Contributions à la recherche d'information dans des systèmes distribués, ouverts, intégrant des participants autonomes

Philippe Lamarre

École Centrale de Nantes Université de Nantes École des Mines de Nantes

ÉCOLE DOCTORALE STIM

« SCIENCES ET TECHNOLOGIES DE L'INFORMATION ET DES MATÉRIAUX »

Année 2010

Contributions à la recherche d'information dans des systèmes distribués, ouverts, intégrant des participants autonomes

Rapport scientifique

pour l'obtention de l'

HABILITATION À DIRIGER LES RECHERCHES EN INFORMATIQUE

Philippe LAMARRE

le 27 novembre 2009

au LINA

devant le jury ci-dessous

Président :	Michel RAYNAL, Professeur	Université Rennes 1
Rapporteurs :	Bernd AMANN, Professeur	Université Paris 6
	Gabriella PASI, Professore Associato Confermato	Università degli Studidi Milano Bicocca
	Christophe SIBERTIN-BLANC, Professeur	Université Toulouse 1
Examinateurs:	Marc GELGON, Professeur	Université de Nantes
	Pascale KUNTZ-COSPEREC, Professeur	Université de Nantes
	Patrick VALDURIEZ, Directeur de Recherche	INRIA Sophia Antipolis - Méditerranée

Directeur de recherche : Patrick VALDURIEZ

Laboratoire : LINA (Laboratoire d'Informatique de Nantes Atlantique)

CONTRIBUTIONS À LA RECHERCHE D'INFORMATION DANS DES SYSTÈMES DISTRIBUÉS, OUVERTS, INTÉGRANT DES PARTICIPANTS AUTONOMES

Philippe LAMARRE

favet neptunus eunti

Philippe LAMARRE

Contributions à la recherche d'information dans des systèmes distribués, ouverts, intégrant des participants autonomes

xii+216 p.

Ce document a été préparé avec LaTeX2$_\varepsilon$ et à partir de la classe `these-IRIN` version 0.92 de l'association de jeunes chercheurs en informatique LOGIN, Université de Nantes. La classe `these-IRIN` est disponible à l'adresse :

`http://login.lina.sciences.univ-nantes.fr/`

Impression : HDR_PLamarre.tex – 29/1/2010 – 15:27

Révision pour la classe : $Id: HDR-IRIN.cls,v 1.3 2009-09-27 09:41:25 lamarre-p Exp

Remerciements

Tant au point de vue scientifique qu'humain, la rédaction de ce rapport a été l'occasion de prendre du recul sur les années passées. J'en retire un vif sentiment de gratitude envers ceux, nombreux, dont le rôle a été prépondérant tout au long de ce chemin.

Je ne peux maleuresement lister ici tous les membres des équipes dont j'ai été membre, mais je les remercie chaleureusement pour la qualité des échanges qui m'ont toujours été profitables. Je ne citerai ici que les responsables de ces équipes : Esther Pacitti, Patrick Valduriez, Henry Briand, Jean-François Nicaud et Michaël Griffith ; et les directeurs de laboratoire : Pierre Cointe, Frédéric Benhamou, Jean-François Nicaud et Michaël Griffith. Ils m'ont fait confiance, accompagné et supporté. En particulier, Patrick m'a indiscutablement beaucoup apporté durant ces dernière années. Sa vision scientifique, son écoute, son ouverture et sa capacité à créer un environnement de travail agréable et efficace ont été autant de chances dont j'ai bénéficié.

Je remercie vivement les membres du jury qui m'ont fait l'honneur et le plaisir de prendre sur leur temps pour juger mon travail : les rapporteurs Bernd Amann, Gabriella Pasi et Christophe Sibertin-Blanc, et les examinateurs Marc Gelgon, Pascale Kuntz-Cosperec, Michel Raynal et Patrick Valduriez.

Je remercie très particulièrement Sylvie Cazalens pour ses contributions, ainsi que Christine Jacquin et Emmanuel Desmontils avec lesquels j'ai eu beaucoup de plaisir à travailler lors de notre reconversion thématique qui n'était pas évidente et que nous avons résssi à mener à bien tous les quatre ensemble.

Mes remerciements vont à Sandra Lemp, Jorge-Arnulfo Quiané-Ruiz et Anthony Ventresque dont l'investissement, en particulier durant leur doctorat, a été prépondérant pour l'obtention des résultats présentés ici. Je n'oublie pas tous les étudiants de maîtrise, DEA, DESS, Master première et deuxième années qui ont activement contribué à nos travaux.

Sans pouvoir les nommer tous, je remercie les collègues du département d'Informatique et du LINA avec lesquels j'ai toujours plaisir à échanger et collaborer tant en recherche qu'en

enseignement. Un grand merci aussi aux personnels techniques et administratifs qui font tout pour rendre possible ce qui ne le semble pas toujours.

Je n’oublie pas les enseignants, du primaire à l’université, grâce à la formation et au dévouement desquels j’ai pu me développer et m’épanouir. Je tiens à remercier encore Luis Fariñas del Cerro et Yoav Shoam qui m’ont initié, formé et fait aimer la recherche.

Enfin, les mots me manquent pour exprimer tout ce que je dois aux membres de ma famille, cette source permanente de soutien inconditionnel et de réconfort. Sans vous, rien ne serait possible.

Sommaire

Préambule

L'objectif de ces lignes est de situer, par rapport au déroulement de ma carrière, les travaux présentés dans ce document, consécutifs à une conversion thématique.

En 1992, j'obtiens ma thèse intitulée "*Étude des raisonnements non monotones : apports des logiques des conditionnels et des logiques modales*" sous la direction de Luis Fariñas del Cerro à l'Université Paul Sabatier de Toulouse. Le cadre général de ces travaux est la représentation des connaissances et le raisonnement. Les principaux apports de cette thèse sont : une étude parallèle des formalismes non-monotones, des logiques des conditionnels et des logiques modales [Lam91, CL92]; un démonstrateur automatique pour une classe de logiques des conditionnels [Lam92a, Lam92b].

En janvier 1993, je rejoins l'équipe de Yoav Shoham à l'Université de Stanford pour travailler sur les notions de connaissance et de conditionnels. Mon principal résultat a été de proposer un formalisme logique regroupant les notions connaissance, certitude et croyance.

En septembre 1993, je prends mon poste de Maître de Conférences à l'Université de Nantes dans l'équipe d'*Intelligence Artificielle* dirigée par Michaël Griffiths. Je continue mes recherches sur la représentation des connaissances [LS94] mais je participe aussi très activement aux activités liées au contexte : création du département d'informatique, création du laboratoire. Je prends aussi la responsabilité de la création du site Internet de Faculté des Sciences et des Techniques, premier site Internet de l'Université de Nantes (de l'information des collègues des différentes disciplines à la charte graphique en passant par la mise en œuvre), et j'assume pendant trois ans la responsabilité du Centre Informatique d'Enseignement (CIE) de la Faculté des Sciences et des Techniques de Nantes (gestion financière, direction des personnels, stratégie de développement, dossiers de financement, etc).

