiation sur les services proposés par les fournisseurs.

Le présent travail s'intéressera à ces intermédiaires entre acteurs du Cloud Computing, qui sont appelés "Cloud Service Brokers" ou "Brokers de services". Un Cloud Service Broker (CSB) rend plus faciles, plus sécurisés et plus productifs pour les entreprises, les opérations de recherche, d'intégration, d'utilisation, d'adaptabilité et de maintenance des services délivrés dans un environnement de Cloud Computing; et ce plus particulièrement lorsqu'elles souhaitent obtenir les services de plusieurs fournisseurs. D'ici 2015 et selon Gartner, au moins 20% de tous les services du Cloud Computing seront obtenus via des Brokers plutôt que directement auprès des fournisseurs, contre moins de 5% à l'heure actuelle.

Dans ce travail, nous aborderons une étude sur les Cloud Service Brokers en tant qu'intermédiaire entre les fournisseurs de SaaS et utilisateurs de SaaS. Ainsi, nous débuterons celui-ci en nous demandant comment pouvons-nous décrire plus précisément la conception du Cloud Computing, avec un SaaS Broker comme intermédiaire entre ces deux acteurs. Il s'en suivra une étude de l'état actuel des recherches dans ce domaine. Plus précisément, nous nous concentrerons sur le rôle de la sélection de SaaS qu'un Broker peut effectuer suite à une demande d'un utilisateur précisant les fonctionnalités et les qualités de services attendues.

En effet, vu que l'IT prend une part considérable dans les activités des organisations ou des entreprises, et que les choix des solutions IT peuvent être déterminants pour l'efficacité de leurs activités business, nous pouvons nous demander comment évaluer ces applications SaaS pour proposer, à l'utilisateur, le service le plus pertinent envers ses attentes et ses exigences. Nous nous interrogerons donc sur les spécificités des applications logicielles délivrées par l'Internet. Ainsi, sur base de ces spécificités, nous effectuerons une recherche sur les méthodes d'évaluation des SaaS et de leurs fournisseurs. Cela nous amènera à étudier quels modèle et architecture nous pouvons mettre en place afin de pouvoir évaluer et comparer ces services; et de surcroît, assister l'utilisateur dans son choix. Nous nous demanderons également comment nous pouvons créer un environnement de confiance entre fournisseurs et utilisateurs, de plus basés sur la satisfaction de ces derniers.

Nous tenterons d'apporter des réponses à ces questions tout au long de ce travail, dont l'objectif principal est de proposer un modèle de sélection et d'évaluation des services et des fournisseurs. Typiquement, ce modèle devrait permettre de mesurer la confiance d'un utilisateur envers un fournisseur à partir d'expériences personnelles et de celles de ses pairs. Cette évaluation peut être réalisée sur base de paramètres objectifs (monitoring) et sur base de paramètres subjectifs (feedbacks) collectés auprès des utilisateurs.

Le présent Mémoire s'articule autour d'une large revue de la littérature concernant le Cloud Computing et les Cloud Services Brokers. Dans un premier temps, ces différents travaux nous ont permis d'établir un cadre théorique (*Chapitre 1*). Celui-ci exposera les différents concepts et les différentes notions que nous utiliserons tout au long de notre travail d'étude vers la proposition d'un nouveau Broker. Ce cadre théorique présentera notamment les concepts du Cloud Computing, de Cloud Service Brokers, de Service Level Agreement et des environnements Inter-Cloud.

Ensuite, nous établirons, au *Chapitre 2*, un état de l'art le plus complet possible, bien que non exhaustif, résultant de notre analyse des différents travaux parcourus lors de notre revue de la littérature. Cet état des connaissances actuelles met en lumière les différentes recherches et les différents projets existants dans le domaine des Cloud Services Brokers, ainsi que des méthodes de sélection et de comparaison de services. Nous citerons également quelques systèmes commerciaux.

Cette analyse de l'existant nous permettra de prendre conscience des limites et des lacunes, particulièrement celles concernant les Brokers d'applications SaaS.

Ceci a porté notre réflexion sur la proposition d'un nouveau type de Broker pour applications SaaS, accompagné d'une méthode de sélection de ces applications et d'un modèle de qualité pour évaluer les services obtenus auprès de fournisseurs SaaS dans le *Chapitre 3*. Tout cela amenant à la description haut-niveau d'un modèle d'architecture pour un SaaS Broker.

Dans le dernier chapitre, le *Chapitre 4*, nous proposerons quelques réflexions quant à cette étude préalable. Nous donnerons des indications et des pistes d'orientation pour la continuité de cette investigation à la réalisation d'un SaaS Broker en vue d'une sélection appropriée de SaaS et d'évaluer ceux-ci.

Ce Mémoire est principalement un travail théorique axé autour d'une large revue de la littérature. Le travail se pose avant tout en tant que première recherche dans l'étude d'un nouveau type de Broker pour applications SaaS. Il veut proposer un mécanisme théorique de haut niveau pour l'évaluation et la sélection de services tout en permettant un réseau de confiance basé sur une communauté d'évaluation de la satisfaction.

La *Bibliographie* de ce travail pourra, en outre, servir au lecteur de référence sur le sujet en proposant une compilation de la plupart des travaux relatifs à ce domaine.

Chapitre 1

Cadre Théorique

Le Cloud Computing est un paradigme informatique récent dont il existe une variété de visions et de définitions. De même, il existe plusieurs concepts et termes utilisés dans la littérature relatives à ce sujet. Ces variétés de notions peuvent rendre les intéressés confus ou amener à une compréhension malaisée ou erronée des concepts.

Le but de ce chapitre est donc d'une part de clarifier les concepts en la matière, et d'autre part d'apporter un cadre théorique définissant d'autres concepts sur lesquels nous nous reposerons dans la suite de ce travail.

Ainsi, nous présenterons dans un premier temps le paradigme du Cloud Computing. Nous introduirons ensuite le concept des Cloud Service Brokers comme nouveaux acteurs dans ce paradigme ainsi que les notions qui leurs sont relatives telles que le Brokerage de services, la sélection de services et les Service Level Agreements. Nous exposerons finalement l'interopérabilité et les architectures Inter-Cloud.

1.1 Qu'est-ce que le paradigme du Cloud Computing ?

Le terme “*Cloud Computing*” ou “*Cloud*” est souvent usité comme “buzzword”¹ à des fins de marketing et dans le jargon technologique (e.g. termes d'accroche pour les nouveaux produits et nouvelles solutions des grandes entreprises IT, sujets de nombreux articles de sites relatifs au monde technologique, etc.).

Jusqu'il y a peu, le terme était utilisé comme maître mot ou qualificatif d'une nouvelle technologie et méthode de stockage des données à distance et de leur accès via l'Internet depuis n'importe quel endroit (e.g. Dropbox, Google Drive). Il est aussi utilisé pour la location à la demande de machines virtuelles et serveurs hébergés dans des datacenters externalisés (e.g. Amazon) afin d'effectuer les activités informatiques d'une organisation. À présent, ce concept ou paradigme prend de l'ampleur et se retrouve souvent au centre de discussions relatives aux craintes liées à l'externalisation des données sensibles. Une compréhension habituelle du terme “Cloud Computing” peut signifier “les services informatiques délivrés par l'Internet”. Cependant, cette définition est bien trop vague et simpliste, et donc il est légitime de se demander ce qu'est réellement le Cloud Computing d'un point de vue scientifique et technologique. Pour parvenir à une définition claire et complète du paradigme et de ses concepts, nous avons parcouru de nombreux travaux relatifs à ce sujet.

1. “Mot à la mode”

Au cours de nos recherches, nous avons été confrontés à une variété de définitions exprimant et définissant le paradigme du Cloud Computing, son fonctionnement, ses concepts et son environnement.

Hormis la définition/vision de Berkeley (Armbrust et al. [1, 2]) et celle des travaux du National Institute of Standards and Technology (NIST, [3, 4, 5]) principalement utilisées comme référence(s), la littérature scientifique et l'industrie emploient des définitions multiples et variées dont l'angle de vue peut largement différer.

Le fait que nous ayons été confrontés à cette multitude de définitions, pas assez complètes et précises à notre convenance pour expliquer le paradigme du Cloud Computing, nous a conduit à répertorier quelques-unes d'entre elles afin de construire notre propre définition de ce paradigme dans le cadre de notre étude. Parmi celles-ci, remarquons entre autres la définition de Buyya et al. (CLOUDS Laboratory de Melbourne [6]), celle des groupes d'étude Forrester et Gartner, celle de Zhang et al.[7], celle de Vaquero et al.(HP Labs, [8]), ainsi que celle du NIST (Mell et Grance [3]). Le lecteur intéressé trouvera l'exposé de ces différentes définitions à l'Annexe A.

Afin de développer notre propre définition, nous utiliserons comme référence principale la vision et la définition de Berkeley [2]. Celle-ci s'affirme comme étant l'une des références les plus citées dans la littérature² et présente la vision qui nous semble la plus pertinente comme base pour développer la notion de Cloud Service Broker.

Pour ces auteurs, le *Cloud Computing* fait référence à deux types de services : d'un côté aux applications délivrées comme services (as-a-Service) sur l'Internet, et d'un autre côté au Hardware et System Software³ des datacenters qui fournissent les services d'infrastructures pour le déploiement des applications.

Ces services de livraison d'applications sont dénommés Software-as-a-Service (SaaS) qui est le terme utilisé dans le domaine depuis une dizaine d'années pour les décrire.

Les auteurs mentionnent aussi les termes Infrastructure-as-a-Service (IaaS) et Platform-as-a-Service (PaaS) comme mentionnés par le NIST. Le premier est utilisé pour décrire les services relatifs à la mise à disposition d'infrastructures informatiques, et le second pour décrire les plate-formes de déploiement d'applications logicielles. Cependant, ces auteurs nous préviennent qu'il est préférable de ne pas utiliser ces termes IaaS et PaaS car leur définition varie dans la littérature et diffère selon le fournisseur. En outre, la frontière entre les deux aspects reste assez floue. Ces deux types de services sont fortement liés, et bien souvent ils seront fournis ensemble.

Les datacenters pour le Hardware et le System Software forment la partie "*Infrastructure Cloud*" du paradigme tandis que les services d'infrastructures proposés et vendus sont nommés par le terme d' "*Utility Computing*".

La partie "*Cloud*" est rendue disponible au public général sous un modèle à la demande et payant à l'utilisation appelé "on-demand and pay-per-use model". Ce modèle de Cloud porte le nom de "*Public Cloud*".

2. 7000+ citations selon Google Scholar, 2000+ pour celle du NIST

3. Logiciels conçus pour effectuer et contrôler la partie matérielle (hardware) informatique et fournir une plate-forme d'exécution des programmes d'applications. Il existe différents types de Logiciels Système : les systèmes d'exploitation (OS), les programmes utilitaires (outils de monitoring du système, antivirus, gestionnaires de fichiers, etc.), les bibliothèques et les programmes de transformation de codes sources (assembleur, interpréteur, compilateur) [source : wikibooks.org]

Le terme “*Private Cloud*” est utilisé pour faire référence aux datacenters internes à une organisation et qui ne sont pas disponibles au grand public. De tels Private Clouds sont conçus de façon à bénéficier des avantages du Cloud Computing.

L’avantage du Cloud Computing, nous disent ces auteurs, est de permettre aux fournisseurs de services d’applications et à leurs utilisateurs d’avoir à portée de main l’installation et la maintenance logicielle ainsi qu’un contrôle de performance centralisé. Les utilisateurs finaux peuvent accéder à leurs services “n’importe où et n’importe quand”, partager leurs données et collaborer plus facilement tout en conservant leurs données en sécurité dans une infrastructure décentralisée et gérée par des professionnels. Le Cloud Computing donne aux fournisseurs d’applications la possibilité de déployer leurs services applicatifs en tant que Software-as-a-Service sans devoir faire l’acquisition de leur propre datacenter ou de “racks”⁴ de serveurs. De plus, les utilisateurs de ces Software-as-a-Service peuvent donc à présent déléguer les problèmes survenant au niveau logiciel à leur fournisseur d’application(s) logicielle(s), de même que ces derniers peuvent se reposer sur leur fournisseur d’Infrastructure Cloud.

Nous noterons cependant que ce paradigme n’est pas sans inconvénients. En effet, parmi ceux-ci, nous pouvons citer l’exposition aux risques liés à la sécurité du réseau sur l’Internet et la violation de la confidentialité, à la dépendance envers la qualité du réseau et à la qualité des services offerts par les fournisseurs d’applications. Cette qualité de services est en corrélation avec les services de l’Utility Computing offerts par les fournisseurs d’Infrastructures Clouds. Les organisations doivent faire confiance aux tiers dans les garanties de l’utilisation de leurs données. Des questions juridiques peuvent aussi se poser sur la localisation des données et les lois en vigueur pouvant s’y appliquer.

4. Armoires métalliques pour brancher de multiples serveurs ou appareils réseau

Cette conception du Cloud Computing telle que proposée par Berkeley (Armbrust et al., 2009) peut être représentée par la Figure 1.1 et perçue de la façon suivante :

Le Cloud Computing est l'ensemble des Software-as-a-Service (applications logicielles) et de l'Utility Computing (services d'Infrastructures Clouds) délivrés via l'Internet ou un réseau, mais ne prenant pas en considération les datacenters de petites et moyennes tailles (Private Cloud), même si ceux-ci reposent sur une virtualisation utilisée pour leur gestion. Le Cloud Computing rassemble les fournisseurs de Cloud (Cloud Providers) et les fournisseurs SaaS (SaaS Providers – aussi appelés Cloud Users –). Les utilisateurs de SaaS ou les utilisateurs finaux (SaaS Users/End-Users) utilisent les services applicatifs (applications SaaS) offerts par les fournisseurs SaaS (SaaS Providers).

Cloud Computing =
(SaaS délivrés par SaaS Providers)
+
(Utility Computing délivrés par Cloud Providers, sauf Private Cloud)

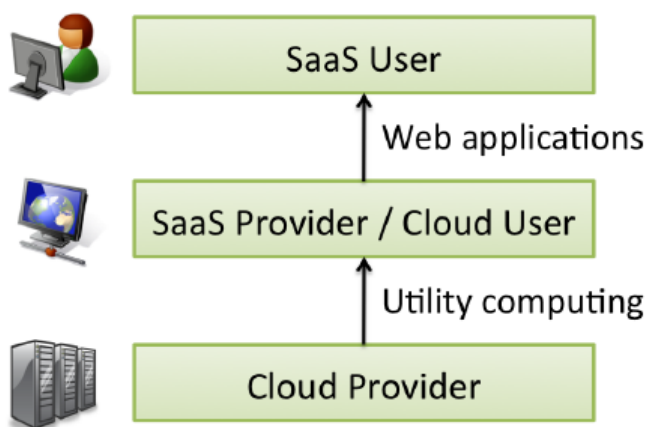


FIGURE 1.1 – Paradigme du Cloud Computing, vision de Berkeley [1].

1.2 Clarification des concepts du paradigme du Cloud Computing

Afin de clarifier et de transmettre notre définition pour les différents concepts utilisés par le Cloud Computing, nous allons détailler chacun d'entre eux ci-dessous.

1.2.1 Les services du Cloud Computing

Le concept de “*service*” est largement utilisé dans le domaine économique. Un service est un bien intangible qu'il faut différencier des produits ou des marchandises. C'est une prestation qui consiste en la mise à disposition d'une capacité technique ou intellectuelle. Un service du Cloud Computing est un ensemble de fonctionnalités et de capacités rendues disponibles à un groupe d'utilisateurs par un fournisseur dont le but est de concrétiser les objectifs du client-utilisateur.

Les services sont consommés dans le même temps nécessaire pour les produire. Pour cette raison, ils sont considérés comme n'étant pas stockables (Encyclopédie Larousse).

Les *services IT* font référence aux applications business permettant aux entreprises la création, la gestion, l'optimisation, l'accès aux informations et aux processus business. Ces services sont fournis par des organisations ou des technologies et peuvent être des services du Cloud Computing (Gartner).

Les services du Cloud Computing sont d'une part les services relatifs à l'infrastructure Cloud (hardware et system software) et d'autre part les applications logicielles (software, services web) offertes à travers le Cloud Computing.

L'entreprise cliente-utilisatrice n'a pas besoin d'acquérir et de déployer ses propres ressources, ni d'allouer une équipe IT à la gestion et la maintenance de l'infrastructure et des applications. Celles-ci sont gérées professionnellement par les fournisseurs respectifs. Ce type de services peut être adapté dynamiquement selon les besoins de l'entreprise utilisatrice.

Ces services s'adressant à un besoin business spécifique d'un utilisateur peuvent être simples mais également très complexes, c'est-à-dire qu'un service peut être une composition de multiples services qui visent à répondre conjointement à un besoin spécifique.

A. Services d'Infrastructure Cloud

Les Infrastructures Clouds proposées sous forme de services (Utility Computing) ont pour but d'offrir une solution adéquate selon les besoins en ressources informatiques requis par un utilisateur telles que les serveurs de stockage et de backup, les ressources de calcul (CPU, Mémoire), les machines virtuelles, les environnements et outils de développement, de déploiement et d'exécution d'applications (OS, plate-formes Infrastructure as a service), les infrastructures réseaux, et toute autres ressources informatiques fondamentales.

Ces services permettent aux utilisateurs le déploiement et l'exécution d'applications diverses (Stockage de données, Calcul, SaaS, Services Web, etc.) sur les infrastructures Cloud détenues et gérées par le fournisseur.

Cette solution permet aux utilisateurs de se décharger du contrôle et de la gestion des infrastructures tout en gardant la maîtrise des applications déployées et la configuration de leur environnement, de leurs paramètres et du stockage de leurs données.

De nombreux fournisseurs proposent différents types et solutions d'Infrastructure Cloud tels que Amazon EC2 et S3, VMWare, Citrix, Windows Azure, Rackspace, OVH, etc.

Les services d'Infrastructures Clouds proposés sont multiples : serveurs, infrastructures

réseaux, machines virtuelles (VMs), ressources informatiques fondamentales (CPU, Mémoire, OS, etc.), bases de données, outils de développement et de tests, intégration, déploiement d'application, backup et recovery, Content Delivery Networks, etc.

B. Software-as-a-Service (SaaS)

Les Software-as-a-Service (SaaS) suscitent de plus en plus d'intérêt dans les mondes industriel et académique car ils représentent une tendance du futur dans le Cloud Computing. Leur adoption ne cesse de s'accroître et évolue sur le marché des applications à destination des entreprises. Celles-ci y voient un grand intérêt car les restrictions des capitaux et des budgets demandent des alternatives plus souples et moins coûteuses mais tout aussi performantes (Gartner⁵).

En opposition aux logiciels traditionnels où les utilisateurs achetaient une licence pour un usage perpétuel ou défini dans le temps, les SaaS reposent sur le paradigme de la distribution de logiciel par l'Internet dont les organisations clientes-utilisatrices ne paient pas pour acquérir et détenir le logiciel mais paient pour l'utilisation de ce logiciel comme un service (à la demande) et dont la facturation est définie à l'usage et temporellement. Les SaaS Providers doivent maintenir et faire évoluer le logiciel, mais également assurer le support technique et l'assistance face aux problèmes potentiels.

Les utilisateurs "louent", "souscrivent à", "sont assignés" ou "ont les droits d'accès à" une application développée et déployée par le fournisseur sur une Infrastructure Cloud, et sont alors libérés de toute gestion complexe logicielle et matérielle. Les applications sont accédées depuis divers programmes clients tels qu'une interface d'un client léger (e.g. un navigateur Web) ou bien une API (Application Programming Interface).

Ce modèle d'accès aux logiciels réduit drastiquement les coûts. Les SaaS peuvent être déployés avec un minimum d'effort, être accessibles dans de très court délais par un large panel d'utilisateurs qui peuvent accéder aux ressources partagées sans créer des interférences entre eux. Cette approche d'hébergement centralisé des logiciels devrait rendre les mises à jour et le redéploiement transparent pour les utilisateurs.

La réduction du "Total Cost of Ownership", la rapidité de déploiement, la fiabilité, la sécurité des données, le disaster recovery, la mise à jour continue (upgrade) peuvent être cités comme avantages du modèle SaaS. Ce modèle permet aussi de réduire les coûts des supports up-front car il n'est plus nécessaire de supporter de multiples plate-formes et versions (IBM).

Les utilisateurs SaaS ont également la possibilité d'adapter dynamiquement la configuration du logiciel selon leurs besoins sans que cela exige des interventions complexes. La demande d'ajustement peut être réalisée par un interface pré-programmée.

Les applications SaaS sont construites sur une architecture "multi-tenant"⁶ qui est conçue pour supporter un large nombre de clients. Les SaaS vont automatiquement servir tous les clients sur base de l'architecture technique de l'application SaaS, ainsi le coût de chaque nouveau client est minimal. Cette architecture multi-tenant est établie sur une Infrastructure Cloud permettant ainsi aux SaaS d'être dynamiquement et automatiquement adaptable selon les besoins courants de l'utilisateur.

Des exemples typiques sont les applications de bureaux en ligne (e.g. Microsoft Office 365),

5. <http://www.gartner.com/newsroom/id/1406613>

6. Principe d'architecture logicielle permettant à un logiciel de servir plusieurs organisations clientes à partir d'une seule installation

les services de stockage (e.g. Dropbox), les services de messageries, et diverses applications Business telles les applications collaboratives, les ERP (Enterprise Resource Planning), les CRM (Customer Relationship Management), les HRM (Human Resource Management), les Content Management, et les logiciels “Sales”, “Financials”, “Billing”, etc.

La Figure A.2 à l’Annexe A montre quelques exemples de services du Cloud Computing.

Selon Armbrust et al., une définition large des SaaS peut comprendre tout logiciel à la demande, ce qui inclut également les logiciels qui s’exécutent localement mais dont l’utilisation est contrôlée et facturée à distance.

Nous différencions les SaaS des Services Web de la façon suivante : Un SaaS est une application (ou un ensemble de services) solide et mature qui fournit des applications logicielles sous forme d’utilité à des utilisateurs, alors qu’un Service Web a tendance à effectuer une fonctionnalité spécifique (e.g. exécuter une requête, effectuer un calcul, collecter des données) faisant souvent partie d’une suite plus large de services.

De plus, nous pourrions ajouter que les Services Web ne fournissent en général que les données produites (souvent aux format XML) par le service devront être ensuite utilisées par une application adaptée de manière à être visualisées, analysées, traitées, etc.

1.2.2 Les acteurs du Cloud Computing

Le paradigme du Cloud Computing rassemble fournisseurs et utilisateurs de services. Nous nommerons les fournisseurs de services du Cloud Computing les *Service Providers* et les utilisateurs de ces services les *Service Users*.

Dans la vision de Berkeley présentée ci-dessus, nous distinguons les trois types d’acteurs suivants : les fournisseurs d’Infrastructures Clouds, les fournisseurs de services logiciels, et les utilisateurs de ces services logiciels que nous nommerons respectivement “*Cloud Providers*”, “*SaaS Providers*” et “*SaaS Users*”.

Notons que les fournisseurs de services logiciels sont également utilisateurs d’une Infrastructure Cloud “*Cloud Users*”.

L’interaction entre ces trois types d’acteurs est montrée à la Figure 1.2.

Nous définissons plus précisément ces trois acteurs clés de la manière suivante :

A. Cloud Providers :

Nous appelons Cloud Providers, les organisations ou entités qui offrent des services d’infrastructures et de ressources informatiques – tels que réseaux, stockage et exécution de programmes système – qui prennent en charge la gestion des plate-formes Cloud pour l’exécution d’applications logicielles. Ils sont responsables de la mise à disposition de ces infrastructures physiques et de leurs logiciels système, et assurent leur disponibilité sous de bonnes performances.

Les Cloud Providers louent et mettent à disposition ces ressources suivant le principe du modèle de l’Utility Computing (on-demand et pay-per-use).

Les activités des Cloud Providers peuvent être classifiées en cinq domaines majeurs : le domaine de déploiement des services, celui de l’orchestration des services, celui de la gestion

de services Cloud, celui de la sécurité et celui de la privacy.

Les demandes des Cloud Users/SaaS Providers et les performances garanties doivent être satisfaites selon les accords négociés et décrits dans un Service Level Agreement (voir 1.5).

Nous considérons que le rôle de Cloud Providers regroupe les fournisseurs prenant en charge les IaaS et les PaaS distingués séparément dans d'autres travaux.

Amazon (EC2 et S3), Google (Apps Engine), Microsoft (Windows Azure), VMWare, Rackspace, GoGrid, IBM, Oracle, Fujitsu sont quelques exemples de Cloud Providers.

B. SaaS Providers / Cloud Users :

Les SaaS Providers sont des organisations ou des personnes entretenant une relation business avec un ou plusieurs Cloud Providers dans le cadre de l'utilisation de ressources clouds et de services d'Infrastructure Cloud tels que l'exécution de programmes, le stockage de données et les réseaux dans le but de développer, déployer et exécuter leurs services d'applications logicielles SaaS qu'ils fournissent à des organisations ou utilisateurs finaux (SaaS Users/End-Users). Ils sont donc à la fois des utilisateurs de Cloud (Cloud Users) approvisionnés chez un Cloud Provider et fournisseurs d'applications SaaS (SaaS Providers). De tels fournisseurs SaaS s'occupent de la gestion du déploiement, de la configuration, des mises à jour, de la maintenance et des accès – ce qui inclut sécurité, disponibilité et performance – de leurs applications logicielles rendues disponibles sur une Infrastructure Cloud. Et par là, ces fournisseurs s'assurent que les SaaS peuvent respecter les garanties de services attendues et requises par les utilisateurs (Service Level Agreement, voir 1.5).

Comme exemples de SaaS Providers, nous pouvons citer Amazon (AWS), Salesforce (CRM), Oracle (On-Demand CRM), SAP (ERP et CRM), Workday, Concur, Google Apps, etc. Les principales applications SaaS actuelles concernent des applications telles que les logiciels de collaboration, les ERP, les CRM, les HRM, etc.

C. SaaS Users/End-Users :

Les SaaS Users sont les organisations ou les personnes entretenant une relation business avec un ou plusieurs SaaS Providers afin d'acquérir des services logiciels ou applications SaaS hébergés par ces derniers et y accéder par l'Internet. Ces applications sont en général des suites de logiciels de bureautiques et de messageries mais aussi des solutions IT complètes de ERP, CRM, HRM, etc.

Les utilisateurs de SaaS peuvent être le personnel de l'organisation qui emploie le service, mais aussi les clients de cette organisation appelés utilisateurs finaux (End-Users) qui utilisent directement le SaaS.

1.3 Cloud Computing – Que retenir ?

Suite aux démarches et clarifications effectuées précédemment, nous proposons notre définition du Cloud Computing en guise de récapitulatif. Nous nous baserons sur celle-ci dans la suite de ce travail.

Cloud Computing

Nous admettons que le *Cloud Computing* est un modèle de distribution de ressources informatiques sous forme de *services*. Celui-ci remplace le modèle traditionnel de livraison sous forme de produits.

Nous considérons deux types de services mis à disposition par les fournisseurs (Service Providers) pour les utilisateurs (Service Users) : d'une part l'*Utility Computing*, et d'autre part les *Software-as-a-Service*.

L'*Utility Computing* représente les services d'infrastructures informatiques et ressources informatiques fondamentales proposés et vendus par l'Internet.

Ces services sont délivrés par les *Datacenters* qui mettent à disposition le Hardware (infrastructure matérielle et de ressources informatiques fondamentales) et les System Software (logiciel système tel que l'OS et les plate-formes nécessaires au déploiement des applications logicielles).

Ces Datacenters (Hardware et System Software, infrastructures et ressources) forment ce qu'on appelle l'*Infrastructure Cloud* ou "Cloud".

L'*Utility Computing* offre plusieurs caractéristiques qui rendent ce principe attrayant pour les fournisseurs de services logiciels :

- (a) Aucun investissement initial n'est nécessaire car l'*Utility Computing* repose sur le modèle de tarification à l'utilisation. Un fournisseur SaaS n'a donc besoin que de seulement louer les infrastructures et les ressources informatiques nécessaires auprès des fournisseurs de Cloud et selon ses besoins ;
- (b) Réduction du coût global de possession (TCO) car les ressources dans un environnement de Cloud Computing sont adaptables et flexibles suivant la charge de travail ("elasticity"). Elles peuvent être allouées ou désallouées rapidement selon que la demande de services augmente ou diminue. Il n'est donc pas nécessaire d'acquérir des ressources supplémentaires pour un pic de travail temporaire ;
- (c) Dynamiquement évolutif car les fournisseurs d'infrastructure Cloud rendent disponible et facilement accessible une grande quantité de ressources des datacenters. Cette offre permet aux fournisseurs de SaaS de faire évoluer facilement leur service et les infrastructures nécessaires afin de répondre à leur croissance et de s'ajuster par rapport aux demandes.

Les *Software-as-a-Service* (SaaS) sont les applications logicielles mises à disposition en tant que services et accédées par les utilisateurs via l'Internet à l'aide d'un navigateur web par exemple. Ces applications sont déployées sur les Infrastructures Clouds et maintenues par leur fournisseur. L'utilisateur ne doit donc pas s'occuper de la gestion de l'infrastructure et de la maintenance du logiciel.

Les services Clouds et SaaS sont délivrés aux utilisateurs *via l'Internet*, ce qui les rend accessibles à n'importe quel moment et depuis n'importe quel endroit dans le monde. Ces services sont mis à disposition en tant que *commodité* ("Utility", tout comme l'eau, l'électricité et le gaz), ce qui permet :

- (a) un accès à la demande d'infrastructures, ressources et applications informatiques,
- (b) un modèle de facturation selon l'utilisation effectuée ("pay-per-use" ou "pay-as-you-go") ou un plan tarifaire flexible et adapté qui remplace le modèle de paiement forfaitaire ("flat-rate").

Ces services peuvent être rapidement mis en place selon les besoins courants de l'utilisateur et avec un minimum d'interaction et d'intervention humaine, et ils peuvent donc être dynamiquement et facilement configurables et extensibles/flexibles. Cinq facteurs essentiels peuvent ainsi caractériser un environnement de Cloud Computing : services offerts à la demande, accès via un réseau (l'Internet), mise en commun de ressources informatiques

(datacenters, “pooling”, architecture “multi-tenant”), adaptation rapide (“elasticity”) et utilisation mesurée des services (contrôle, monitoring et optimisation).

Le Cloud Computing regroupe trois types d’acteurs :

- Les *Cloud Providers* gèrent et maintiennent les Infrastructures et ressources Clouds qu’ils mettent à disposition selon le principe de l’Utility Computing.
- Les *SaaS Providers/Cloud Users* peut obtenir l’infrastructure et les ressources informatiques nécessaires au déploiement et à l’exécution des services d’applications logicielles (SaaS) auprès d’un ou plusieurs Cloud Providers; et ce, afin d’offrir leurs Software-as-a-Service aux SaaS Users/End-Users. Remarquons qu’un SaaS Provider peut également être un SaaS User.
- Les *SaaS Users/End-Users* accèdent aux Software-as-a-Service via l’Internet. Ils ne doivent dès lors plus s’occuper de l’installation ni de la maintenance du logiciel et des ressources informatiques nécessaires pour exécuter les applications.

Pour terminer, notons que les conditions de prestation du services sont spécifiées dans un contrat appelé Service Level Agreement (voir Section 1.5, p. 39) établi entre le fournisseur et l’utilisateur.

La Figure 1.2 schématise cette définition du Cloud Computing.

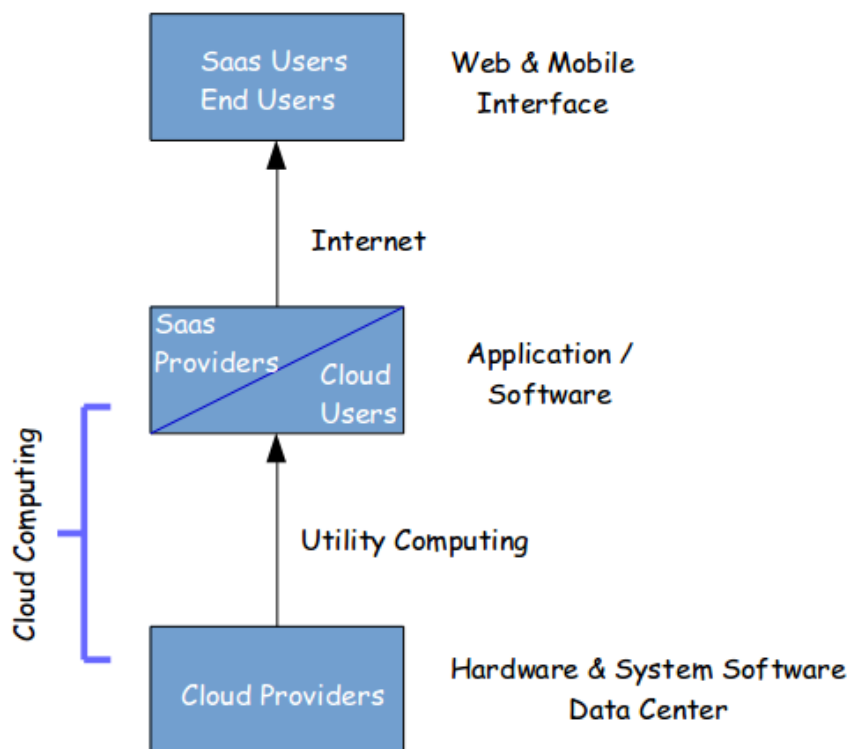


FIGURE 1.2 – Schématisation de la définition du Cloud Computing.

1.4 Cloud Service Brokers et Cloud Brokerage

Comme nous l'avons déjà fait remarquer, le Cloud Computing devient de plus en plus populaire auprès des entreprises comme manière d'accéder à des ressources informatiques via l'Internet (infrastructures et applications). Au vu de son émergence et de son adoption massive par les organisations, il en découle la venue sur le marché d'un nombre continuellement croissant de Service Providers et des services qu'ils proposent.

Dans le modèle Service Providers–Service Users, les utilisateurs dépendent de leurs fournisseurs et ne peuvent exercer que peu de contrôle sur les services reçus. Il n'est donc pas aisé pour les utilisateurs d'obtenir pleine assurance quant à l'efficacité des services reçus en comparaison avec les technologies utilisées localement. D'un point de vue historique de tels modèles fournisseurs–utilisateurs ont souvent introduit et fait l'usage d'un intermédiaire, appelés trader ou broker pour palier à ces problèmes, tel le courtier dans le secteur des assurances. Ainsi, les intermédiaires de services jouent un rôle de coordination des relations entre fournisseur et utilisateur dans l'industrie IT.

Bien que le Cloud Computing apporte un changement fondamental dans le modèle de consommation IT, il n'enlève pas le besoin d'intermédiaire ; au contraire, il l'augmente. En effet, il y a de plus en plus de fournisseurs et de parties prenantes, et les nombreuses relations sont à gérer minutieusement par la nécessité d'une assurance de qualité et d'interopérabilité avec les différentes fonctionnalités déjà présentes.

Notre but dans cette section est de mettre en lumière une définition claire et complète des notions de Cloud Service Broker et de Brokerage en vue de les utiliser comme références lors de notre état de l'art et de notre contribution à l'apport d'un nouveau Service Broker et de son architecture pour l'évaluation et la sélection du meilleur service à mettre en place chez le Service User concerné en adéquation avec ses exigences et ses besoins demandés.

Lors de notre recherche, nous avons remarqué que la définition et la description de Gartner et celle du NIST sont celles qui sont les plus communément reprises dans la littérature scientifique ainsi que par les différents fournisseurs (Service Providers). Cependant, ces définitions ne nous semblent pas assez précises et complètes. Nous tenterons donc de les enrichir afin d'établir un cadre de référence.

1.4.1 Pourquoi utiliser des Brokers ?

Nous relevons donc l'importance d'un intermédiaire pouvant gérer les relations avec de multiples fournisseurs et la variété des services disponibles via l'Internet. De nombreux fournisseurs apparaissent sur le marché et apportent infrastructures, plate-formes et applications logicielles liées au Cloud Computing, mais en réalité il n'y a que très peu de standards et les interopérabilités entre les différents services sont complexes.

Les utilisateurs qui souhaiteraient acquérir les services auprès de plusieurs fournisseurs dans le but de satisfaire des exigences business précises se retrouvent bloqués face à ce manque de standardisation et d'interopérabilité (voir Section 1.6.1, p. 41) entre les solutions des différents fournisseurs.

Tant les utilisateurs que les fournisseurs ont donc besoin d'une solution transparente pouvant combler ce manque d'interopérabilité entre les services provenant de plusieurs fournisseurs.

Les *Cloud Services Brokers* (CSB) – “Brokers de services dans le Cloud Computing” – sont ainsi introduits en tant qu’intermédiaires (ou third-party) entre les Services Providers et les Services Users (voir Figure 1.3). Leur rôle principal est d’offrir une intermédiation de services en y ajoutant une plus-value (e.g. personnalisation, agrégation de services, sécurité, gestion des identités) ainsi que de permettre l’intégration et l’interopérabilité avec plusieurs autres services.

De plus, avec les offres de services qui se multiplient, la tâche des organisations dans le choix du/des service(s) à adopter et du/des fournisseurs est rendue plus laborieuse pour arriver à sélectionner le(s) service(s) qui satisferont et s’adapteront le mieux aux exigences et besoins requis.

En effet, chaque Service Provider offre des services plus ou moins similaires substituables mais dont les prix et les niveaux de performances avec différents ensembles de caractéristiques peuvent grandement varier. Alors qu’un fournisseur peut très bien être bon marché dans l’offre d’un certain service (e.g. stockage, accès réseau rapide), il peut tout aussi bien être plus coûteux dans l’offre d’un autre service (e.g. location de VMs). De plus, des compromis peuvent apparaître entre les exigences fonctionnelles et non fonctionnelles remplies par les Providers [9].

Cette complexité pour les Service Users dans l’évaluation des différents services disponibles requiert des efforts et des ressources d’analyses non négligeables, et ce, de manière objective demandant qualité, fiabilité et sécurité. Afin de faciliter l’analyse en vue de la sélection des services adaptés pour les besoins d’un Services Users, ces démarches pourraient être automatisées par un système expert de prise de décision. C’est exactement ce que peut être mis en place par un Broker. Ainsi, l’entreprise ne doit plus consacrer de ressources (humaines et financières) dans l’analyse et les démarches pour la sélection du/des fournisseur(s) et du/des service(s) qui lui correspond le mieux parmi une panoplie de fournisseurs et de services. Elle peut s’enregistrer auprès d’un Broker et lui déléguer les démarches sur base de ses exigences et besoins.

Le Cloud Service Broker opère automatiquement et dynamiquement, avec un minimum d’intervention humaine, les services d’intermédiation, d’interopérabilité et d’aide à la décision en vue d’une acquisition de services, ce qui est appelé “*Service Brokerage*”. Grâce à de tels Brokers, nous dit D. Plummer (Gartner), une entreprise ne doit plus se spécialiser en matière de sélection et d’approvisionnement de services (Cloud) ni simplement se contenter d’obtenir ses services auprès d’un fournisseur unique en se fiant aveuglément à la manière dont ses services devraient fonctionner [10]. Les Brokers permettent donc d’éviter le “lock-in” de certains fournisseurs et d’assurer la réalisation des exigences des organisations (Service Users).

Les Brokers proposent et fournissent donc une place/plate-forme unique pour l’intermédiation, l’interopérabilité, la recherche et la sélection, ainsi que la fourniture de services adaptés auprès d’un ou plusieurs fournisseurs. Ils permettent également d’assurer aux Service Users que leurs exigences décrites sous contrat (voir Service Level Agreement, Section 1.5) sont correctement respectées en réalisant des activités de monitoring et de suivis de performances. Il est important d’évaluer pour chaque service désiré quelle est l’offre la plus pertinente et dans cette optique, les CSB peuvent être efficaces pour collecter, évaluer et fournir les résultats et la sélection d’un ou plusieurs services chez un ou plusieurs fournisseurs, sur base des exigences souhaitées.

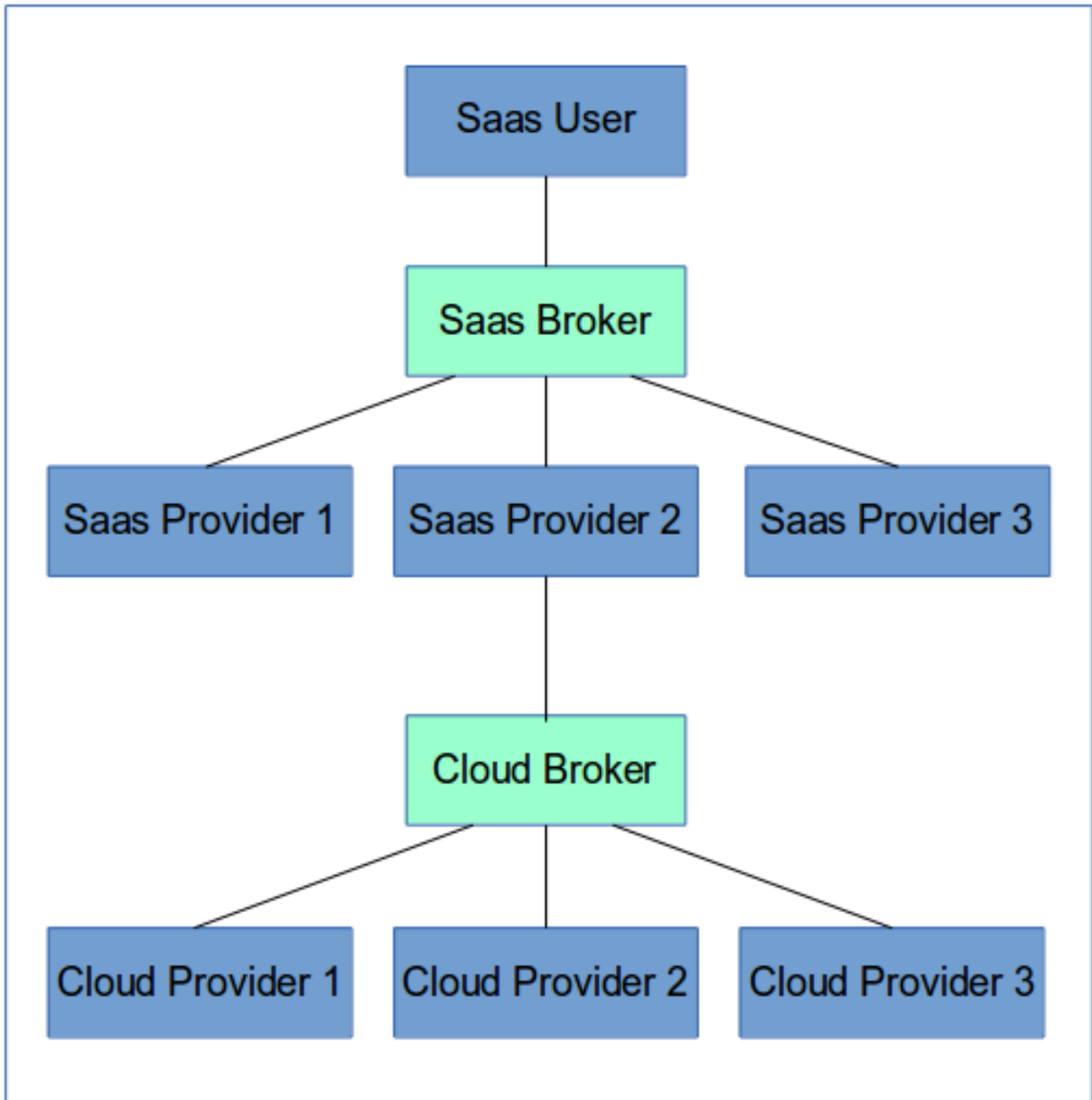


FIGURE 1.3 – Service Brokers dans le Cloud Computing.
 Chaque SaaS Providers (SP) repose sur un ou plusieurs Cloud Provider(s) (CP). Chaque CP opté par un SP fournit une Utility Computing différente. Le SaaS Brokers réalise les opérations de Brokerage pour le compte des SaaS Users, tandis que le Cloud Broker effectue le Brokerage pour les SaaS Providers. Remarquons que les CPs pour SP1 et SP3 ne sont pas représenté sur le schéma. Nous supposons qu'ils suivent la même structure que pour SP2. Ils peuvent également passer par le même Broker.

L. Frank Kenney (Gartner) affirme que “le futur du Cloud Computing s’accompagnera de la notion de Brokers négociant les relations entre fournisseurs de services et les consommateurs de ces services (Service Users), et permettra l’allocation de ressources auprès de multiples datacenters. Dans un tel contexte, un Broker peut être un logiciel, une plate-forme ou des suites de technologies qui améliorent les services de base disponible à travers le Cloud”.

1.4.2 Revue des caractéristiques des Cloud Service Brokers

Lors de nos recherches sur les Cloud Services Brokers, nous avons remarqué que la définition de Gartner et celle du NIST sont celles les plus souvent citées et reprises dans la littérature ainsi que par les différents fournisseurs.

Cependant, afin de faire preuve d’esprit critique et de prendre connaissance plus en détail de ce qu’il est dit dans la littérature, nous nous sommes penchés sur différents travaux afin d’en dégager les multiples caractéristiques propres aux Cloud Service Brokers.

En utilisant donc les différentes définitions rencontrées lors de nos recherches, nous tenterons de les approfondir et de les enrichir à l’aide des caractéristiques repérées. Ceci dans le but de pouvoir établir notre propre définition du concept et ainsi fournir une référence sur laquelle nous baser dans la contribution à l’apport d’un nouveau type de Service Broker.

Le lecteur pourra retrouver à l’Annexe B les différentes définitions collectées en vue de cette analyse.

Selon les travaux réalisés par Plummer et al. pour Gartner [11, 12], un Cloud Service Broker est à la fois un *rôle IT* et un *Business Model* dans lequel une entreprise ou toute autre entité ajoute de la valeur aux services pour le compte d’utilisateurs (Services Users). Forrester [13] ajoute que ce Business Model est un modèle qui exploite les compétences et les capacités des trois Business Models traditionnels que sont ceux du logiciel, de la consultance et de l’infrastructure.

Un Cloud Service Broker joue un rôle d’*intermédiaire* positionné entre utilisateurs de services (Services Users) et fournisseurs de services (Services Providers). Il peut être une entité logique ([14]) ou une plate-forme créant un point unique d’accès aux services de différents Providers et de leur gestion ([15, 16, 17, 18]).

Cette plate-forme de confiance ([15]) peut fournir simplement le service demandé en y apportant la valeur ajoutée (*intermédiation*) ou profiter de sa position d’intermédiaire ([17]) pour réaliser différentes opérations ([12, 14, 15]) :

- L’*Intégration* de plusieurs services provenant de différents Service Providers, et en assurer l’*Interopérabilité* afin de compenser la complexité due au manque de standardisation et ainsi fournir de manière transparente et aisée un accès à plusieurs services provenant de différentes plate-formes [14, 16, 18].
- L’*Agrégation* de plusieurs services interconnectés. D’une part, le Cloud Service Broker peut rassembler tous les services proposés afin de les rendre disponibles pour ses utilisateurs-clients en un seul et unique endroit, typiquement une plate-forme de marché. D’autre part, un Service User pourrait demander des services ou conditions qui ne peuvent pas être remplis par un unique Service Provider. Dans ce cas, le Broker réalise une agrégation de plusieurs services disponible pour construire un service adapté à la demande du Service User [16, 17].
- La *Sélection* du meilleur service adapté aux besoins et aux exigences du Service User de

sorte à réduire la charge de travail dans l'analyse des différents services disponibles. Le Broker permet ainsi d'aider les Service Users et de réduire la complexité de la sélection (e.g. fonctionnalités et qualités) et de la gestion des services (customisation, intégration, agreements) [15, 17, 18]).

- Créer un *Répertoire* des services disponibles et les catégoriser selon leurs similarités et leurs propriétés pour ensuite proposer aux Service Users une liste de services potentiels correspondant à leur requête [17].
- Parmi ces opérations, nous remarquons également que le Broker permet la *Customisation* (personnalisation) des services [12], la gestion des identités et des accès, l'audit et le monitoring des services [14, 18], ainsi que la création d'un environnement sécurisé [15].

La valeur ajoutée aux services grâce à l'intermédiation concerne la gestion des performances, des identités, de la sécurité, de la distribution des services et de leur utilisation ainsi que la gestion efficace des relations entre Service Providers et Service Users à l'aide d'un contrat Service Level Agreement (voir Section 1.5).

Gartner et le NIST classifient les Cloud Service Brokers en catégories selon le rôle de Brokerage qu'ils effectuent et les valeurs ajoutées qu'ils donnent aux services des Providers. Le Broker peut jouer un rôle primaire unique ou, en général, offrir une combinaison de ces rôles.

Pour le NIST, les Brokers peuvent jouer les rôles suivant : "Service Intermediation", "Service Aggregation" et "Service Arbitrage".

Gartner propose une classification des rôles de Brokerage de la manière suivante : "Customization Brokerage", "Aggregation Brokerage", "Integration Brokerage" et "Arbitrage Brokerage".

Kandpal ("*Rôle of Service Brokers in Cloud Computing*" [16]) indique également le "Governance Brokerage" effectuant la gestion de la Qualité de Service (voir Section 1.5), typiquement du monitoring et suivi de performance, ainsi que la gestion des Service Level Agreements.

La gestion des SLAs est souvent prise en charge par le Broker.

Finalement, précisons que les Service Brokers sont également bénéfiques aux Service Providers. En effet, ces intermédiaires sont également un excellent moyen pour aider les fournisseurs de services à se faire connaître et augmenter leur visibilité auprès des utilisateurs [11, 18].

1.4.3 Proposition d'une définition pour le concept de Cloud Service Broker

À partir des caractéristiques relevées ci-dessus, nous allons à présent proposer notre conception et définition pour les notions de Cloud Service Broker et de Brokerage qui sont au centre de ce travail. Nous prendrons pour référence ces définitions lors de notre contribution à l'apport d'un nouveau type de Broker.

A. Définition de Cloud Service Broker

Un *Cloud Service Broker* (CSB) est à la fois un Business Model et un rôle d'intermédiaire IT (ou de third-party) entre des Services Providers (fournisseurs) et des Services Users (utilisateurs).

Cet intermédiaire va assurer une plate-forme sécurisée et de confiance rassemblant les fournisseurs (Service Providers) et les utilisateurs de services (Service Users) en une plate-forme unique.

Il exécute, pour le compte des utilisateurs, des traitements appelés *Brokerage de services* (Service Brokerage) en vue d'apporter une valeur ajoutée aux services de base tels que proposés par leur fournisseur. Comme acteur intermédiaire, le Broker a pour but premier d'instaurer un environnement sécurisé pour la fourniture des services d'Infrastructure Clouds ou la fourniture des SaaS, et d'apporter une solution transparente d'interopérabilité entre plusieurs services de multiples fournisseurs. Celle-ci permet de combler les différences d'architectures et le manque de standardisation. Il permet également l'intégration du service demandé avec ceux déjà en place chez l'utilisateur.

De plus, il peut livrer entre autres la personnalisation/customisation des services, la gestion des identités et des accès, ainsi que l'audit, le suivi et le monitoring de la qualité des services. La gestion des Service Level Agreements (voir Section 1.5, p. 39) est souvent prise en charge par le Broker, ce qui comprend leur négociation des conditions de la qualité du service demandé et des niveaux de performances, les conditions financières ainsi que d'autres conditions liant l'utilisateur (Services User) à son fournisseur (Service Provider) en concordance avec les besoins requis par l'utilisateur lors de sa requête de recherche et d'obtention de ce service.

B. Les rôles de Brokerage

Profitant de sa position d'intermédiaire, le Cloud Service Broker peut simplement jouer le rôle principal d'intermédiation ou, en général, proposer une combinaison de plusieurs autres rôles de Brokerage donnant aux services d'origine (ou fonctionnalités) une valeur ajoutée orienté business.

En effet, le Broker peut délivrer à l'utilisateur une agrégation de plusieurs services de base dont la valeur ajoutée pour l'utilisateur réside alors dans leur combinaison. De même, puisque l'environnement de Cloud Computing comporte des risques liés à la qualité de service, à l'assurance de la sécurité et aux responsabilités, le Broker peut s'en porter garant.

Il possède les moyens pour répertorier, catégoriser et analyser les différents services disponibles et leurs propriétés. Il peut donc assister les Service Users à choisir, gérer, coordonner plus facilement les services en accord avec les exigences dont ils ont besoin pour réaliser leurs activités business. Ainsi, par la mise en place d'un tel système de Brokerage, le Broker est capable d'offrir une meilleure sélection de services pour le compte des utilisateurs.

Nous proposons donc cinq rôles clés de Brokerage de services joués par les Cloud Services Brokers : Intermédiation de services ("Intermediation"), Intégration de services ("Integration"), Agrégation de services ("Aggregation"), Arbitrage-Sélection de services

(“Arbitrage”) et Gestion des Service Level Agreements (“SLA Management”). Notons encore qu’un même Broker réalisera généralement les opérations provenant de plusieurs de ces rôles. En général, les Brokers concentrent leur expertise et leurs opérations en se basant sur les Qualités de Service (“QoS”, voir Section 1.5) délivrées, les plans tarifaires (“pricing”), et la gestion des SLAs. Certains d’entre eux proposent également d’effectuer le monitoring des performances, ainsi que le suivi de la réputation des fournisseurs (“Trust reputation”).

- **Intermédiation de services.** Opérations et services offerts par le Broker, pour le compte des utilisateurs, qui améliorent la qualité et les fonctionnalités des services de base tels que proposés par les fournisseurs, et fournissent des fonctionnalités supplémentaires à valeur ajoutée. Parmi ces fonctionnalités, citons la gestion des identités et des accès (e.g. le Single Sign-On⁷), l’amélioration de la sécurité, le suivi et le monitoring de la qualité et des performances, etc. L’intermédiation offre également la possibilité de personnalisation/customisation des services en modifiant ou améliorant un service avec une/des nouvelle(s) fonctionnalité(s) pour des besoins propres à un utilisateur.
- **Intégration de services.** Opérations et solutions d’interopérabilité permettant à plusieurs services séparés et indépendants de fonctionner et d’interagir ensemble, ce qui comprend l’intégration entre services du Cloud Computing d’une part, et d’autre part entre services du Cloud Computing et les services/systèmes de l’infrastructure IT de l’utilisateur. Le gain de valeur apporté par le Broker réside alors dans l’intégration et l’interopérabilité.
- **Agrégation de services.** Opérations pour combiner et inclure plusieurs services de différents fournisseurs et les rassembler en un seul ou plusieurs nouveaux services à mettre à disposition d’un utilisateur. Les services de base ne sont pas améliorés, le gain de valeur est obtenu par leur agrégation. Le Broker doit également garantir l’interopérabilité entre ces différents services. Il s’agit du principe typique de l’interopérabilité entre Clouds ou SaaS, et des architectures Inter-Cloud dont l’objectif est de créer un environnement standardisé permettant la communication entre de multiples systèmes tel que présenté et défini à la section 1.6 (voir p. 41).
- **Arbitrage/Sélection de services.** Opérations permettant la sélection du/des service(s) adéquat(s) dans un ensemble de services et de fournisseurs potentiels selon les fonctionnalités et exigences demandées. L’Arbitrage rend les utilisateurs plus flexibles dans leur choix et leur donne la possibilité de naviguer entre les différents services et fournisseurs disponibles. Il s’agit d’une fonction complémentaire à celle offerte par l’Agrégation. L’Arbitrage offre aux utilisateurs la possibilité de choisir le meilleur service parmi un classement des services potentiels sur base des fonctionnalités, des exigences et de différents critères requis. Par exemple, le Broker peut effectuer une sélection de services correspondant aux besoins demandés par l’utilisateur, et lui proposer un service de credit-scoring sous forme de classement des services potentiels de façon à assister l’utilisateur dans ses démarches pour la sélection du service qui lui convient le mieux.
- **Gestion des SLAs.** Processus de négociation et contrôle des qualités, niveaux de performance et conditions relatives à la fourniture du service pour le compte de l’utilisateur. Les Brokers proposent donc en général une gestion sécurisée, automatisée

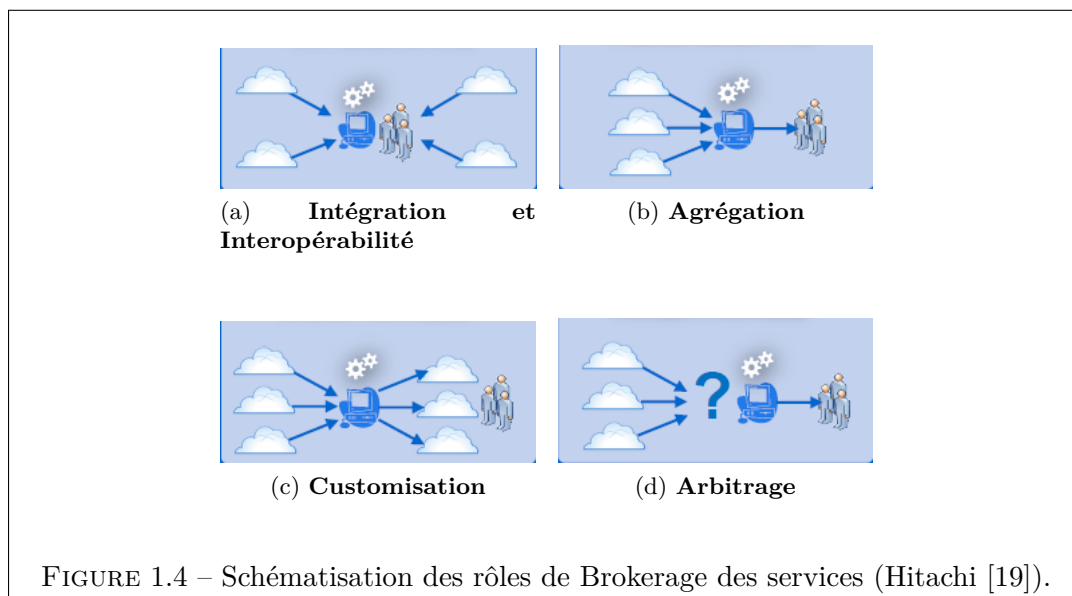
7. Le SSO offre la possibilité aux utilisateurs d’avoir accès à plusieurs services sous l’identité d’un seul compte

et précise des Services Level Agreements selon les exigences requises par l'utilisateur. La gestion des SLAs implique un suivi de la qualité et des performances du service à l'aide par exemple d'outils de monitoring.

Une représentation graphique de ces différents rôles de Brokerage est présentée à la Figure 1.4. Des exemples seront donnés et analysés dans notre État de l'Art ⁸.

Pour les utilisateurs, se référer à un Broker permet un gain de ressources, de temps et de coûts. il est plus facile, plus sécurisé et plus productif dans leurs opérations de recherche, sélection, intégration, consommation et maintenance de services clouds ou SaaS, et ce, particulièrement lorsqu'ils proviennent de multiples et divers fournisseurs de services. Les Brokers permettent également aux fournisseurs de plus facilement et mieux se faire connaître auprès des utilisateurs. De plus, le Broker peut aider les fournisseurs à mieux s'adapter à la clientèle ciblée en récoltant des feedbacks et ainsi augmenter la qualité des services. Le Broker peut aussi créer une place de fédération ou de partenariat entre fournisseurs. De cette manière, ceux-ci auront la possibilité de se compléter dans leurs offres de services ou de proposer une qualité de leur service améliorée. D'autre part, cette place de marché unique qu'instaure un Broker peut amener une concurrence saine (i.e. gérée et contrôlée) entre les fournisseurs donnant accès aux utilisateurs à des services de meilleure qualité.

À partir de ces différents rôles de Brokerage qui peuvent être réalisés par un Broker, nous distinguons trois catégories de Brokers : les *Brokers d'Intermédiation* ("Intermediation Brokers"), les *Brokers d'Agrégation* ("Aggregation Brokers") et les *Brokers d'Arbitrage* ("Arbitrage Brokers"). En général, ces Brokers effectuent également les opérations d'Intégration des services et la gestion des SLAs.



8. Voir Chapitre 3, Section 2.2, p. 2.2

C. Les types de Brokers

À partir de la définition du Cloud Computing que nous avons établie précédemment (section 1.1), nous relevons deux types de Cloud Service Brokers.

D'une part, nous avons les *Cloud Brokers* qui agissent en tant qu'intermédiaire entre les Cloud Providers et les Cloud Users (SaaS Providers); et d'autre part, nous avons les *SaaS Brokers* mis en place entre les SaaS Providers (Cloud Users) et les SaaS Users (également End-Users dans la chaîne des services Cloud).

Les Services Users ont recours à un Service Broker et lui envoient une requête contenant les exigences fonctionnelles et non-fonctionnelles attendues dans l'obtention d'un service Cloud ou d'un SaaS.

Le Cloud User envoie ses requêtes et interagit avec un Cloud Broker qui lui-même interagira avec les Cloud Providers; tandis que le SaaS User envoie ses requêtes et interagit avec un SaaS Broker qui est responsable du Brokerage de services et des interactions avec les SaaS Providers.

La Figure 1.5 montre l'architecture d'un environnement de Cloud Computing intégrant les deux types de Services Brokers. Ci-dessous, nous définissons plus précisément ces deux types de Service Brokers utilisés dans ce travail :

- Le **Cloud Broker** joue le rôle d'intermédiaire entre les Cloud Providers, fournisseurs d'Infrastructures Clouds, et les Cloud Users/SaaS Providers qui déploient leurs services et applications SaaS sur ces Clouds. Ce type de Brokers a pour but d'aider les Cloud Users dans l'intégration, l'interopérabilité et la sélection de leurs Infrastructures Clouds, ainsi que les ressources informatiques fondamentales pour leurs activités business. Ces services Clouds sont principalement basés et évalués sur des critères de qualité de service (QoS) et de prix. Les Cloud Brokers peuvent aussi réaliser la gestion des Service Level Agreements, et être responsables du suivi et du monitoring des niveaux de QoS.
- Le **SaaS Broker** agit comme un intermédiaire entre les SaaS Providers, fournisseurs d'applications SaaS, et les utilisateurs de ces SaaS (SaaS Users/End-Users). Ils assistent les Users dans la sélection, l'intégration, l'interopérabilité des SaaS et offrent une gestion automatisée des Service Level Agreements.

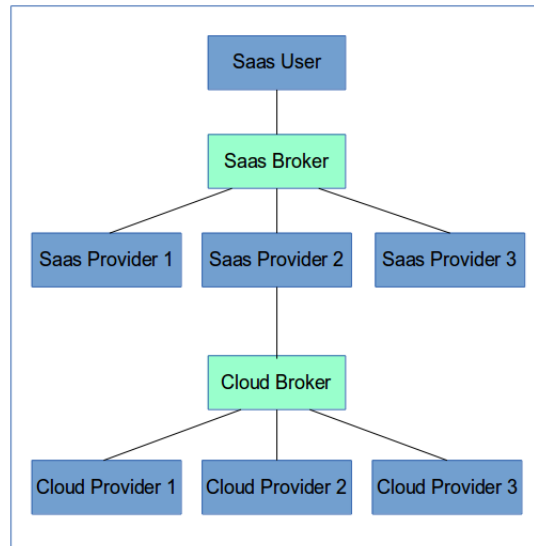
Ces SaaS Brokers sont à même de collecter et répertorier les services offerts par de multiples SaaS Providers. Ils permettent ensuite la recherche et la sélection de SaaS qui correspondent le mieux aux exigences fonctionnelles et non-fonctionnelles de l'utilisateur. Étant donné la multitude des services SaaS proposés sur le Web, ces SaaS Brokers facilitent ainsi le travail des utilisateurs dans leur choix et l'intégration des SaaS.

Les SaaS Brokers se portent donc responsables pour assister l'utilisateur dans sa requête de sélection, de fourniture et d'intégration de SaaS.

À notre meilleure connaissance, il n'existe qu'un nombre très limité de proposition de SaaS Brokers ou de tels frameworks. De tels Brokers visent à combler deux problèmes :
(a) obtenir un environnement persistant qui permet l'enregistrement d'applications SaaS par leur fournisseur ainsi que leur recherche, acquisition et exécution par les utilisateurs ;
(b) la possibilité de proposer un même environnement dans lequel les consommateurs peuvent accéder à leurs ressources et applications sur l'Internet.

Notons que la notion de "SaaS Broker" est, à notre meilleure connaissance, rarement retrouvée dans la littérature, car il n'existe actuellement que très peu de recherches

FIGURE 1.5 – Service Brokers dans le Cloud Computing



concernant le Brokerage pour les Software-as-a-Service. Les principes de Brokerage restent les mêmes, la différence réside dans le type de services traités.

Beaucoup de travaux concernent des recherches liées aux SaaS, mais seulement un nombre très limité d'entre eux traite d'une plate-forme rassemblant les applications SaaS en un environnement unique et sécurisé facilitant la recherche, la sélection et l'intégration. En effet, il existe de très nombreuses études sur les Cloud Brokers ou architectures similaires permettant la sélection et l'intégration de services d'Infrastructures clouds; mais au niveau des SaaS rien n'est encore réellement proposé et abouti.

Pour terminer, nous remarquerons que nous devons différencier les termes "Broker" et "Brokerage" quelques fois utilisés à tort comme synonymes. Un éclaircissement de ces deux termes est présenté par le travail de Plummer et al. [12].

Selon ces auteurs, le terme "Brokerage" fait référence à un Business Model décrivant les activités d'une entité qui opère le rôle d'intermédiaire comme étant son but premier. Ce terme, qui renvoie à l'objectif d'un business jouant le rôle d'intermédiaire entre fournisseurs et utilisateurs, est assez vague, et peut être utilisé pour indiquer tout type d'"intermédiation" qui donne de la valeur ajoutée à l'utilisation d'un service par un utilisateur. Il peut en être de même pour le business des fournisseurs.

De manière générale, le terme "Broker" est utilisé erronément comme abréviation pour signifier l'action de "Brokerage"; mais du point de vue rôle IT, "Broker" signifie une entité qui effectue des instances de "Brokerage". Le Broker rassemble l'architecture et l'application des mécanismes qui réalisent l'"intermédiation" entre l'utilisateur et le fournisseur, et il agit en tant que third-party dans cet environnement de relations.

1.5 Qualité de Service et Service Level Agreement

Nous avons à plusieurs reprises mentionné les termes “Qualité de Service” et “Service Level Agreement” dans nos explications ci-dessus. Afin de bien comprendre ces notions importantes qui seront utilisées dans la suite du travail, nous exposons ici leur définition.

Qu’est-ce que la Qualité de Service ?

La Qualité de Service (QoS – Quality of Service) peut être définie comme étant la performance générale d’un service (un réseau informatique, un service fourni via l’Internet) telle que perçue par les utilisateurs.

Par ses caractéristiques elle désigne la capacité du service à répondre aux différentes exigences des utilisateurs. Elle permet aux fournisseurs de s’engager formellement auprès de leurs utilisateurs sur les caractéristiques du service qui sera presté, leur garantissant ainsi un niveau de performance qui devra être constamment assuré.

La qualité d’un service est une notion qui peut varier selon le type de service et les critères qui lui sont propres. Elle pourra concerner des critères tels que la disponibilité, le débit, le temps de réponse, la fiabilité, les capacités, la sécurité, le temps de résolution des problèmes, etc.

Les principales composantes de la qualité de service seront fournies par des métriques caractérisées par un type, une unité et une fonction de calcul. Ces métriques doivent pouvoir être facilement mesurées et quantifiées entre autre à l’aide d’outils de monitoring.

Dans un modèle avec un Broker, celui-ci pourra mettre en place ces outils de monitoring afin de vérifier pour le compte de l’utilisateur si le niveau de qualité du service presté par le fournisseur correspond constamment à celui qui avait été convenu.

Qu’est-ce qu’un Service Level Agreement ?

Au vu du nombre grandissant d’utilisateurs du Cloud Computing qui délèguent la gestion de leurs ressources informatiques à de nombreux fournisseurs, les Service Level Agreements entre ces deux parties deviennent un aspect clé non négligeable pour s’assurer de la qualité des services prestés, ainsi que pour le bon fonctionnement du Cloud Computing.

La négociation et la gestion des niveaux de performance réalisés peuvent être des tâches fastidieuses à opérer manuellement. Les Brokers proposent donc en général une gestion sécurisée, automatisée et précise des Services Level Agreements selon les exigences requises par l’utilisateur.

Un Service Level Agreement (SLA) est un document formel et légal qui est négocié entre fournisseur (prestataire de service) et utilisateur. Ce contrat décrit le service presté et définit, en termes quantitatifs et qualitatifs, le niveau de performance des opérations (la QoS) qui sera offert et maintenu par le fournisseur envers l’utilisateur.

Le prix du service, la méthode de facturation et le plan tarifaire doivent être aussi clairement exprimé dans le contrat.

Chaque métrique incluse dans un SLA doit pouvoir être mesurée de manière régulière. Typiquement, le SLA couvre : les parties prenantes, la période de validité, les services couverts, les objectifs du niveau de performance de ces services (typiquement les QoS), les garanties, les restrictions, les responsabilités, etc. qui doivent être assurés en permanence. Il couvre également les informations relatives à la sécurité et à la facturation. Il est également

important que les SLAs comportent les obligations et les actions/pénalités qui seront prises en cas de non-respect ou violation des termes spécifiés dans le contrat ⁹.

Les SLAs doivent être renégociés chaque fois qu'un service est sujet à une modification de ses caractéristiques ou de ses performances, ou bien si un attribut du contrat est dans l'incapacité d'être réalisé (ITILv2).

Les fournisseurs utilisent les SLAs afin d'optimiser l'usage des ressources disponibles en leur possession, et de s'assurer que les niveaux de services définis dans ce SLA et nécessaires aux utilisateurs sont effectivement correctement délivrés. Les utilisateurs utilisent les SLAs pour s'assurer de recevoir le niveau de service minimum attendu en concordance avec leurs besoins [21].

Notons que la gestion et la négociation des SLAs font l'objet de nombreuses recherches dans le cadre des services du Cloud Computing. Bien que la gestion automatisée des SLAs soit bien connue dans le cadre de l'Utility Computing, leur adoption dans le cadre du modèle SaaS n'en est qu'à ses débuts.

La gestion des SLAs, par un Broker par exemple, implique deux phases : (a) la (re)négociation du contrat suivie de la validation des engagements et (b) le monitoring du respect de ceux-ci. Selon Wu et al. dans "*Service level agreement (SLA) in utility computing systems*" [22], le cycle de vie d'un SLA pour les systèmes de l'Utility Computing comporte six étapes réparties en trois phases :

- la recherche des Service Providers, la définition du SLA et l'acceptation des termes pour la phase de Création ;
- le monitoring des services et le suivi du respect des termes pour la phase de Réalisation ;
- la terminaison du SLA et l'application des sanctions en cas de violation pour la phase de Terminaison.

En outre, de nombreux travaux de recherche se concentrent sur la gestion automatisée des SLAs. Nous pouvons ainsi citer les initiatives les plus répandues dans le monde de la recherche et de l'industrie : Web Service Level Agreement (WSLA) par IBM et WS-Agreement par l'Open Grid Forum ; mais aussi SLAng, QML et WSOL. Une brève description et comparaison de ces frameworks et langages standardisés peut être trouvée dans le travail de Wu et al. [22].

Dans l'État de l'Art ¹⁰ nous donnons quelques travaux ("SLA-based Brokers") détaillant un mécanisme pour la gestion des SLAs.

9. Une étude d'analyse pour la minimisation du nombre de violations SLA dans un environnement Cloud est réalisée par Wu et al. dans [20]

10. Voir Chapitre 3, Section 2.2

1.6 Interopérabilité et Architectures Inter-Cloud

Le modèle standard du Cloud Computing où un utilisateur se lie à un seul fournisseur comporte des limites. En effet, l'utilisateur n'est pas à l'abri d'une indisponibilité du service ou d'une limitation des capacités demandées, mais aussi d'une cessation des activités de son fournisseur. Dans de telles situations, l'utilisateur se retrouverait bloqué dans la poursuite de son business.

Buyya et Grozev font remarquer l'utilité de pouvoir combiner plusieurs Clouds et d'avoir recours à plusieurs fournisseurs dans le but d'atteindre une meilleure qualité de service, ainsi qu'une meilleure fiabilité et flexibilité [23].

À l'image de l'Internet qui est le réseau des réseaux, l'idée est de relier plusieurs Clouds entre eux pour former un réseau de Clouds. Cette utilisation de multiples Clouds forment un "Cloud de Clouds" qui est nommé "*Inter-Cloud*". Afin de pouvoir rassembler ces multiples Clouds et d'être flexible entre les différents fournisseurs, il est nécessaire d'instaurer des moyens d'interopérabilité.

Dans ce contexte, le concept de Broker se révèle un intermédiaire intéressant pour proposer des adaptateurs et des APIs standardisées en vue de l'échange standardisé de données et la cohabitation de multiples architectures ou services (interopérabilité). De plus, les Brokers peuvent faciliter l'Inter-Cloud.

1.6.1 Interopérabilité entre Clouds – Cloud Interoperability

L'interopérabilité entre Clouds (Cloud Interoperability) permet à plusieurs Clouds de domaines différents de communiquer ensemble et d'échanger des données standardisées.

Selon Loutas et al. [24], l'interopérabilité dans le Cloud Computing est la possibilité d'écrire du code qui fonctionne simultanément avec plus d'un (Cloud) Provider, sans tenir compte des différences d'architecture et de standards entre ces Providers.

Chaque fournisseur a sa manière propre pour définir l'architecture de ses services et la façon dont un utilisateur ou un autre service interagit avec son système. Les architectures et les APIs sont en général propres à chaque fournisseur. L'utilisateur est donc souvent limité dans ses choix et bloqué chez un même fournisseur pour des raisons de portabilité et d'intégration de nouveaux services avec ceux déjà en place. Ce phénomène qui empêche les utilisateurs de s'adresser à d'autres fournisseurs est appelé "lock-in".

L'interopérabilité des services jouera un rôle important dans le modèle du Cloud Computing. Elle permettra aux services (infrastructures Clouds, applications et données) d'interagir et d'être facilement intégrables malgré leurs différences de standard et d'architecture, mais également d'être facilement interchangeable en rendant les utilisateurs indépendants des différentes architectures utilisées par les fournisseurs.

Pour atteindre cette interopérabilité, l'Open Grid Forum a proposé comme première initiative, la définition d'une API standardisée nommée OCCi (Open Cloud Computing interface¹¹).

11. <http://occi-wg.org/>

1.6.2 Inter-Cloud : Multi-Cloud & Cloud Federation

La notion d'Inter-Cloud et les recherches associées sont récentes. La terminologie utilisée varie selon les auteurs et nous nous retrouvons parfois face à des ambiguïtés. Nous proposons donc une définition pour le concept d'environnement Inter-Cloud en nous basant sur une étude du domaine réalisée par Grozev et Buyya ("Inter-Cloud Architectures And Application Brokering : Taxonomy and Survey [25]).

A. Définition

Du point de vue des utilisateurs, l'accès à un environnement Inter-Cloud permet de facilement diversifier et répartir les services d'infrastructures et d'applications parmi les offres de multiples fournisseurs.

Du point de vue des fournisseurs, la participation à un Inter-Cloud donne la possibilité d'améliorer la garantie de qualité de services afin de satisfaire au mieux les exigences convenues avec les utilisateurs, et de ne pas se retrouver limité en cas de demandes de charge de travail excessive. Grâce à la mise à disposition des ressources excédentaires entre Clouds par leur location en cas de besoin, les fournisseurs peuvent momentanément se partager et étendre leurs ressources et par là même proposer de meilleures conditions de performance dans les SLAs.

Le Global Inter-Cloud Technology Forum ¹² (GICTF, 2010) définit formellement l'Inter-Cloud comme *"Un modèle de Cloud qui, dans le but de garantir la qualité de service (QoS), telle que la performance et la disponibilité de chaque service, permet la réassignation à la demande des ressources et le transfert de la charge au moyen d'une interconnexion de système Clouds, provenant de différents Cloud Providers, basés sur la coordination des exigences de QoS de chaque utilisateur telles que précisées dans les SLAs et ceci en faisant usage d'interfaces standardisées"*.