En 1999, avec trois collègues, nous amorçons une conversion thématique vers les concepts et techniques pour la recherche d'information et de documents distribués sur Internet. Le groupe de travail est composé de quatre permanents : Sylvie Cazalens (thème d'origine : représentation

des connaissances), Emmanuel Desmontils (thème d'origine : synthèse d'image), Christine Jacquin (thème d'origine : traitement automatique du langage naturel), et moi-même. J'en assure la responsabilité sous la direction de Jean-François Nicaud alors responsable de l'équipe Intelligence Artificielle. Le groupe mène des travaux dans deux directions : l'indexation sémantique des documents et la recherche de documents en environnement distribué ouvert.

En 2002, suite à une réorganisation du laboratoire l'équipe Intelligence Artificielle disparaît. Notre groupe de travail intègre l'équipe *CID (Connaissance, Informations, Données)*, dirigée par Henri Briand. Avec Francky Trichet et Michel Leclère nous y composons la thématique "*Ingénierie des Connaissances*".

En 2004, une nouvelle réorganisation du laboratoire conduit à scinder le groupe de travail. Christine Jacquin et Emmanuel Desmontils qui ont principalement travaillé sur l'indexation sémantique rejoignent l'équipe TALN (Traitement Automatique du Langage Naturel) sous la direction de Béatrice Daille alors que Sylvie Cazalens et moi-même dont les travaux sont centrés sur la recherche d'information en environnement distribué ouvert rejoignons l'équipe *ATLAS-GDD (Gestion de Données Distribuées)* sous la direction de Patrick Valduriez. J'ai par la suite été intégré à *l'Equipe Projet INRIA ATLAS* où je suis actuellement en délégation INRIA.

Les travaux concernant la représentation des connaissances et le raisonnement sont absents de ce document. Nous présentons ici ceux qui ont été menés depuis notre conversion thématique des années 1999-2000.

Ces travaux ont été menés en équipe et les résultats présentés résultent de diverses collaborations. Ce document est donc rédigé à la première personne du pluriel.

CHAPITRE 1

Introduction

Les deux dernières décennies ont vu naître et évoluer Internet. Les échanges d'informations qu'il permet ont un impact qui s'étend bien au delà du monde scientifique et atteint l'industrie, le commerce, l'économie, l'enseignement, l'administration, les particuliers, etc. Le mouvement est toujours en marche, en passe de s'étendre aux objets du quotidien via l'Internet des objets [WOO *60*]. Cette révolution a des conséquences multiples et ouvre de nouvelles problématiques scientifiques et technologiques mais aussi sociétales telles que la protection de la vie privée, la neutralité des opérateurs, la liberté et la capacité d'accès aux ressources, ou encore la gouvernance.

Le volume d'information présent sur Internet, ainsi que son organisation rendent indispensable l'utilisation d'outils de recherche. Ils sont donc au cœur d'enjeux économiques, politiques et sociaux particulièrement importants. Sans toujours pouvoir répondre aux interrogations qui en découlent, il est important de garder présent à l'esprit les enjeux liés aux problématiques scientifiques que nous abordons.

C'est dans cette optique que nous nous sommes intéressés à la recherche d'information sur Internet en voulant proposer une alternative aux approches centralisées des moteurs de recherche. Les spécificités d'Internet introduisent bien des problématiques. Ainsi l'obstacle du passage à l'échelle est d'autant plus difficile à franchir qu'il concerne simultanément le nombre de documents (volume de données à traiter), de sources (très nombreuses et très largement réparties) et de requêtes (extrêmement élevé). Fiabilité et performances constituent des points particulièrement difficiles qui justifient de lourds investissements des acteurs du domaine. Un point qui nous semble mériter plus d'attention est le degré d'autonomie accordé aux participants. Ces derniers, personnes physiques ou morales, décident par eux-mêmes de contribuer au système en fonction des objectifs qu'ils poursuivent individuellement tels que diffusion d'infor-

mations, publicité, obtention de marchés, etc. Ce ne sont pas forcément des bénévoles altruistes. Ils utilisent leurs propres moyens pour arriver à leurs fins (serveurs, ressources humaines pour la production, mise en forme et publication des informations, etc). Or, les moteurs de recherche, seuls outils disponibles pour se rendre visible, ne proposent aucune autonomie à ces participants. En effet, sans produire aucune information par eux-mêmes, ils les collectent sur les sites des participants et les classent pour les proposer en réponses aux requêtes des utilisateurs. Ainsi, les participants n'ont pas d'autre choix que d'autoriser ou d'interdire l'indexation de leur site et n'ont que très peu de maîtrise sur la manière dont les moteurs diffusent leurs informations. Or, les propriétaires de sites déploient des efforts considérables pour palier cet état de fait, en particulier pour améliorer le rang de leurs pages pour certaines requêtes. Le récent succès du service AdWords [WAW39] de Google qui exploite le besoin des fournisseurs de préciser individuellement les requêtes qui les intéressent confirme aujourd'hui le besoin d'un degré d'autonomie plus important au sein du système.

L'intégration de participants autonomes dans un système est naturellement liée à la propriété "d'ouverture" des systèmes. Dans les années 1980, un système était qualifié "d'ouvert" lorsqu'il permettait le développement et l'installation de logiciels tiers. De tels systèmes se caractérisaient alors par un souci de portabilité et d'interopérabilité entre les applications. Cela a conduit à la mise en place de standards logiciels permettant une intégration plus aisée. Dans le même temps, le terme ouvert a aussi été utilisé pour caractériser les systèmes autorisant les utilisateurs à se connecter de manière anonyme. Un système peut donc être ouvert en direction des logiciels ou des utilisateurs. Un système distribué est maintenant qualifié d'ouvert lorsqu'il autorise l'intégration et la disparition à la volée de nouveaux composants/services. Dans le cadre d'un système largement distribué, ces composants pouvant appartenir à différents propriétaires, les deux facettes de l'ouverture sont présentes. Pour être efficace, un système distribué ouvert doit donc minimiser l'impact lié à l'intégration ou au départ d'un participant. Cette propriété et celle du passage à l'échelle sont certainement à la base (même si ce ne sont pas les seules raisons) de l'entrée remarquée tant dans le monde académique qu'industriel des systèmes pair-à-pair. Ces systèmes partagent avec les grilles de calcul ou de données, certains systèmes multi-agents, etc, la caractéristique d'être ouverts, et construits à partir de moyens apportés par des participants

autonomes.

Notre intérêt pour la recherche d'information sur Internet nous a donc conduit à considérer des systèmes qui sont non seulement largement distribués, mais qui présentent aussi la particularité d'intégrer des participants que l'on peut qualifier d'autonomes. Cette caractéristique est partagée par d'autres applications. Leur diversité nous conduit à peu préciser le terme "autonomie" de sorte qu'il puisse être décliné de différentes manières. Cependant deux dimensions peuvent être distinguées : l'autonomie vis-à-vis du système et l'autonomie à l'intérieur du système.