Pour Buyya et Grozev [25], les bénéfices d'un environnement Inter-Cloud sont :

- La flexibilité entre le choix de services, une meilleure élasticité, et l'assurance contre la cessation d'activités d'un fournisseur ;
- Le contournement du lock-in des fournisseurs et la possibilité de transférer ou répartir le travail entre eux en cas de changements des conditions contractuelles ;
- Diverses localisations géographiques permettant le respect des conditions administratives et des conformités législatives en vigueur ;

Pour Demchenko et al., l'architecture Inter-Cloud s'occupe de trois formes de problèmes architecturaux [26] :

- Les applications reposant sur de multiples fournisseurs et utilisant plusieurs Clouds, ce qui inclut l'intégration et l'interopérabilité avec les services d'infrastructure ;
- L'interopérabilité et le monitoring des infrastructures intra-fournisseurs ;
- La notion de fédération (Cloud Federation) que nous décrivons plus loin.

12. <http://www.gictf.jp>, "Use Cases and Functional Requirements for Inter-Cloud Computing"

B. Classification architecturale

La définition apportée par le GICTF ne spécifie pas si les Providers collaborent volontairement ensemble afin de former un Inter-Cloud ou non. Nous trouvons dans la littérature deux types d'Inter-Cloud qui peuvent nous aider à faire cette distinction : (a) le *Cloud Federation* et (b) le *Multi-Cloud* [25].

(a) Le terme **Cloud Federation**, ou fédération de Clouds, signifie une coopération entre un ensemble de Providers qui interconnectent volontairement leurs infrastructures afin de permettre la combinaison des données et le partage des ressources entre eux. Une telle fédération constitue un environnement dont le but est de surmonter une augmentation momentanée d'utilisation effective des ressources, et d'éviter le problème de lock-in associé au choix d'une solution propriétaire unique [27, 28]. La Fédération doit respecter les différents niveaux du Cloud Computing (Utility Computing et SaaS) et être transparente pour l'utilisateur tout en respectant le SLA [29, 28].

Une Fédération Cloud Horizontale (Horizontal Cloud Federation dans [30]) permet aux fournisseurs de se porter mutuellement assistance afin de faire face aux accroissements de capacité et de gérer des requêtes lorsque le service d'origine a atteint sa pleine capacité ou est incapable de traiter celles-ci au niveau de qualité de service (QoS) requis.

Dans le but d'épargner à l'utilisateur la complexité du traitement avec plusieurs fournisseurs simultanés, une première étape est de permettre leur interopérabilité en développant des équivalences entre les différentes APIs des Clouds qu'elles soient open-sources ou propriétaires, ainsi qu'en développant des traducteurs entre toutes les spécifications et formats de données [31]. La Fédération permet l'adaptation des ressources selon le contexte, la gestion des SLAs, la confiance et la sécurité, ainsi que l'intégration horizontale et verticale. Elle accroît la facilité de fournir des services sous forme de commodité. Ce modèle d'environnement Cloud va de pair avec les Brokers.

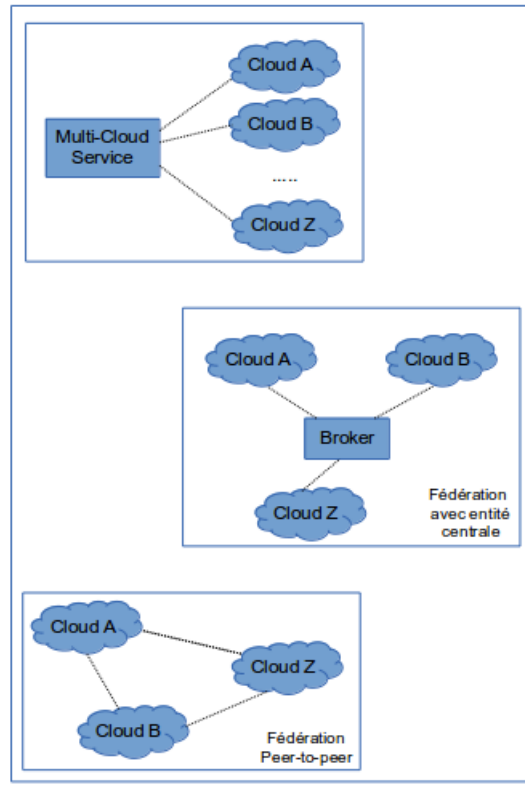
L'environnement de fédération peut être réalisé chez un Cloud Provider ou auprès d'un Cloud Broker. Un Broker ne possède pas proprement les ressources, mais il met en relation les utilisateurs et les fournisseurs sur base des conditions (non fonctionnelles) du SLA telles que convenues avec l'utilisateur [29]. Une fédération peut introduire un Broker entre les différents Clouds afin de réaliser ou de faciliter la distribution des ressources et des services (Fédération avec entité centrale). En général, le Broker jouera le rôle de répertoire des ressources enregistrées et disponibles. Il peut également jouer le rôle d'une place de marché. Sans une telle entité intermédiaire, les Clouds communiquent et négocient directement entre eux (Fédération Peer-to-Peer).

Une représentation des fédérations est montrée à la Figure 1.6.

(b) Le **Multi-Cloud** représente l'agrégation par un utilisateur, un service ou un Broker, de multiples Clouds indépendants tel que présenté par la Figure 1.6. Un environnement Multi-Cloud n'implique pas une interconnexion et un partage volontaire des infrastructures et ressources des fournisseurs contrairement à la Fédération. Les utilisateurs sont directement responsables de la gestion de l'acquisition et de la planification de leurs ressources.

Le Multi-Cloud peut être assimilé à une Fédération Cloud Verticale (Vertical Cloud Federation dans [30]), qui permet aux utilisateurs de services de rechercher individuellement leurs propres fournisseurs de services. Dans une approche de type Multi-Cloud, chacun des SaaS Providers déploie ses services chez plusieurs Cloud Providers. Chaque Cloud Provider opté par le SaaS Provider leur permet d'obtenir des conditions différentes. L'architecture Multi-Cloud rend donc possible que chaque SaaS Provider ait accès à un choix plus flexible à partir du moment où ils ne sont pas restreints à un seul Cloud Provider [30].

FIGURE 1.6 – Représentation des Fédérations Clouds et des Multi-Cloud.



C. Application au niveau des SaaS

La Fédération au niveau SaaS, à distinguer de la Cloud Federation, est analogue à l’usage de sous-traitants ou de pairs qui peuvent fournir des services (en parties ou en entier) finaux équivalents à ceux du fournisseur principal/responsable, offrant ainsi la garantie de fournir un service adéquat. De plus, lorsqu’un SaaS requiert une augmentation temporaire de performances, le SaaS Provider va déléguer l’acquisition de ressources supplémentaires à la partie “Cloud”. Le Cloud Provider alloue alors les ressources nécessaires de son Cloud “local”. Si il ne peut fournir les ressources supplémentaires demandées, alors il tentera de les obtenir en location temporaire auprès d’un autre Cloud Provider participant à la Fédération de Clouds.

Une vue synthétisée de l’approche Inter-Cloud est représentée par la Figure 1.7. La Figure 1.8 montre la taxonomie des architectures Inter-Cloud et donne quelques exemples de projets qui seront présentés dans notre État de l’Art¹³. Quelques plate-formes ont été développées afin d’apporter des mécanismes pour supporter l’Inter-Cloud telles que OpenNebula¹⁴ et Eucalyptus¹⁵.

13. Voir Chapitre 3, Section 2.2

14. <http://www.opennebula.org/>

15. <http://open.eucalyptus.com/> et www.eucalyptus.com

FIGURE 1.7 – Vue schématisée de l'approche Inter-Cloud [25].

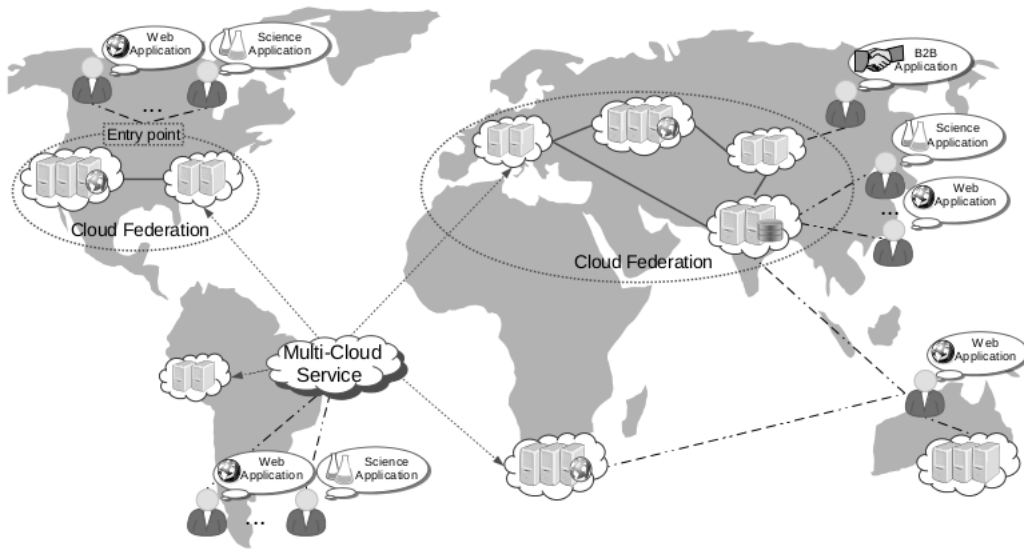
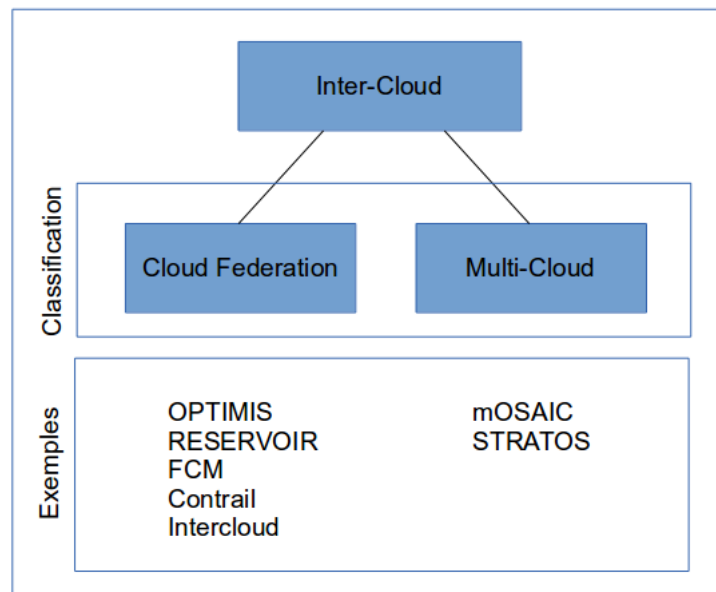


FIGURE 1.8 – Taxonomie des architectures Inter-Cloud.



Chapitre 2

État de l'art des Cloud Service Brokers

Dans ce chapitre, nous tentons d'établir une présentation, le plus complète possible, et une analyse des différents Cloud Service Brokers (projets, frameworks et méthodes) dans le but de constituer l'État de l'Art du domaine des Brokers de services du Cloud Computing. Durant notre parcours de la littérature, nous avons remarqué que les termes "Cloud Service Broker" et "Brokerage" ne sont pas des termes standards et communément utilisés pour décrire la plate-forme ou l'agent entre fournisseurs et utilisateurs offrant des mécanismes d'intermédiation, d'interopérabilité, d'agrégation, de sélection de services et de gestion des SLAs tels que nous l'avons discuté au chapitre précédent. Nous donnerons d'abord la méthodologie employée pour réaliser cet État de l'Art. Ensuite, nous présenterons les différentes études concernant les Services Brokers et les mécanismes de Brokerage. Nous terminerons ce chapitre par une comparaison des travaux en guise de récapitulatif, et en présentant les directions de recherches possibles.

2.1 Méthodologie

Afin de mieux maîtriser le sujet du Brokerage de services dans le Cloud Computing et de prendre connaissance de l'état actuel de ce domaine et des projets existants, nous avons effectué de nombreuses lectures dans littérature. Notre but est donc de présenter les résultats de notre large revue de la littérature scientifique sur les Cloud Service Brokers. Nous exposerons ainsi des travaux issus de la littérature, mais aussi quelques systèmes commerciaux, afin d'obtenir une vision globale de l'existant en matière de Cloud Services Brokers et de mécanismes de Brokerage. Remarquons que nous ne pouvons pas être exhaustifs et recenser tous les projets, études et systèmes existants.

Dans le but de mieux connaître le sujet et les opportunités de nouvelles études relatives, nous avons porté nos recherches sur un maximum de travaux traitant des systèmes intermédiaires (Brokers) entre utilisateurs et fournisseurs de services du Cloud Computing (Infrastructure et SaaS), et discutant de mécanismes d'interopérabilité, d'agrégation, de sélection de services et de gestion des SLAs. Nous avons pu de cette façon établir un large panel de projets, frameworks et méthodologies concernant le Brokerage de services. En outre, nous nous sommes intéressés plus particulièrement aux systèmes de sélection de

services et aux modèles de réputation (“trust” et “feedback”).

Dans un premier temps, nous avons donc parcouru la littérature à la recherche de nombreuses sources concernant les Brokers de services (Infrastructure et SaaS), les mécanismes de sélection et modèles de réputation. Ensuite, nous avons sélectionné les travaux nous semblant les plus pertinents en vue d’une analyse plus approfondie. Pour chaque projet ou modèle analysé, nous avons alors rédigé une présentation en essayant de refléter au mieux la conception du Brokerage ou du modèle étudié. Ce sont ces descriptions qui constituent notre État de l’Art. Finalement, nous avons établi une comparaison entre ces différents travaux exposés afin de faire ressortir les aspects similaires étudiés dans les travaux ainsi que les ouvertures de recherches laissant la possibilité à une nouvelle étude.

Dans les travaux que nous avons collectés, nous avons répertorié d’une part de nombreuses études concernant les services d’Infrastructure Cloud, alors que d’autre part nous n’avons pu répertorier au contraire que très peu de travaux de recherche et d’investigation relatifs au Brokerage d’applications et SaaS Brokers.

Bien que nous concentrons nos recherches sur les SaaS Brokers, nous avons donc décidé d’inclure les travaux concernant les Cloud Brokers et la sélection de services d’Infrastructure Cloud dans notre État de l’Art afin de l’étoffer et posséder une solide base de connaissances. En effet, nous pensons que ces travaux peuvent être intéressants pour notre travail dans le sens où ils nous fournissent des exemples et des réflexions sur des architectures et des mécanismes de Brokerage. De plus, ces travaux et ces architectures pourraient être adaptés aux SaaS, et donc nous être utiles pour la suite de notre travail.

La revue de la littérature et la constitution de cet État de l’Art nous ont permis d’obtenir un enrichissement de nos connaissances sur les notions et architectures relatives au domaine du Brokerage de services du Cloud Computing. Cette partie nous amène également des éléments et des réflexions qui nous serviront de support lors de notre proposition d’un modèle de SaaS Broker.

2.2 Principaux Cloud Service Brokers

Dans cet État de l’Art, nous essayerons d’exposer un maximum de Cloud Services Brokers pertinents et de mécanismes relatifs au Brokerage de services en reprenant les travaux les plus aboutis. Nous présenterons d’abord les études relatives aux *SaaS Brokers* : un framework de SaaS Broker pour la sélection d’applications basée sur les SLAs (Badidi) et un Broker pour l’agrégation et l’intégration de SaaS (Moore et Mahmoud).

Nous exposerons ensuite les travaux concernant les Infrastructures Clouds présentant des *Cloud Brokers* réalisant l’agrégation et/ou la sélection de services. Par ceux-ci, nous analyserons entre autres quelques projets du EU-FP7¹ :

OPTIMIS qui permet la gestion d’une Infrastructure Cloud ainsi que le déploiement et l’exécution d’applications sur cette infrastructures ;

1. Le European Union 7th Framework Programme for Research and Technological Development (EU-FP7) est un programme de recherche derrière de nombreux travaux de recherche dans le domaine du Cloud Computing et l’acquisition de service Cloud. Ces projets sont à l’origine de recherches concernant l’obtention d’Infrastructure Cloud, le Brokerage de services, la gestion des SLAs, le cloud monitoring, etc.

SLA@SOI qui est un framework pour la gestion automatisée des SLAs ainsi que le monitoring des services pour des architectures et infrastructures orientées services (SOA et SOI) ;
mOSAIC qui est un large projet qui développe une API et plate-forme open-source pour le Multi-Cloud ;
RESERVOIR qui explore la notion de Clouds Fédérés et propose une architecture pour créer une mise en commun des ressources et des infrastructures Clouds ;
ainsi que Contrail qui est un framework ayant pour but une fédération de Clouds et développant une approche de gestion des identités et d'accès à l'aide d'un identifiant unique. Nous expliquerons également STRATOS, un projet de recherche financé par Amazon et des organisations canadiennes, qui est un Cloud Service Broker pour faciliter le Multi-Cloud, le déploiement et les modifications à l'exécution de plate-formes d'applications cloud ;
Intercloud qui propose un environnement de Clouds Fédérés et une gestion des SLAs (Buyya et al.) ;
Federated Cloud Management, qui est une architecture pour l'agrégation de plusieurs Cloud Providers ;

En plus de ces projets, nous passerons en revue quelques méthodes indépendantes ou frameworks pour *la sélection de services* : la proposition d'une approche de sélection basée sur la qualité de service (Wang et al.),
une approche par décision multi-critères pour la priorisation et la sélection d'application SaaS (Godse et al.),
une architecture basée sur la création d'un index des Cloud Providers (CSP-index) en vue de leur sélection suivant certains critères (Sundareswaran et al.) ; un framework pour une sélection de services basé sur l'agrégation de feedbacks et de mesures quantitatives à l'aide (Qu et al.) ; un framework pour la recommandation de services sur base de critères propres à l'utilisateur et permettant de collecter les feedbacks (Yan et al.) ; et un plate-forme de Brokerage pour allouer les ressources Clouds à partir de multiples datacenters (Houidi et al.).

En plus de ces travaux traitant des Brokers, nous exposons également quelques *outils pour la comparaison* de services Clouds :
CloudCmp qui est un comparateur de performance et de coût des fournisseurs d'infrastructures cloud (Cloud Providers) ;
SMICloud qui est un framework pour la comparaison et le ranking de services ; et
CloudAdvisor qui permet de trouver une configuration optimale pour une Infrastructure Cloud selon les préférences de l'utilisateur.

Pour finir, nous présentons une comparaison de ces différents Brokers à la Section 2.3.

2.2.1 Cloud Service Broker for SLA-based SaaS, E. Badidi

E. Badidi présente dans “*A Framework for SaaS selection and provisioning*” l’un des rares frameworks de SaaS Broker pour l’acquisition et la sélection d’applications SaaS [32]. La sélection des applications opérée par le Broker est effectuée sur base du Service Level Agreement [33].

Le framework exposé permet :

- l’expression des exigences fonctionnelles et non-fonctionnelles par les utilisateurs des SaaS ;
- l’intermédiation de services ;
- l’implémentation d’une sélection de SaaS Providers basée sur les Qualités de Service (QoS) offertes ;
- la négociation des SLAs ;
- l’abstraction pour les utilisateurs de l’hétérogénéité des APIs des SaaS Providers (interopérabilité) ;
- le monitoring et l’évaluation des accords et des conditions décrits dans le SLA.

Le framework est composé de quatre entités principales : les consommateurs d’applications SaaS (SaaS Users), le Cloud Service Broker (SaaS Broker), l’infrastructure de Monitoring (Monitoring Infrastructure), et les fournisseurs d’applications SaaS (SaaS Providers).

Le composant principal du framework est le “*Cloud Service Broker*”. Ce SaaS Broker est en charge de la médiation entre les SaaS Users et les SaaS Providers dans le but d’atteindre les accords et les conditions qui décrivent les fonctionnalités attendues des services et les niveaux de qualité de service offerts. La négociation SLA, la sélection des SaaS Providers appropriés, la gestion des tâches et le monitoring des QoS sont tous délégués au composant *Cloud Service Broker*. L’architecture du Cloud Service Broker comprend également dans son rôle d’Intermédiation les opérations de la gestion des identités et des accès, la gestion des règles stratégiques (policies) ainsi que l’acquisition de SaaS. L’algorithme de sélection utilise une fonction d’utilité agrégée linéaire²) afin de classer les offres des applications SaaS potentielles en les comparant avec les exigences de qualité de service demandées par l’utilisateur, et d’en sélectionner l’application SaaS optimale (Arbitrage). Les différents paramètres de qualité de service sont supposés indépendants.

Le composant *Selection Manager* pour les opérations d’acquisition des services implémente différentes règles pour la sélection des SaaS Providers adéquats. Ces règles sont basées sur les exigences de QoS des utilisateurs et les offres de QoS des fournisseurs. C’est ce Selection Manager qui évalue la fonction d’utilité agrégée et détermine si les offres des SaaS Providers sont acceptables ou non pour l’utilisateur. L’auteur fait référence à des travaux relatifs à la sélection de services Cloud, tels que celui de Wang et al. “*Cloud Model for Service Selection*” [34] et celui de Li et al. “*Comparing public-cloud providers*” [35] que nous présenterons plus loin dans ce chapitre (voir 2.2.18 et 2.2.14).

2. “linear aggregate utility function” de la forme suivante : $U = w_1x_1^{\beta_1} + w_2x_2^{\beta_2} + \dots + w_nx_n^{\beta_n}$

Le *SLA Manager* effectue la négociation des SLAs entre le SaaS User et un SaaS Provider sélectionné. Il détermine ensuite si le fournisseur peut assurer le niveau de service requis. Si le fournisseur sélectionné est incapable de délivrer le niveau de service adéquat, alors le Broker sélectionne un autre fournisseur et réitère le processus de négociation. Le *Profile Manager* est responsable des profils des utilisateurs (Identity and Access Management), qui inclut également leurs préférences de SaaS et leurs QoS requises. Le *Policy Manager* garde les différentes sortes de règles telles que les politiques d'autorisation et les politiques de sélection des fournisseurs sur base des QoS. Le composant *Monitoring Infrastructure* inclut des Monitoring Adapters et Measurement Services. Pour chaque *Measurement Service*, il existe un Monitoring Adapter. Les *Measurement Services* permettent aux SaaS Providers d'avoir des monitorings indépendants de leurs offres, et permet aux utilisateurs des services d'être certains de recevoir le niveau de service promis. L'auteur cite quelques exemples pour la mesure et le monitoring des services : Keynote (www.keynote.com), Monitis (www.monitis.com) et Uptrends (www.uptrends.com). Les *Monitoring Adapters* mettent en lien les métriques des ressources en paramètres de SLA, assurent le suivi du niveau de service courant et évaluent leur adéquation avec le SLA. Les Adapters utilisent les APIs des Measurement Services afin d'obtenir les données de la QoS courante.

Notons que le processus de négociation des SLAs, le modèle de négociation multi-attributs ainsi que l'algorithme de sélection sont amplement détaillés dans les travaux de l'auteur [33] et [32].

La figure 2.1 présente le framework du Cloud Service Broker et ses quatre composants pour la sélection de SaaS.

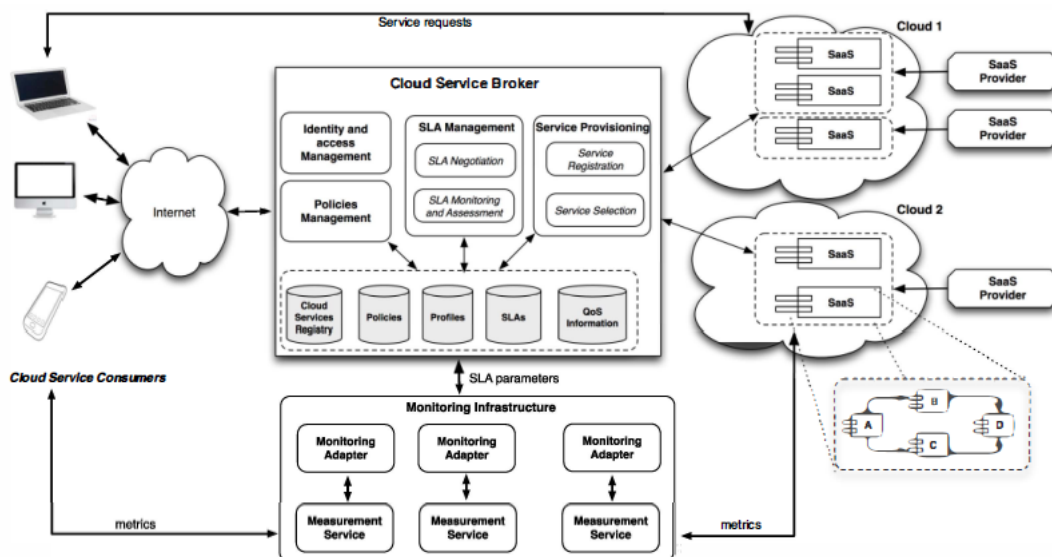


FIGURE 2.1 – Framework d'un CSB pour la sélection de SaaS basée sur le SLA, E. Badidi [32]

2.2.2 A Services Brokerage and Business Model for SaaS, Moore & Mahmoud

Moore et Mahmoud proposent dans “*A Service Brokerage and Business Model for SaaS*” un Broker de Services “fiable” pour applications SaaS [36].

Ce Broker réalise l’Intermédiation et l’Agrégation d’applications SaaS, et il agit comme un “container pour la publication d’applications SaaS hétérogènes par des fournisseurs indépendants” (Interopérabilité) en gérant les problèmes d’intégration de données qui surviennent lors des échanges (de données) entre les différents systèmes.

La technologie principale derrière ce Cloud Service Broker est celle des Services Web au vu des nombreuses applications SaaS communiquant principalement par Services Web à l’époque (2009). D’un côté, ce Broker a pour but de gérer un environnement persistant (une plate-forme) qui permet aux applications SaaS d’être inscrites, retrouvées, souscrites par des utilisateurs pour être ensuite exécutées. D’un autre côté, il vise à fournir un environnement qui donne aux utilisateurs la possibilité d’accéder à leurs ressources sur Internet. Les auteurs centrent leur travail sur trois aspects de “fiabilité” :

- (a) La confiance à travers un système sûr et fiable mis en place (Trust) : authentification, communication sécurisée, disponibilité des services, intégrité des données, performances cohérentes avec leur description ;
- (b) La validation des données par un mapping approprié des données et des techniques de communication (Data Mapping) ;
- (c) Une recherche de SaaS plus précise grâce à la proposition d’une “Interface Attributes”, une interface permettant aux SaaS Providers d’ajouter des contraintes, des conditions et des informations supplémentaires au fichier WSDL (voir Figure 2.3).

Pour les auteurs, il n’y avait pas de réels standards pour les communications entre applications SaaS. Ainsi, ils ont développé un standard de communication pour le mapping d’objets de données via des extensions aux fichiers WSDL connus dans la technologie des Services Web. En plus d’obtenir dynamiquement les interfaces et les détails de localisation des Services Web, leur Broker est également capable de relier des données variées à différents Web Services.

Ce Broker crée un environnement sécurisé et fiable pour accéder aux ressources et gérer les applications SaaS.

L’architecture haut-niveau, représentée à la figure 2.2, se compose des éléments suivant : *Local UDDI Registry*, *UDDI Database*, *Client Interface*, *Service Providers*, *Data Integration Manager*, *Web Service Broker*, *WSDL Manager* et *Trust Manager* décrit dans le travail des deux auteurs [36].

Le composant *Broker* aide à gérer les communications et l’échange de données entre les utilisateurs – qui désirent accéder au système – et les fournisseurs de SaaS – qui désirent promouvoir leurs services dans un environnement de confiance – .

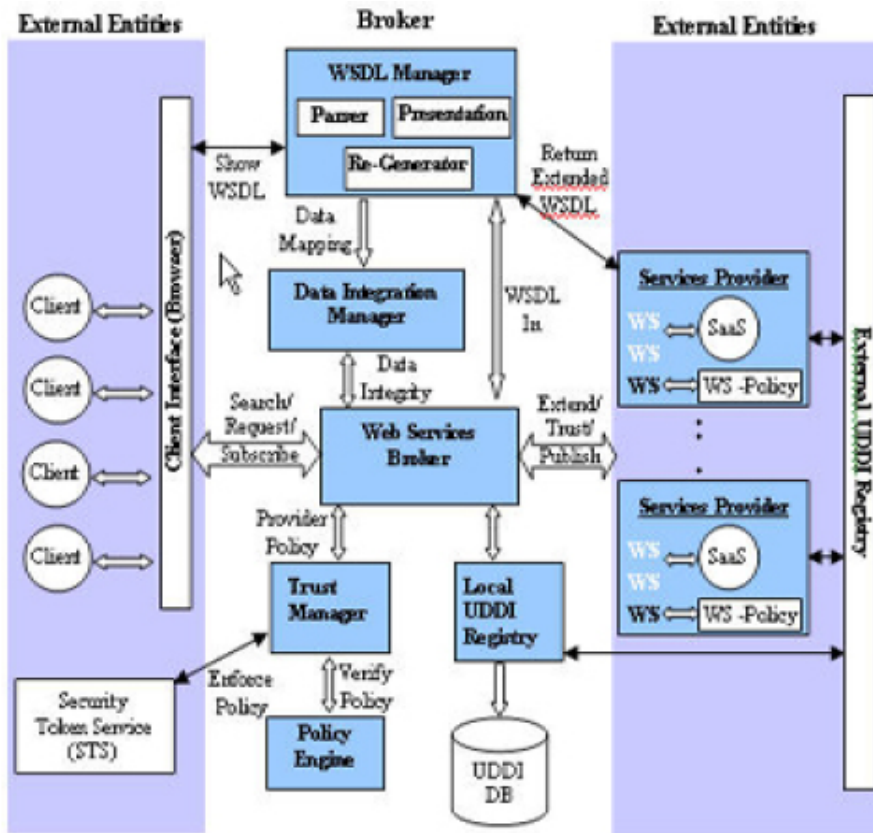


FIGURE 2.2 – Architecture haut-niveau du Broker, Moore & Mahmoud [36].

WSDL File: http://localhost/implementation/mortgagecalc.aspx?wsdl

Service Name: MortgageCalc

Method: CalculateMthlyPayment

Name	Data Type
Interest Rate	Float
Payment Frequency	String
Mortgage Amount	Currency
Amortization Period	Integer

Name	Data Type
Mortgage Payment	Currency

Name	Value	Description
Cost	\$0.60	This rate is per use
Currency	US	Based on current Exchange rates
Availability	24/7	This service has a 99.9% up time

Update Reset Cancel

FIGURE 2.3 – Écran du formulaire “Add Interface Attribute”, Moore & Mahmoud [36].

2.2.3 OPTIMIS, EU-FP7

OPTIMIS (Optimized Infrastructure Services³) est un projet de l'EU-FP7 concernant l'offre de services d'Infrastructure Cloud. Ce projet a pour but de permettre aux organisations d'externaliser automatiquement leurs services et leurs applications chez des Cloud Providers de confiance et auditables [37].

OPTIMIS concerne tant les Cloud Providers que les SaaS Providers.

OPTIMIS vise à optimiser les services clouds en apportant un framework d'architecture ouvert et un toolkit de développement qui couvrent le cycle de vie entier des services clouds de la manière suivante :

- Construction de l'application : le SaaS Provider construit l'application SaaS et la configure pour son déploiement et son exécution sur l'infrastructure fournie ;
- Déploiement sur le Cloud : L'application SaaS est placée sur l'infrastructure la plus adéquate pour son exécution. La négociation des termes du SLA entre le SaaS Provider et le Cloud Provider est un processus important durant le déploiement ;
- Exécution des opérations et interactions : d'un côté, le SaaS Provider réalise un ensemble d'opérations de gestion dans le but d'atteindre des objectifs spécifiés durant la construction de l'application. D'un autre côté, le Cloud Provider réalise les actions pour consolider et distribuer les charges de travail de l'application, répliquer et redistribuer les ensembles de données, ainsi que déclencher les actions d'adaptation des capacités afin d'adhérer aux conditions convenues dans le SLA.

OPTIMIS donne aux SaaS Providers la possibilité de prendre les bonnes décisions de déploiement pour leurs applications, d'orchestrer facilement leurs Infrastructures Clouds et d'exécuter leurs applications dans un environnement Cloud sur base de règles stratégiques (policies) et préférences requises en termes de confiance, risque, efficacité écologique et coûts. OPTIMIS nomme ces quatre catégories de préférences : (a) Trust (Reputation Management), (b) Risk (Risk Assessment), (c) Eco-efficiency (Green Assessment) et (d) Cost (Cost and economical sustainability). Cet ensemble de critères de préférences est appelé "*TREC policies*".

Le projet développe le "Toolkit OPTIMIS" qui comprend un ensemble d'outils destinés aux Cloud Providers, SaaS Providers, développeurs d'applications et SaaS Users. Ces outils doivent permettre de construire et d'exécuter des services clouds dans ce que les auteurs appellent "*the Best-Execution Venues under the TREC policies*" (le meilleur endroit d'exécution sous les critères TREC).

En utilisant le Toolkit OPTIMIS déployé dans les datacenters en complément de la gestion du Cloud et de l'orchestration des plate-formes, les SaaS Users peuvent facilement disposer d'infrastructures Multi-Cloud, et donc optimiser l'utilisation de leurs ressources auprès de multiples fournisseurs ; ce qui se fait de manière transparente, interopérable et indépendante des architectures [38].

Le Toolkit se compose d'un ensemble de cinq composants fondamentaux et du Basic Toolkit (voir Figure 2.4). Ces cinq composants⁴ sont : le *Service Builder* (SB), l'*Admission Controller* (AC), le *Deployment Engine* (DE), le *Service Optimizer* (SO) et le *Cloud Optimizer* (CO). Par leur implémentation chez les SaaS Providers et les Cloud Providers, ils donnent accès

3. www.optimis-project.eu

4. Ces composants sont décrits dans [39]

à une variété d'architectures de type Inter-Cloud : architectures fédérées, architectures multi-cloud, agrégation de ressources, architecture hybrid cloud. Une représentation de ces architectures est montrée par la Figure 2.4.

Le *Basic Toolkit* réalise des analyses quantitatives et qualitatives afin d'amener à des prises de décisions optimales quelque soit le composant invoqué. Les composants à mettre en place diffèrent selon que l'organisation joue le rôle de SaaS Provider ou de Cloud Provider.

OPTIMIS propose trois scénarios d'utilisation afin d'expliquer les capacités offertes [37] :

- (a) Un scénario Cloud Programming Model démontre comment le framework OPTIMIS peut être utilisé pour les étapes du cycle de vie des services telles que la migration, le développement et l'exécution.

- (b) Un scénario Cloud Brokerage démontre l'utilisation d'un Broker pour un Multi-Cloud et une Fédération entre de multiples Cloud Providers (Agrégation).

- (c) Un scénario Cloudbursting montre comment les organisations peuvent adapter leur infrastructure en utilisant diverses ressources provenant de fournisseurs tiers sur base d'un ensemble de critères "TREC".

Dans un scénario de Cloud Brokerage, les composants sont utilisés pour permettre le déploiement Multi-Cloud d'une application impliquant plusieurs Cloud Providers. Un Broker réalise alors l'agrégation des ressources à partir de plusieurs Cloud Providers pour les mettre à disposition des SaaS Providers (voir Figure 2.5).

Le Broker fournit la possibilité d'obtenir des environnements qui sont les mieux adaptés au déploiement des composants selon un ensemble donné de ressources physiques (mémoire, CPU, bande passante, etc.) et un ensemble de contraintes et/ou préférences donné par les critères TREC (prise de décision). Et cela en utilisant un ou plusieurs Cloud Providers fournissant les services compatibles (Agrégation).

Le Broker réalise également la négociation et la gestion de SLAs, ainsi que les mécanismes pour faire correspondre les fonctionnalités proposées par les Cloud Providers avec les besoins des SaaS Providers. Selon OPTIMIS, le fait de permettre aux SaaS Providers d'opter pour plusieurs Cloud Providers dans l'hébergement des différents composants de leurs applications donne lieu à un environnement Cloud plus compétitif et dynamique.

Le Broker joue le rôle d'agrégation de services en offrant la possibilité d'un Multi-Cloud et d'une Fédération de Clouds entre les Cloud Providers qui peuvent en interne utiliser les services d'autres Cloud Providers afin d'assurer la flexibilité des demandes. Il propose également l'intermédiation de services en s'occupant de la gestion des identités, de la sécurité et de l'intégrité des données ainsi qu'en supportant l'encodage end-to-end entre les multiples Cloud Providers.

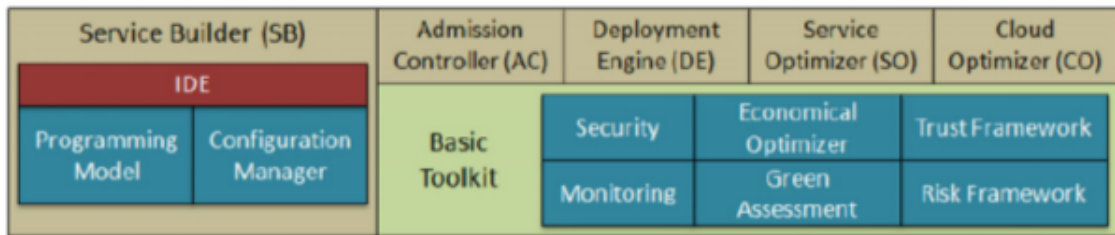


FIGURE 2.4 – Vue haut-niveau des composants du Toolkit OPTIMIS, Ferrer et al. [39]

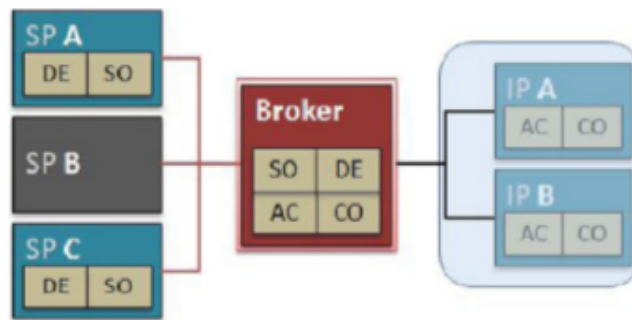


FIGURE 2.5 – Dans un scénario de Cloud Brokerage, le Broker réalise l'agrégation des ressources depuis plusieurs fournisseurs (IP) pour les mettre à disposition des utilisateurs (SP). Ferrer et al. [39]

2.2.4 SLA@SOI, EU-FP7

Le projet *SLA@SOI* (<http://sla-at-soi.eu/>) fait également partie du programme EU-FP7 (Juin 2008-Juillet 2011) et est réalisé en coopération avec un consortium d'industries et d'instituts académiques européens. Ce projet se concentre sur la gestion automatisée des Service Level Agreements et sur le monitoring du cycle de vie des services clouds (SaaS et Infrastructure Cloud).