Un participant est autonome vis-à-vis d'un système lorsque qu'il peut choisir d'y participer ou non. Cette autonomie n'est pas maîtrisée par le système. Si elle existe, elle est imposée par le contexte et le système ne peut que l'acter et s'y adapter. D'un point de vue technique, cela se traduit par la nécessité de prendre en compte les intégrations et les départs volontaires des participants. Cependant, ce point est loin de couvrir tous les aspects de la situation. S'intégrer à un système nécessite des efforts quelquefois importants qu'il faut justifier. Qu'une personne rationnelle et autonome prenne une telle décision ne s'explique que si elle considère que cela la rapproche de ses objectifs. Le départ volontaire d'un participant d'un système s'explique de la même manière. Pour être précis, il faudrait ici distinguer la personne (physique ou morale) de l'artefact informatique. Les objectifs sont des attributs naturels d'une personne et nous ne les attribuons à l'artefact que dans la mesure où il la représente dans le système. En partant de l'hypothèse que ses participants l'intègrent volontairement pour lui apporter leurs ressources, un système distribué ouvert intègre donc des participants autonomes vis-à-vis de lui. Par là même, il se condamne à devoir satisfaire les objectifs de ses participants. En cas d'échec, la sanction sera de constater leur départ. L'hétérogénéité des participants constitue un élément de difficulté pour ce problème. L'hétérogénéité peut en effet se concrétiser sur les représentations des informations qu'ils utilisent, mais aussi sur leurs objectifs et leurs intérêts. Différents participants peuvent donc manifester des intérêts différents. Dans certains cas, une homogénéité des objectifs, peut-être plus simple à gérer, est certainement possible, mais nous la croyons très peu probable dans un cadre général. Pourtant, lorsqu'ils considèrent implicitement les objectifs, bien des systèmes actuels les supposent homogènes. Ainsi, un initiateur de requête est supposé

avoir uniquement des intérêts fixes et précis, comme “les réponses doivent être correctes et complètes”, “le temps de réponse doit être le plus court possible”. Les intérêts des fournisseurs ne sont quant à eux pas pris en compte ou très rarement. En effet, les autres critères utilisés dans les systèmes distribués sont relatifs au fonctionnement du système et n’impactent sur les intérêts des fournisseurs qu’indirectement (ex. répartition de charge). Ces politiques introduites dans les systèmes distribués n’ont de toute façon pas pour objectif de prendre en compte des intérêts individuels.

La deuxième dimension est relative à la marge de manœuvre laissée à un participant à l’intérieur du système. En effet, un système peut dicter ce qu’il doit faire à chaque participant, ou au contraire, lui laisser assez de liberté pour adapter son comportement et agir en fonction ses intérêts et de sa stratégie propre. Différents degrés d’autonomie sont donc envisageables à l’intérieur d’un même système, et deux systèmes fonctionnellement équivalents peuvent faire des choix différents à ce niveau. Intuitivement, trop d’autonomie peut engendrer un système complexe à comportement chaotique. Au contraire une autonomie trop faible, ne laissant pas assez de place aux intérêts individuels des participants, peut les conduire à quitter un système dont ils ne sont pas captifs. Trouver le bon équilibre est un enjeu d’autant plus important que les participants sont autonomes vis-à-vis du système. S’il souhaite attirer des participants et les conserver, il semble donc important pour un système ouvert qu’il trouve le bon équilibre d’autonomie de ses participants pour qu’ils puissent répondre à leurs besoins.

Les deux dimensions de l’autonomie présentées sont relativement indépendantes. Un participant autonome vis-à-vis d’un système peut très bien se satisfaire d’un rôle “d’esclave” à l’intérieur de ce même système (i.e. n’avoir aucune autonomie à l’intérieur). Bien que pouvant sembler paradoxale, cette situation est acceptable si le système impose un rôle en adéquation avec les intérêts individuels du participant. Cela suppose de la part du système une connaissance approfondie des intérêts de chacun que seuls les participants eux-mêmes sont à même de préciser.

La notion d’autonomie offre aussi une perspective intéressante pour observer les solutions apportées à certains problèmes connus. Cela peut s’avérer intéressant, en particulier pour ceux présents dans les systèmes distribués ouverts. L’hétérogénéité des participants fait partie de

ceux-là. Une solution normative à ce problème consiste à imposer une représentation du domaine d'application dont l'adoption par un participant est un pré-requis à son intégration dans le système. Cette approche fait entièrement supporter son coût à un participant lors de son intégration et impacte fortement sur son fonctionnement sans offrir aucune autonomie. Une autre approche consiste à permettre à chacun de conserver sa propre représentation du domaine et de la faire évoluer comme bon lui semble. Les échanges d'informations seront certainement plus coûteux car nécessitant plus de ressources qu'avec une représentation unique. Cependant, ce coût est plus supportable par les participants car mieux réparti dans le temps. Surtout, chacun conserve une autonomie qui peut être mise à profit pour investir sur la représentation utilisée et en tirer un avantage concurrentiel. Si une représentation commune doit apparaître, elle émergera naturellement suite aux interactions entre les participants. Lorsque cette autonomie porte sur la représentation sémantique, nous parlons d'autonomie sémantique.

Pour résumer, notre intérêt pour la recherche d'information sur Internet nous conduit à considérer des systèmes largement distribués, ouverts, intégrant des participants autonomes. Prenant acte du besoin d'autonomie des participants, nous souhaitons concevoir des systèmes qui la prennent en compte et s'y adaptent. Plus précisément, il nous semble important qu'un système soit régulé en tenant compte des objectifs individuels et que chaque participant puisse conserver la maîtrise de la sémantique qu'il utilise. Enfin, notre démarche se veut à la fois théorique et pratique. En effet, toutes nos propositions sont programmées pour obtenir une validation expérimentale que ce soit dans le cadre d'un simulateur ou d'un prototype.

Nous avons été confrontés à de nombreux problèmes auxquels nous avons dû apporter des réponses concrètes : organisation permettant un passage à l'échelle, algorithmes de routage des requêtes et des réponses, allocation des requêtes, représentation des données, des participants et des groupes de participants, hétérogénéité des schémas de données ou hétérogénéité sémantique, outils de développement distribués, de programmation et de simulation. Ces travaux sont donc au croisement de plusieurs domaines. Le domaine des Bases de Données en s'intéressant à la représentation et au stockage de données structurées ou semi-structurées ainsi qu'aux langages d'interrogation associés et à la mise en oeuvre efficace de ces techniques a naturellement été confronté au problème de la distribution ainsi qu'à celui de l'hétérogénéité de la représen-

tation des données (schémas) [ÖV99, ÖV04]. Le domaine des Systèmes Distribués [TvS06] propose de relier des machines en réseau pour obtenir un système qui du point de vue de l'utilisateur se comporte comme une seule et même entité. Un système distribué ouvert doit en plus tenir compte des objectifs individuels de ses participants. Les Systèmes Multi-Agents [BD01] proposent de résoudre bien des problèmes grâce à l'autonomie des agents qui est considérée dans ce cadre comme un support permettant l'émergence de solutions. Le domaine de la Recherche d'Information s'intéresse plus particulièrement à la représentation des documents non structurés et des requêtes, mais aussi à l'étude de l'efficacité de ces méthodes et à la production de corpus de tests pour des évaluations centralisées [BYRN99]. De nombreux travaux sont menés dans le Web Sémantique pour la modélisation des connaissances, mais aussi sur l'hétérogénéité sémantique et en particulier l'alignement d'ontologies [ES07]. Enfin des domaines plus orientés applications tels que les objets distribués, les intergiciels et les services web offrent des solutions pratiques pour la programmation de nos prototypes et simulateurs.