Au cours de ce projet SLA@SOI, Theilmann et al. ont développé un framework et une architecture de référence ([40, 41]) pour la gestion automatisée des Service Level Agreements. L'architecture a pour objectif d'apporter une solution pour

- (a) supporter la gestion des SLAs dans des environnements multi-niveaux (Cloud et SaaS) et leur intégration dans différentes infrastructures orientées service (Service-Oriented Infrastructure),
- (b) couvrir le cycle de vie complet des SLAs et des services,
- (c) être utilisée dans des scénarios couvrant de multiples domaines industriels.

Le projet envisage une infrastructure orientée services prête à l'emploi qui valorise l'économie de services de manière flexible et fiable.

Une telle infrastructure couvre trois caractéristiques et challenges majeurs [42] :

- la prédiction et la fiabilité par des caractéristiques de service qui doivent être prédictifs et réalisables à l'exécution ;
- la gestion transparente des Service Level Agreements qui définissent les conditions exactes sous lesquelles les services sont consommés ;
- l'automatisation de l'entièreté du processus de négociation des SLAs, ainsi que celle de l'obtention, l'utilisation et le monitoring des services tout en permettant une consommation de ces services hautement dynamique et flexible.

Un exemple de scénario typique est le cas d'un Service Provider souhaitant offrir des services sous des conditions de SLAs différentes, fiables et ajustables dont la négociation avec l'utilisateur est réalisée de manière automatisée. Les caractéristiques de SLAs peuvent contenir de multiples critères non-fonctionnels tels que la sécurité, la performance, la disponibilité et la fiabilité du service.

Le projet comprend des recherches reposant sur quatre scénarios clés orientés sur le monde de l'industrie : ERP Hosting, Entreprise IT, Service Aggregation et e-Gouvernement.

Dans le but de proposer une telle architecture dans les quatre scénarios soulignés ci-dessus, un composant *Resource Manager* est nécessaire afin de supporter un certain nombre d'APIs fonctionnelles.

L'infrastructure de gestion des SLAs requiert diverses opérations telles que le *Services Offer Listing*, le *Service Instance Creation*, le *Service Instance Monitoring* et le *Service Instance Adjustment*.

Les fonctionnalités innovantes du framework sont :

- (a) le framework de création de contrats électroniques (SLA) automatisé⁵,
- (b) la compréhension des opérations business supportées par l'infrastructure,

5. Les auteurs soulignent que Amazon EC2 n'utilise pas de SLAs négociables ni exécutés électroniquement.

- (c) l’exploitation des technologies de virtualisation au niveau de l’infrastructure pour l’exécution des SLAs,
- (d) des méthodes d’ingénierie pour la création et la gestion de services s’exécutant de manière prévisibles.

Un SLA d’infrastructure (Infrastructure SLA – iSLA) dans le framework SLA@SOI défini (a) les attributs de l’infrastructure, (b) les objectifs du niveau de services, (c) les règles pour une gestion dynamique⁶.

2.2.5 RESERVOIR, EU-FP7

RESERVOIR (REsources and SERvices VirtualizatiOn wIthout baRriers) est un des multiples projets de l’EU-FP7.

Ce projet permet le déploiement de SaaS à grande échelle et la gestion des services IT complexes au sein de diverses plate-formes informatiques réparties géographiquement. *RESERVOIR* développe des technologies afin d’aider les SaaS Providers à gérer les problèmes de flexibilité rencontrés lors d’un approvisionnement chez un unique Cloud Provider [43].

RESERVOIR explore la notion de Clouds Fédérés en partant du principe qu’aucun fournisseur ne peut, à lui tout seul, opérer et satisfaire complètement toutes les demandes des utilisateurs en même temps. Le projet vise donc à créer un modèle dans lequel de multiples Cloud Providers indépendants peuvent coopérer en mettant à disposition des autres Providers l’excédent de leurs capacités, dans une situation où ceux-ci nécessiteraient des ressources additionnelles (Agrégation, Fédération Cloud). Les Cloud Providers contrôlent des sites faisant partie de *RESERVOIR*, et ils forment une fédération de collaborateurs, détenant et gérant des infrastructures physiques sur laquelle s’exécutent les applications SaaS. Ensemble, ils créent un “réservoir” de Clouds [44]. Ces Cloud Providers peuvent ainsi partager leurs ressources à un prix convenu avec d’autres fournisseurs [45]. Par là, *RESERVOIR* essaye de briser les barrières qui peuvent exister entre les environnements Clouds incluant les problèmes de sécurité, de gestion et géographiques.

Les services d’Infrastructures Clouds peuvent être répartis entre différents sites pour profiter d’un Inter-Cloud tout en respectant le SLA attachés à ces services [42]. Cette vision d’une architecture qui offre des partenariats dynamiques entre Cloud Providers a pour objectif de créer un pool “infini” de ressources d’infrastructures IT tout en préservant l’autonomie des prises de décisions technologiques et business.

Cette architecture fournit une séparation claire des intérêts et des responsabilités entre les couches opérant à différents niveaux d’abstraction dans le but de cacher les détails de l’infrastructure bas niveau et les décisions de la gestion haut-niveau des SaaS Providers.

L’architecture du site *RESERVOIR* est composée, du plus haut niveau au plus bas niveau d’abstraction, par un *Service Manager* (SM), un *Virtual Execution Environment Manager* (VEE Manager) et un *Virtual Execution Environment Host* (VEE Host) telle que montrée à la Figure 2.6.

Ces composants peuvent être décrit de la manière suivante :

- Le *Service Manager* est le plus haut niveau d’abstraction (niveau application). Il interagit avec les SaaS Providers afin de recevoir les besoins en ressources des services, de négocier les prix et de traiter la facturation.

6. e.g. règles pour l’adaptation de l’utilisation du CPU.

Ce composant effectue le déploiement et la mise à disposition des Virtual Execution Environments (VEEs) sur base des besoins en ressources et infrastructures.

Le Service Manager est également en charge de la tenue des accords spécifiés dans le SLA en ajustant la capacité des instances des VEEs et des ressources nécessaires. Pour ce faire, il effectue le monitoring de l'état courant des différents systèmes.

- Le *Virtual Execution Environment Manager* interagit avec le Service Manager au niveau supérieur d'abstraction (niveau application), avec les VEEHosts au niveau inférieur (niveau infrastructure), et avec le VEEManager des autres sites permettant la Fédération Cloud. Le VEEM est responsable du choix optimal des VEEs au sein des VEEHosts sujets aux contraintes déterminées par le Service Manager. Il est aussi responsable de la fédération des sites distants. RESERVOIR utilise OpenNebula⁷ comme VEEManager. OpenNebula est un moteur d'infrastructure virtuelle open source qui fournit des fonctionnalités de contrôle, déploiement et monitoring de machines virtuelles [46].
- Le *Virtual Execution Environment Host* interagit avec le VEEManager pour réaliser les décisions de la gestion IT parmi un ensemble de plate-formes de virtualisation. Le VEEHost est responsable du contrôle et du monitoring des VEEs et de leurs ressources.

RESERVOIR peut être vu comme un environnement idéal pour le déploiement d'applications dont les besoins en ressources diffèrent dans le temps.

La combinaison des standards ouverts et logiciels open-source peut être considérée comme ROI (Return On Investment) de la contribution de RESERVOIR dans le développement économique et scientifique.

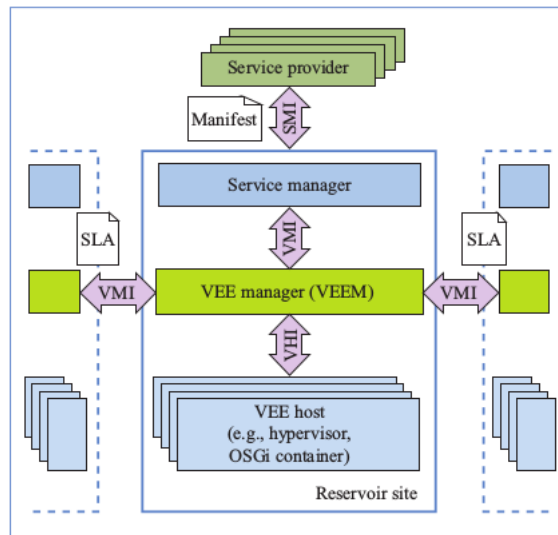


FIGURE 2.6 – Architecture de RESERVOIR : composants principaux et interfaces [44]. (OSGi : Open Services Gateway initiative; SMI : Service Management Interface; VMI : VEE Management Interface; VHI : Virtual Host Interface).

7. openebula.org

2.2.6 mOSAIC, EU-FP7

mOSAIC – Open source API and platform for multiple clouds – est un large projet faisant partie du programme EU FP7. L’objectif de ce projet est d’améliorer l’état des connaissances du Cloud Computing par la création, la promotion et l’exploitation d’une plate-forme ainsi que d’une API open-source pour le Cloud. Cette plate-forme et cette API ont pour objectifs de pouvoir développer des applications orientées Multi-Cloud, ainsi que la négociation de services d’Infrastructure Cloud en fonction des besoins de ces applications et des requêtes effectuées par les utilisateurs. [47].

mOSAIC se veut transparent et simple dans l’accès à des ressources provenant de différents fournisseurs afin d’éviter des solutions propriétaires fermées sur l’extérieur (lock-in) [48].

Par mOSAIC, les applications sont capables de décrire leurs fonctionnalités, qualités de services et interfaces grâce à une “ontologie cloud” (détaillée dans [47]), qui leur permet de communiquer entre elles via l’API développée. L’ontologie se base sur plusieurs taxonomies Cloud, et elle est construite sur le langage OWL (Web Ontology Language). Cette ontologie peut être utilisée pour une recherche sémantique de services ou leur composition.

L’API et la plate-forme rendent possibles le développement et le déploiement d’applications dans un environnement reposant sur de multiples Clouds. La plate-forme implémente un Brokerage multi-agents, et elle met automatiquement à disposition de l’utilisateur les services clouds selon un ensemble de Clouds prédéfinis, et cela sans implication directe de l’utilisateur. Le Brokerage recherche les services qui vont correspondre aux requêtes des applications/utilisateurs. Il peut composer différents services requis dans le cas où une solution directe n’est pas trouvée (Agrégation).

Le *Broker* développé pour mOSAIC est défini comme une entité qui gère l’utilisation, la performance et la mise à disposition de services d’Infrastructure Cloud. Il négocie également les relations entre les Cloud Providers (fournisseurs) et les Cloud Users (utilisateurs-consommateurs). Les fonctionnalités du Broker sont réalisées par le “Cloud Agency”, qui est le composant central de mOSAIC.

Les principaux composants et services de mOSAIC détaillés dans [49] sont les suivants :

- Le “*Ressource Broker*” qui est responsable de la médiation entre utilisateurs et fournisseurs. Il contrôle les niveaux de services à partir d’indicateurs de performance par rapport à ceux définis dans le SLA. Le Ressource Broker contient un “Cloud Agency” et un “Client Interface”.
- Le “*Cloud Agency*” dont la tâche principale est la recherche et la sélection dynamique des Infrastructures Clouds qui répondent le mieux aux ressources requises par les applications des utilisateurs (Cloud Users). Ce Brokerage est réalisé à partir de l’ensemble des ressources mises à disposition par les différents fournisseurs (Clouds Providers). Le Cloud Agency réalise également, pour le compte des utilisateurs, toutes les tâches de gestion des SLAs et leur re-négociation avec les fournisseurs, ainsi que le monitoring et le suivi de l’utilisation des ressources.
- Le “*Client Interface*” s’occupe de demander les ressources additionnelles pour le composant Application Executor.
- L’ “*Application Executor*” gère le déploiement, l’exécution et le monitoring des services au sein de ces ressources allouées.

L'architecture de mOSAIC repose sur le critère de prix/coûts via le Cloud Agency qui prend en considération les différents plans tarifaires proposés par les Cloud Providers. Cette architecture est représentée à la Figure 2.7. Par ailleurs, mOSAIC permet à ces Cloud Providers de facilement se faire connaître auprès de utilisateurs, créant par la même occasion une concurrence entre eux.

En somme, le projet mOSAIC vise à promouvoir des applications Multi-Clouds en fournissant une plate-forme qui permet l'interopérabilité entre différents services Cloud. Cette système facilite la portabilité des applications développées sur différentes plate-formes et permet la recherche de services et leur composition, ainsi que la négociation et la gestion des SLAs. Du point de vue des utilisateurs, le composant important est le Cloud Agency implémentant les fonctionnalités de base : gestion des SLAs, déploiement des services Clouds ainsi que la sélection des services.

mOSAIC émet l'hypothèse que l'architecture des applications est divisée en composants et utilise uniquement l'API proposée. Il s'agit d'une architecture orientée service (SOA). Les développeurs d'applications doivent ainsi spécifier pour chaque composant les besoins en ressources requis (unités de calcul, stockage, communication).

Notons que de nombreuses publications liées au projet mOSAIC peuvent être trouvées dans la littérature et sur le site web du projet⁸. Elles concernent des sujets tels que le Brokerage de services, la gestion des SLAs, le monitoring de services, le multi-cloud, etc.

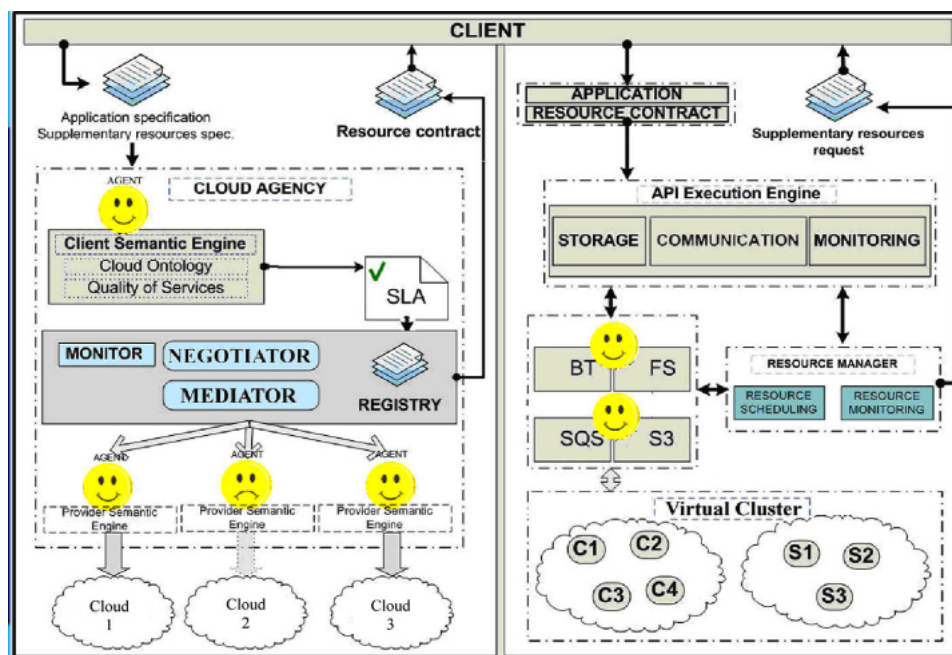


FIGURE 2.7 – Architecture de mOSAIC et du Cloud Agency.

8. <http://www.mosaic-cloud.eu/>

2.2.7 STRATOS, Canada

STRATOS fait partie d'un programme de recherche financé par les fonds canadiens du NSERC, différentes autres organisations canadiennes, et Amazon Web Services (AWS).

STRATOS est un Cloud Service Broker qui se positionne au début de la phase de développement d'applications. Il est présenté par Pawluk et al. dans [50].

Ce Broker est l'un des premiers travaux souhaitant mettre en avant une acquisition de ressources et infrastructures clouds automatisée dans un environnement Multi-Cloud, avec une plate-forme de type Intercloud telle que celle imaginée par Buyya et al. dans [51].

Ce Cloud Service Broker vise à faciliter le déploiement et les modifications à l'exécution des topologies⁹ pour applications SaaS en utilisant des services d'Infrastructure Cloud alloués "on the fly" par différents fournisseurs, et en concordance avec les exigences haut-niveau des SaaS Providers.

Le Broker utilise des éléments Clouds et des services de multiples Cloud Providers en adéquation avec les besoins de déploiement spécifiés par des objectifs haut-niveau. L'allocation optimale des ressources parmi les multiples Clouds est réalisée à l'exécution.

Nous pouvons dire que le Broker effectue ainsi des opérations d'intermédiation et d'agrégation.

Pour obtenir des solutions optimales, le Broker doit effectuer les mesures adéquates des caractéristiques de performances des Cloud Providers. À cet effet, STRATOS utilise le Service Measurement Index (SMI) et des métriques définies par Garg et al. dans le framework SMICloud (voir 2.2.15 et [9]). Les exigences haut-niveau décrivant les objectifs sont exprimées en termes de KPIs. Lorsqu'une requête pour l'acquisition de ressources est envoyée au Broker, celui-ci évalue la requête de manière standardisée grâce à l'utilisation du SMI. Ensuite, il confronte les besoins avec toutes les offres des Cloud Providers afin de sélectionner le(s) offre(s) de ressources qui convien(nen)t le mieux aux besoins.

Le point d'entrée dans le système STRATOS est le composant *CloudManager*. Ce CloudManager reçoit la description des exigences et la topologie des applications par un fichier *Topology Descriptor File* (TDF). Le CloudManager contacte ensuite le composant *Broker* qui détermine l'allocation optimale des ressources parmi les Clouds. Le fichier TDF peut être considéré comme un document SLA entre l'utilisateur et le service de STRATOS. Ce fichier au format XML spécifie les contraintes et les objectifs de configuration de la topologie de déploiement des applications [23].

Le *Broker* résout des problèmes de décision d'acquisition de ressources (Resource Acquisition Decision – RAD).

Un problème RAD se présente lors de deux situations :

(a) à l'initialisation de la topologie de l'environnement de l'application, et (b) à l'exécution lorsqu'un changement dans l'allocation des ressources est exigé selon la règle d'élasticité du serveur d'applications tier.

Le problème RAD est introduit pour la prise de décision et il est lié à un problème d'optimisation multi-critères. Le Broker vise donc à automatiser la décision d'acquisition des ressources entre les Users et les Providers. Il est utilisé pour agréger la connaissance sur les différents services offerts par tous les fournisseurs et apporte ainsi une interface

9. Topologie : environnements d'applications gérés tels que les clusters, noeuds, VMs et leurs caractéristiques

unifiée. Le Broker doit être connecté aux fournisseurs sélectionnés afin d’adapter et de gérer les ressources. Il requiert deux informations de la part des utilisateurs afin de résoudre un problème RAD particulier : la configuration désirée et l’ensemble des objectifs.

Le *CloudManager* réalise les décisions (allocations) “élastiques” en cours via l’*ApplicationManager* tandis que le processus de décision est effectué par le Broker. Tout deux font usage des informations issues du monitoring. Après le déploiement initial, le *CloudManager* peut prendre la décision d’ajouter ou de retirer des ressources si nécessaire. L’*ApplicationManager* contrôle la gestion de l’application à l’exécution selon le modèle des exigences établi.

Afin de pouvoir supporter de multiples Cloud Providers et faire abstraction des différences entre les Clouds, STRATOS utilise l’API delta-cloud¹⁰.

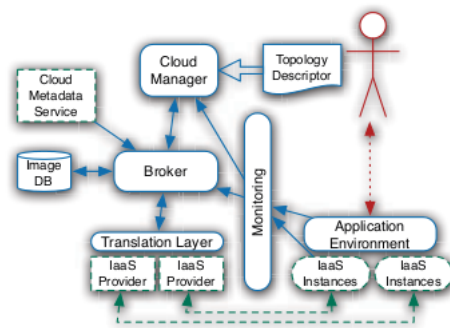


FIGURE 2.8 – STRATOS – Vue architecturale haut-niveau du framework. L’Application Manager n’est pas montré [50].

```

1 <configuration>
2 <topology name="Awesome Cloud" id="topology1">
3 <cluster name="Web Cluster" id="Cluster_Web_1">
4 <node name="Web Host" id="" type="worker" config="
5     small"
6     imgs="img-ef38fa86" publicIP="" privateIP="" region="
7     us-east-1" band-in="" band-out="500">
8 <container name="Tomcat 6" id="">
9 <service name="Simple_Application" id="tom1" />
10 <service name="SNMP" id="snmp" />
11 </container>
12 </node>
13 </cluster>
14 <!-- additional clusters omitted -->
15 <dependencies>
16 <!-- Connect Simple App to MySql -->
17 <dependency from="tom1" to="mysql" />
18 <!-- Connect Load Balancer to Tomcat -->
19 <dependency from="ball" to="tom1" />
20 </dependencies>
21 </topology>
22 <broker>
23 <configurations>
24 <config name="small" id="small">
25 <property name="disk" value="160" unit="GB"/>
26 <property name="cpu" value="2" unit="units"/>
27 <property name="ram" value="4" unit="GB"/>
28 </config>
29 </configurations>
30 <objective name="cost"
31 class="strato.broker.objectives.CostObjective" />
32 </broker>
33 </configuration>

```

FIGURE 2.9 – STRATOS – Contenu d’un fichier Topology Descriptor File (TDF) [50].

10. <http://deltacloud.apache.org/>

2.2.8 InterCloud, Buyya et al.

Dans [51], Buyya et al. introduisent la création d'un environnement de Cloud Computing fédéré, qu'ils appellent "*InterCloud*". Un tel environnement vise à faciliter l'obtention opportuniste, just-in-time et flexible de services tout en atteignant les qualités de service visées sous des conditions variables de charges de travail, ressources et accès réseaux.

Intercloud a donc pour objectif de créer un environnement informatique qui supporte des adaptations dynamiques de services et d'applications (VMs, stockage, bases de données, etc.) auprès de multiples Cloud Providers, qui sont répartis en des endroits géographiques différents, dans le but de répondre à deux problèmes de localisation des ressources :

(1) la difficulté pour les utilisateurs d'infrastructure (Cloud Users/SaaS Providers) de déterminer à l'avance le meilleur endroit pour héberger leurs applications SaaS puisqu'ils pourraient ne pas connaître l'origine des utilisateurs de leurs propres services (SaaS Users), et (2) la difficulté pour les Cloud Providers d'offrir les qualités de services attendues pour tous leurs utilisateurs de services répartis géographiquement.

Les auteurs avancent que ces problèmes nécessitent donc la réalisation de mécanismes pour une fédération sans frontières des datacenters des Cloud Providers et/ou des SaaS Providers. Ces datacenters doivent alors disposer d'applications dynamiquement adaptables dans de multiples domaines (intégration et interopérabilité) dans le but d'atteindre les QoS ciblées par les utilisateurs.

Grâce à leur recherche, les auteurs ont pour objectifs de répondre aux besoins de la "prochaine génération du Cloud Computing". Pour eux, les Cloud Providers devraient être capables de :

- (a) Adapter dynamiquement la capacité de leurs ressources lors d'augmentations soudaines des demandes en charge de travail grâce à la location des capacités de stockage et de traitements informatiques (computational capabilities) chez d'autres Cloud Providers ;
- (b) Fonctionner comme un marché orienté vers une fédération de location des ressources où les SaaS Providers hébergent leurs applications basées sur des Service Level Agreements et sous l'influence des prix compétitifs du marché ;
- (c) Fournir à la demande, des services fiables et rentables basés sur des technologies de virtualisation et qui respectent des standards de QoS élevées avec des coûts minimisés.

Les composants clés pour permettre cette Fédération de Clouds et l'auto-adaptation des applications SaaS sont : (a) les Cloud Coordinators, (b) le Broker et (c) le Cloud Exchange. L'architecture d'Intercloud est centralisée et construite autour du Cloud Exchange. Cette entité centrale *Cloud Exchange* (CEx) joue le rôle d'une place de marché où les fournisseurs et les utilisateurs vont interagir. Les ressources y sont délivrées aux utilisateurs de Cloud qui peuvent être également d'autres Cloud Providers ou Brokers dans l'optique de la fédération [25]. Le CEx joue le rôle d'un registre d'information qui enregistre les coûts d'utilisation, les ressources disponibles et les demandes actuelles de Clouds. Il réalise également l'agrégation des demandes d'Infrastructures Cloud provenant des Brokers et les compare aux ressources publiées par les Cloud Coordinators et qui sont disponibles à ce moment. De plus, le Cloud Exchange donne aux Cloud Coordinators et aux Cloud Brokers la possibilité de localiser les fournisseurs et les utilisateurs concernés par les offres adéquates.

Les *Cloud Coordinators* (CCs) gèrent les membres de la fédération en effectuant les

communications avec le CEx. Ils mettent à jour périodiquement leurs disponibilités, leurs prix, et leurs critères de SLA dans le registre d'information des ressources Clouds maintenu par le CEx. Chacun des Cloud Providers participant dans l'Intercloud doit posséder un Cloud Coordinator.

Un *Cloud Broker* travaille pour le compte des utilisateurs et identifie les Cloud Providers appropriés via le CEx. Il négocie ensuite avec les CCs pour une allocation des ressources qui satisfont les qualités de service demandées par les utilisateurs. Chaque utilisateur de la fédération doit instancier un Cloud Broker qui peut dynamiquement établir des contrats avec les CCs via des négociations réalisées par le CEx. Les Cloud Brokers envoient les requêtes au registre d'information du CEx afin de prendre connaissance des offres de SLA, des ressources disponibles auprès des Cloud Providers de la fédération et de leurs prix. Un mapping vers les services des Cloud Providers les plus adéquats sera finalement réalisé.

L'architecture du *Broker et ses composants* (User Interface, Core Services, Execution Interface, Persistence) sont décrit dans [51].

Le Brokerage dans cet environnement InterCloud est réalisé sur base du SLA et du prix. La répartition géographique des ressources étant connue du Cloud Exchange, un Brokerage sur la localisation des ressources est également possible.

2.2.9 Contrail, EU FP7

Contrail – Open Computing Infrastructures for Elastic Services – est un framework, issu du programme européen FP7, pour une fédération de Clouds qui développe une approche de gestion des identités d'accès telles que l'authentification et les contrôle d'accès [31, 52].

Contrail offre à ses utilisateurs l'accès à de multiples ressources Clouds via un seul point d'accès et les décharge de la gestion d'accès différents auprès de chaque Cloud Providers individuellement. Ces ressources clouds permettent aux Cloud Users/ SaaS Providers l'exécution de leurs applications et du stockage de leurs données.

Les services de la Fédération effectuent le Brokerage des accès aux infrastructures clouds sur base de SLA et assurent que la meilleure ressource disponible soit sélectionnée ainsi que les diverses ressources puissent être accédées sans interruption.

Ce Cloud Federation Broker réalise l'agrégation des services en assurant leur interopérabilité, leur performance et leur sécurité.

Contrail se concentre également sur une technologie de Single Sign-On afin que les utilisateurs ne s'identifient qu'une seule fois pour travailler avec l'entière de la fédération.

Le framework est construit autour d'une entité centralisée qui agit comme point d'entrée unique à la fédération des Cloud Providers. Cette entité observe les états des ressources des Cloud Providers dont il négocie les relations avec les Cloud Users par l'utilisation de SLAs. Des heuristiques d'optimisation des coûts et des performances sont implémentés pour réaliser la sélection des Cloud Providers adéquats sur base des spécifications du SLA et de la minimisation des coûts. La situation géographique et d'autres méta-informations pertinentes relatives à chaque Cloud Provider sont également connues du système. Le Brokerage est réalisé sur base des spécifications du SLA.

Un module *Federation Runtime Manager* (FRM) est responsable de la sélection du service en fonction du SLA et de la minimisation des coûts.

Des composants *Adapters Layers* constituent des adaptateurs pour chaque Cloud Providers de la fédération et permettent la communication entre les composants de gestion de la fédération et les Clouds : les *Internal Adapters* sont utilisés pour les Clouds sous le software Contrail et les *External Adapters* pour les Clouds qui ne s'exécutent pas sous le software Contrail.

De son côté, l'utilisateur définit des règles de politique d'utilisation : limites sur les quantités de ressources à consommer, exigences en terme de privacy, etc. [23].

2.2.10 Federated Cloud Management : an Architecture for Integrating IaaS Cloud Systems, Marosi et al.

Federated Cloud Management (FCM) est une architecture pour l'agrégation des capacités de plusieurs Cloud Providers afin de créer des systèmes d'infrastructures clouds intégrés [53]. Cette architecture joue un rôle de point d'entrée pour une fédération de clouds, et incorpore les concepts de méta-brokering, de cloud brokerage et de déploiement de services à la demande. Cette architecture traite également le problème d'élasticité en mettant en place des cloud brokers qui vont gérer les machines virtuelles en accord avec les demandes courantes des applications utilisateurs.

Le composant de méta-brokering fournit une exécution de service transparente pour les utilisateurs en rendant possible l'interconnexion du système avec les différentes solutions du Cloud Broker disponibles dans le système (intégration et interopérabilité). Les Cloud Brokers sont responsables de la gestion du nombre et de la localisation des instances des machines virtuelles utilisées pour les requêtes de services clouds reçues.

2.2.11 Cloud service delivery across multiple cloud platforms, Houdi et al.

Houdi et al. proposent un framework dont l'objectif est d'allouer, à partir de multiples datacenters, les ressources et services clouds aux utilisateurs en faisant la requête par l'intermédiaire d'un Cloud Broker [54].

Le Broker décompose les requêtes des utilisateurs et assure l'obtention des ressources à partir de multiples fournisseurs. Les auteurs ont ainsi développé un algorithme de décomposition ("*split algorithm*") pour distribuer de manière efficiente les requêtes des utilisateurs auprès de plusieurs Cloud Providers en vue de diminuer les coûts pour ces utilisateurs.

La décomposition est formulée sous forme de Mixed Integer Program (MIP). Lorsque la requête est divisée par le Broker, chacun des Cloud Providers reçoit une sous-requête pour l'acquisition et l'instanciation des ressources en utilisant son propre Cloud Manager (OpenStack, OpenNebula) de datacenter.

Ce travail se concentre sur l'instanciation des liens Inter-Clouds entre les réseaux des fournisseurs. Le but est de trouver la décomposition optimale des requêtes auprès des multiples Cloud Providers tout en minimisant le prix que l'utilisateur devra payer pour la totalité des services. Le problème de décomposition ne considère que l'aspect financier (prix) indiqué par les Cloud Providers.

2.2.12 A brokerage-based approach for cloud service selection – CSP-index, Sundareswaran et al.

Sundareswaran et al. proposent une architecture de Brokerage où un Cloud Broker est responsable de la sélection de services [17]. Le Broker comprend une technique d'indexation appelée “*Cloud Service Provider Index – CSP-index*”.

Cette méthode d'indexation peut gérer des informations provenant d'un large nombre de Cloud Providers. En plus de cette méthode, ils développent des algorithmes pour une sélection efficace des services. Ces algorithmes sont capables d'établir un classement des Cloud Providers potentiels selon des critères donnés et de les agréger si nécessaire (Agrégation).

Cette architecture comporte trois types d'acteurs : les Cloud Providers, le Cloud Broker et les Cloud Users.

Le Cloud Broker possède des contrats avec les Cloud Providers et collecte leurs propriétés (e.g., type de service, coût unitaire et ressources disponibles) ainsi que les exigences des utilisateurs pour ensuite analyser et indexer les Cloud Providers selon les similarités avec leurs propriétés. Lorsqu'une requête est reçue d'un utilisateur pour la sélection d'un service, le Broker va parcourir l'index afin d'identifier une liste ordonnée des Providers candidats selon leur pertinence envers les exigences de l'utilisateur. Cette liste forme la base pour la décision finale de l'utilisateur.

Le CSP-index est un B^+ -Tree qui est représenté de la façon suivante :

$\langle Key_{sp}, SID, p_1, p_2, \dots, p_{10} \rangle$, où Key_{sp} est la clé d'indexation, SID est l'identité du Cloud Provider, p_i est la $i^{ème}$ propriété du Cloud Provider.

Ces propriétés sont le type de service, la sécurité/privacy (haute, moyenne, faible sur base de la sécurité des services), la qualité de service (haute, moyenne, faible sur base de cotation), l'unité de mesure, l'unité de prix (e.g. heure, mois, année), la taille des instances (micro, small, medium, large, extra-large), le système d'exploitation (Linux, Windows), le prix (effectif à l'utilisation), la sensibilité du prix selon les régions et les sous-traitants.

L'algorithme de construction de l'index comporte trois étapes majeures. Premièrement, les propriétés du Cloud Provider sont encodées. Ensuite vient l'encodage des relations entre les Cloud Providers construites par les sous-traitants. Finalement, les Cloud Providers sont regroupés (“clusterisés”) sur base de leur encodage afin de construire la clé d'index. L'indexation aide le Broker à organiser les Cloud Providers de manière à faciliter un accès rapide à l'information.

La sélection de service est réalisée de la façon suivante :

Un utilisateur envoie une requête au Broker pour la sélection d'un service. Cette requête spécifie quelles sont les propriétés et les valeurs attendues des Cloud Providers. Une requête de sélection de services d'Infrastructure Cloud est un t -uple de la forme :

$\langle Q = (QP_1 : D_1), (QP_2 : D_2), \dots, (QP_k : D_k) \rangle$, où QP_i et $(1 \leq i \leq k)$ est la propriété que l'utilisateur demande aux fournisseurs de posséder, et où D_i décrit les valeurs attendues pour cette propriété QP_i .

Cet algorithme de recherche “Cloud Service Selection” comporte quatre phases :

(1) encodage de la requête, (2) recherche des k voisins les plus proches, (3) raffinement, (4) prise en considération de critères spéciaux.

Les résultats d'une étude de la performance de cet index seront trouvés dans ce même travail [17].

2.2.13 Cloud service recommendation and selection for enterprises, Yan et al.

Yan et al. proposent dans “*Cloud service recommendation and selection for enterprises*”[55], un framework dont l’objectif est de fournir aux entreprises des recommandations sur base de cotations, de même que des outils de sélections basés sur des règles de comportements afin d’accélérer et automatiser l’acquisition de services d’Infrastructure Cloud.

Ces recommandations et sélections doivent être réalisées en concordance avec les exigences business des entreprises et respecter leurs règles de politiques (“policies”) et de standards.

Le framework se compose de deux modules principaux, le module Policy Implementation Engine et le module Cloud Intelligence, ainsi que d’un catalogue de services.

Le *Catalogue de services* est utilisé comme répertoire pour enregistrer toutes les spécifications des différentes offres de services des Cloud Providers provenant soit d’un Cloud Privé ou soit d’un Cloud Public. Il classe les services clouds et extrait les caractéristiques et les fonctionnalités catégorisées afin de faciliter l’analyse en vue de l’optimisation de la sélection.

Le *Policy Implementation Engine* est conçu de façon à considérer les interdépendances et les conformités envers les besoins et exigences des utilisateurs, au même titre que les règles de stratégies des entreprises et les régulations gouvernementales. Les points communs entre les fonctionnalités extraites des services clouds rendent possible les comparaisons.

Les exigences business des utilisateurs peuvent être traduites en un ensemble de critères basés sur les fonctionnalités communes des services. La fonction d’utilité d’un ensemble de services clouds sera représentée par un indice de performance sur le critère dérivé des exigences utilisateurs.

Dans cette approche, une détection automatique de conflit et son explication sur base des règles de conformité est d’abord fournie par le Policy Implementation Engine (PIE).

Après cette analyse de conflit, le module PIE génère toute les solutions faisables dans un domaine fini, ou bien il aide l’utilisateur à donner plus de précision quant à ses exigences si aucune des solutions n’est réalisable. Les solutions doivent satisfaire les exigences et respecter les règles données par l’entreprise.

La sélection de service dans le module PIE est modélisée à l’aide d’un *problème de satisfaction par contrainte* (Constraint Satisfaction Problem – CSP).

Le Policy Implementation Engine automatise la procédure de résolution en utilisant des techniques de programmation par contrainte (Constraint Programming – CP).

Le module *Cloud Intelligence* est le composant clé du framework. Son but est d’évaluer et d’optimiser les solutions proposées par le Policy Implementation Engine selon les préférences des utilisateurs qui sont basées sur de multiples critères à partir de métriques quantitatives pour les services d’infrastructure cloud (prix, SLA, temps de réponse, etc.), d’une intelligence collective à partir d’un intranet et de médias sociaux, de feedbacks et cotation d’utilisateurs internes, et de préférences individuelles.

Le mécanisme de cotation et de feedback donne aux utilisateurs la possibilité de partager leurs expériences sur l’utilisation de différents services clouds du catalogue. Les notations et les feedbacks deviennent alors des critères de valeur pour l’évaluation du module Cloud

Intelligence.

Ces nombreux critères pris en considération offrent une recommandation et une sélection optimale pour les utilisateurs.

Plusieurs algorithmes MCDM peuvent être appliqués au module Cloud Intelligence afin de maximiser la fonction d'utilité de l'utilisateur, sur base des exigences et des besoins pour ses activités business. Les entreprises ont la possibilité d'ajuster, selon leurs préférences, le poids des différents critères provenant de multiples sources connectées au Cloud Intelligence pour les services clouds à sélectionner.

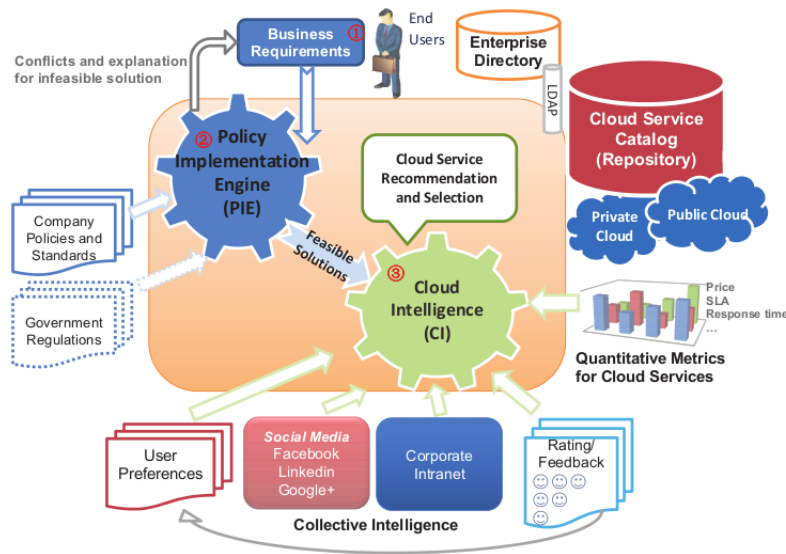


FIGURE 2.10 – Framework pour la sélection et la recommandation de services Cloud, Yan et al. [55].

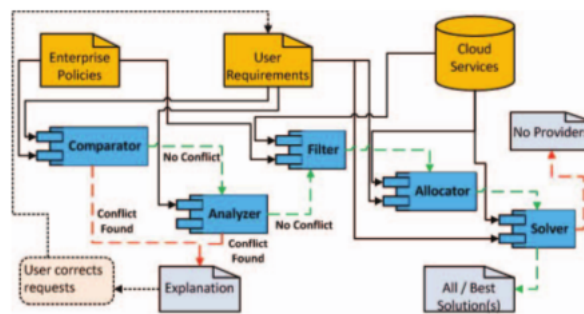


Fig. 2. Policy Implementation Engine (PIE)

FIGURE 2.11 – Policy Implementation Engine, Yan et al. [55].

2.2.14 CloudCmp, Li et al.

Li et al. examinent le problème de la comparaison de performance des Cloud Providers et développent un framework et outil qu'ils appellent CloudCmp [56, 57].

CloudCmp est un framework de comparaison de performance et de coût dont le but est d'aider les utilisateurs dans leur sélection de fournisseurs. L'outil présenté mesure la flexibilité des ressources informatiques, le stockage permanent et les services réseaux offerts par un fournisseur grâce à des métriques qui reflètent directement l'impact des performances sur les applications de l'utilisateur. CloudCmp veut assurer la représentativité, la conformité et l'équité ("fairness") de ces mesures tout en limitant leur coûts.

CloudCmp offre aux utilisateurs la possibilité d'estimer la performance et les coûts d'une application tels qu'ils seraient en cours d'exécution sans devoir réaliser un déploiement effectif.

L'outil principal contient un ensemble de sous-outils de mesure des benchmarks qui vont comparer les services similaires offerts par les différents Cloud Providers pour l'exécution d'une application, et utiliser les résultats de références ("benchmarking results") afin de prédire la performance et les coûts de cette application si elle était déployée chez un fournisseur en particulier.

Le framework caractérise les services offerts par les différents Cloud Providers et compare la performance et les coûts de ces services grâce aux benchmarks. Cette comparaison est réalisée à l'aide d'un ensemble d'interfaces de service communes. Un exemple d'interface de comparaison est montré à la Figure 2.12 et la Figure 2.13 .

Cette approche pour la prédiction de performance ressemble aux approches fortement utilisées pour évaluer différents systèmes informatiques à haute performance.

Un Cloud User potentiel peut donc utiliser ces résultats afin de comparer les différents Cloud Providers. Ensuite, il peut décider si oui ou non il devrait migrer ses applications sur le Cloud et par la même occasion prendre connaissance du fournisseur qui lui est le mieux adapté pour ses applications.

CloudCmp effectue les estimations de performance et de coûts d'une application en trois étapes[57] : (a) le *benchmarking du service* ("service benchmarking"), (b) la *collecte de la charge de travail de l'application* ("application workload collection"), et (c) la *prédiction de la performance* ("performance prediction").

Le benchmarking du service génère un profil de performance et de coûts pour un service Cloud qui inclut le temps de terminaison d'une tâche et les coûts associés.

En plus de la représentation de la charge de travail d'une application, les traces des requêtes de l'application et chaque chemin d'exécution d'une requête sont également collectés. Par exemple, la trace d'une requête devrait inclure l'adresse IP source de cette requête, le moment où cette requête est récupérée, ainsi que sa taille.

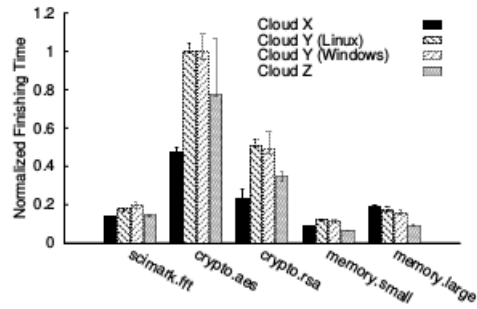
La prédiction de la performance indiquera le temps d'exécution et les coûts pour chaque requête collectée en utilisant un profil de performance et de coûts pré-généré pour chaque service cloud. Ces profils permettent d'estimer le temps passé et les coûts dépensés sur chaque composant du chemin d'exécution. Ils permettent ensuite de combiner ces temps d'exécution et coûts individuels en vue d'obtenir le temps et les coûts globaux.

Les auteurs identifient des objectifs clés pour mener une comparaison significative. Ils utilisent également leur outil CloudCmp afin d'évaluer quelques Cloud Providers qui se différencient largement dans leur implémentation mais qui, ensemble, dominent le marché des Infrastructures Clouds.

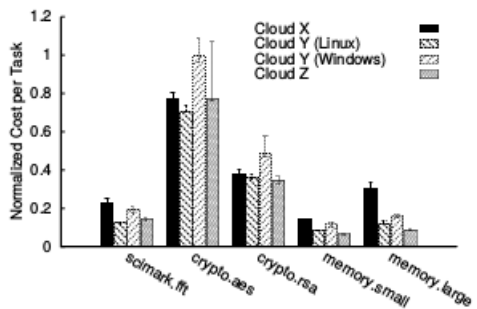
Ces objectifs fonctionnels clés ("design goals") pour CloudCmp sont les suivants :

- Guider un utilisateur dans son choix de fournisseur sur base de métriques de performance et de coûts ;
- Être pertinent envers les Cloud Providers afin de les aider à identifier leurs services qui sont sous les performances réalisées par les autres fournisseurs ;
- Effectuer une comparaison équitable entre tous les fournisseurs en les caractérisant chacun à l'aide du même ensemble de métriques et charges de travail ;
- Maintenir une comparaison détaillée avec un coût de mesure limité ;
- Couvrir un ensemble représentatif de fournisseurs tout en maintenant un coût de développement restreint ;
- Être en conformité avec les politiques (policies) des fournisseurs.

Grâce à cet outil, les auteurs ont pu observer, lors d'une évaluation de trois Cloud Providers représentatifs supportant des applications web, des variations significatives et non négligeables des performances et des coûts dans les offres d'instances virtuelles, de services de stockage et de transferts réseaux. Ces auteurs soulignent donc la pertinence indéniable de l'utilisation d'un tel framework de comparaison.



(a) Normalized finishing time



(b) Normalized cost per task

FIGURE 2.12 – CloudCmp : Exemple d’interface de comparaison de performance et de coûts (1), Li et al.[57]

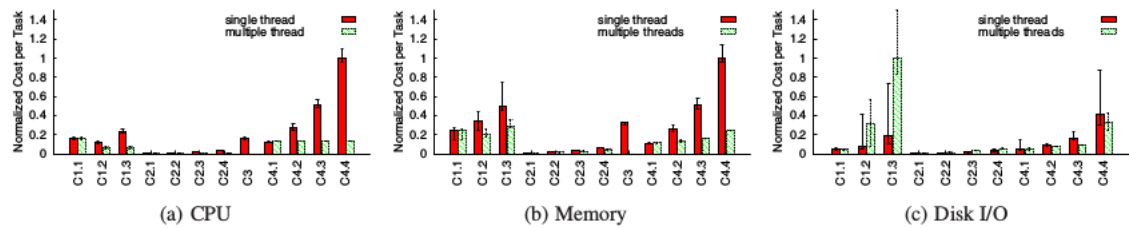


FIGURE 2.13 – CloudCmp : Exemple d’interface de comparaison de performance et de coûts (2), Li et al. [56]

2.2.15 SMICloud, Garg et al.

Garg et al. proposent *SMICloud*, un mécanisme et un framework qui mesure la qualité des services dans le but d'une comparaison et d'un classement par priorisation de services d'Infrastructure Cloud les mieux adaptés aux besoins d'un utilisateur [9]. Pour ces auteurs, les processus de mesures sont important dans l'évaluation d'un service Cloud. De ce fait, ils proposent de mettre à disposition des utilisateurs un outil dans le but de comparer différents services d'Infrastructures Clouds.

D'une part, le framework *Service Measurement Index Cloud* – SMICloud – a pour objectif d'aider les utilisateurs (Cloud Users) à trouver le fournisseur (Cloud Provider) le plus adapté à leurs besoins. D'autre part, il a pour but de créer une compétition/concurrence saine entre les Cloud Providers et de les forcer à respecter les SLAs tout en améliorant leurs Qualités de Services proposées.

Pour leur outil, les auteurs utilisent un index de mesure, le *Service Measurement Index* (SMI) développé par le Cloud SMI Consortium¹¹ (CSMIC). Celui-ci peut être trouvé à l'Annexe C.

Les attributs de mesures du SMI sont basés sur les Standards ISO fournis par le CSMI Consortium. Ils consistent en un ensemble de Key Performance Indicators (KPIs), qui apportent une méthode standardisée pour effectuer les mesures et les comparaisons des performances du service.

Le SMICloud permet aux utilisateurs de comparer et de classer les différentes offres d'Infrastructure Cloud selon des exigences fonctionnelles et non-fonctionnelles de qualité de service qui se base sur :

- (a) la responsabilité (accountability/risk)
- (b) l'agilité/adaptabilité (agility)
- (c) les garanties (assurance)
- (d) l'aspect financier et les coûts (financial/cost)
- (e) les performances (performance)
- (f) la sécurité et la protection des données (security and privacy)
- (g) l'utilisabilité (usability)

Cependant, certaines d'entre-elles ne sont pas facilement mesurables étant donné leur nature d'attributs qualitatifs. Elles ne pourront donc pas être formellement mesurées.

La réalisation du classement des services est un problème de prise de décision multi-critères (Multi-Criteria Decision Making – MCDM). Pour résoudre un tel problème, les auteurs proposent d'utiliser un processus d'analyse hiérarchique (Analytical Hierarchical Process – AHP). Il est basé sur un mécanisme de classement (ranking) pour résoudre le problème d'assignation de poids aux fonctionnalités en considérant une interdépendance entre elles, ce qui fournit donc une base des besoins quantitatifs les plus désirés pour établir le classement des services cloud.

11. <http://csmic.org/>

L'architecture du framework comporte trois composants principaux :

- Le *Broker SMICloud* qui est responsable des interactions avec les utilisateurs et collecte les besoins en fonction des critères du SMI. Il réalise la recherche et le classement des services adéquats aux fonctionnalités demandées. Il comporte un composant *SLA Management* pour la gestion des SLAs et leur historique, un composant *Ranking System* qui classe les services sélectionnés par le Broker, et un composant *SMI Calculator* qui calcule les KPIs utilisés par le Ranking System pour la priorisation des services.
- Le composant *Monitoring* retrouve les services potentiels qui peuvent satisfaire la demande d'un utilisateur, et ensuite effectue le monitoring de ces services.
- Le *catalogue de services* (Service Catalog) répertorie les services et leurs propriétés telles qu'annoncées par les Cloud Providers.

Le processus réalisé par l'outil est le suivant :

Les utilisateurs transmettent à l'outil leurs exigences désirées en fonction des critères de l'index. Le Broker SMICloud reçoit les requêtes des utilisateurs concernant le déploiement d'une application. Le composant Monitoring va alors retrouver les services potentiels pour les fonctionnalités demandées afin de les évaluer. Un classement des services les plus adéquats est ensuite établi en utilisant le SMICalculator et le Ranking System. Suite à cette évaluation et ce classement, l'outil retourne une liste priorisée de services d'Infrastructure Cloud sur lesquels les utilisateurs peuvent déployer leur application en adéquation avec les critères qu'ils ont introduits.

La Figure 2.14 résume l'exemple du cas d'étude donné dans [9].

Le framework présenté dans [9], et schématisé à la Figure 2.15, est un modèle de QoS pour les services d'Infrastructure Cloud (analyse des Cloud Providers) basés sur les KPIs du SMI. Cependant, les auteurs précisent que ce modèle pourrait être étendu au SaaS (analyse des SaaS Providers).

Top level QoS Groups (Weights)	First level Attributes (Weights)	Second Level Attributes (Weights)		Service 1 (S1)	Service 2 (S2)	Service 3 (S3)	Value Type	User Required Value
Accountability (0.5)	level:0-10 (1)			4	8	4	Numeric	4
Agility (0.1)	Capacity (0.6)	CPU (0.5)	0.5	9.6	12.8	8.8	Numeric	4x1.6 GHZ
		Memory (0.3)	0.3	15	14	15	Numeric	10 GB
		Disk (0.2)	0.2	1690	2,040	630	Numeric	500 GB
	Elasticity (.4)	Time (1)	0.4	80-120	520-780	20-200	Range	60-120 sec
Assurance (0.2)	Availability (0.7)	0.7		99.95%	99.99%	100%	Numeric	99.9%
	Service Stability (0.2)	Upload Time (0.3)	0.3	13.6	15	21	Numeric	
		CPU (0.4)	0.4	17.9	16	23	Numeric	
		Memory (0.3)	0.3	7	12	5	Numeric	
	Serviceability (0.1)	Free Support (0.7)	0.7	0	1	1	Boolean	
Type of Support (0.3)		0.3	24/7,Diagnostic Tools, Phone, Urgent Response	24/7,Diagnostic Tools, Phone, Urgent Response	24/7, Phone, Urgent Response	Unordered set	24/7, phone	
Cost (0.3)	On-Going Cost (1)	VM Cost (0.6)	0.6	0.68	\$0.96	0.96	Numeric	< 1 dollar/hour
		Data (0.2)	inbound	10	10	8	Numeric	100 GB/month
			outbound	11	15	18		200 GB/month
	Storage (0.2)	0.2	12	15	15	Numeric	1000 GB	
Performance (0.3)	Service Response Time (1)	Range (0.5)	0.5	80-120	520-780	20-200	Range	60-120 sec
		Average Value (0.5)	0.5	100	600	30	Numeric	
Security (0.05)	level: 0-10 (1)			4	8	4	Numeric	4

FIGURE 2.14 – SMICloud – Interface d’une étude de cas, Garg et al. [9]

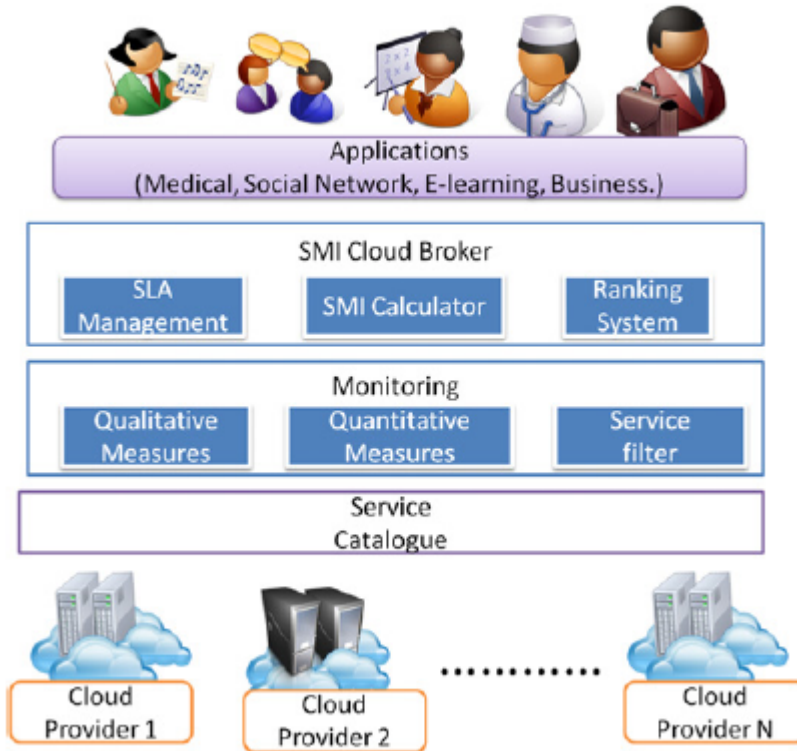


FIGURE 2.15 – Framework SMICloud, Garg et al. [9]

2.2.16 CloudAdvisor : A Recommendation-as-a-Service Platform for Configuration and Pricing, Jung et al.

CloudAdvisor est une plate-forme de “Recommendation-as-a-Service” qui aide à trouver une configuration optimale pour les infrastructures clouds sur base d’une charge de travail et de préférences données [58]. Cette plate-forme permet d’explorer différentes configurations recommandées sur bases de préférences telles que un budget, les performances attendues, les économies d’énergie, etc. Les utilisateurs sont capables de comparer les prix offerts, selon leurs préférences et pour une charge de travail donnée, avec d’autres offres à l’aide de benchmarks. Les résultats fournis sont estimés par une méthode de résolution par optimisation de contraintes (constraint optimization solving method) qui prend en compte les préférences des utilisateurs et les ressources disponibles (en correspondance avec les bons supports hardware and software). La méthode de recherche “best-first graph” basée sur l’algorithme A^* est utilisée pour résoudre le problème d’optimisation par contrainte (CSP). Ce solveur recherche les configurations des infrastructures clouds qui correspondent aux préférences de l’utilisateur tout en considérant les disponibilités logicielles et matérielles. Le problème de recherche est encodé sous forme du problème du sac-à-dos (“knapsack problem”). CloudAdvisor fournit une interface graphique où les utilisateurs ne doivent pas entrer des détails de configurations spécifiques mais où ils fournissent seulement le niveau de préférence pour cinq critères à l’aide de curseurs dans le but de sélectionner le niveau désiré : Usage, Load, Performance, Cost, Energy Consumption.

2.2.17 Cloud Service Selection Based on the Aggregation of User Feedback and Quantitative Performance Assessment, Qu et al.

Lors d’une récente recherche (2013), Qu et al. présentent un modèle de sélection de services clouds se basant d’une part sur le feedback des utilisateurs et d’autre part sur des mesures de performance [59]. Ce modèle de sélection de services joue un rôle d’agrégation d’informations issues d’évaluations subjectives venant des utilisateurs et d’indicateurs de performance reçu d’un tiers de confiance fournissant des mesures objectives (“benchmark testing”).

Les auteurs appliquent un système de grandeurs additives floues (“fuzzy additive weighting system”) pour considérer des situations du monde réel dans le but de permettre une normalisation et une agrégation de tous les types d’attributs d’un service qu’ils soient subjectifs ou objectifs. De cette façon, certains aspects spécifiques de performance peuvent être pris en compte selon les exigences des utilisateurs.

Le système d’addition de grandeurs additives simples (Fuzzy Simple Additive Weighting System – FSAWS) établi par Chou et al. [60] converti toutes les valeurs des attributs en de simple valeur de cotation. Ensuite, un score est calculé pour chaque alternative correspondante au service cloud en question, et le score le plus élevé correspond à la meilleure performance entre toutes alternatives possibles.

Le framework de Qu et al. se compose de quatre composants, chacun jouant le rôle d’un service particulier : (a) le “cloud selection service”, (b) le “benchmark testing service”, (c) le “user feedback management service”, et (d) le “assessment aggregation service”.

La Figure 2.16 schématise ce framework.

Le *service de sélection de cloud* (cloud selection service) a la charge d'accepter et de traiter au préalable les requêtes pour la sélection d'un service cloud émanant de Cloud Users potentiels. Quand un utilisateur potentiel soumet une requête pour recevoir le service le plus adapté à ses besoins, le service de sélection choisit d'abord parmi une liste de services candidats, un ensemble de services clouds qui pourraient satisfaire toutes les exigences objectives et quantitatives (e.g., le type des services, la spécification des VMs et les coûts) . Ensuite, selon des exigences plus précises de l'utilisateur, le composant envoie la requête au service de test des benchmarks et au service de gestion de feedback pour obtenir les résultats de clouds alternatifs. Ces résultats enregistrés sont alors envoyés au service d'agrégation des évaluations. En agrégeant ces enregistrements-résultats grâce au modèle proposé, le service d'agrégation des évaluations peut transmettre au service de sélection de Cloud un score final pour chaque alternative. Tous ces résultats sont alors montrés à l'utilisateur potentiel pour sa sélection.

Le *service de test des benchmarks* (benchmark testing service) est réalisé par une partie tiers de confiance ("a trusted third party"). Ce tiers réalisera les scénarios de tests pour les aspects communs de performances d'un service cloud tels que la disponibilité, la flexibilité, le temps de réponse, le coût par tâche, etc. grâce à des suites de benchmarks standards.

Le *service de gestion de feedback de l'utilisateur* (user feedback management service) est utilisé pour collecter et gérer le feedback des utilisateurs qui consomment des services clouds. Pour chaque aspect de performance d'un service, l'utilisateur donne une évaluation subjective liée à son expérience. Chaque aspect que les utilisateurs évaluent sont considérés comme un attribut subjectif du service cloud. Ces attributs subjectifs sont exprimés avec des variables simples dans un langage naturel : "bon", "mauvais", "moyen", "correct", "faible", etc. Chaque service Cloud est associé à une paire ($u \leq s; u \leq o$), où s représente le nombre d'attributs subjectifs, o le nombre d'attributs objectifs et u le nombre d'attributs associés à un service cloud. Les services d'après vente, la disponibilité, le temps de réponse, la privacy, peuvent être des attributs subjectifs tirés de feedback d'utilisateurs. La disponibilité, le temps de réponse, le calcul cryptographique peuvent être quelques uns des attributs issus des tests de benchmarks réalisés par le third party. Les attributs associés sont les attributs subjectifs et objectifs regroupés.

Lorsque toutes les valeurs des attributs subjectifs et objectifs des services clouds alternatifs ont été rassemblées, les valeurs relatives aux attributs subjectifs sont converties en cotations par traduction des variables en langage naturel vers des nombres flous (fuzzy numbers).

Les valeurs des attributs objectifs sont également converties en cotation en les comparant aux valeurs des mêmes attributs objectifs des services alternatifs. Au final, le système obtient d'une part plusieurs ensembles de valeurs pour les attributs subjectifs provenant des différents feedbacks des utilisateurs, et d'autre part un seul ensemble de valeurs pour les attributs objectifs reçu des tests du third party.

Le *service d'agrégation d'évaluation* (assessment aggregation service) agrège les valeurs des attributs subjectifs du feedback, issus du services de gestion de feedback, avec les attributs objectifs issus du services de test de benchmarks. Il va ensuite calculer le score final pour chacune des alternative de service cloud selon l'importance des poids qui ont été donné par l'utilisateur sous forme de variables dans le langage naturel.

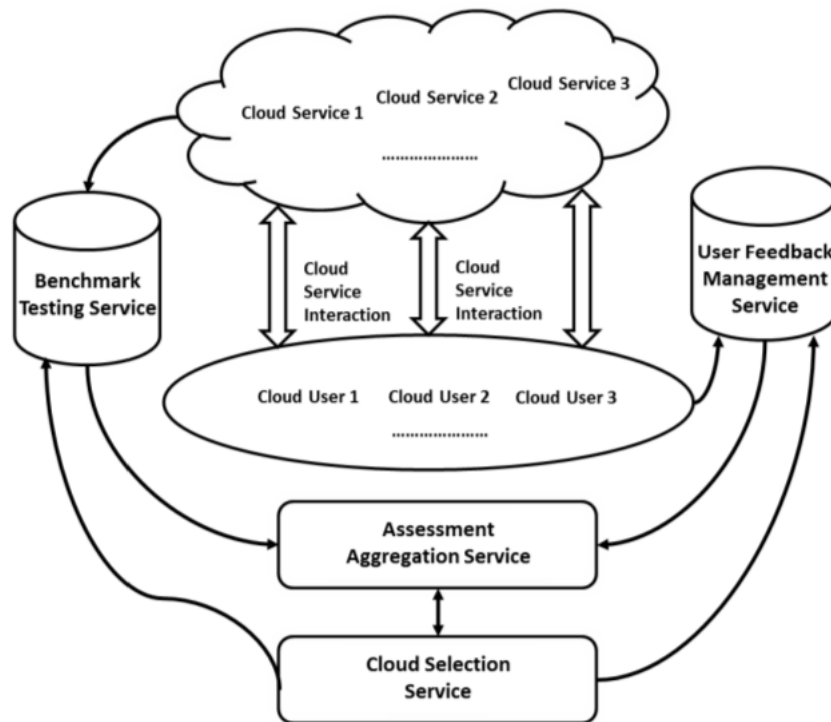


FIGURE 2.16 – Framework pour la sélection de service basé sur le feedback et l'évaluation de performances quantitatives, Qu et al [59].

2.2.18 Cloud Model for Service Selection, Wang et al.

Wang et al. présente un modèle basé sur une approche de sélection de Services Web [34](2011). Cette approche met en oeuvre un modèle pour calculer l'incertitude des qualités de service. Elle utilise la méthode Mixed Integer Programming (MIP) pour identifier le Service Web le plus adéquat.

Dans ce travail, les auteurs font référence à certains travaux apparentés concernant la composition de services web et leur sélection.

Ils adressent le problème de la sélection de services web et démontrent l'influence des qualités de services dans le processus de sélection. Ils décrivent également un concept appelé "QoS uncertainty computing". Ce concept modélise les incertitudes intrinsèques des qualités de service. Sur base de ce modèle, les auteurs présentent une approche de sélection de service rapide et fiable portant sur les qualités de services.

2.3 Comparaison des Cloud Service Brokers présentés

Pour souligner les différentes caractéristiques des Cloud Services Brokers apportées par les différents travaux étudiés, nous proposons ici une comparaison de plusieurs Brokers et des méthodes exposés ci-dessus.

Comme mentionné au début de notre État de l'Art, nous pouvons classifier ces projets en trois catégories : (a) les projets traitant d'un SaaS Broker (voir Tableau 2.1), (b) les projets décrivant un Cloud Broker ou une méthode similaire (voir Tableau 2.2) et (c) les projets présentant un framework de comparaison des services d'Infrastructure Clouds (voir Tableau 2.3).

Les projets de SaaS Broker proposent tous les deux une Intermédiation et Interopérabilité entre services. Le projet de Badidi [2.2.1] expose un rôle d'Arbitrage des services reposant sur les QoS du SLA dans un environnement Multi-Cloud, tandis que celui de Moore et Mahmoud [2.2.2] réalise sur l'Agrégation de services.

Les projets de Cloud Brokers proposent, outre l'Intermédiation et l'Interopérabilité de base, une Agrégation de services. mOSAIC [2.2.6], Contrail [2.2.9], Sundereswaran et al. [2.2.12], Yan et al. [2.2.13] fournissent un mécanisme d'Arbitrage et/ou de sélection. Hormis OPTIMIS [2.2.3] et l'étude de Yan et al. qui permettent également un Brokerage sur des critères qualitatifs (feedbacks, confiance, règles stratégiques), les travaux présentent des Brokers qui effectuent leurs opérations sur base des critères de prix et de QoS qui peuvent être inclus dans un SLA.

mOSAIC et STRATOS [2.2.7] proposent un environnement Multi-Cloud, tandis que RESERVOIR [2.2.5], FCM [2.2.10], Intercloud [2.2.8] et Contrail donnent lieu à une Fédération de Clouds.

Différentes méthodes pour la sélection de services sont présentées comme, par exemple, le problème de décision d'acquisition de ressources – RAD (STRATOS) – et le problème de décision multi-critères – MCDM (Yan et al.) – .

Trois grandes études de modèles et outils pour la comparaison, le classement et la sélection de services peuvent être retenus.

CloudCmp [2.2.14] effectue la comparaison des services Clouds par évaluation de benchmarks concernant la performance et les coûts.

CloudAdvisor [2.2.16] vise à fournir la configuration optimale des ressources sur base d'une charge de travail donnée et de critères de préférences en utilisant la méthode du "best first graph".

SMICloud [2.2.15] donne un outil de comparaison et de classement des services d'Infrastructures en utilisant les critères de l'index SMI. Selon ses auteurs, cet outil pourrait être adapté aux SaaS. Notons que STRATOS reprend également cet index.

Finalement, le travail de Qu et al. [2.2.17] démontre un modèle de sélection de services se basant d'une part sur le feedback des utilisateurs et d'autre part sur des mesures de performance, et réalise l'agrégation de ces informations afin de proposer un score pour chaque alternative de service. Les auteurs utilisent le principe du "Fuzzy Simple Additive Weighted System".

TABLE 2.1 – Tableau comparatif des Brokers (partie 1)

Services	Projet	Architecture Cloud	Rôles Brokerage	Critères	Mécanisme	Éléments d'Architecture	Autres Caractéristiques
SaaS	Badidi [33]	Multi-Cloud	Intermédiation, Sélection	QoS, SLA	Linear aggregate utility function	SLA-, Profile-, Policy-Manager, Measurement Service, Monitoring Adapters	Monitoring
	Moore & Mahmoud [36]	Cloud	Intermédiation, Agrégation	QoS	“Trust”, “Data Mapping”, “Interface Attribute”	Local UDDI Registry, UDDI Database, Client Interface, Service Providers, Data Integration Manager, Web Service Broker, WSDL Manager et Trust Manager	Extension WSDL
Cloud/SaaS	OPTIMIS	Fédération Cloud	Intermédiation, Agrégation	Quantitatifs (QoS, prix) & Qualitatifs (TREC, géo-loc.)	Configuration, déploiement et interaction de l'application, optimisation des ressources	API, Basic Toolkit (TREC), Service Builder, Admission Controller, Deployment Engine, Service Optimizer, Cloud Optimizer	Transparence, ouvert, private/hybrid cloud
	SLA@SOI	SOI	Gestion des SLAs	SLA	Prédiction et fiabilité, gestion et automatisation SLA	infrastructure-SLA Manager, Framework de gestion de SLA multi-couches et multi-domaines	OCCI, hybrid cloud
	RESERVOIR	Fédération Cloud	Agrégation	Pricing, SLA	Règles Service Manifest, OCCi	Service Manager, VEEM, VEEH, Agreement, VEE	Transparent, private/public/hybrid cloud

TABLE 2.2 – Tableau comparatif des Brokers (partie 2)

Services	Projet	Architecture Cloud	Rôles Brokerage	Critères	Mécanisme	Éléments d'Architecture	Autres Caractéristiques
Cloud	mOSAIC	Multi-Cloud	Agrégation, Sélection	QoS, prix	CSP, Pareto optimum	Cloud Agency , Cloud ontology	Transparence, ouvert, hybrid cloud
	STRATOS	Multi-Cloud	Agrégation	Critères SMI, Géo-loc., Politiques	Problème RAD, Topology Descriptor File	CloudManager, Application Manager, API deltacloud	Transparent, gestion SLA
	Intercloud	Fédération Cloud	Agrégation	SLA, Pricing, Géo-loc.	Adaptation dynamique des services, plate-forme de fédération	Cloud Broker, Cloud Coordinator, Cloud Exchange	-
	Contraïl	Fédération Cloud, Fédération	Agrégation, Sélection	SLA, coûts, performances	heuristiques d'optimisation	Federation Runtime Manager, Adapters, Gestion des accès	Single Sign-On
	FCM	Fédération Cloud	Agrégation	Resources, VMs	Broker Prop. Descr. Lang. Scheduling	FCM repository, WSDL, Generic Meta-Broker Service (GMBS)	Transparent
	Sundareswaran et al. [17]	Cloud	Agrégation, Arbitrage, Sélection	QoS, pricing	CSP-index, Cloud Service Selection algorithm	3 acteurs : Cloud Providers, Cloud Broker, Cloud Users	-
	Yan et al. [55]	Cloud	Arbitrage, Sélection	Pricing, SLA, feedback	CSP, MCDM	Policy Implementation Engine, Cloud Intelligence	Private cloud

TABLE 2.3 – Modèles de comparaison et de sélection étudiés

Services	Projet	Architecture Cloud	Comparaison / Sélection	Critères	Mécanisme	Éléments d'Architecture	Autres Caractéristiques
Cloud	CloudCmp	Cloud	Comparaison	Performance, coûts	Benchmarks	Service Benchmarking, Workload Collection, Performance Prediction	-
	SMICloud	Cloud	Comparaison, Ranking, Sélection	Critères SMI	MCDM-AHP	SMI Framework : Broker, Monitoring, Catalog	adaptable aux SaaS
	CloudAdvisor	Cloud	Sélection	Usage, Load, Performance, Cost, Energy Consumption	CSP, best-first graph	-	GUI
	Qu et al. [59]	Cloud	Sélection	QoS & Feedback	Fuzzy Additive Weighted system	Framework : Cloud selection, feedback management, assessment aggregation	-

2.4 Quelques Brokers commerciaux

Dans un but de complétude, nous présentons très brièvement quelques Services Brokers et SaaS Marketplace commerciaux. Nous jugeons important de les mentionner car ils nous apportent des pistes de réflexions quant leurs lacunes laissées dans le monde industriel ; comme par exemple un manque de comparateurs et de systèmes d'évaluation fiables.

- Jamcracker (www.jamcracker.com) est un des principaux Cloud Services Broker commerciaux. L'architecture de son Cloud Broker offre des supports pour tout le cycle de vie des services Clouds à destination des organisations qui souhaitent déployer et gérer des clouds privés, publiques ou hybrides répartis géographiquement. Le Jamcracker Services Delivery Network (JSDN) rassemble dans un catalogue les services clouds (infrastructures clouds et Business applications) de ses partenaires et leur API. Il fournit une plate-forme d'intermédiation pour l'acquisition de ces services multi-clouds et leur accès, en assurant également les aspects relatifs à la sécurité, au paiement, à l'administration et au support. Il ne permet pas la recherche ni la comparaison des services.
- Amazon AWS Marketplace (aws.amazon.com/marketplace/) est la plate-forme d'Amazon.com pour l'acquisition de Software Infrastructure, Development Tools et Business Software (SaaS). Elle propose la recherche et l'acquisition de services, ainsi qu'une méthode d'évaluation du service par l'utilisateur (cotation et commentaire).
- CloudCommons (www.cloudcommons.com) est le marketplace de CA Technologies, il offre aux utilisateurs la possibilité de rechercher et d'obtenir des services logiciels et de sécurité développés par CA ou ses partenaires. Il permet la recherche et l'obtention des services, ainsi que l'évaluation (cotation et commentaire). Il ne permet pas la comparaison.
- Cloud Sherpas (cloudsherpas.com) est un Cloud Broker qui aide les organisations à déployer et exécuter des applications, ainsi que leur intégration avec d'autres systèmes clouds. Il offre les solutions Clouds de ses partenaires Salesforce, Google, ServiceNow.
- CloudSurfing (www.cloudsurfing.com/rank/) est un marketplace proposant la recherche, la comparaison et recommandation, ainsi que l'évaluation d'applications Web et de SaaS.
- SaaSMAX (www.saasmax.com) est un marketplace pour SaaS qui a pour but de créer un écosystème entre vendeurs d'applications (SaaS Providers), revendeurs d'applications (SaaS Providers, Brokers) et acheteurs-utilisateurs (SaaS Users). Il offre également des ressources afin de développer une stratégie business en se basant sur les SaaS. Il propose la recherche, l'achat et l'évaluation des SaaS. Il permet également de mettre en contact l'utilisateur avec un expert de l'application.
- SaaS-Guru (www.saas-guru.com/) est une plate-forme-marketplace proposant des solutions SaaS. Elle permet de s'informer sur les SaaS ainsi que de comparer et consulter les solutions disponibles dans son catalogue. Elle s'adresse plus particulièrement aux PME. Cette plate-forme offre la recherche, la comparaison, l'acquisition, l'évaluation des SaaS.
- Arjuna Agility (www.arjuna.com/cloud-services-brokerage) fournit un framework à destination des Cloud Services Brokers afin de gérer les SLAs et de permettre une fédération de services Clouds.

2.5 Possibilités de recherche

Nous remarquons que la grande majorité des travaux analysés précédemment s'intéressent aux services d'Infrastructures Clouds, et que très peu d'entre eux prennent en considération le Brokerage de SaaS pour le compte de SaaS Users. De plus, nous notons que ces travaux se concentrent d'une part sur les critères de qualité de service (QoS) et des coûts, impliquant le monitoring des performances et des tests de benchmarks, et d'autre part sur la négociation de ces critères via les SLAs.

Puisque ces multiples projets actuels s'intéressent principalement aux services Clouds et à leurs QoS, coûts et SLA, nous y voyons des opportunités de recherche concernant la sélection, la comparaison et l'évaluation des SaaS.

Nous envisageons d'étudier le domaine des SaaS Brokers en vue de proposer un modèle pour un nouveau type de Broker qui porterait attention à des caractéristiques supplémentaires aux QoS traditionnelles, comme par exemple le feedback des Users et la réputation des Providers, la qualité logicielle, les règles de conformité législative, l'efficacité d'utilisabilité, la sécurité et la privacy.

Nous pensons également qu'un modèle d'évaluation de la réputation et de la confiance accordée aux fournisseurs peut trouver sa place dans la sélection de services SaaS et peut faire l'objet d'une étude. Celle-ci étant un sujet peu abordé dans la littérature.