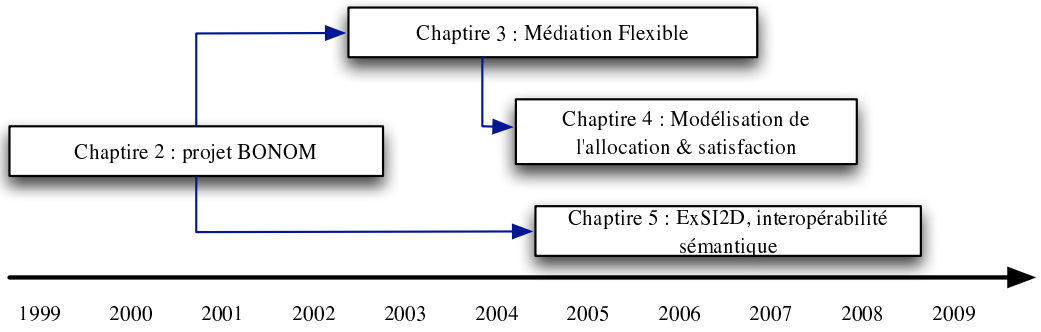


Figure 1.1 – Liens entre les travaux présentés dans les différents chapitres

La suite de ce document est organisée en quatre chapitres concernant des résultats dont les liens sont représentés figure 1.1. Le chapitre 2 présente le projet *BONOM* avec les premiers travaux que nous avons menés dans ce domaine et qui est à l'origine des développements suivants. Plus précisément, nous présentons ici l'organisation proposée pour permettre un passage à l'échelle du système tout en prenant en compte les intérêts éventuellement divergeant des différents participants. Le chapitre 3 présente la *médiation flexible*, une solution au problème de l'allocation de requête dans le cadre d'un système distribué ouvert intégrant des participants autonomes. En tenant compte des intérêts des uns et des autres, la médiation flexible cherche à

maintenir un équilibre entre les intérêts des différents participants. Cette première proposition nous a permis de mieux appréhender le problème qui est revisité au chapitre 4. Ce chapitre présente deux apports : tout d'abord, une formalisation d'un système de médiation prenant en compte les différents points mis en évidence par les travaux précédents ; ensuite, une nouvelle proposition pour l'allocation de requêtes induite par les réflexions liées à la formalisation. Enfin, le chapitre 5 propose une solution originale au problème de l'hétérogénéité sémantique qui vient en complément des alignements entre ontologies. En améliorant la compréhension entre acteurs sémantiquement hétérogènes elle permet d'améliorer l'autonomie sémantique des participants. Enfin, le chapitre 6 conclut et présente nos travaux en cours ainsi que nos perspectives de recherche.

CHAPITRE 2

Vers une infrastructure pour la recherche d'information distribuée

Résumé : *Ce chapitre est consacré à la recherche d'information sur Internet. Le problème est abordé avec l'objectif d'offrir une certaine autonomie aux contributeurs (sources de données) leur permettant de mieux gérer leurs relations avec le système. La solution proposée est totalement décentralisée. Chaque source d'information gère elle-même l'indexation de ses données et la réponse aux requêtes. Nous nous concentrons ici sur l'architecture distribuée permettant de répondre au problème du passage à l'échelle tout en offrant une autonomie importante à tous les participants.*

Ces travaux ont été menés entre 1999 et 2003 au sein de l'équipe *Intelligence Artificielle* sous la direction de Jean-François Nicaud, puis dans l'équipe *CID (Connaissances, Informations, Données)* sous la direction d'Henri Briand par un groupe de travail composé de quatre personnes : Sylvie Cazalens, Christine Jacquin, Emmanuel Desmontils et moi-même (groupe *Bonom* dont j'assurais la direction au sein du thème Ingénierie des Connaissance).

A noter, les quatre permanents constituant le groupe *Bonom* étaient tous en reconversion thématique au départ du projet (cf. préambule de ce document).

Encadrements relatifs à ce projet :

- C. Daurat, F. Jouhannel, B. Poussin, S. Pressense, *Corba et projet Bonom, étude de faisabilité*, projet de fin d'étude du DESS de Nantes, 1999-2000.

- [Cha02] Cédric Champeau, *Étude et définition de mécanismes de 'matchmaking'*, DEA, 2001-2002.
  a été embauché par l'entreprise *e-Manation* en tant qu'ingénieur.
- [Lem03] Sandra Lemp, *Processus d'allocation équitable de tâches dans un système multi-agent*, DEA, 2002-2003.
  a poursuivi en thèse au LINA.
- [Gro04] Guillaume Grondin, *Passage à l'échelle dans le cadre de la recherche d'information*, DEA, 2003-2004.
  a poursuivi en thèse à l'Ecole des Mines de Douai.

Ces travaux ont été partiellement supportés par :

- Transfert de technologie avec la société ***e-Manation*** [WWW 55].
- Projet ***BonomCV*** financé par la ***Fondation VédiorBis*** [WWW 53] sous l'égide de la Fondation de France [WWW 54].
- Projet RNTL ***MassCV*** labellisé en 2002.

Travaux liés :

- ***CorbaTrace*** [WWW 66], outil d'interception et de trace des communications entre objets CORBA distants. Publié sous licence LGPL. Mots-clés : CORBA, débuggage, trace, log, monitoring.
- ***ASU*** (Agents au Service des Utilisateurs). Outil d'aide à la gestion des données personnelles transmises sur Internet via des formulaires. Mots-clés : production et saisie de formulaires électroniques, sémantique, gestion et utilisation des données personnelles.

2.1 Problème et objectifs

La problématique générale porte sur la recherche d'information dans un système très largement distribué, ouvert. La solution alternative aux moteurs de recherche centralisés par mots-clés doit permettre d'améliorer la pertinence des réponses obtenues, mais aussi l'autonomie des fournisseurs d'informations que les moteurs de recherche centralisés utilisent comme des esclaves au sens de la terminologie client/serveur.

La recherche d'information sur Internet est un cas concret, riche et particulièrement intéressant. Cependant, nous sommes convaincus que les systèmes largement distribués sont amenés à se développer de manière importante. A l'image du système d'information actuellement présent sur Internet, nous pensons qu'au moins certains d'entre eux fonctionneront grâce aux ressources apportées par leurs participants. Le point de vue de ces participants qui contribuent au système, le composent, voire le constituent, est donc important à prendre en considération.

Concernant l'amélioration de la recherche d'information sur Internet, plusieurs travaux tendent à montrer les avantages d'une approche sémantique [FDES98, HHL99a] pour palier les inconvénients des recherches par mots-clés. Une amélioration substantielle de la qualité des réponses peut être espérée en adoptant une approche sémantique pour l'indexation des documents et des requêtes.

En second lieu, le nombre de sources d'information présentes sur Internet est de plus en plus important et croit rapidement. Les moteurs de recherche généralistes ont donc quelques difficultés pour tenir à jour leurs index. En conséquence, les réponses qu'ils apportent aux requêtes comportent des documents qui ont été supprimés et omettent ceux ajoutés trop récemment. Si le dernier n'est pas perceptible par l'utilisateur, le premier génère une erreur (erreur 404) lorsque l'initiateur de la requête tente d'accéder à la donnée. En réponse à ce problème, certains moteurs de recherche ont développé une stratégie de mise en cache de tous les documents indexés [WWW40]. Ils peuvent ainsi garantir à un initiateur de requête qu'il peut accéder à toutes les données qui ont été proposées en réponse à sa requête. Une course à la puissance de stockage et de calcul s'est donc engagée entre les moteurs. Cela passe par la mise en oeuvre de solutions distribuées (clusters et multi-sites) permettant de supporter la montée en charge tant du volume de données à traiter et à stocker que celui du nombre de requêtes.