Chapitre 3

Vers un nouveau SaaS Broker

À présent que nous avons proposé un cadre théorique, et analysé l'état actuel des recherches dans le domaine du Cloud Computing et plus particulièrement dans celui des Brokers de services, nous souhaitons apporter notre contribution à cette matière par la proposition d'un SaaS Broker innovant. Cette contribution vise d'une part à combler les lacunes relevées lors de notre analyse de l'État de l'Art, et d'autre part à proposer une architecture de Brokerage pour la sélection de SaaS ainsi que l'évaluation de la satisfaction et de la confiance des utilisateurs concernant l'utilisation de ces services logiciels. Dans ce chapitre, nous étudierons les critères à prendre en considération dans le choix et l'évaluation des SaaS, ainsi que les fonctionnalités d'un SaaS Broker et des différents composants du système.

Suite à notre analyse des travaux sur les systèmes de Brokerage réalisée précédemment, nous exposerons les raisons de notre étude pour la proposition d'un nouveau SaaS Broker. Ensuite, nous présenterons les propriétés clés relatives aux SaaS. Nous pourrions alors par après définir les paramètres à prendre en considération lors d'une sélection et d'une évaluation de SaaS. Ces paramètres seront exposés au sein d'un modèle proposé de la qualité des SaaS. Nous décrirons ensuite les fonctionnalités indispensables que nous retenons pour proposer le modèle d'un nouveau type de Broker SaaS. Sur base de celles-ci, nous proposerons un modèle d'architecture dans lequel nous décrirons les différents composants haut-niveau.

3.1 Étude préalable pour un SaaS Broker

Cette section présente nos motivations à entreprendre une étude portant sur la réalisation d'un nouveau SaaS Brokers, nos objectifs dans l'investigation d'un tel Broker ainsi que les fonctionnalités que nous désirons y intégrer.

3.1.1 Pourquoi un nouveau SaaS Broker ?

Notre objectif à l'entame de cette étude était de travailler sur un agent intermédiaire entre SaaS Providers et Cloud Providers en vue d'investiguer comment établir un modèle d'évaluation des fournisseurs de services. Typiquement, cette évaluation devrait permettre de mesurer la confiance d'un Service User envers un Service Provider à partir d'expériences personnelles et de celles des autres Service Users.

Cependant, suite au travail effectué en vue de présenter l'État de l'Art (Chapitre 2), nous

avons relevé différents incitants nous conduisant à la proposition d'un nouveau SaaS Broker :

- (a) Tout d'abord, le manque d'études sur les SaaS Brokers nous conduit à investiguer sur les fonctionnalités et les mécanismes que nous pourrions mettre en place pour un tel Broker.
- (b) Il est en de même pour l'étude des aspects clés à prendre en compte lors de l'obtention et l'utilisation de SaaS. En effet, le choix d'une telle solution IT implique des réflexions quelques peu différentes de celles requises dans le choix d'un produit logiciel classique.
- (c) De plus, selon des prévisions réalisées en 2013 par Gartner, le marché des SaaS ne cesse de croître et nous pouvons constater la venue de nombreux SaaS Providers sur le marché offrant ainsi une multitude de SaaS aux propriétés et qualités de service différentes, compliquant de la sorte les démarches des SaaS Users dans leur choix.
- (d) Par ailleurs, nous constatons également qu'une des difficultés dans ces démarches est le manque de méthodes et de modèles pour comparer et évaluer la qualité d'un SaaS¹. Or, nous jugeons que l'assistance dans le choix d'une solution SaaS conforme aux besoins des utilisateurs ne devrait pas être négligée car la sélection d'un SaaS peut demander l'analyse de nombreuses qualités et de nombreux critères qui rendent la comparaison difficile à réaliser manuellement.
- (e) Nous pensons donc que la mise en place d'un Broker effectuant un Arbitrage des services disponibles pourrait faciliter la prise de décision des utilisateurs dans leur choix ; de même, nous pensons que la négociation et l'automatisation de la gestion des Services Level Agreements (SLA) sont des fonctionnalités à intégrer à ce type de Broker.
- (f) Ajoutons encore que les nombreuses architectures actuelles de Brokerage se concentrent essentiellement sur des critères et objectifs de qualité de service (mesurables) et de prix. Nous y voyons donc par là l'opportunité de proposer un Arbitrage de services tenant compte de critères supplémentaires comme par exemple des informations sur des règles de conformité, de qualité logicielle et d'évaluations subjectives (feedbacks, réputation).

De ce fait, nous considérons que le processus de sélection d'un SaaS ne peut être totalement identique à celui opéré pour l'acquisition de services d'Infrastructures Clouds, car des caractéristiques propres aux applications logicielles devraient être pris en considération.

- (g) Finalement, tel l'Inter-Cloud pour les services d'Infrastructures Clouds ou encore l'Agrégation de services, nous pensons qu'une communauté ou fédération de SaaS Providers et/ou de SaaS Brokers pourrait être bénéfique. À notre meilleure connaissance, un tel sujet ne fait actuellement pas l'objet d'études.

3.1.2 Quels sont nos objectifs dans l'étude d'un nouveau Broker ?

Notre objectif est d'investiguer sur les principaux paramètres ainsi que les composants et les mécanismes d'une architecture haut-niveau d'un SaaS Broker. Celui-ci aura pour but principal d'assister les SaaS User dans leur choix (Arbitrage) et leur obtention de services tout en étant assurés par un Service Level Agreement consistant et conforme à leurs attentes (Gestion SLA). Nous porterons plus particulièrement notre attention sur la sélection de services nous basant sur les critères quantitatifs traditionnels de qualité de service (QoS) mais également sur des propriétés qualitatives additionnelles telles que la qualité logicielle, les règles de conformité, ou encore des informations collectées à l'aide d'un mécanisme d'évaluation de feedback/satisfaction. Par là, notre contribution veut apporter une étude préalable à la

1. Nous jugeons insuffisant les comparateurs et les évaluations subjectives des services par les utilisateurs que certains marketplaces proposent.

réalisation d'un tel Broker et définir un modèle haut-niveau en vue d'une validation et d'une implémentation future.

3.2 Étude des propriétés clés dans la sélection de SaaS

Adopter une application logicielle sous forme de services, ou autrement dit un "Software-as-a-Service" nécessite une réflexion quelque peu différente de celle entreprise lors de l'acquisition d'un produit logiciel traditionnel ou d'une solution IT hébergé(e) sur l'infrastructure IT locale de l'utilisateur/End-User.

En effet, les SaaS présentent certains aspects, propriétés et certaines fonctionnalités clés ("key features") à prendre en considération avant l'achat de tels services. Bien que de telles applications logicielles peuvent être obtenues "à la demande" et utilisées d'une manière plus flexible, le SaaS User devra tout de même être attentif aux critères de qualité et aux conditions de prestation du service (localisation et durée de vie des données, fiabilité du fournisseurs, etc.). Lors d'une sélection de SaaS potentiels et du choix du service le plus adéquat en correspondance avec ses besoins, exigences, politiques/stratégies et règles de conformité légales, le SaaS User devra donc prendre en considération de multiples caractéristiques clés et les propriétés des services délivrés.

Comme nous l'avons vu précédemment, l'utilisateur peut se faire assister par un SaaS Broker ou lui déléguer les opérations de sélection et de gestion des SLAs reprenant toutes propriétés essentielles pour l'efficacité des services prestés ; tant pour l'utilisateur qui souhaite recevoir le niveau de qualité du service logiciel attendu sous les conditions exigées, que pour le fournisseur qui doit satisfaire ses utilisateurs/clients. Remarquons encore que certaines de ces propriétés typiques rappellent certaines bonnes pratiques de réflexion pour l'acquisition d'un produit logiciel traditionnel.

Dans cette section, nous étudions donc ces propriétés en vue de pouvoir dégager les paramètres d'évaluation de la qualité des SaaS, tant fonctionnelle et non-fonctionnelle que objective et subjective. Cette étude nous permettra ensuite de construire un modèle de qualité qui sera utilisé dans notre modèle de sélection et d'évaluation des SaaS pour notre Broker.

Nous proposons ci-dessous une présentation et description des différentes conditions essentielles à prendre en considération dans la réflexion en vue de la sélection et l'obtention d'un SaaS afin d'opter pour le service qui apportera une efficacité maximale à l'utilisateur lors son l'exécution : (a) *Fonctionnalités du SaaS et QoS*, (b) *Accès au services, Customisation et Configuration*, (c) *Support et Formation*, (d) *Intégration et Interopérabilité*, (e) *Stockage des données et Privacy*, (f) *Négociation et Gestion du SLA*, (g) *Sécurité du logiciel et des données*, (h) *Évolution, mises à jour et maintenance des SaaS*, (i) *Plans tarifaires, Licences, Engagements*, (j) *Identification du Provider et de sa réputation* et (k) *Autres critères*.

Notons que nous utilisons le terme "organisation", plus familier, pour signifier toute entreprise ou personnes jouant le rôle de SaaS Users.

A. Fonctionnalités du SaaS et QoS

L'organisation ou SaaS User doit s'assurer que les fonctionnalités apportées par le SaaS soient similaires à toutes celles qui sont remplies par le produit logiciel à substituer et déployées sur une infrastructure locale, ou bien qui pourraient être mises en place par l'acquisition d'un tel nouveau produit à intégrer aux services internes existants.

Les fonctionnalités proposées par les SaaS peuvent être substituables, mais les niveaux de

service peuvent diverger et doivent être analysés selon ses besoins requis.

Ces niveaux de services (bas/moyen/élevé) se répercutent en général sur le plan tarifaire ou le prix de la licence “on-line”. L’organisation doit s’assurer que les qualités de service offertes (QoS) seront en adéquation avec les besoins de ses activités.

L’organisation doit également s’interroger sur la nécessité ou non de disposer d’un accès à certaines données (critiques ou non) en mode hors-ligne.

Dans la multitude de SaaS disponibles sur le marché, l’organisation devra également porter attention au secteur d’activité et/ou au type d’organisation auquel le SaaS est destiné. Par exemple, certains SaaS pourraient être spécialement adressés à des PME ou au secteur médical, et donc les conditions de service pourraient être mieux adaptées.

B. Accès au services, Customisation et Configuration

L’organisation doit obtenir à l’avance les informations relatives à l’accès du service (comment accédera-t-elle au service ?) et aux possibilités de customisation et de configuration des paramètres.

L’organisation doit se demander si l’application possède la capacité de s’adapter à certaines exigences propres à ses besoins ou son fonctionnement.

Bien que les SaaS Providers auront tendance à adapter leurs services selon leurs propres intérêts, c’est-à-dire être le plus général possible pour satisfaire une multitude d’utilisateurs, si ils n’offrent pas une personnalisation minimale et une adaptation à certains besoins spécifiques, l’utilisateur devra se contenter d’utiliser le service, et par là, cela limitera l’adoption des SaaS.

La configuration d’un service est un aspect important car il prend en compte la structure de l’organisation cliente, l’interface utilisateur qui sera mise en place, le contrôle des données et des accès, etc.

C. Support et Formation

L’organisation doit se renseigner sur les méthodes de support et d’helpdesk mises à disposition, de même que le format de la documentation. Elle doit également prendre en compte les disponibilités, délais d’assistance et conditions du fournisseur en cas de problèmes, ainsi que les possibilités de formation pour certains logiciels (e.g. e-learning).

D. Intégration et Interopérabilité

L’intégration avec les services existants déployés sur l’infrastructure locale de l’organisation doit pouvoir s’effectuer sans difficulté. L’organisation doit donc être en mesure de connaître les besoins nécessaires pour une intégration et interopérabilité efficiente du SaaS avec les applications déjà en place. Par exemple, le format des données utilisé par le SaaS doit pouvoir être conforme et cohérent envers celui des données locales. L’intégrité des données et des paramètres de sécurité doit être préservée.

E. Stockage des données et Privacy

Le stockage des données, leur accès ainsi que le cadre d’utilisation de ces données sont des problèmes critiques pour les organisations. Il convient donc pour l’organisation de se renseigner à l’avance avec soin sur les conditions exactes d’utilisation ainsi que sur les droits de propriété des données une fois dans le “Cloud” (data ownership). L’organisation devrait donc avoir recours aux informations répondants aux questions suivantes :

- Où seront stockées les données ;
- Des backups sont-ils réalisés et quelles sont les conditions d'un disaster recovery ;
- À quelles données le fournisseur posséderait un accès ;
- Comment et sous quelles conditions le fournisseur pourrait utiliser les données de l'organisation ;
- Les règles politiques "Privacy" sont-elles en adéquation avec celles de l'organisation ou de ses attentes en regard de l'utilisation de ses données ;
- Les conditions légales de conformité et de protection de données en vigueur dans le pays du fournisseur et de ses installations sont-elles conformes aux règles de l'organisation ;
- Le fournisseur détruira-t-il toutes les données de ses installations, ou ce qu'advient-il de celles-ci lorsque que le contrat prendra fin ou si le fournisseur devait cesser ses activités.
- Les données sont-elles stockées dans un format qui permet l'exportation vers une autre infrastructure ;
- Le fournisseur permet-il l'importation et l'exportation de données depuis/vers le SaaS, ou depuis/vers une infrastructure locale ou celle d'un autre fournisseur, autrement dit les données sont-elles transférables ?

F. Sécurité du logiciel et des données

La sécurité autour du service et celle de l'application, de ses accès et de ses données sont également critiques.

L'organisation devrait avoir la possibilité de prendre connaissance des mécanismes et des mesures de sécurité mis en place par le fournisseur et si celui-ci respecte les normes de sécurité en vigueur.

La manière dont sont chiffrés les échanges et les stockages de données, la protection des données contre les intrusions, ainsi que le système de sécurité des accès et d'authentification mis en place, sont également à prendre en considération. Il en est de même pour l'existence et la réalisation des tests de sécurité et des audits.

G. Négociation et Gestion du SLA

L'utilisation d'un Service Level Agreement n'est pas à négliger. Sous la protection d'un SLA, l'organisation peut s'assurer que les qualités du service attendues seront conformes aux exigences. Par ailleurs, la gestion de SLA peut être prise en charge par un Broker pour une meilleure efficacité et un suivi de la qualité du service presté.

L'organisation doit donc s'assurer que :

- (a) Le SLA gouverne la prestation du service ; (c) Le SLA est clairement établi avec les termes et les conditions négociés et validés par les deux parties ; en concordance avec les besoins des activités de l'organisation.
- (b) Les termes de ce contrat comprendront entre autre les qualités du service fourni, le plan tarifaire appliqué et les compensations et/ou pénalités possibles en cas de violation des termes du SLA.
- (c) Un suivi ou monitoring des qualités de service ainsi que des conditions est réalisé par le SaaS Provider ou un Broker afin d'assurer que les termes agréés soient maintenus.

H. Évolution, Mises à jour et Maintenance des SaaS

Comme tout logiciel durant son cycle de vie, les SaaS doivent évoluer et être maintenus. L'organisation doit donc par-exemple vérifier :

- (a) La façon dont le fournisseur réalise les mises à jour sur les services ainsi que leur maintenance ;
- (b) Si les mises-à-jour lui seront notifiées et par quel moyen ;
- (c) Si les mises à jour et la maintenance s'opère de manière transparente ;
- (d) Si elle contribuera aux tests des upgrades et des modules des nouvelles versions du logiciel.

I. Plans tarifaires, Licences, Engagements

Les modes d'achats/paiements pour l'obtention des SaaS ou leurs modèles de licence diffèrent de celui d'un logiciel traditionnel. Ils diffèrent également selon les fournisseurs. En effet, l'organisation ne va pas acquérir ou détenir un logiciel. Les dépenses ne sont donc plus estimées en terme d'appropriation du produit, mais en termes d'utilisation d'un service selon un modèle de tarification défini par le fournisseur.

Ainsi, le fournisseur établit un ou plusieurs plan(s) tarifaire(s), la fréquence de facturation ainsi que le nombre d'utilisateurs supporté. La politique tarifaire peut être comptabilisée par période d'utilisation (e.g., au mois), par nombre d'invocations du service ou par quantité de ressources utilisées, par nombre d'utilisateurs, etc.

Une période d'engagement (contrat) peut être demandée par le fournisseur de même qu'il peut proposer une période d'essai gratuite.

L'organisation devra donc évaluer le modèle qui lui correspondra le mieux selon ses activités et ses besoins d'utilisation prévus.

J. Identification du Provider et de sa réputation

Une organisation serait souvent tentée de se diriger vers les solutions SaaS des grands fournisseurs bien connus. Cependant, elle pourrait trouver des services lui étant mieux adaptés chez des fournisseurs moins connus. Afin de réduire les craintes liées à l'incertitude d'un fournisseur, l'organisation pourrait se renseigner ou demander des informations sur l'entreprise. Par exemple, une description des activités réalisées, son statut, sa viabilité à long-terme, sa stabilité financière, l'évaluation donnée par ses clients, le nombre de ses clients SaaS ainsi que ses principaux clients et partenaires ou références. De cette manière, l'organisation pourrait mieux cerner la réputation et la confiance que dégage un fournisseur moins connu. Ce genre d'information pourrait typiquement être proposé par un Broker et cela afin d'éviter le lock-in des vendeurs les plus connus².

K. Autres critères

Il pourrait par ailleurs être intéressant de pouvoir (a) fournir une évaluation et un suivi pour le service reçu, (b) vérifier si le logiciel proposé respecte les normes de qualités logicielles (e.g. ISO 25000, IEEE 1061) et (c) si le Fournisseur d'Accès à Internet (FAI) permet d'assurer dans des conditions optimales l'utilisation du service par rapport aux qualités proposées.

2. Ceci pourrait permettre également de stimuler le marché de SaaS et de PME.

3.3 Étude d'un Modèle de Qualité pour SaaS

Sur base des caractéristiques et spécificités des SaaS que nous avons exposées précédemment, nous établissons ici un modèle de qualité pour l'évaluation des SaaS et de leur conformité avec les exigences qui pourraient être demandées par les utilisateurs en vue d'une sélection du meilleur service adéquat.

Nous tentons donc d'établir un modèle de qualité pour les SaaS qui apporte une complémentarité au modèle de qualité issu de la norme ISO 25000 SQuaRE³ dans le sens où notre modèle veut intégrer une évaluation de la satisfaction des utilisateurs, ce qui comprend son expérience personnelle avec le service et sa confiance accordée à son fournisseur.

Notre modèle définit deux grandes classes de paramètres permettant d'évaluer la qualité d'un Software-as-a-Service du point de vue de l'utilisateur. Il peut en quelques sortes être comparé au *Service Measurement Index (SMI) Framework* (voir Annexe C). Les deux classes, "*Paramètres Quantitatifs*" et "*Paramètres Qualitatifs*", proposent une série de critères et de propriétés pour lesquels l'utilisateur pourra juger bon de déterminer l'importance et le niveau/propriété selon les exigences requises pour ses activités business, ses règles stratégiques et légales. Ce modèle est réalisé dans le but de proposer à l'utilisateur une interface claire et structurée, lui permettant d'observer et d'évaluer les différents paramètres lors d'une sélection ou d'une évaluation d'un service.

3.3.1 Concepts

Avant de proposer le modèle annoncé, nous (re)définissons brièvement quelques concepts utilisés.

Service. Le Service fait référence au Software-as-a-Service qui est offert par le Fournisseur à l'Utilisateur ; et ce via un SaaS Broker.

SaaS Broker. Le SaaS Broker est l'intermédiaire entre les Fournisseurs et les Utilisateurs qui propose dans son intermédiation la sélection et l'évaluation des Services à l'aide du modèle de qualité que nous définissons présentement.

Fournisseur. Le Fournisseur fait référence au SaaS Provider qui offre un service d'applications logicielles SaaS à des Utilisateurs.

Utilisateur. L'Utilisateur fait référence au SaaS User qui réalise le choix et l'évaluation du Service ; mais également à l'utilisateur final (End-User) de ce Service.

SLA. Le SLA est le contrat spécifiant les paramètres tel que requis par l'Utilisateur et convenus avec le Fournisseur ; il comprend les paramètres quantitatifs et qualitatifs.

3. Software Quality Requirements and Evaluation qui veut harmoniser les normes ISO 9126 et ISO 14598 (qualité du produit logiciel et évaluation du produit logiciel)

Paramètres Quantitatifs Propriétés techniques de QoS mesurées par des méthodes d'audit et des outils de monitoring, permettant d'évaluer objectivement et d'analyser quantitativement le niveau de la qualité d'un service telles que la performance, la disponibilité et la fiabilité du service ainsi que toutes les autres propriétés de QoS quantitatives spécifiées dans le SLA. Pour chaque propriété, une probabilité ou une mesure quantitative et objective est réalisée. Ces mesures peuvent ainsi être évaluées avec des KPIs et des benchmarks.

Paramètres Qualitatifs. Propriétés subjectives concernant la qualité d'expérience d'utilisation du service, la satisfaction envers le service et son fournisseur. Ces propriétés peuvent renvoyer par exemple aux besoins de conformités, de légalités, de fiabilité, de confiance, de sécurité, etc. Elles font référence à des propriétés spécifiques à certains besoins et certaines demandes propres à l'utilisateur qui seront spécifiées dans le SLA. Elles ne sont pas quantifiables ou mesurables de manière formelle et automatisée. Elles sont propres à l'expérience d'utilisation du service par le SaaS User et de la satisfaction qu'il en retire. Ces propriétés doivent donc être collectées auprès des utilisateurs à l'aide d'un système d'évaluation par Feedback. Celui-ci peut être transmis via une méthode de cotation (expliquée si possible) et des commentaires.

Feedback. Le Feedback est l'évaluation subjective propre à un Utilisateur sur son retour d'expérience personnelle envers le Service obtenu. Cette évaluation peut être réalisée sous forme de cotations des paramètres sur une échelle de valeur, ou sous forme de commentaires textuels.

Satisfaction. La Satisfaction d'un Utilisateur dépend du niveau de la qualité du service reçu par l'Utilisateur et de son appréciation personnelle sur l'utilisation et les conditions générales de ce service.

Confiance (Trust). La Confiance indique le sentiment de sécurité et d'assurance envers un Fournisseur sur la qualité et la fiabilité de ses services et du respect des conditions convenues dans le SLA.

Réputation. La Réputation est un processus fondé sur la satisfaction, la fiabilité, l'efficacité et les expériences antérieures. Toutefois, nous devons veiller à ce que cette réputation soit la moins subjective possible car elle repose sur des perspectives humaines temporelles pouvant exposer, volontairement ou involontairement, des évaluations malhonnêtes [61]; d'où notre intérêt pour l'accès à une communauté d'évaluation.

3.3.2 Paramètres du Modèle de Qualité

-
Dans cette partie, nous nous intéressons de plus près aux différents paramètres (critères et propriétés) qu'un SaaS User pourrait prendre en considération lors de sa recherche d'un SaaS ou de l'évaluation de la qualité offerte pour celui-ci. Ces paramètres ont été relevés à l'aide de différents travaux et de normes de qualité logicielle [9, 62, 63, 64].

Pour plus de précisions et pour la définition des métriques, nous proposons une recherche ultérieure. Celle-ci pourra se baser sur différents travaux de la littérature tel que les

définitions⁴ des critères proposées par le SMI (voir Annexe C) et l'étude SMICloud (Chapitre 3, [9]) ainsi que dans [62, 65, 63, 20]. Notons que nous avons pris connaissance de l'état d'avancement de ce framework après avoir réalisé une première étude sur les différents paramètres.

En ce qui concerne les paramètres qualitatifs, nous proposons une investigation ultérieure dans la continuité de ce travail, ainsi que la construction d'un modèle d'évaluation de la satisfaction et de calcul du score.

Nous distinguons deux classes de paramètres (critères et propriétés) pour la sélection de SaaS : les critères quantitatifs reprenant des qualités de service mesurables (ou surveillées) (voir Tableau 3.1), et les propriétés qualitatives qui représentent une appréciation de l'utilisateur (voir Tableau 3.3). Ces paramètres devraient pouvoir refléter les exigences de qualités souhaitées et exigées par les SaaS Users.

Nous présentons ci-dessous chacune de ces propriétés ; leur métrique et la façon dont nous pouvons les obtenir devra faire l'objet d'une étude ultérieure. Le lecteur pourra trouver dans le Framework SMI (Annexe C) une définition pour les paramètres. Concernant l'évaluation des propriétés qualitatives, nous proposerons un système de cotation et de commentaires. Le système de cotation consisterait à assigner une valeur (sur une échelle) afin d'indiquer le niveau de satisfaction obtenu. Chaque valeur de l'échelle serait accompagnée d'une description présentant le niveau de qualité propre à la propriété concernée.

A. Paramètres Quantitatifs Voir Tableau 3.1

B. Paramètres Qualitatifs Voir Tableau 3.3

4. Voir également *Service Measurement Index Framework Version 2.1*, http://csmic.org/wp-content/uploads/2014/07/SMI_Overview_TwoPointOne1.pdf

TABLE 3.1 – Tableau des paramètres quantitatifs.

Critères/QoS	Définitions
<p>Performance</p> <p>Disponibilité (Availability) Fiabilité (Reliability) Temps de réponse / de transaction (Response / Transactional time) Scalabilité et élasticité (Scalability et Elasticity) Throughput Accuracy</p>	
<p>Sécurité</p> <p>Authentification Rôles d'accès Data Privacy Chiffrement des données Robustesse Backup - Disaster Recovery</p>	<p>ex. : RBAC</p> <p>Quelles sont les méthodes de Backup et de Recovery ?</p>
<p>Architecture</p> <p>Intégration & Interopérabilité</p> <p>Portabilité Adaptation</p>	<p>Abilité d'un service à s'intégrer facilement avec les autres applications et services.</p>

TABLE 3.2 – Tableau des paramètres qualitatifs.

Critères/Propriétés	Définitions
Fonctionnalités (Capabilities et suitability)	Capacité des fonctionnalités du SaaS à répondre correctement aux attentes de l'utilisateur
Économique Plan tarifaire - Licence Prix - Coûts Suivi de consommation Période d'essai Engagement	Flexibilités et méthodes de facturation Rapport qualité/prix-coût du SaaS Une période d'essai est-elle offerte ? L'utilisateur est-t-il tenu à s'engager pour une période déterminée ?
Qualité Logicielle Utilisabilité (Usability) Importation / Exportation des données Customisation - Personnalisation User Interface Mode Off-line Support - Helpdesk Documentation Update / Upgrade Maintenance Secteur d'activité Structure organisation visée	Productivité, efficience dans les tâches, accessibilité, satisfaction, compréhension, apprentissage Vers le SaaS, vers l'infrastructure locale, vers d'autres fournisseurs ? Intuitif, facile d'usage, esthétique, etc. Mécanisme qui permet aux utilisateurs de travailler avec un système hors-ligne et de se synchroniser une fois connecté. Support pour différentes "devices", support en cas de problème Manuels utilisateur, modules e-Learning, etc. Mise à jour suivie de l'application Transparence de la maintenance Secteur ciblé pour l'utilisation du service Taille de l'organisation en nombre de personnes.
Conformité / Certification / Legal / Politiques Sécurité Géo-localisation Legal Privacy "Sustainability"	Respect des normes de sécurité en vigueur Hébergement des applications et des données Est-ce conforme à la juridiction ? Impact sur l'économie, la société et l'environnement (e.g. eco/energy-efficiency, normes, chartes)

TABLE 3.3 – Tableau des paramètres qualitatifs.

Critères/Propriétés	Définitions
<p>Réputation et Confiance</p> <p>Identité du fournisseur Contacts du fournisseur Popularité du fournisseur Nombre de clients Croissance Users Rating Partenaires Satisfaction générale Garanties Respect des SLAs Résilience SLA Négociation SLA</p>	<p>Popularité du fournisseurs auprès des utilisateurs du Broker Indique le niveau d'utilisation (nouveau service ou service bien établi) Croissance du nombre d'utilisation au cours du temps Cotation des utilisateurs Exemples de partenariats créés</p> <p>Garanties qu'offre le fournisseur Niveau de respect des SLAs L'utilisateur peut-il résilier son contrat ? Le contrat est-il négociable ?</p>

3.4 Objectifs et Fonctionnalités du SaaS Broker

Nous décrivons les fonctionnalités que nous souhaitons intégrer à notre étude du modèle de SaaS Broker.

Nous envisageons la proposition d'offrir un mécanisme pour la sélection d'une solution SaaS conforme aux paramètres et aux propriétés définies par le SaaS User, ainsi qu'une plate-forme pour l'évaluation de la satisfaction qui sera basée sur les feedbacks.

Distinguons deux catégories de clients qui utiliserons le Broker : les SaaS Providers et les SaaS Users.

Le Broker apporterait aux SaaS Providers une meilleure visibilité auprès des SaaS Users. Avec un système de feedback et de réputation, le Broker vise à entraîner une compétition (saine) entre les SaaS Providers dont le but est d'obtenir une augmentation des performances et des qualités, ainsi que mieux comprendre les besoins des organisations en termes de fonctionnalités et de qualités. Les SaaS Providers se font connaître auprès du Broker et enregistrent les SaaS qu'ils mettent à disposition des SaaS Users. Ils fournissent également les informations précises pour la description du service.

Pour les SaaS Users, le Broker a pour objectifs de conduire à une sélection efficiente de services, tout en permettant une interopérabilité, une meilleure gestion des SLAs entre les parties, ainsi qu'assurer une plate-forme sécurisée d'échange et une gestion d'identité unique pour l'ensemble des services souscrits. Les SaaS Users s'inscrivent auprès du Broker afin d'avoir accès aux services qu'il propose ainsi qu'au catalogue de SaaS. Ils peuvent ainsi effectuer leur requête en vue de la sélection d'un SaaS ou transmettre l'évaluation du service.

Remarquons également que des systèmes externes pourraient venir se connecter au Broker tels que le third party de monitoring et d'autres Brokers dans le cas d'une Fédération.

3.4.1 Quels sont les objectifs du SaaS Broker ?

Nous désirons proposer un système intermédiaire entre Service Users et Service Providers de type SaaS Broker (Voir Figure 3.1) proposant l'Arbitrage de SaaS et l'évaluation de la satisfaction des utilisateurs, car nous pensons qu'une des difficultés à l'heure actuelle dans la sélection d'applications SaaS est le manque de méthodes et de modèles pour évaluer et comparer la qualité d'un SaaS. Notre Broker a donc pour objectif de palier ce manque.

L'objectif principal de ce SaaS Broker est donc de rendre plus efficient le choix des SaaS Users dans l'achat et l'utilisation d'un Software-as-a-Service. L'enjeu est de proposer un système de sélection de SaaS affiné donnant la possibilité aux utilisateurs de ce système d'être précis dans leur requête tant au niveau des fonctionnalités et des qualités de services (QoS) traditionnelles qu'au niveau des règles stratégiques (conformités, assurance, utilisation des données, etc.) exigées par ses activités. Ces critères et propriétés ont été exposées précédemment.

De plus, le Broker doit permettre la gestion du Service Level Agreement pour le compte du SaaS User afin de lui assurer la réalisation du service conformément à ses attentes. Pour vérifier cette conformité, le Broker réalise (ou délègue) le monitoring et le suivi de ces QoS et de la conformité des propriétés demandées par le SaaS User. En plus de ces deux objectifs, le Broker propose un système d'intelligence collective permettant de rassembler les évaluations des utilisateurs dans le but de créer une communauté d'évaluation de fournisseurs et de créer un réseau de confiance et de réputation.

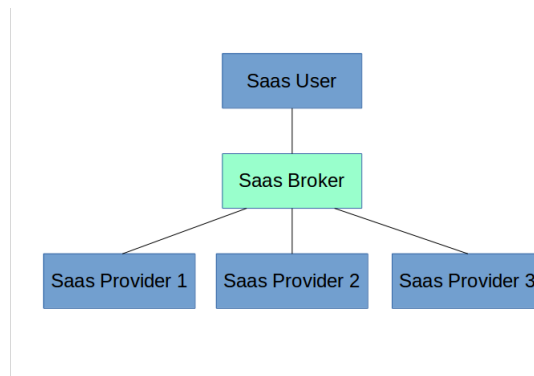


FIGURE 3.1 – SaaS Broker entre Users et Providers.

3.4.2 Quelles sont les fonctionnalités du SaaS Broker ?

Le Broker proposé pour les applications Software-as-a-Service présente donc les fonctionnalités suivantes : (a) *Plate-forme sécurisée*, (b) *Catalogue de fournisseurs et de SaaS*, (c) *Sélection de SaaS*, (d) *Gestion des Service Level Agreements*, (e) *Monitoring et Suivi des qualités*, (f) *Intelligence collective – Communauté d'Évaluation* et (g) *Gestion des identités et des accès*.

Nous définissons ces fonctionnalités principales de notre Broker de la façon suivante :

A. Plate-forme sécurisée

Le Broker agit en tant que plate-forme intermédiaire sécurisée entre SaaS Providers et SaaS Users.

Les SaaS Providers s'y rendent visibles à une communauté de SaaS Users et y enregistrent leurs SaaS ainsi que la description des différentes propriétés du service qui sera délivré (fonctionnalités, QoS, conformités, stratégies, etc.).

Les SaaS Users s'inscrivent auprès du Brokers afin d'avoir accès à un catalogue de SaaS et de recevoir du Broker son expertise pour la sélection d'un SaaS parmi ce catalogue.

Le Broker propose aux SaaS Users une liste de services correspondant à leur requête, mais c'est à eux que revient la décision finale. Il peut également fournir les valeurs ajoutées traditionnelles de Brokerage (Intégration, Interopérabilité).

Cette plate-forme collecte les évaluations des utilisateurs et réalise un suivi de leur satisfaction à propos des services délivrés par les différents fournisseurs.

Ceci a pour but (a) de permettre une nouvelle méthode de vérification la qualité d'un SaaS afin d'aider l'utilisateur dans sa sélection ; et (b) de créer une concurrence saine entre les fournisseurs et de ce fait améliorer la qualité et les conformité des services délivrés.

B. Catalogue de fournisseurs et de SaaS

Le Broker établit et maintient un catalogue étoffé de fournisseurs et d'applications. Ce catalogue permet de catégoriser les différents fournisseurs et SaaS selon leurs caractéristiques et propriétés. La description d'un fournisseur contient entre autre son identité et les informations de contacts, la description de ses activités, son nombre de clients SaaS ainsi que ses principaux clients de références.

La description des SaaS contient notamment les informations claires et précises concernant

les fonctionnalités, les domaines d'applications visés, la tarification, les qualités de services, les conformités respectées et les formalités légales, la fiabilité basée sur les évaluations, etc. Une définition des différents critères et des propriétés concernant la qualité des SaaS qui peuvent être prises en compte lors de la sélection ont été décrites précédemment.

C. Sélection de SaaS

La proposition d'une liste classée selon la pertinence et le score accordés aux SaaS potentiels est la fonctionnalité centrale du Broker.

À l'aide d'une interface, le système permet aux SaaS Users d'effectuer leur requête pour la sélection d'une liste de SaaS. L'interface propose à l'utilisateur de fournir au Broker les fonctionnalités souhaitées pour une application recherchée (pour le SaaS Users opérer ses activités business) ainsi que le niveau de qualité pour chaque critère et conditions proposées par l'interface ou un ensemble de ceux-ci.

La requête est analysée par le Broker qui sélectionne dans le catalogue la liste de fournisseurs et SaaS potentiellement conformes aux conditions demandées. Selon la pertinence avec ces conditions, le Broker établit un classement en accordant un score à chaque service (Ranking). La décision finale du service à obtenir revient aux SaaS Users.

D. Gestion des Service Level Agreements

Le Broker réalise la gestion (automatisée) du SLA pour le compte des SaaS Users. Il va négocier et conclure les conditions de prestation du service avec le SaaS Provider.

Durant l'exécution du service, Il assure pour le compte des SaaS Users le suivi de la conformité et de la maintenance des conditions spécifiées dans le SLA.

Ce suivi est réalisé par les opérations de monitoring et les évaluations de l'utilisateur et de ses pairs.

E. Monitoring et Suivi des qualités

Afin d'évaluer, de vérifier et de s'assurer que les qualités de service (QoS) quantitatives des SaaS et les conditions du SLA soient effectivement réalisées et maintenues, le Broker effectue le monitoring et le suivi des conditions de ces services prestés. Il peut déléguer ces opérations à un third-party spécialisé.

F. Intelligence collective – Communauté d'Évaluation

Le Broker instaure un système d'intelligence collective (Community of Evaluation – CoE) pour l'évaluation de la qualité des différents SaaS, de la satisfaction de ses clients et de la confiance envers les fournisseurs.

Le Broker récolte donc régulièrement les feedbacks et les évaluations des utilisateurs et des fournisseurs afin d'établir les niveaux de qualité, de confiance et de réputation. Il s'agit du suivi des propriétés non mesurables quantitativement par un système de monitoring.

Notons que ce système doit veiller à l'honnêteté de l'évaluation en mettant en place un modèle d'incitants et de pénalités. La "Communauté d'Évaluation" est un groupe de SaaS Users ou utilisateurs qui sont unis par des intérêts communs issus de l'évaluation des services en vue d'obtenir satisfaction et confiance dans leur choix de SaaS.

G. Gestion des identités et des accès

Les Providers et les Users possèdent chacun un compte unique personnalisé ainsi qu'un identifiant unique pour l'accès à tous les SaaS. Le Broker effectue donc la gestion des identités et des authentifications des SaaS Users auprès de leurs Providers par un identifiant unique (Single Sign-On) et permet des contrôles d'accès aux applications et à leurs données.

H. Autres Fonctionnalités

L'intégration et l'interopérabilité avec les applications locales des utilisateurs sont réalisées. Le Broker assure également la sécurité de la plate-forme et des échanges. Un système peut être mis en place pour la facturation des opérations du Broker, mais également pour l'achat des SaaS. Le Broker réalise alors la facturation pour le compte des Providers selon le SLA.

3.5 Modèle d'Architecture du SaaS Broker

Dans cette section, nous présentons notre proposition de modèle pour notre SaaS Broker. Nous poserons tout d'abord quelques hypothèses sur le système étudié. Nous décrirons ensuite les différents agents et acteurs de l'environnement du système. Après cela, nous pourrions enfin exposer et détailler chacun des composants et mécanismes haut-niveau du modèle théorique et architectural de Broker.

3.5.1 Hypothèses sur le système

Ci-dessous, nous posons quelques hypothèses sur le fonctionnement et l'architecture du système afin de simplifier notre propos et de nous concentrer sur certaines fonctionnalités essentielles de l'architecture haut niveau du système. Elles permettent également une bonne exécution des opérations. Ces hypothèses sont les suivantes :

- La plate-forme est sécurisée et fiable.
- Le système comporte déjà un certain nombre de SaaS Providers et de SaaS Users enregistrés et fidélisés, ainsi qu'un catalogue de SaaS étendu et un large nombre d'évaluations.
- Les critères et les propriétés de qualités ainsi que les conditions ont été validées par une étude, et sont compréhensibles pour les SaaS Users.
- Les propriétés du service demandés aux SaaS Users sont soit pertinentes pour leur requête soit facultatives. Les SaaS Users ne sont pas obligés de fournir un niveau ou une condition pour chaque propriété. Un profil par défaut peut être choisi.
- Les SaaS Providers disposent de leur Private Cloud. Par cette hypothèse, nous supposons qu'ils sont les seuls responsables de l'hébergement et de la qualité de leurs applications et services. De cette manière, les possibles responsabilités des Cloud Providers ne sont pas impliquées dans la qualité de service des applications. Nous pourrions également considérer, dans le cas où les SaaS Providers ont recours aux services des Cloud Providers, que les conditions spécifiées dans leur SLA sont toujours respectées.
- Dans un premier temps, nous admettons que le monitoring des critères de qualité de service quantitatifs (QoS) est délégué à un third party de confiance. Il transmet les résultats au Broker en vue de leur évaluation par rapport aux SLAs.

- Les feedbacks et les évaluations sont des jugements honnêtes. Un mécanisme d’incitants et de pénalités pourrait être instauré afin de soulever l’hypothèse.
- Lors de la sélection des applications SaaS potentielles pour une requête, nous considérons qu’elles réalisent toutes les mêmes fonctionnalités ; et donc qu’elles sont substituables dans leurs fonctionnalités.
- Le catalogue de SaaS contient toujours une application qui satisfera la requête d’un utilisateur. Le Broker peut toujours offrir une solution à l’utilisateur.
- Nous n’investiguerons pas sur la manière dont les SaaS sont délivrés. Nous supposons qu’ils sont directement exécutés entre le SaaS Provider sélectionné et l’utilisateur.

Bien que notre SaaS Broker a pour rôle l’arbitrage de services qui implique une agrégation possible de plusieurs services dans une situation où aucun service ne pourrait à lui seul convenir aux besoins d’un SaaS User, nous ne prenons pas en compte cette caractéristique. Nous prenons donc pour hypothèse que les SaaS Users introduisent uniquement des requêtes pour des SaaS dont les fonctionnalités donnent accès à une solution complète existante dans le catalogue SaaS. Nous pensons qu’une étude sur l’agrégation de SaaS devrait être réalisée au préalable. Cependant, si nous considérons que des SaaS peuvent être proposés par des Services Web, une recherche et proposition de composition de Services Web pertinents pourrait être mise en place. Cela pourrait faire l’objet d’un travail ultérieur.

3.5.2 Acteurs du système

Ci-dessous, nous présentons les acteurs participant à l’environnement. La Figure 3.2 montre les différentes interactions dans l’environnement du système.

Les SaaS Users - Communauté d’Évaluation (CoE)

Les SaaS Users s’enregistrent auprès du Broker afin de profiter de l’expertise du mécanisme de sélection de SaaS et d’avoir accès à la *Communauté d’Évaluation* (Community of Evaluation – CoE) et d’y participer en proposant un suivi et une évaluation (*Review*) des services obtenus auprès des SaaS Providers. Lors de leur inscription, ils fournissent leur identité ainsi que leurs préférences en termes de qualité de services et de conditions d’exécution des services (e.g. qualité logicielle, conformités légales, etc.) Ils transmettent leur requête pour une sélection de services au Broker via une “interface de requête” (*Request Interface*, typiquement une interface web). Ils sélectionnent ainsi via cette interface le niveau des qualités requises et les conditions demandées. Ils sont tenus à participer à la CoE en donnant des évaluations et des suivis honnêtes. La CoE ainsi formée permet aux SaaS User la consultation des avis de leurs pairs et la mise en place d’un réseau de confiance et de réputation entre la tri-partite Provider-Broker-User. Les évaluations de satisfaction réalisées poussent à la comparaison des fournisseurs (proposé par le Broker lors du ranking) et peut conduire à une concurrence saine entre ceux-ci.

Le SaaS Broker

Le SaaS Broker réalise les opérations d’intermédiation, d’Arbitrage de services et de gestion des SLAs entre les SaaS Users et les SaaS Providers. Il maintient un catalogue (*Service Repository*) de fournisseurs et de SaaS. Il permet aux SaaS Users d’effectuer une requête de sélection de services via la *Request Interface*. Il recherche et sélectionne (*Selection Management*) alors dans son catalogue les SaaS pertinents par rapport à la demande, et

procède au calcul d'un score de conformité avec les conditions/propriétés demandées pour établir le *Ranking* et fournir un classement des services à l'utilisateur. Celui-ci pourra alors comparer les différents SaaS et choisir celui qui lui convient le mieux (en principe le mieux côté).

Une fois le choix effectué, le Broker va réaliser une négociation du SLA avec le SaaS Provider et conclure l'achat du service.

Par la suite, il vérifie également que les propriétés effectivement délivrées sont conformes au SLA grâce aux résultats du monitoring reçus et des feedbacks des utilisateurs.

Il maintient dans un registre les préférences des SaaS Users afin de leur proposer des profils types (*Profile Management*). Son composant *Identity and Access Management* permet la gestion des identités et des contrôles d'accès à l'aide d'un identifiant unique (Single Sign-On). Il gère également la *Community of Evaluation* en proposant une plate-forme (*CoE platform*) d'évaluations/de suivis et de feedbacks sur les fournisseurs et leurs services obtenus : qualités et conformités effectives par rapport aux exigences spécifiées dans le SLA ou annoncées auprès du Broker. Les SaaS Users pourront donc y effectuer leurs *Review* via la *Review Interface*.

Les SaaS Providers

Les SaaS Providers peuvent proposer plusieurs applications SaaS qu'ils enregistreront auprès du SaaS Broker. Lors de l'enregistrement des SaaS, ils fournissent les spécifications de l'application, les qualités de service (QoS, ils peuvent au préalable réaliser un monitoring de ces qualités) ainsi que les diverses conditions du service. Lors de leur inscription auprès du Broker, ils veillent à transmettre les informations sur leur identité et leurs activités commerciales.

Lors d'une baisse apparente ou potentielle de QoS, ils s'assurent d'allouer les ressources nécessaires afin de maintenir le niveau conforme aux conditions fournies et conclues par les SLAs. Ils implémentent un *SLA Manager* afin de négocier et de gérer les conditions du SLA d'une requête avec le SaaS Broker.

L'infrastructure de Monitoring - Third Party

L'infrastructure de Monitoring réalise pour le Broker les mesures des qualités de service (QoS) quantifiables.

Le Third Party transmet les résultats au Broker qui effectuera l'évaluation de la conformité de ces qualités par rapport aux spécifications des SLAs. L'usage d'un même outil de monitoring permet une évaluation équitable entre les différents Providers en vue de leur comparaison.

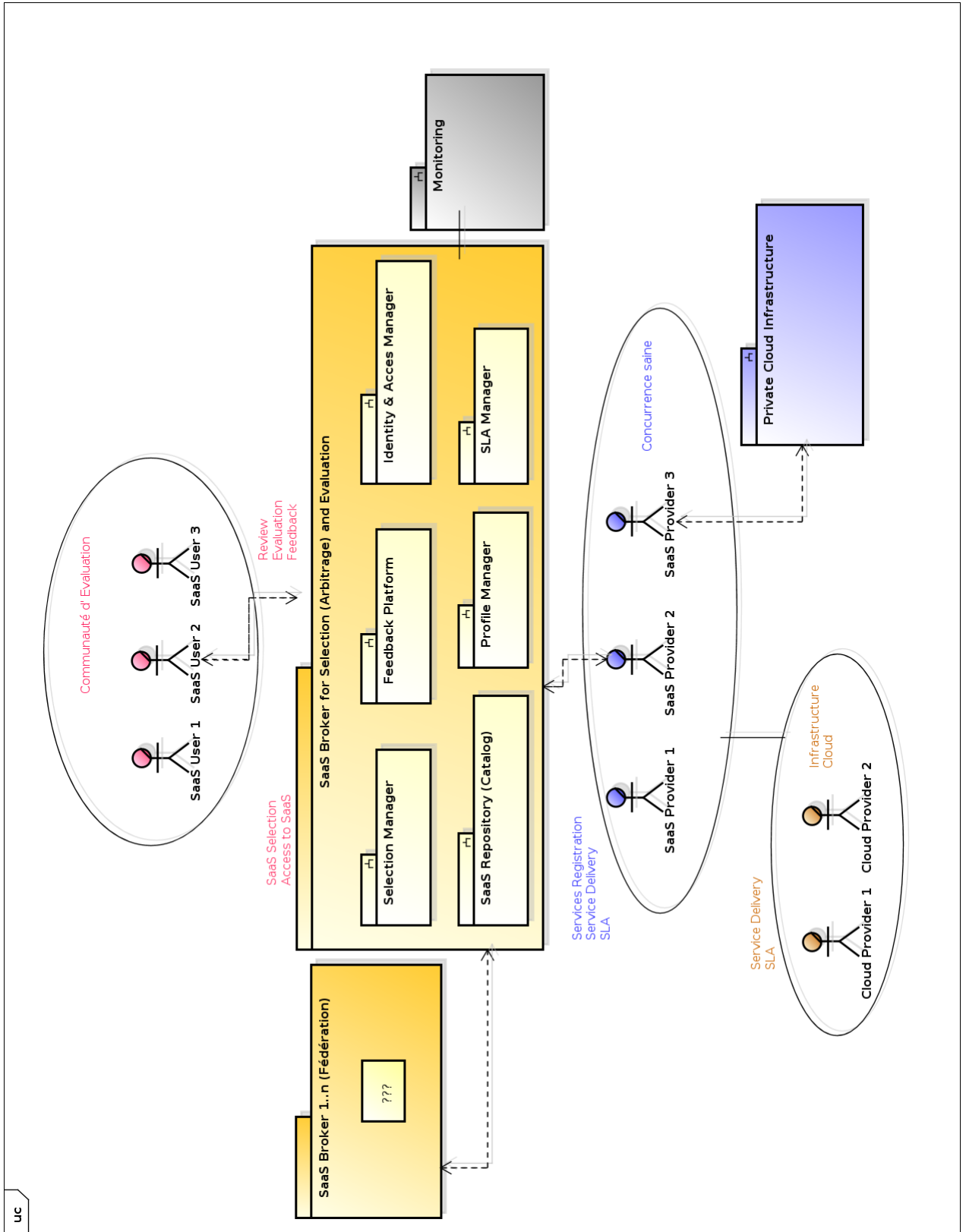


FIGURE 3.2 – Environnement du SaaS Broker.

3.5.3 Composants haut-niveau du Broker

Nous présentons ici les différents composants principaux du Broker.

SaaS Repository

Le *SaaS Repository*, ou catalogue des SaaS et des fournisseurs, regroupe l'ensemble des applications SaaS qui ont été enregistrées par les SaaS Providers afin de rendre celles-ci disponibles par la plate-forme du Broker.

Ces SaaS sont répertoriés (par exemple selon une ontologie définie) sémantiquement selon leur(s) fonctionnalité(s) et propriétés décrites par leur Provider. Ce catalogue peut ainsi référencer l'ensemble des SaaS Providers et leurs informations, et fournir une liste de leurs services offerts. Il sert donc de base de données-répertoire des différentes applications SaaS enregistrées ainsi que leurs spécifications, qui sont catégorisées selon divers principaux critères tels leur(s) fonctionnalité(s), le secteur d'activité, le type d'organisation pour lesquels elles sont destinées etc. Cette catégorisation permet une première analyse et sélection d'une liste des SaaS appropriés lors d'une nouvelle requête faite par un SaaS User. L'enregistrement des services et de leurs spécifications pourrait être réalisée via une interface web.

Selection Manager

À l'aide d'une interface dans un navigateur web, le SaaS User effectue le choix des fonctionnalités logicielles pour lesquelles il désire acquérir un SaaS. Le SaaS User sélectionne la solution la mieux adaptée parmi une liste de fonctionnalités répertoriées dans le catalogue des SaaS (e.g. CRM, ERP, HRM, Collaboration, etc.).

Il peut affiner cette sélection en précisant un secteur d'activité et/ou la taille de l'organisation pour laquelle le service pourrait être fourni. Ensuite, le SaaS User se voit proposer, toujours via une interface, les différents paramètres de qualité dont il peut, soit lui-même définir une valeur, soit laisser le paramétrage proposé par défaut par le système. Pour chaque paramètre, il peut attribuer un poids spécifiant ainsi l'importance qu'il accorde à celui-ci.

La requête envoyée par le SaaS User est récupérée et analysée par le système qui recherche dans le SaaS Repository une liste de service correspondants d'une part aux fonctionnalités désirées et d'autre part aux différents paramètres de qualité et aux propriétés souhaitées. Chaque requête est décomposée en deux parties : l'une en vue de rechercher, dans le catalogue de SaaS, les services relatifs aux fonctionnalités demandées et d'en ressortir une liste, l'autre en vue de traiter le paramétrage des exigences de qualité demandé en vue d'un classement selon la pertinence avec ces dernières.

Le composant établit une liste de SaaS correspondants aux fonctionnalités demandées et réalise un classement sur base du score attribué à chacun. L'utilisateur peut ensuite effectuer son choix parmi la liste de services proposés.

Plate-forme d'évaluation La plate-forme d'évaluation est une interface interactive qui permet aux utilisateurs de transmettre leurs évaluations à propos des services délivrés. Ces derniers peuvent également consulter les évaluations effectuées par les autres utilisateurs de ces mêmes services.

SLA Manager Le SLA Manager négocie et garantit le respect des termes du SLA pour le compte de l'utilisateur. Il va s'assurer de construire le SLA propre à l'utilisateur sur base d'un

contrat type et des paramètres spécifiques sélectionnés par l'utilisateur. Ensuite, il négociera automatiquement ce SLA avec le fournisseur choisi par l'utilisateur et avertira celui-ci du résultat de la négociation. Si la négociation échoue, il propose à l'utilisateur de choisir un autre fournisseur. Par la suite, il prend en charge la vérification du respect des termes du SLA.

Identity and Acces Manager – SSO L'authentification des accès aux services et aux ressources est un aspect essentiel pour sécuriser l'environnement et l'utilisation des SaaS. Le gestionnaire des identités doit permettre une authentification pour chacun SaaS auprès de chaque fournisseur. Les différents services utilisés par un SaaS User (c'est-à-dire, chaque entreprise ou organisation) sont groupés et le système doit gérer un identifiant unique.

Composants supplémentaires

Relevons encore la nécessité de module pour les opérations suivantes :

- L'enregistrement des SaaS Providers et des SaaS Users auprès de la plate-forme (système).
- La gestion des comptes, des profils et des préférences (Profile Manager).
- Gestionnaire de profils “par défaut” pour la sélection des qualités d'un SaaS lors d'une requête et des SLAs associés. Ils sont construits sur base de l'historique des requêtes et des feedbacks reçus.
- Interopérabilité – API.

3.5.4 Illustrations du modèle

La Figure 3.3 et la Figure 3.4 illustrent le modèle proposé ainsi que les principaux composants du Broker dont nous avons discuté ci-dessus.

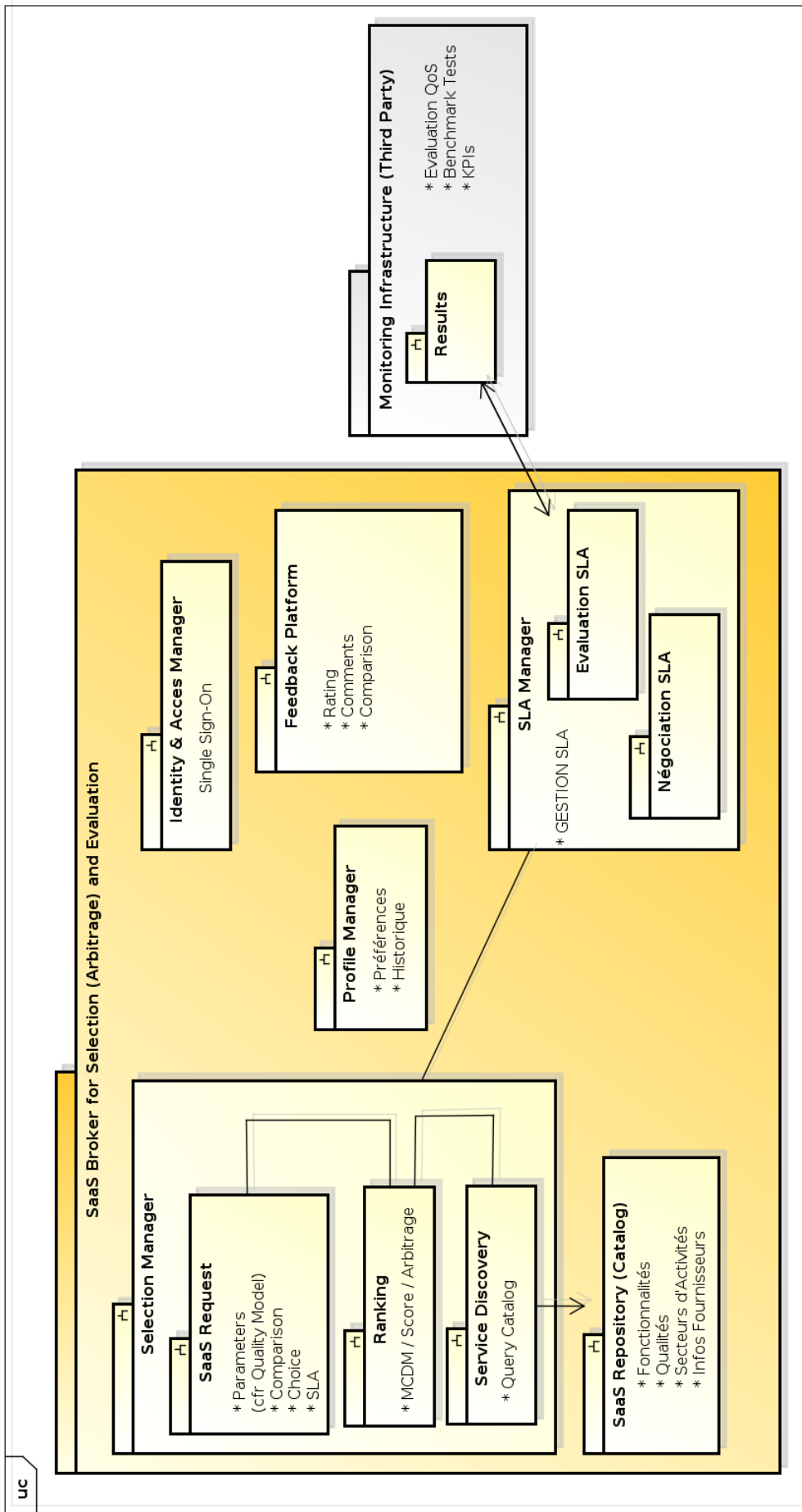


FIGURE 3.3 – Composants du SaaS Broker.

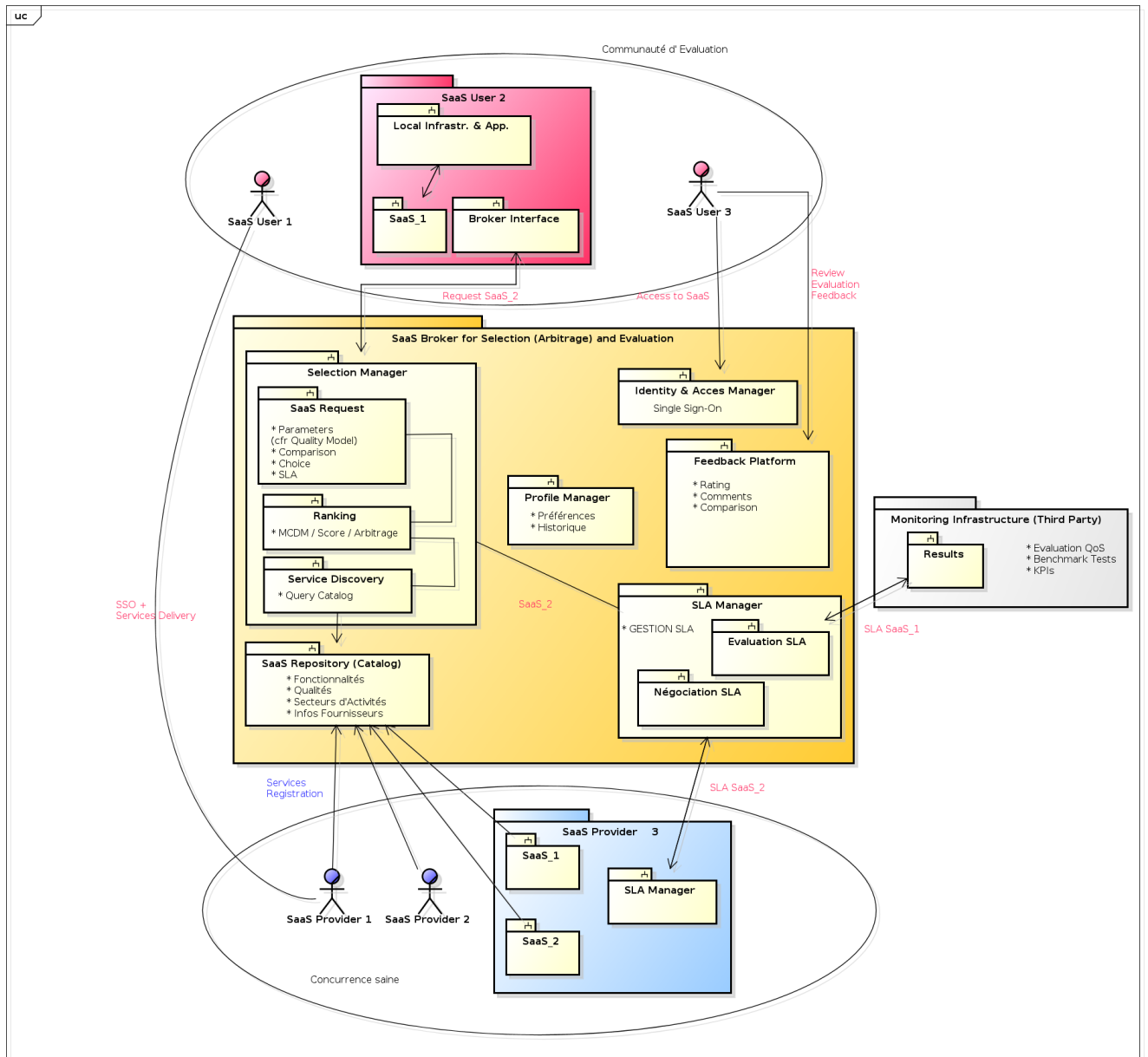


FIGURE 3.4 – Modélisation de l'architecture et du fonctionnement du SaaS Broker.

Chapitre 4

Discussion

Pour terminer ce travail, nous proposons un chapitre de discussion autour des réflexions et des hypothèses prises au cours de notre étude. Nous décrirons également les perspectives du Broker proposé ainsi que les limites de ce travail. À cet effet, nous proposerons des orientations pour la continuité de cette étude de recherche à propos de ce SaaS Broker.

4.1 Avantages et différenciation de notre Broker

Le Broker proposé permet une comparaison efficace des différents SaaS correspondants à la demande de l'utilisateur ainsi qu'une assistance automatisée dans le choix d'un de ceux-ci. Ce système offre la possibilité à l'utilisateur de gérer tous ses SaaS en un même endroit et d'y accéder à l'aide d'un identifiant unique. Il comble ainsi les lacunes des SaaS marketplaces existants. De plus, grâce à la plate-forme d'évaluation de la qualité des services et de leur fournisseur, il favorise la création d'une place et d'un réseau de confiance.

4.2 Étendue de la recherche

Nous proposons un modèle de qualité pour les SaaS complémentaire aux modèles connus (ISO 25000 et SMI) et nous souhaitons l'orienter vers l'évaluation de la satisfaction des SaaS Users envers les Providers. En ce sens, nous voudrions y ajouter des propriétés subjectives (feedback). Cependant par manque de temps, nous n'avons pu établir ce modèle et définir clairement les différents paramètres que nous désirions ajouter. Pour les futures recherches, nous suggérons donc de compléter cette étude par les paramètres qualitatifs à ajouter au modèle SMI. Celui-ci pourrait dans un premier temps être largement suffisant. Il s'agirait ici, d'une amélioration pour un modèle à destination des services d'applications logicielles (SaaS).

Vu l'ampleur de la problématique, nous avons dû également simplifier nos démarches dans l'étude du modèle d'architecture du Broker. Nous avons donc proposé une première version et les premières pistes pour le modèle de qualité et l'architecture haut niveau du Broker. Concernant l'architecture, une modélisation ou schématisation pourrait être donnée et complétée.

Nous avons entrepris le développement d'un nouveau modèle de qualité pour les SaaS. Cependant, nous n'avons pas eu l'opportunité de vérifier l'existence d'une norme pour la qualité des services informatiques au sens large.

Nous n'avons pas eu l'occasion de nous interroger sur le développement d'une possible agrégation des SaaS entre eux ni sur l'opinion des fournisseurs à ce propos. De même, la possibilité de créer une fédération de Brokers reste un point ouvert.

Bien que nous travaillons avec des SaaS, nous remarquons que de nombreux aspects de qualité (notamment ceux des infrastructures Clouds) sont de la responsabilité des Cloud Providers en lieu et place des SaaS Providers. Ainsi, nous notons qu'une coopération entre Cloud Brokers et SaaS Brokers peut se révéler efficace afin d'assurer cette qualité vis-à-vis de l'utilisateur [30]. Par exemple, un Cloud subissant une panne ou une perte de données est de la responsabilité du Cloud Provider mais aura une répercussion sur toute la chaîne et donc sur la responsabilité du SaaS Provider.

Finalement, nous avons remarqué que la portée de ces recherches et de cette étude s'étendent bien au-delà du cadre d'un Mémoire et du temps mis à la disposition pour la réalisation de celui-ci. Pour la continuité de ce travail, nous proposons donc de diviser cette recherche selon les différents composants et mécanismes constituant l'architecture présentée.

4.3 Perspectives de développements ultérieurs

Nous suggérons de diviser le travail de recherche en différentes phases, chaque composant du modèle présenté faisant l'objet d'une étude particulière. Ainsi, nous obtenons des études ciblées et permettant d'approfondir les recherches. Concernant notre modèle de qualité, nous proposons de continuer la recherche dans la direction des paramètres qualitatifs proposés qui s'appuient sur le feedback et le retour d'expérience des utilisateurs concernant les propriétés qui ne peuvent pas être mesurées. D'un point de vue architectural, nous proposons de poursuivre l'investigation réalisée dans ce travail par un enrichissement du modèle en spécifiant de manière plus précise les mécanismes de chaque composant. Par exemple, lors de nos recherches, nous avons entre autre pris connaissance de différents mécanismes de sélection tels que les résolutions de problèmes multi-critères (MCDM et AHP, [9, 62]), de modèles d'évaluation de confiance et d'automatisation de la gestion des SLAs.

4.3.1 Comment implémenter l'architecture du Broker ?

Comme mentionné précédemment, nous proposons d'étudier séparément les différents composants et mécanismes de l'architecture du Broker.

Ainsi, le mécanisme de sélection pourrait faire appel à une implémentation d'un programme de résolution de problèmes à l'aide d'un algorithme AHP.

Les interfaces web d'interaction entre les acteurs et le Broker pourraient faire l'objet de Services Web.

Concernant la gestion des SLAs, certains travaux proposent une étude de l'automatisation des processus de négociation et d'évaluation de leur conformité.

4.3.2 Comment évaluer le modèle de Broker proposé ?

Actuellement, nous ne pouvons pas évaluer et valider nos propos. Nous proposons une évaluation théorique en comparaison avec divers travaux réalisés par le Brokerage d'Infrastructures Clouds ainsi que par la réalisation d'une étude de cas théoriques suivie d'une mise en pratique.

Nous suggérons de réaliser une enquête auprès des entreprises et des organisations (SaaS

Users), utilisant ou souhaitant adopter des Software-as-a-Service, afin de prendre conscience de leurs besoins effectifs dans la sélection et la comparaison des services. Nous pensons par là à des questionnaires demandant la pertinence des paramètres (essentiel / non-essentiel) d'évaluation proposés ainsi que celles des informations que les SaaS Users souhaiteraient obtenir pour effectuer leur choix de SaaS.

Une autre enquête pourrait également porter sur l'intérêt de rejoindre une Communauté d'Évaluation afin de créer un réseau de confiance avec les différents fournisseurs et de partager l'information avec les SaaS Users pairs.

Une enquête supplémentaire auprès des fournisseurs (SaaS Providers) pourrait éventuellement être organisée afin de cibler la possibilité de diffuser les informations relatives aux paramètres d'évaluation.

Après avoir conçu un modèle théorique abouti et implémenté un premier prototype de framework fonctionnel, nous proposons de réaliser une étude de cas théorique suivie d'une mise en situation réelle du prototype si l'étude de cas se révèle concluante. Ensuite, toujours dans un principe d'expérimentation, nous proposons la mise en place d'une communauté rassemblant quelques PME ou entreprises volontaires se vouant à l'étude de la concrétisation du projet et de son évolution.

Conclusion

Nous voici arrivés au terme de notre étude théorique, préalable à la réalisation d'un nouveau type de Service Broker pour applications SaaS.

À l'heure où les Cloud Service Brokers vont tenir un rôle essentiel dans le paradigme du Cloud Computing, ce travail constitue un premier apport dans ce que nous souhaitons être un projet de recherche continué, tant d'un point de vue architectural que d'un point de vue économique (Business Model).

Tout d'abord, nous avons établi un cadre théorique rassemblant les différents concepts clés à la base de ce travail. Celui-ci constitue un recueil de définitions autour du concept de Cloud Computing et des Service Brokers.

Le chapitre "état de l'art" propose un large éventail des travaux de recherches actuels dans le domaine.

Le point de départ de nos recherches était l'étude d'un mécanisme d'évaluation des fournisseurs SaaS. Cependant, malgré le large éventail des différents travaux de recherches actuels mis en avant et analysés lors de la revue de la littérature (cfr l'état de l'art), nous avons constaté qu'il existait un manque de travaux sur les Brokers d'applications SaaS.

Afin de pouvoir proposer cette évaluation des fournisseurs et de leurs services, nous avons choisi d'aborder la problématique en commençant par l'étude des spécificités des SaaS et de leur implication quant aux paramètres clés à prendre en considération pour leur comparaison et sélection, mais aussi pour l'évaluation de la satisfaction du service obtenu.

Ceci nous a amené à définir un modèle de qualité à intégrer dans un mécanisme de comparaison en vue de la sélection mais aussi pouvant servir pour réaliser une plate-forme d'évaluation rassemblant une communauté de SaaS Users. Cette plate-forme d'évaluation a pour but de créer un réseau de confiance et de réputation entre fournisseurs et utilisateurs. En nous basant sur ce modèle de qualité, nous avons pu concevoir un modèle théorique pour définir la mise en oeuvre d'une architecture haut niveau d'un nouveau type de SaaS Broker.

Pour permettre la mise en oeuvre effective de cette architecture, nous pensons qu'il faudrait approfondir ce travail en menant des études complémentaires sur chacun des composants et des mécanismes qui y sont proposés. Le but de ces études serait de préciser et d'implémenter le fonctionnement de chacun de ces composants.

Ce travail nous a permis d'aborder une nouvelle problématique et nous a donné l'occasion de côtoyer de plus près le monde de la recherche.

Nous osons croire qu'il aura pu apporter un éclairage sur les différents concepts relatifs aux

Services Brokers, et plus précisément aux SaaS Brokers, permettant d'avoir une meilleure vue d'ensemble et une meilleure compréhension du sujet.

Nous espérons également que les bases jetées pour la mise en œuvre d'un nouveau type de SaaS Broker permettront à ceux qui s'en inspireront de mener à bien la concrétisation de cette solution.

Annexe A

Revue des définitions pour le paradigme du Cloud Computing

Présentons un passage en revue des définitions les plus courantes pour le paradigme du Cloud Computing.

Tout d’abord, les entreprises de recherches et d’études Forrester et Gartner exposent respectivement le paradigme du Cloud Computing comme étant un *“potentiel IT standardisé (services, software ou infrastructure), délivré via Internet de manière self-service et pay-per-use”*; ainsi qu’un *“style d’informatique dans lequel des capacités IT massivement adaptables sont délivrées en tant que service à des clients externes utilisant les technologies de l’Internet avec lesquelles les services sont suivis par des métriques définies et convenues avec chacun des utilisateurs afin de permettre des modèles de déploiement multiples”*.

Buyya et al. nous indique que le monde informatique est en transformation vers un monde où les logiciels développés seront à consommer “as-a-Service”, plutôt que de les installer et les faire exécuter sur des ordinateurs individuels. Dans [6, 66], ils décrivent le Cloud Computing comme étant une infrastructure *“depuis laquelle les aspects business et les utilisateurs sont capables d’accéder à des applications, à la demande, depuis n’importe quel endroit dans le monde”*. Ils ajoutent que cette infrastructure est *“un type de système parallèle et distribué qui consiste en une collection d’ordinateurs interconnectés et virtualisés qui sont eux-mêmes provisionnés, accédés dynamiquement et présentés comme une ou plusieurs ressources informatiques unies”*. Ces ressources se basent sur des Service-Level Agreements (SLA) établis à travers des négociations entre le fournisseur de services et le consommateur-utilisateur final. De plus, dans *“Cloud Computing : Principles and Paradigms”* [67], publié avec J. Broberg et A.M. Goscinski, Buyya décrit un modèle business pour une livraison à la demande et offrant des ressources informatiques, du stockage et des logiciels “as-a-Service” pour lesquels le consommateur-utilisateur paye les fournisseurs à l’utilisation. Ce modèle est appelé l’*“Utility Computing”* et est similaire à la manière dont nous obtenons les commodités publique (*“public utility”*) telles que l’eau, l’électricité et le gaz.

Zhang et al. [7] parlent également d’Utility Computing comme le concept représentant *“le modèle d’acquisition de ressources à la demande et chargeant le client de payer sur base de son utilisation plutôt qu’à partir d’un montant forfaitaire”*. Ces auteurs voient dès lors le

Cloud Computing comme une “*réalisation de l’Utility Computing*” qui adopte un schéma de prix basé sur l’utilisation des ressources.

Vaquero et al., dans leur travail [8], nomment “Clouds”, les vastes pools de ressources virtuelles facilement utilisables et accessibles (hardware, plate-formes de développement, services) dans lesquels ces ressources sont exploitées via un modèle pay-per-use et qui peuvent être reconfigurées dynamiquement pour s’ajuster à une utilisation variable et optimale des ressources. Les garanties de service sont maintenues telles qu’indiquées par le fournisseur dans un contrat SLA personnalisé.

Mell et Grance définissent le Cloud Computing en tant que “*modèle facilitant l’ubiquité¹, la facilité (commodité) et l’accès à la demande au réseau pour joindre un pool partagé de ressources informatiques configurables (e.g. réseaux, serveurs, stockage, applications, services) qui peuvent être rapidement provisionnées et libérées après utilisation avec un minimum d’effort de gestion ou d’interactions avec le fournisseur de services (Service Provider)*”. Cette vision de modèle de Cloud Computing se compose de

- caractéristiques essentielles : “on-demand”, “self-service”, “network”, “resource pooling”, “fast elasticity”, et “measured services”;
- modèles de service : Software-as-a-Service (SaaS), Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS);
- modèles de déploiement : Private - Public - Community et Hybrid Cloud.

Une définition de chaque terme sera trouvée dans “*The NIST Definition of Cloud Computing*” [68].

La Figure A.1 représente la vision du NIST pour la définition du Cloud Computing.

La Figure A.2 montre quelques exemples de services du Cloud Computing.