Enfin, le passage par un moteur centralisé n'est pas toujours satisfaisant pour les fournisseurs de données. Outre la latence importante nécessaire pour la prise en compte d'une action de suppression, de modification ou d'ajout de données par les moteurs centralisés, ces derniers n'offrent à une source aucune possibilité de contrôle direct sur la diffusion de ses propres données. La stratégie de mise en cache ne fait qu'amplifier cette perte de contrôle. Une information supprimée de leur site reste disponible sur le moteur jusqu'à ce que celui-ci soit mis à jour ;

certaines pages nécessitant théoriquement une identification pour y accéder peuvent être directement disponibles via ces caches [WW40]. Certaines sources font donc le choix d'interdire l'accès de leur site aux moteurs de recherche. Elles fournissent aux internautes un moteur spécifique à leur site et maîtrisent ainsi les outils d'analyse et d'action nécessaires à la gestion de leur communication. L'installation de tels outils n'est plus un obstacle technologique. Cependant, les moteurs centralisés rendent un service unique dont la plupart des sites ne peuvent se passer pour être visibles sur la Toile. Ces derniers déploient donc des efforts importants pour s'adapter aux techniques utilisées par les moteurs. Leur principal objectif est que, pour des requêtes particulières généralement bien identifiées, leurs informations soient présentes dans les réponses fournies par le moteur et en suffisamment bonne place pour qu'elles soient lues.

Après avoir analysé les problèmes posés, qui se sont révélés de natures différentes, notre groupe de travail a conduit des investigations dans deux directions. Pour schématiser, Christine Jacquin et Emmanuel Desmontils ont fait porter leurs efforts sur l'indexation sémantique, et Sylvie Cazalens et moi même nous sommes concentrés sur l'organisation et l'architecture du système pour permettre l'accès aux sources d'informations en mode distribué. Ce rapport étant centré sur mes travaux, ceux sur l'indexation sémantique ne seront pas présentés car je n'y ai tenu qu'un rôle secondaire.

L'abandon de l'interrogation par mots-clés pour une approche sémantique introduit le problème de l'hétérogénéité des sources (représentation des données, langage d'interrogation, thématiques abordées...). Cet obstacle n'est pas résolu par la proposition présentée ici, mais il est abordé au chapitre 5. Offrir aux différentes sources la possibilité de répondre elles-mêmes aux requêtes nécessite une approche distribuée qui n'est pas sans poser problèmes. Celui du passage à l'échelle est particulièrement difficile. Le nombre de requêtes à traiter est bien trop important pour qu'il soit possible de solliciter systématiquement toutes les sources de données. Outre que cela créerait une charge réseau bien trop importante, les sources elles-mêmes ne sont certainement pas dimensionnées pour recevoir et traiter toutes les requêtes, pas plus qu'un initiateur n'est dimensionné pour recevoir et traiter les réponses de tous les fournisseurs. Il est donc indispensable de ne solliciter pour chaque requête qu'un nombre limité de fournisseurs, et il est clairement souhaitable que ceux sollicités soient pertinents par rapport à la requête.

De nombreuses questions se posent alors : "*Comment déterminer les fournisseurs pertinents ?*", "*Comment déterminer parmi eux ceux qui vont effectivement traiter la requête ?*","*Comment router les requêtes vers ces fournisseurs ?*", "*Comment router les réponses des fournisseurs vers les initiateurs de requêtes ?*", "*Comment intégrer de nouveaux participants ?*", "*Comment gérer le départ de participants ?*", et "*Quelle architecture adopter et comment la gérer ?*".

2.2 État de l'art

Quand nous commençons ces travaux, peu de systèmes distribués sont pensés pour supporter un nombre très important de fournisseurs d'informations et de requêtes. Deux problèmes majeurs sont : la définition d'une architecture passant à l'échelle, et la résolution du problème de connexion entre les initiateurs de requêtes et les fournisseurs d'information, parfois appelé "problème de liaison"[DSW97b]. Deux grands domaines se sont intéressés à ces problématiques : les bases de données et les systèmes multi-agents.

Très tôt le domaine des bases de données a proposé des solutions pour l'intégration de sources multiples. Pour résoudre le problème de l'hétérogénéité des schémas, variante du problème de connexion, des solutions à base de médiateur ont été proposées [Wie92c, TRV98b, WJW*51*] . Ces solutions sont intéressantes lorsque le nombre de fournisseurs n'est pas trop important. L'usage de plusieurs médiateurs est indispensable pour envisager un passage à l'échelle, cependant, leur gestion peut s'avérer délicate [NBN99].

Le domaine des systèmes multi-agents, en s'appuyant sur l'hypothèse d'autonomie des agents, développe une approche compatible avec notre vision. Là aussi le problème de liaison a été beaucoup étudié et des agents intermédiaires ont été définis [DSW97b, WS00]. Le rôle des agents dits "facilitateurs" [FFMM94, FLM97, WJW*52*] est de coordonner les interactions entre les autres agents. Les plus connus sont les "Matchmakers" et "Brokers" [KH95a, NBN99]. Ils déterminent quels sont les agents fournisseurs capables de traiter une requête à partir de déclarations de capacités effectuées par ces agents [Klu99b]. Ces déclarations peuvent être syntaxiques ou sémantiques [SKW99, NBN99]. Le Matchmaker renvoie à l'utilisateur la liste des fournisseurs adéquats en lui laissant la responsabilité du choix final. Au contraire, le Broker interroge

lui-même le(s) fournisseur(s) qu'il a choisi et fait suivre les résultats à l'initiateur de la requête. Un des avantages avancé de l'approche Broker est qu'il protège la confidentialité. Un fournisseur ne sait pas pour qui il travaille et un initiateur ne sait pas quel(s) fournisseur(s) a(ont) travaillé sur sa requête. Dans certains cas cela peut effectivement présenter un avantage, mais il est clair que cela va à l'encontre de l'autonomie des participants. Un Broker contrôle entièrement l'allocation, ce qui lui permet d'influer sur le comportement du système. Au contraire, une approche "Matchmaker", en laissant toute latitude à l'initiateur de requête pour choisir les fournisseurs qu'il désire, n'a aucun moyen de contrôler le système. Que ce soit avec un Matchmaker ou un Broker, la situation d'un fournisseur est très proche de celle d'un esclave exécutant les requêtes qui lui sont imposées. Les seules différences sont liées à l'identité de l'autorité qui les impose et à la connaissance des initiateurs des requêtes, mais cela ne lui apporte pas une grande marge de manœvure.

En ce qui concerne les architectures ou organisations de systèmes, une solution consistant à faire circuler les requêtes de proche en proche, d'un fournisseur à l'autre, et qui serait maintenant qualifiée de pair-à-pair non structuré, est proposée dans [She99]. Cette solution est particulièrement intéressante car elle permet à un groupe de participants de gérer eux-mêmes le routage sans dépendre d'un tiers. Cependant, elle nous semble peu adaptée à notre problématique car chaque fournisseur serait alors sollicité au moins une fois pour chaque requête, ce qui représente une charge bien trop importante. Deux architectures majeures utilisant des agents intermédiaires ont été proposées. Le projet *INFOSLEUTH* [NBN99, WWW 70] propose une architecture pour déployer des applications agents qui focalisent sur la collecte et l'analyse d'informations à partir de réseaux dynamiques de sources d'informations. Le projet *RETSINA* [SPVG03, WWW 71] a commencé par s'intéresser aux agents intermédiaires [DSW97b] et n'a proposé une infrastructure finalisée que bien plus tard. Ces deux projets considèrent un champ très vaste de problèmes allant du langage de communication entre les agents, à la gestion de l'intéropérabilité sémantique, en passant par les problèmes de déclaration de capacité et de services nécessaires pour répondre à n'importe quel type de demande. Les solutions sont donc relativement complexes et les applications dans lesquelles elles sont utilisées sont de taille modeste en nombre de fournisseurs.

2.3 Approche

Notre approche propose d'utiliser les ressources de calcul des serveurs de données pour indexer *localement* les pages publiées. Pouvant être alors maintenus sur chaque site, au plus proche des données, au fur et à mesure des modifications des informations publiées, les index seront à jour pour un coût modeste.

Pour conserver la maîtrise de la communication de leurs données, les serveurs répondent eux-mêmes aux requêtes. Le traitement d'une requête se confronte alors à un système très largement distribué avec des fournisseurs autonomes qui contrôlent données et ressources. Cette démarche serait actuellement qualifiée de pair-à-pair.

Pour répondre à la question "*Comment déterminer quels sont les fournisseurs adéquats ?*", nous proposons de profiter de l'approche sémantique conduite dans le projet. En considérant son index sémantique, un fournisseur peut déterminer, de manière automatique, semi-automatique, ou manuelle, quels sont les domaines concernés par les documents qu'il propose. Cela est suffisant pour organiser les fournisseurs en fonction d'une hiérarchie de domaines (cf. figure 2.1). Chaque participant a alors en charge de déterminer les domaines qui le concerne : les fournisseurs pour les documents et les initiateurs pour les requêtes. Cette technique simple permet pour chaque requête de ne solliciter que les fournisseurs des thématiques indiquées. Sous l'hypothèse que chacun renseigne correctement les domaines, les requêtes sont proposées aux fournisseurs compétents sans que les autres fournisseurs soient sollicités. Contrairement à la mise en place d'une ontologie commune, cette approche n'impose aucune contrainte sur la sémantique utilisée par les participants. Elle peut donc être utilisée que le milieu soit sémantiquement homogène ou hétérogène. L'hétérogénéité sémantique reste donc un problème ouvert qui n'est pas abordé dans le cadre du projet ***Bonom***, mais au chapitre 5 de ce document.

L'organisation logique en hiérarchie de thèmes est supportée par des agents intermédiaires qui gèrent la circulation des requêtes et des réponses. Ils sont fédérés en groupes composées d'un ou plusieurs agents, chaque groupe étant relatif à un domaine (cf. figure 2.1). Ces agents ont la responsabilité de collecter les requêtes pour leur groupe. Ils proposent ces requêtes aux fournisseurs de leur groupe ainsi qu'aux groupes associés à des sous-thématiques. Le fonc-

tionnement original de ces agents nous a conduit à les appeler des "*BaGates*", résultat de la contraction de "Bag" ("sac de requêtes") et "Gate" ("porte d'entrée dans le groupe"). Suivant ce schéma, au sein d'un groupe, un fournisseur s'approvisionne en requêtes auprès d'un (ou plusieurs) BaGate(s). Les BaGates d'un groupe sont eux aussi clients des BaGates du groupe associé à la thématique hiérarchiquement supérieure. Il est intéressant de noter que le traitement d'une requête par un BaGate est particulièrement simple : pas de traitement sémantique, mais un simple traitement de chaîne de caractères permet de déterminer les thématiques associées à la requête par l'initiateur, et donc à qui proposer la requête pour qu'elle suive un chemin lui permettant d'atteindre sa destination. Cette approche évite aussi toute analyse de la capacité des fournisseurs à traiter la requête puisque ce sont les fournisseurs qui choisissent les requêtes.

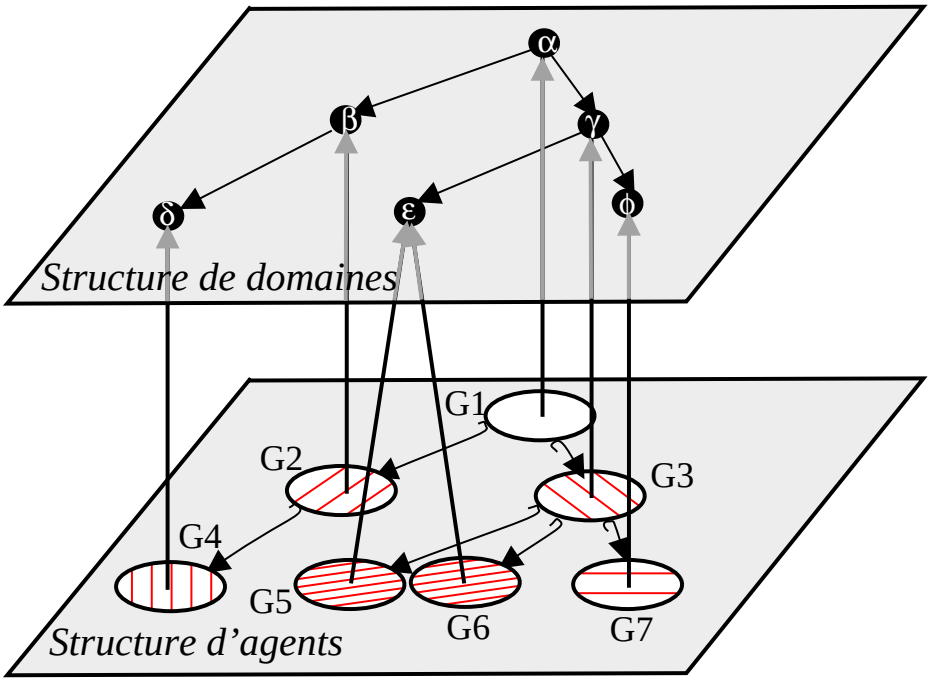


Figure 2.1 – Architecture ***Bonom*** : structuration en groupes et domaines.

Pour répondre à la question "*Comment déterminer parmi eux ceux qui vont effectivement traiter la requête ?*", nous divisons cette question en deux. La première, "Au maximum, combien de fournisseurs doivent traiter la requête par groupe ?", est posée à l'initiateur de la requête. Il est en effet le mieux placé pour déterminer ce nombre en fonction de ses attentes et en prenant