1. Faculté d’être présent en plusieurs lieux à la fois, omniprésence

Visual Model Of NIST Working Definition Of Cloud Computing
<http://www.csrc.nist.gov/groups/SNS/cloud-computing/index.html>

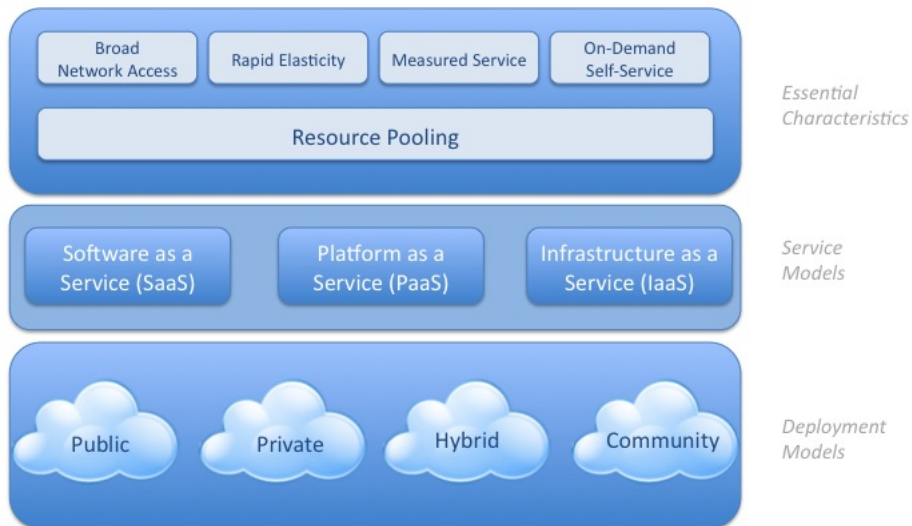


FIGURE A.1 – Définition du Cloud Computing : modèle visuel, NIST

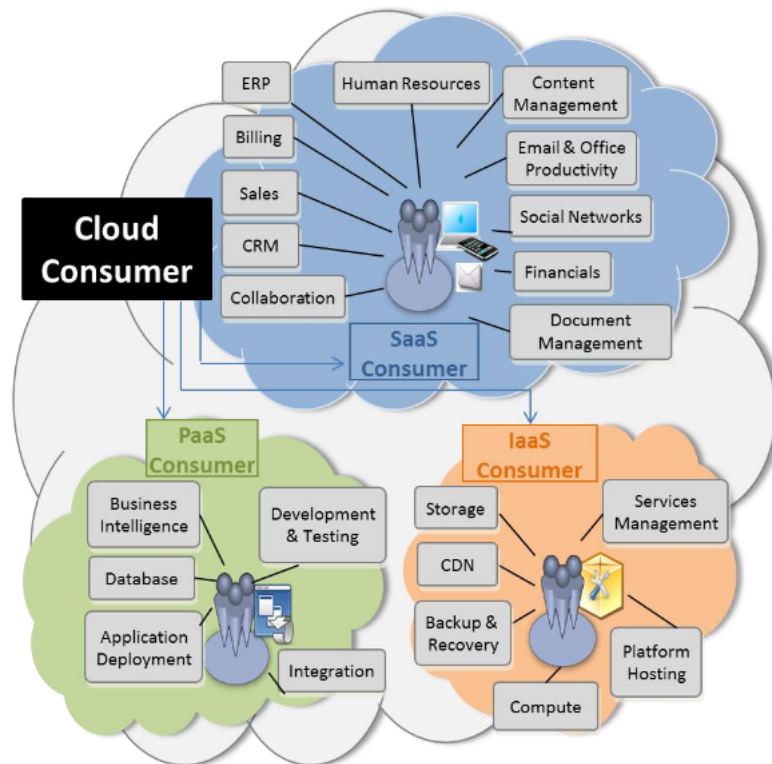


FIGURE A.2 – Services du Cloud Computing, NIST

Annexe B

Revue des définitions pour le concept de Cloud Service Broker

Dans ce qui suit, nous apportons une revue des quelques principales définitions trouvées concernant les Cloud Brokers que nous avons utilisées pour repérer quelles en sont les grandes caractéristiques et pour pouvoir établir notre propre définition du concept.

- Selon les travaux de Gartner réalisés par Plummer et al. [11, 12], un Cloud Service Broker est un rôle IT et un business model dans lequel une entreprise ou toute autre entité qui ajoute de la valeur aux services Cloud pour le compte de consommateurs de ces services. Cette valeur ajoutée peut être réalisée via un rôle primaire ou une combinaison de rôles joués par le Broker : Brokerage par agrégation, intégration ou customisation (personnalisation) de services.

Un Broker est donc un type de fournisseur de services qui joue un rôle d'intermédiaire dans le Cloud Computing et qui rend cet environnement moins cher, plus facile, plus sécurisé et plus productif pour les entreprises afin de naviguer, intégrer, consommer, étendre et maintenir des services Cloud, et ce particulièrement lorsqu'ils couvrent/proviennent de multiples et divers fournisseurs de services Cloud.

- La définition provenant du NIST Reference Architecture [4], nous dit qu'un Cloud Broker est "une entité gérant l'utilisation, les performances et la distribution de services Cloud, ainsi que la négociation des relations entre Cloud Providers et Cloud Consumers/Cloud Users".
- La définition rédigée par la Cloud Security Alliance indique qu'un Cloud Service Broker est "une entité logique qui est responsable d'offrir un service d'intermédiation, de monitoring, de transformation, de portabilité, de gouvernance, de provisionnement et d'intégration de services, de même que la négociation des relations entre différents Cloud Providers et les consommateurs/consumers" [14].
- Forrester précise qu'un CSB est un "business modèle qui exploite les compétences et capacités de tous les trois business modèles traditionnels du logiciel, de la consultance et de l'infrastructure, afin d'offrir une contribution de valeur étendue dans l'espace Cloud émergent".
- Chez Nair et al. [15], nous retrouvons une définition générique. Celle-ci nous explique qu'un Cloud Service Broker crée une plate-forme Cloud de gestion gouvernée et sécurisée dans le but de simplifier la distribution de services Cloud complexes à des clients de

services Cloud.

Les CSB permettent aux utilisateurs (consommateurs) de réaliser/d'acquérir le potentiel maximal qu'un Cloud Provider peut offrir en créant, entre le Cloud Provider et le Consommateur de services Cloud, une plate-forme Cloud de gestion sécurisée, gouvernée et de confiance. Ils aident et assurent également le bon fonctionnement des politiques IT et traitent avec efficacité les contrats SLAs entre Cloud Providers et Cloud Consumers. Pour ces auteurs, le terme CSB est un terme mal défini qui est utilisé aux besoins pour définir différents modèles.

- Kandpal [16] décrit les Cloud Service Brokers comme étant des intermédiaires qui sont positionnés entre un Cloud service Consumer et un ou plusieurs Cloud Service Providers de façon à ce que les consommateurs puissent avoir une seule et unique vue cohésive de tous les services fournis. Les CSB peuvent aussi réaliser la consolidation ou ajouter de la valeur à ces services délivrés par un ou plusieurs Cloud service Providers. Le rôle d'un Broker de service devient critique dans des situations qui requièrent l'intégration de multiples systèmes de manière telle à ce qu'ils dirigent/rendent possible l'interopérabilité entre de multiples solutions Cloud, et donc compense le manque de standards.
- Un Cloud Broker, pour Sundareswaran et al. [17], est un intermédiaire entre utilisateurs (Users) et Cloud Service Providers qui aide les utilisateurs à choisir les services ajustés à leurs besoins et visant à sélectionner les Cloud Service Providers appropriés sur base de leurs exigences. Les auteurs ajoutent qu'un Cloud User peut avoir des besoins ou exigences de services qui ne peuvent pas être remplies par un unique service Provider, et que par conséquent cela requiert une agrégation de plusieurs Service Providers. Le Cloud Broker, qui a un contrat avec les Cloud Service Providers, collecte leurs propriétés (e.g. type de service, coût unitaire, ressources disponibles) et les exigences de service des consommateurs. Alors, le Broker analyse et indexe les Providers selon les similarités de leurs propriétés. Une fois qu'il reçoit une requête d'un Service User pour une sélection de service, le Cloud Broker va chercher dans son index afin d'identifier une liste ordonnée de potentiels Providers candidats sur base de la manière dont ils correspondent avec les exigences. Cette liste forme une base pour les End-Users dans leur décision finale.
- Hitachi [18] considère que les Cloud Service Brokers permettent de donner une valeur de business au Cloud en tant qu'intermédiaires tant pour les consommateurs que pour les fournisseurs. Ils vont aider les consommateurs à gérer la complexité des services Cloud (choix, sélections, accords, qualité, personnalisation, etc.) et aider les fournisseurs à se faire connaître. Les Brokers fournissent des accès simples et faciles au Cloud par différentes solutions sur différentes plateformes, et agrègent le Cloud en un simple "one-stop-shop" pour se procurer et gérer les Cloud Providers adéquats ainsi que leurs services. Un CSB peut faire disparaître la complexité du Cloud et la charge de travail pour les entreprises ainsi que supprimer le besoin d'un expert Cloud pour la sélection et la procurement de services Cloud.

Annexe C

CSMIC Service Measurement Index Framework

Cette annexe présente le framework du SMI pour mesurer et comparer la qualité des services du Cloud Computing.

Service Measurement Index Framework Version 2.1

July 2014



SMI

CSMIC
Carnegie Mellon University Silicon Valley
Moffett Field, CA USA



Introducing the Service Measurement Index (SMI)

The Service Measurement Index (SMI) is a set of business-relevant Key Performance Indicators (KPIs) that provide a standardized method for measuring and comparing a business service regardless of whether that service is internally provided or sourced from an outside company. It is designed to become a standard method to help organizations measure cloud-based business services based on their specific business and technology requirements. The Service Measurement Index is currently being developed by the Cloud Services Measurement Initiative Consortium (CSMIC).

The SMI is a hierarchical framework. The top level divides the measurement space into 7 Categories. Each Category is further refined by 4 or more Attributes. Then within each Attribute a set of KPIs are defined that describe the data to be collected for each measure/metric. Some of these KPIs will be service specific while others will apply to all services (BPaaS, IaaS, PaaS, and SaaS).

Version 2.1 of the CSMIC SMI contains the first 2 levels of this hierarchy, the Categories and Attributes. Work is currently ongoing on the definition of the KPIs and measures related to several attributes. See Figure 1: High-Level Overview to the right for a high-level overview of the SMI. A more detailed view (Figure 2: Detailed View) is provided on the last page of this document. For more information see: csmic.org

The Attribute definitions provided here are all written using these basic terms.

Client

An organization that acquires one or more cloud-based services for its users

Cloud Service Provider (CSP)

An organization that delivers one or more cloud-based services to one or more clients.

Service

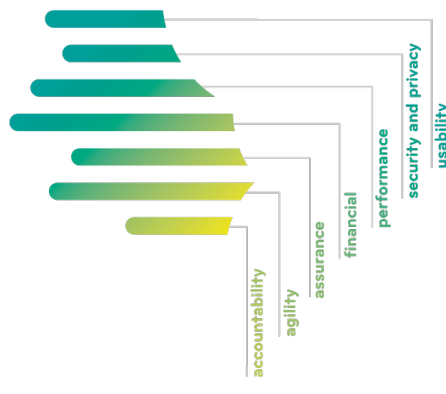
A set of functions or capabilities made available to a group of users by a cloud service provider, in order to achieve client objectives.

SLA (Service Level Agreement)

An agreement between a cloud service provider and a client documenting quantitative service level commitments.

User

The people that interact directly with a service.



Definitions

Accountability

This category contains attributes used to measure the properties related to the cloud service provider organization. These properties may be independent of the service being provided.

Attribute	SMI Attribute Definition
Auditability	The ability of a client to verify that the cloud service provider is adhering to the standards, processes, and policies that they follow.
Compliance	Standards, processes, and policies committed to by the cloud service provider are followed.
Contracting experience	Indicators of client effort and satisfaction with the process of entering into the agreements required to use a service.
Ease of doing business	Client satisfaction with the ability to do business with a cloud service provider.
Governance	The processes used by the cloud service provider to manage client expectations, issues and service performance.
Ownership	The level of rights a client has over client data, software licenses, and intellectual property associated with a service.
Provider business stability	The likelihood that the cloud service provider will continue to exist throughout the contracted term.
Provider Certifications	The cloud service provider maintains current certifications for standards relevant to their clients' requirements.
Provider Contract/SLA Verification	The cloud service provider makes available to clients SLAs adequate to manage the service and mitigate risks of service failure.
Provider Ethicality	Ethicality refers to the manner in which the cloud service provider conducts business; it includes business practices and ethics outside the scope of regulatory compliance. Ethicality includes fair practices with suppliers, clients, and employees.
Provider Personnel Requirements	The extent to which cloud service provider personnel have the skills, experience, education, and certifications required to effectively deliver a service.
Provider Supply Chain	The cloud service provider ensures that any SLAs that must be supported by its suppliers are supported.
Provider Support	The extent to which the cloud service provider includes or makes available assistance to the client in their efforts to use the service, including answering questions about the service and working around or correcting any problems that may arise.
Sustainability	The impact on the economy, society and the environment of the cloud service provider.

Agility
Indicates the impact of a service upon a client's ability to change direction, strategy, or tactics quickly and with minimal disruption.

Attribute	SMI Attribute Definition
Adaptability	The ability of the cloud service provider to adjust to changes in client requirements.
Elasticity	The ability of a cloud service provider to adjust its resource consumption for a service at a rapid enough rate to meet client demand.
Extensibility	The ability to add new features or services to existing services.
Flexibility	The ability to add or remove predefined features from a service.
Portability	The ability of a client to easily move a service from one cloud service provider to another with minimal disruption.
Scalability	The ability of a cloud service provider to increase or decrease the amount of service available to meet client requirements and agreed SLAs.

Assurance

This category includes key attributes that indicate how likely it is that the service will be available as specified.

Attribute	SMI Attribute Definition
Availability	The appropriateness of the service availability window, as well as the likelihood that the availability window will actually be provided to clients.
Maintainability	Maintainability refers to the ability for the cloud service provider to make modifications to the service to keep the service in a condition of good repair.
Recoverability	Recoverability is the degree to which a service is able to quickly resume a normal state of operation after an unplanned disruption.
Reliability	Reliability reflects measures of how a service operates without failure under given conditions during a given time period.
Resiliency/Fault Tolerance	The ability of a service to continue to operate properly in the event of a failure in one or more of its components.
Service stability	The degree to which the service is resistant to change, deterioration, or displacement.
Serviceability	The ease and efficiency of performing maintenance and correcting problems with the service.

Financial

The amount of money spent on the service by the client.

Attribute	SMI Attribute Definition
Billing Process	The level of integration that is available between the client and cloud service provider's billing systems and the predictability of periodic bills.
Cost	The client's cost to consume a service over time. This includes cost of transition of the service along with recurring costs (e.g., monthly access fees) and usage-based costs.
Financial Agility	The flexibility and elasticity of the financial aspects of the CSP's services
Financial Structure	How responsive to the client's needs are the cloud service provider's pricing and billing components.

Performance
This category covers the features and functions of the provided services.

Attribute	SMI Attribute Definition
Accuracy	The extent to which a service adheres to its requirements.
Functionality	The specific features provided by a service.
Suitability	How closely the capabilities of the proposed service match the features needed by the client.
Interoperability	The ability of a service to easily interact with other services (from the same cloud service provider and from other cloud service providers).
Service Response Time	An indicator of the time between when a service is requested and when the response is available.

Security and Privacy

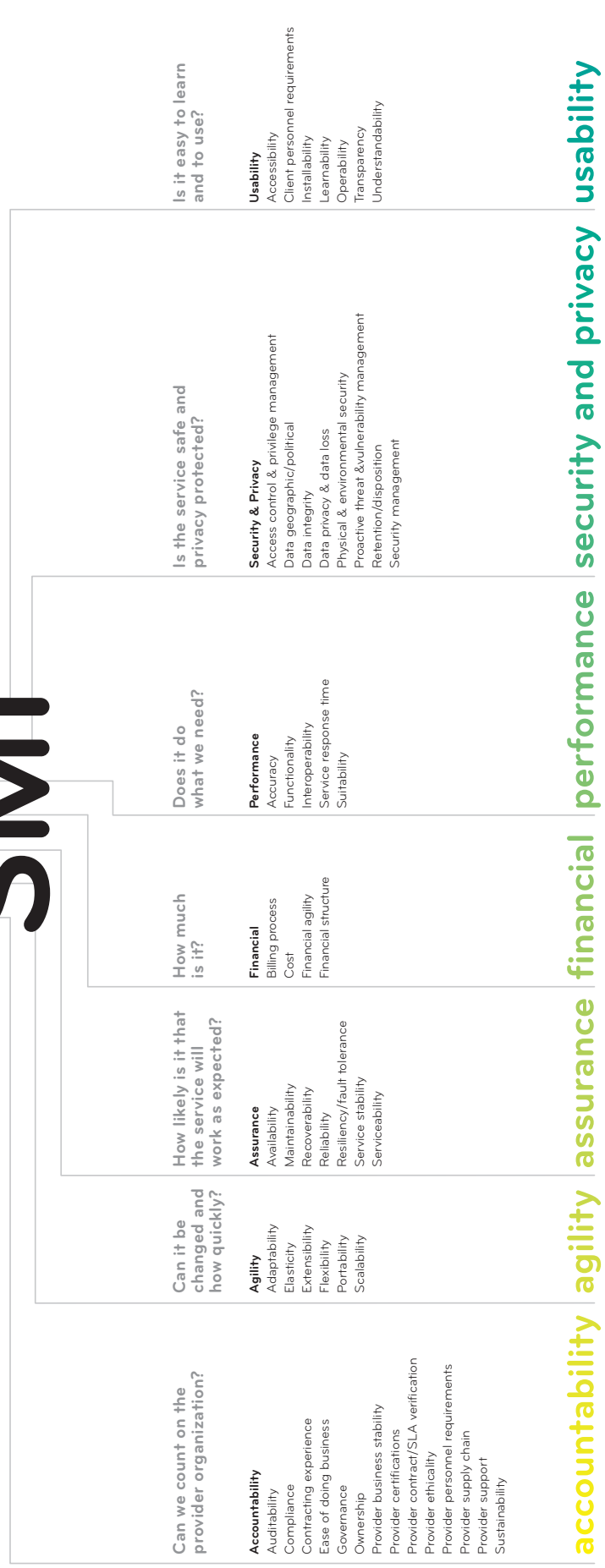
This category includes attributes that indicate the effectiveness of a cloud service provider's controls on access to services, service data, and the physical facilities from which services are provided.

Attribute	SMI Attribute Definition
Access Control & Privilege Management	Policies and processes in use by the cloud service provider to ensure that only the personnel granted appropriate privileges can make use of or modify data/work products.
Data Geographic/Political	The client's constraints on service location based on geographic or political risk.
Data Integrity	Keeping the data that is created, used, and stored in its correct form so that clients may be confident that it is accurate and valid.
Data Privacy & Data Loss	Client restrictions on use and sharing of client data are enforced by the cloud service provider. Any failures of these protections are promptly detected and reported to the client.
Physical & Environmental Security	Policies and processes in use by the cloud service provider to protect the provider facilities from unauthorized physical access, damage or interference.
Proactive Threat & Vulnerability Management	Mechanisms in place to ensure that the service is protected against know recurring threats as well as new evolving vulnerabilities.
Retention/Disposition	The cloud service provider's data retention and disposition processes meet the clients' requirements.
Security Management	The capabilities of cloud service providers to ensure application, data, and infrastructure security based on the security requirements of the client.

Usability

The ease with which a service can be used.

Attribute	SMI Attribute Definition
Accessibility	The degree to which a service is operable by users with disabilities.
Client Personnel Requirements	The minimum number of personnel satisfying roles, skills, experience, education, and certification required of the client to effectively utilize a service.
Installability	Installability characterizes the time and effort required to get a service ready for delivery (where applicable).
Learnability	The effort required of users to learn to use the service.
Operability	The ability of a service to be easily operated by users.
Transparency	The extent to which users are able to determine when changes in a feature or component of the service occur and whether these changes impact usability.
Understandability	The ease with which users can understand the capabilities and operation of the service.



Annexe D

Exemples d'interfaces

Nous présentons ci-dessous deux exemples d'interface afin de donner un aperçu de notre idée d'utilisation des paramètres pour la sélection et la comparaison d'un SaaS (SMI Tool Users Guide¹, Figure D.1), et pour l'évaluation des SaaS (www.cloudsurfing.com, Figure D.2).

1. http://csmic.org/wp-content/uploads/2014/07/SMI-Prototype-Tool-Users-Guide_v147.pdf

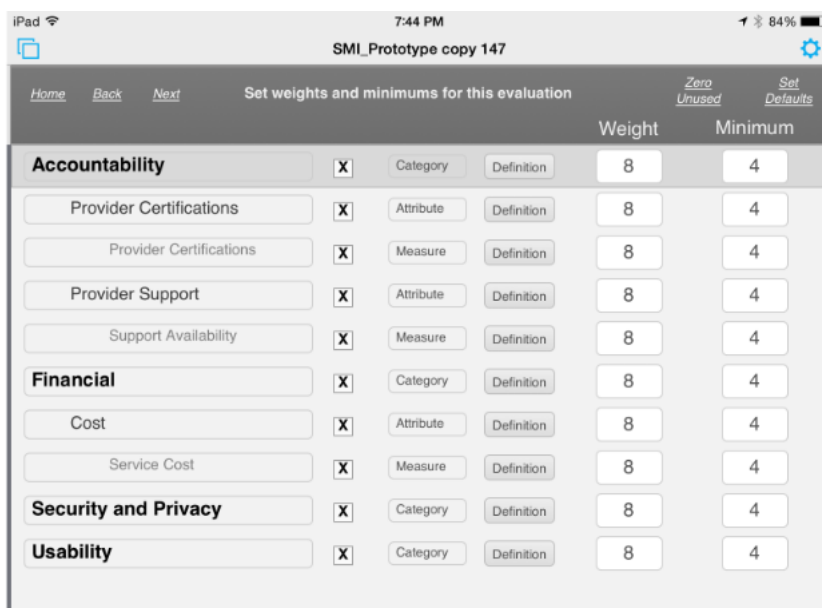


FIGURE D.1 – Interface de l’outil SMI pour choisir les paramètres, leur minimas et leur poids en vue d’une comparaison.

Detailed and Summary Rating

Weight	Categories	Rating	Explanation
20%	Usability	★★★★★	
20%	Popularity	★★★★★	
25%	Value	★★★★★	
10%	Support	★★★★★	
10%	Security	★★★★★	
10%	Integration	★★★★★	
5%	Access	★★★★★	
100%	Overall	★★★★★	Rate This

FIGURE D.2 – Interface pour l’évaluation, CloudSurfing.com .

Table des figures

1.1	Paradigme du Cloud Computing, vision de Berkeley	22
1.2	Schématisation de la définition du Cloud Computing.	28
1.3	Service Brokers dans le Cloud Computing.	31
1.4	Schématisation des rôles de Brokerage des services (Hitachi [19]).	36
1.5	Service Brokers dans le Cloud Computing	38
1.6	Représentation des Fédérations Clouds et des Multi-Cloud	44
1.7	Vue schématisée de l’approche Inter-Cloud	45
1.8	Taxonomie des architectures Inter-Cloud.	45
2.1	Framework d’un CSB pour la sélection de SaaS basée sur le SLA, E. Badidi	51
2.2	Architecture haut-niveau du Broker, Moore & Mahmoud	53
2.3	Écran du formulaire “Add Interface Attribute”, Moore & Mahmoud	53
2.4	Toolkit OPTIMIS	56
2.5	Cloud Brokerage dans OPTIMIS	56
2.6	Architecture de RESERVOIR	59
2.7	mOSAIC – Cloud Agency	61
2.8	STRATOS – Vue architecturale haut-niveau du framework	63
2.9	STRATOS – Fichier TDF	63
2.10	Framework pour la sélection et la recommandation de services Cloud, Yan et al. [55].	69
2.11	Policy Implementation Engine, Yan et al. [55].	69
2.12	CloudCmp : Exemple d’interface de comparaison de performance et de coûts (1), Li et al.[57]	72
2.13	CloudCmp : Exemple d’interface de comparaison de performance et de coûts (2), Li et al. [56]	72
2.14	SMICloud – Interface d’une étude de cas, Garg et al. [9]	75
2.15	Framework SMICloud, Garg et al. [9]	75
2.16	Framework pour la sélection de service basé sur le feedback et l’évaluation de performances quantitatives, Qu et al [59].	78
3.1	SaaS Broker entre Users et Providers.	98
3.2	Environnement du SaaS Broker.	103
3.3	Composants du SaaS Broker.	106
3.4	Modélisation de l’architecture et du fonctionnement du SaaS Broker.	107
A.1	Définition du Cloud Computing : modèle visuel, NIST	117
A.2	Services du Cloud Computing, NIST	117

D.1 Outil SMI	128
D.2 Interface pour l'évaluation, CloudSurfing.com	128

Liste des tableaux

2.1	Tableau comparatif des Brokers (partie 1)	80
2.2	Tableau comparatif des Brokers (partie 2)	81
2.3	Modèles de comparaison et de sélection étudiés	82
3.1	Tableau des paramètres quantitatifs.	94
3.2	Tableau des paramètres qualitatifs.	95
3.3	Tableau des paramètres qualitatifs.	96

Bibliographie

- [1] Armbrust M., Fox A., Griffith R., Joseph AD., Katz R., Konwinski A., Lee G., Patterson D., Rabkin A., Stoica I., et al. Above the clouds : A berkeley view of cloud computing. department of electrical engineering and computer sciences. University of California at Berkeley, Berkeley, CA, 2009.
- [2] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D Joseph, Randy Katz, Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, et al. A view of cloud computing. Communications of the ACM, 53(4) :50–58, 2010.
- [3] Peter Mell and Tim Grance. The nist definition of cloud computing. 2011.
- [4] Fang Liu, Jin Tong, Jian Mao, Robert Bohn, John Messina, Lee Badger, and Dawn Leaf. Nist cloud computing reference architecture. NIST special publication, 500 :292, 2011.
- [5] Michael Hogan, Fang Liu, Annie Sokol, and Jin Tong. Nist cloud computing standards roadmap. NIST Special Publication, 35, 2011.
- [6] Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal, James Broberg, and Ivona Brandic. Cloud computing and emerging it platforms : Vision, hype, and reality for delivering computing as the 5th utility. Future Generation computer systems, 25(6) :599–616, 2009.
- [7] Qi Zhang, Lu Cheng, and Raouf Boutaba. Cloud computing : state-of-the-art and research challenges. Journal of Internet Services and Applications, 1(1) :7–18, 2010.
- [8] Luis M Vaquero, Luis Rodero-Merino, Juan Caceres, and Maik Lindner. A break in the clouds : towards a cloud definition. ACM SIGCOMM Computer Communication Review, 39(1) :50–55, 2008.
- [9] Saurabh Kumar Garg, Steven Versteeg, and Rajkumar Buyya. Smicloud : A framework for comparing and ranking cloud services. In Utility and Cloud Computing (UCC), 2011 Fourth IEEE International Conference on, pages 210–218. IEEE, 2011.
- [10] Daryl Plummer. The business landscape of cloud computing. Gartner Inc., Accessed at <http://www.ft.com/intl/cms/5e231aca-a42b-11e1-a701-00144feabdc0.pdf> on November, 30 :2012, 2012.
- [11] Gartner Inc. Cloud services brokerage (csb). <http://www.gartner.com/it-glossary/cloud-services-brokerage-csb>, consulté le 08/11/2013.
- [12] D Plummer, B Lheureux, and Frances Karamouzis. Defining cloud services brokerage : taking intermediation to the next level. Gartner Research Note, (G00206187), 2010.
- [13] Stefan Ried. Cloud broker-a new business model paradigm. Evolution, 2011.
- [14] Bhumip Khasnabish, Shao Weixiang, and Hu Jie. Cloud service broker. <http://tools.ietf.org/html/draft-shao-opsawg-cloud-service-broker-03>.

- [15] Srijiith K Nair, Sakshi Porwal, Theo Dimitrakos, Ana Juan Ferrer, Johan Tordsson, Tabassum Sharif, Craig Sheridan, Muttukrishnan Rajarajan, and Afnan Ullah Khan. Towards secure cloud bursting, brokerage and aggregation. In Web Services (ECOWS), 2010 IEEE 8th European Conference on, pages 189–196. IEEE, 2010.
- [16] Dolly Kandpal. Role of service brokers in cloud computing. In Cloud Computing, pages 87–105. Springer, 2013.
- [17] Smitha Sundareswaran, Anna Squicciarini, and Dan Lin. A brokerage-based approach for cloud service selection. In Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on, pages 558–565. IEEE, 2012.
- [18] Hitachi Consulting Corporation. Cloud service brokerage model for optimized cloud use, sustainable governance & efficient utilization, April 2012.
- [19] Chad Lawler. An executive view on cloud service brokers, 2013.
- [20] Linlin Wu, Saurabh Kumar Garg, and Rajkumar Buyya. Sla-based resource allocation for software as a service provider (saas) in cloud computing environments. In Cluster, Cloud and Grid Computing (CCGrid), 2011 11th IEEE/ACM International Symposium on, pages 195–204. IEEE, 2011.
- [21] Prashant Khanna and Budida Babu. Cloud computing brokering service : A trust framework. In CLOUD COMPUTING 2012, The Third International Conference on Cloud Computing, GRIDs, and Virtualization, pages 206–212, 2012.
- [22] Linlin Wu and Rajkumar Buyya. Service level agreement (sla) in utility computing systems. arXiv preprint arXiv :1010.2881, 2010.
- [23] Nikolay Grozev and Rajkumar Buyya. Inter-cloud architectures and application brokering : taxonomy and survey. Software : Practice and Experience, 2012.
- [24] Nikolaos Loutas, Eleni Kamateri, Filippo Bosi, and Konstantinos Tarabanis. Cloud computing interoperability : the state of play. In Cloud Computing Technology and Science (CloudCom), 2011 IEEE Third International Conference on, pages 752–757. IEEE, 2011.
- [25] Nikolay Grozev and Rajkumar Buyya. Inter-cloud architectures and application brokering : taxonomy and survey. Software : Practice and Experience, 2012.
- [26] Yuri Demchenko, Canh Ngo, Marc Makkes, Rudolf Stgrijkers, and Cees de Laat. Defining inter-cloud architecture for interoperability and integration. In CLOUD COMPUTING 2012, The Third International Conference on Cloud Computing, GRIDs, and Virtualization, pages 174–180, 2012.
- [27] Christos Tranoris and Spyros Denazis. Federation computing : A pragmatic approach for the future internet. In Network and Service Management (CNSM), 2010 International Conference on, pages 190–197. IEEE, 2010.
- [28] David Villegas, Norman Bobroff, Ivan Rodero, Javier Delgado, Yanbin Liu, Aditya Devarakonda, Liana Fong, S Masoud Sadjadi, and Manish Parashar. Cloud federation in a layered service model. Journal of Computer and System Sciences, 78(5) :1330–1344, 2012.
- [29] Opencloudmanifesto. Cloud computing use cases white paper. <http://www.opencloudmanifesto.org/>.

- [30] Erbin Lim and Philippe Thiran. Communication of technical qos among cloud brokers. In Proceedings of the 2014 IEEE International Conference on Cloud Engineering, pages 403–409. IEEE Computer Society, 2014.
- [31] Massimo Coppola, Patrizio Dazzi, Aliaksandr Lazouski, Fabio Martinelli, Paolo Mori, Jens Jensen, Ian Johnson, and Philip Kershaw. The contrail approach to cloud federations. In Proceedings of the International Symposium on Grids and Clouds (ISGC'12), 2012.
- [32] Elarbi Badidi. A framework for software-as-a-service selection and provisioning. arXiv preprint arXiv :1306.1888, 2013.
- [33] Elarbi Badidi. A cloud service broker for sla-based saas provisioning. pages 61–66, 2013.
- [34] Shangguang Wang, Zibin Zheng, Qibo Sun, Hua Zou, and Fangchun Yang. Cloud model for service selection. In Computer Communications Workshops (INFOCOM WKSHPs), 2011 IEEE Conference on, pages 666–671. IEEE, 2011.
- [35] Ang Li, Xiaowei Yang, Srikanth Kandula, and Ming Zhang. Comparing public-cloud providers. Internet Computing, IEEE, 15(2) :50–53, 2011.
- [36] Brian Moore and Qusay H Mahmoud. A service broker and business model for saas applications. In Computer Systems and Applications, 2009. AICCSA 2009. IEEE/ACS International Conference on, pages 322–329. IEEE, 2009.
- [37] Optimis. Optimis - eu fp7 project. www.optimis-project.eu. consulted on December 2013.
- [38] Srijith Nair et al. Optimis white paper - why use optimis? 2013. <http://www.optimis-project.eu/sites/default/files/content-files/document/optimis-white-paper-may-2013.pdf>.
- [39] Ana Juan Ferrer, Francisco Hernández, Johan Tordsson, Erik Elmroth, Ahmed Ali-Eldin, Csilla Zsigri, Raül Sirvent, Jordi Guitart, Rosa M Badia, Karim Djemame, et al. Optimis : A holistic approach to cloud service provisioning. Future Generation Computer Systems, 28(1) :66–77, 2012.
- [40] Wolfgang Theilmann, Ramin Yahyapour, and Joe Butler. Multi-level sla management for service-oriented infrastructures. In Towards a Service-Based Internet, pages 324–335. Springer, 2008.
- [41] Wolfgang Theilmann, Jens Happe, Costas Kotsokalis, Andy Edmonds, Keven Kearney, and J Lambea. A reference architecture for multi-level sla management. Journal of Internet Engineering, 4(1), 2010.
- [42] Thijs Metsch, Andy Edmonds, and Victor Bayon. Resrvoir - sla@soi : Using cloud standards for interoperability of cloud frameworks.
- [43] Reservoir. Reservoir - eu fp7 project. <http://www.reservoir-fp7.eu/>, consulté en décembre 2013.
- [44] Benny Rochwerger, David Breitgand, Eliezer Levy, Alex Galis, Kenneth Nagin, Ignacio Martín Llorente, Rubén Montero, Yaron Wolfsthal, Erik Elmroth, Juan Caceres, et al. The reservoir model and architecture for open federated cloud computing. IBM Journal of Research and Development, 53(4) :4–1, 2009.
- [45] Benny Rochwerger, David Breitgand, Amir Epstein, David Hadas, Irit Loy, Kenneth Nagin, Johan Tordsson, Carmelo Ragusa, Massimo Villari, Stuart Clayman, et al. Reservoir-when one cloud is not enough. Computer, 44(3) :44–51, 2011.

- [46] OpenNebula.org. Opennebula.org. www.opennebula.org, consulté en décembre 2013.
- [47] Francesco Moscato, Rocco Aversa, Beniamino Di Martino, T Fortis, and Victor Munteanu. An analysis of mosaic ontology for cloud resources annotation. In Computer Science and Information Systems (FedCSIS), 2011 Federated Conference on, pages 973–980. IEEE, 2011.
- [48] Alba Amato, Beniamino Di Martino, and Salvatore Venticinque. Evaluation and brokering of service level agreements for negotiation of cloud infrastructures. In Internet Technology And Secured Transactions, 2012 International Conference For, pages 144–149. IEEE, 2012.
- [49] Salvatore Venticinque, Rocco Aversa, Beniamino Di Martino, Massimiliano Rak, and Dana Petcu. A cloud agency for sla negotiation and management. In Euro-Par 2010 Parallel Processing Workshops, pages 587–594. Springer, 2011.
- [50] Przemyslaw Pawluk, Bradley Simmons, Michael Smit, Marin Litoiu, and Serge Mankovski. Introducing stratos : A cloud broker service. In Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on, pages 891–898. IEEE, 2012.
- [51] Rajkumar Buyya, Rajiv Ranjan, and Rodrigo N Calheiros. Intercloud : Utility-oriented federation of cloud computing environments for scaling of application services. In Algorithms and architectures for parallel processing, pages 13–31. Springer, 2010.
- [52] Contrail. Contrail website. www.contrail-project.eu. consulted on july 2014.
- [53] Attila Csaba Marosi, Gabor Kecskemeti, Attila Kertesz, and Peter Kacsuk. Fcm : an architecture for integrating iaas cloud systems. In CLOUD COMPUTING 2011, The Second International Conference on Cloud Computing, GRIDs, and Virtualization, pages 7–12, 2011.
- [54] Ines Houidi, Marouen Mechtri, Wajdi Louati, and Djamel Zeghlache. Cloud service delivery across multiple cloud platforms. In Services Computing (SCC), 2011 IEEE International Conference on, pages 741–742. IEEE, 2011.
- [55] Shixing Yan, Chunqing Chen, Guopeng Zhao, and Bu Sung Lee. Cloud service recommendation and selection for enterprises. In Network and Service Management (CNSM), 2012 8th International Conference on, pages 430–434. IEEE, 2012.
- [56] Ang Li, Xiaowei Yang, Srikanth Kandula, and Ming Zhang. Cloudcmp : comparing public cloud providers. In Proceedings of the 10th ACM SIGCOMM conference on Internet measurement, pages 1–14. ACM, 2010.
- [57] Ang Li, Xiaowei Yang, Srikanth Kandula, and Ming Zhang. Cloudcmp : shopping for a cloud made easy.
- [58] Gueyoung Jung, Tridib Mukherjee, Shruti Kunde, Hyunjoo Kim, Naveen Sharma, and Frank Goetz. Cloudadvisor : A recommendation-as-a-service platform for cloud configuration and pricing. In Services (SERVICES), 2013 IEEE Ninth World Congress on, pages 456–463. IEEE, 2013.
- [59] Lie Qu, Yan Wang, and Mehmet A Orgun. Cloud service selection based on the aggregation of user feedback and quantitative performance assessment. In Services Computing (SCC), 2013 IEEE International Conference on, pages 152–159. IEEE, 2013.
- [60] Shuo-Yan Chou, Yao-Hui Chang, and Chun-Ying Shen. A fuzzy simple additive weighting system under group decision-making for facility location selection with

- objective/subjective attributes. European Journal of Operational Research, 189(1) :132–145, 2008.
- [61] Hamdi Yahyaoui, Zakaria Maamar, Jamal Bentahar, Nabil Sahli, Said Elnaffar, and Philippe Thiran. On the reputation of communities of web services. In Proceedings of the 8th international conference on New technologies in distributed systems, page 4. ACM, 2008.
- [62] Manish Godse and Shrikant Mulik. An approach for selecting software-as-a-service (saas) product. In Cloud Computing, 2009. CLOUD'09. IEEE International Conference on, pages 155–158. IEEE, 2009.
- [63] Jae Yoo Lee, Jung Woo Lee, Soo Dong Kim, et al. A quality model for evaluating software-as-a-service in cloud computing. In Software Engineering Research, Management and Applications, 2009. SERA'09. 7th ACIS International Conference on, pages 261–266. IEEE, 2009.
- [64] Ahmad Rashidi and Naser Movahhedinia. A model for user trust in cloud computing. International Journal on Cloud Computing : Services and Architecture (IJCCSA), 2(2) :1–8, 2012.
- [65] David Ameller and Xavier Franch. Service level agreement monitor (salmon). In Composition-Based Software Systems, 2008. ICCBSS 2008. Seventh International Conference on, pages 224–227. IEEE, 2008.
- [66] Rajkumar Buyya, Chee Shin Yeo, and Srikumar Venugopal. Market-oriented cloud computing : Vision, hype, and reality for delivering it services as computing utilities. In High Performance Computing and Communications, 2008. HPCC'08. 10th IEEE International Conference on, pages 5–13. Ieee, 2008.
- [67] Rajkumar Buyya, James Broberg, and Andrzej M Goscinski. Cloud computing principles and paradigms. 2011.
- [68] Peter Mell and Tim Grance. The nist definition of cloud computing, 2009. <http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf>, 2012.