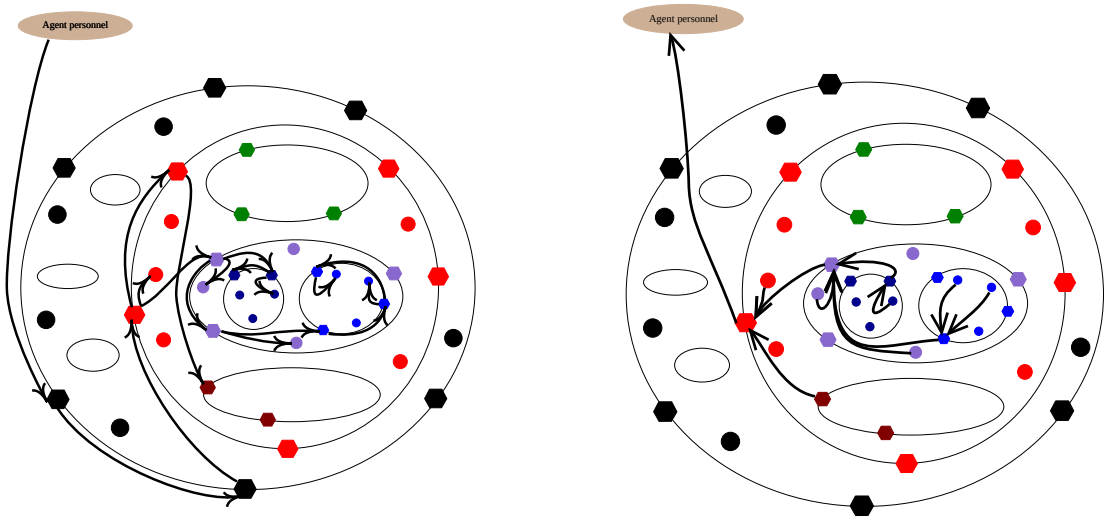
en compte ses capacités réseau et de traitement. Le nombre précisé étant relatif à chaque groupe, le nombre total maximal de réponses s'obtient en multipliant le nombre de groupes répondant par le nombre indiqué par l'utilisateur. Chaque groupe associé à une thématique ciblée par la requête doit donc se conformer à cette contrainte en répondant à la deuxième partie de la question initiale : "*Quels fournisseurs répondent à la requête ?*". Nous proposons ici d'inverser le problème. Un fournisseur choisit lui même les requêtes qu'il souhaite traiter. Ce n'est plus au système de déterminer qui doit traiter quoi, mais à chaque fournisseur de savoir ce qu'il peut et veut traiter. La répartition des requêtes entre les fournisseurs s'effectue simplement en suivant la politique du "premier arrivé, premier servi". Lorsque le nombre maximum de fournisseurs traitant la requête est atteint, elle n'est tout simplement plus proposée aux fournisseurs. A noter que l'initiateur peut explicitement interdire sa requête à certains fournisseurs. Ainsi, si après avoir analysé les réponses à sa requête l'initiateur souhaite obtenir d'autres résultats, il peut la ré-émettre en l'interdisant aux fournisseurs ayant déjà répondu. Par pas successifs, il peut interroger tout le système si besoin. A noter, l'interdiction d'un fournisseur peut aussi être la conséquence d'un jugement négatif suite à une expérience passée ou une évaluation de la réputation du fournisseur. Pour résumer, les initiateurs déterminent donc "combien" et "qui ne traite pas", alors que le groupe des fournisseurs a le contrôle de "qui traite".

Une requête est donc constituée de plusieurs champs. Le premier contient le texte de la requête. Il est exprimé dans un langage d'interrogation (mots-clés, expression booléenne, SQL [GW02], XQuery [WWW], vecteur sémantique [Woo97]...) sur lequel nous ne faisons aucune hypothèse bien que le projet ***Bonom*** s'intéresse aux requêtes sémantiques. Les champs suivants précisent le langage utilisé et l'ontologie de référence dans laquelle ont été puisés les termes de la requête. Les informations liées aux principes énoncés ci-dessus sont regroupées dans l'enveloppe de la requête : initiateur de la requête, liste non vide des domaines ciblés par l'initiateur, liste des sous-domaines interdits (la requête ne les visitera pas), liste des fournisseurs interdits, nombre de fournisseurs maximum à solliciter par domaine (entier supérieur ou égal à 1).

Une fois posés ces principes généraux, pour obtenir un système fonctionnel, il est nécessaire de proposer des solutions à plusieurs problèmes. Nous ne présentons ici que les grands principes des protocoles mis en place pour répondre aux questions : "*Comment router les requêtes*

vers ces fournisseurs ?", et "*Comment router les réponses des fournisseurs vers les initiateurs de requêtes ?*", "*Comment intégrer de nouveaux participants ?*", "*Comment gérer le départ de participants ?*", et "*Comment gérer la structure des agents intermédiaires ?*".

- *routage d'une requête (figure 2.2(a)).*



(a) (b)

Figure 2.2 – Routage des requêtes (a) et des réponses (b).

En fonction du degré de connaissance qu'il a du système, un initiateur de requête peut confier sa requête à un BaGate du domaine "racine" (la liste de ces BaGates est maintenue sur un serveur pour bootstrap), ou à un BaGate d'un domaine plus proche des domaines ciblés. La requête circule de groupe en groupe, en se dupliquant si nécessaire pour parcourir plusieurs domaines ciblés en parallèle, et ce jusqu'à atteindre les groupes gérant les domaines concernés par la requête. A l'intérieur d'un groupe dont le domaine est ciblé par la requête, elle est prise en charge par un seul BaGate à la fois qui la propose aux fournisseurs ainsi qu'aux BaGates des sous-domaines non interdits qui s'approvisionnent chez lui. Dans ce groupe, elle circule ainsi de BaGate en BaGate jusqu'à ce que le nombre requis de fournisseurs soit atteint et qu'elle ait été proposée à tous les sous-domaines possibles ou que tous les BaGates du groupe aient été sollicités. Les informations liées à la gestion de la requête et permettant en particulier d'éviter les soumissions multiples (fournisseur, sous-thème, BaGate) sont ajoutées à l'enveloppe de la requête.

- *Routage des réponses (figure 2.2(b)).*

  Pour éviter une trop grande charge réseau à l'initiateur de la requête, les fournisseurs ne lui renvoient pas les réponses directement. Elles transitent par certains des BaGates visités par la requête. Ceux-ci, dont le rôle est appelé "*concentrateur*", regroupent les réponses en provenance de différentes sources. Sans attendre que tous les fournisseurs aient répondu, les réponses sont renvoyées périodiquement de concentrateur en concentrateur jusqu'à l'initiateur de la requête. Pour chaque requête, les concentrateurs sont choisis lors de sa circulation suivant quelques critères simples : au maximum un concentrateur par groupe, et seuls les groupes où la requête subit un traitement (envoyée à des fournisseurs) ou est envoyée à plusieurs sous-domaines mettent en place un concentrateur. Le premier BaGate d'un groupe qui détecte une situation correspondant à ces critères s'attribue le rôle de concentrateur. La liste des concentrateurs est attachée à la requête. Ainsi, si un concentrateur ne répond pas, par exemple pour cause de panne, les réponses peuvent être redirigées vers un concentrateur de plus haut niveau.
- *Ajout/retrait d'un agent intermédiaire (BaGate).*

  Pour s'intégrer, un BaGate commence par contacter un serveur fiable qui contient les URLs de plusieurs BaGates du groupe "racine" (bootstrap). Il envoie alors un message spécifique ciblé sur le domaine auquel il souhaite contribuer. Ce message circule de groupe en groupe et s'arrête soit en atteignant celui du domaine cible, soit si ce dernier n'existe pas encore, en atteignant le groupe du domaine le plus proche dans la hiérarchie du domaine ciblé. Dans le premier cas, le nouveau BaGate échange avec ses frères BaGates pour obtenir une connaissance de son environnement et signaler sa présence. Il s'abonne auprès d'au moins un BaGate du groupe associé au domaine hiérarchiquement supérieur pour s'approvisionner en requêtes. Dans le deuxième cas, il est le premier de son domaine. Il se contente de s'abonner auprès des BaGates du groupe où son message s'est arrêté pour obtenir des requêtes.

  Pour son départ, un BaGate ne fait rien de particulier. Ses clients et BaGates frères se rendent compte de son départ par eux-mêmes. Les uns cherchent un autre fournisseur, les autres ne lui adressent plus de message. Une difficulté un peu plus importante apparaît

lorsque le BaGate quittant le système est le dernier de son groupe. En attendant l'arrivée d'un nouveau BaGate, ses clients doivent alors faire l'effort d'obtenir les requêtes qui les concernent auprès des BaGates du domaine plus général et périodiquement se mettre en quête d'un BaGate de leur domaine.

- *Ajout/retrait d'un fournisseur.*

  L'ajout d'un fournisseur est identique à celui d'un BaGate à ceci près qu'il n'a pas besoin de signaler sa présence à tous les BaGates du groupe où s'arrête son message de recherche de groupe. Il se contente d'en choisir au moins un auprès duquel il obtiendra ses requêtes. Le départ d'un fournisseur ne nécessite aucun traitement : il cesse simplement de demander des requêtes et donc d'en obtenir.

- *Création d'un nouveau domaine.*

  La création d'un nouveau domaine est effectuée en modifiant la hiérarchie stockée sur un ou plusieurs serveurs stables servant de bootstrap au système. Cette opération ne peut être décidée et réalisée que par des humains autorisés. Les artefacts du système n'ont aucun rôle dans cette opération.

- *Création d'un nouveau groupe.*

  La création d'un nouveau groupe associé à un domaine qui n'en a pas encore est la conséquence de l'insertion dans le système d'un BaGate dédié au domaine en question. Ce n'est donc pas une opération spécifique. La décision est prise par le propriétaire du BaGate inséré.

- *Maintien de la cohérence de l'organisation.*

  Suite à une création simultanée, ou à des problèmes réseaux persistants, plusieurs groupes peuvent être associés au même domaine. Pour remédier à cette situation, les BaGates effectuent régulièrement une recherche de frères du même domaine qu'ils ne connaissent pas. Cette recherche est effectuée en passant par le domaine hiérarchiquement supérieur. Si un BaGate découvre ainsi un frère inconnu, la procédure de fusion est immédiatement amorcée. Elle consiste simplement à échanger et synchroniser les connaissances que les BaGates ont sur leurs groupes respectifs.

2.3.1 Validation

Cette approche a provoqué à l'époque un certain scepticisme quant à sa faisabilité. En effet, la possibilité de mettre en œuvre un système distribué avec participants autonomes était particulièrement mise en question. Les succès remportés depuis par le pair-à-pair, le calcul volontaire, le B2B et B2P ont montré la faisabilité et la potentialité de telles approches. Cependant, pour répondre à ces interrogations de l'époque, nous avons développé un prototype en Java. Pour supporter les échanges entre les participants, deux approches étaient possibles : 1 - utilisation d'un intergiciel (MiddleWare) CORBA pour lequelle nous disposions d'implémentations libres (CORBA a été normé pour Java en 1998), ou 2 - utilisation de la norme "Agent Communication Language" [WWW 57] proposée par la "Foundation for Intelligent Physical Agents" (FIPA) [WWW 58] qui est maintenant une société de standardisation de l'IEEE. L'ACL FIPA, en vogue à l'époque, impose de suivre scrupuleusement la norme pour bénéficier des avantages apportés par la sémantique sous-jacente (exprimée en logique modale). Notre projet ne nécessitant pas de raisonnement de la part des agents lors de leur communication, cela ne présentait pas pour notre projet un avantage décisif. Il en va de même pour le langage de communication entre agents KQML [FFMM94, WWW 56]. Nous avons donc opté pour la solution intergiciel avec l'ORB CORBA Orbacus[WWW 47] pour développer une première version de notre prototype qui fonctionnait en distribué.

Une intégration de ce prototype et de l'outil d'indexation sémantique développée au sein du projet ***Bonom*** a été réalisée dans le cadre d'un transfert de technologie avec la société *e-Manation* [WWW 55]. Bien plus qu'une simulation, la validation a donc consisté à faire fonctionner cette proposition en situation. Le prototype résultant a été présenté par la société e-Manation dans des conditions réelles de distribution de données et évalué par un consortium de sociétés (dont Airbus Industrie). Le consortium a alors apporté son soutien à la société e-Manation.

Le projet ***BonomCV*** financé par la *Fondation Vediorbis* sous l'égide de la *Fondation de France* a été l'occasion de confronter l'approche au problème concret de la mise en relation d'offres d'emplois et de CVs de demandeurs d'emplois. Le projet RNTL ***MassCV***, labellisé mais non financé, poursuivait aussi cet objectif.

2.4 Contribution

Les contributions du projet ***Bonom***, intégrant quatre permanents en reconversion, concernent donc l'indexation sémantique [CDJL02, DJ02a, DJ01b, DJ01a] (non présenté dans ce document) et la définition et l'implémentation d'une architecture pour la gestion de requêtes dans un cadre largement distribué [CDJL02, CL01a, CL01b, CDJL00c]. Alternative aux moteurs de recherche centralisés, la solution offre aux participants une autonomie bien plus importante. Les initiateurs de requêtes peuvent cibler leurs requêtes sur certains thèmes et interdire de traitement certains fournisseurs ou certains sous-thèmes. Mais ce sont les fournisseurs pour lesquels les différences sont les plus marquées. Ils ne sont plus considérés comme des esclaves et gagnent une certaine maîtrise sur la communication de leurs informations. En particulier, ils choisissent eux-mêmes les requêtes qu'ils traitent suivant une stratégie qui leur est propre et qu'ils maîtrisent entièrement. L'approche offre donc une autonomie importante et nouvelle aux fournisseurs d'informations en s'appuyant sur l'hypothèse qu'ils sont en mesure de déterminer eux-mêmes ce qui leur convient le mieux. Par exemple, un fournisseur trop chargé peut décider de ne plus solliciter de requêtes, ou devenir beaucoup plus sélectif dans ses choix, le temps de résorber la charge.

D'un point de vue technique, le rôle des agents intermédiaires (*BaGates*) est central pour le projet. En s'appuyant sur l'hypothèse d'autonomie et la capacité que cela suppose pour chacun de déterminer ce qui lui convient, nous demandons à chaque articipant de caractériser par rapport à une hiérarchie de thème connue le contenu des informations qu'il propose et les requêtes qu'il emmet. Chaque participant a donc en charge l'analyse sémantique qui le concerne. En conséquence, le rôle de BaGate n'a pas à supporter de calcul complexe et peut être tenu par des volontaires n'ayant pas de capacité particulièrement importante. Les fournisseurs peuvent éventuellement jouer ce rôle à condition de vérifier que cela ne pose pas de problème déontologique, en particulier en cas de concurrence entre fournisseurs. Enfin, sauf demande explicite d'un utilisateur, une requête n'impacte pas sur tout le réseau, mais en général sur une faible partie. La charge réseau ainsi que la charge de chaque participant, ont été contenues, que ce soit pour la circulation des requêtes ou celle des réponses.

Enfin, notre approche est particulièrement générique dans la mesure où elle ne nécessite aucune hypothèse que ce soit sur le langage utilisé pour exprimer les requêtes ni sur la sémantique utilisée. Grâce à l'autonomie des participants, notre approche peut être utilisée en milieu hétérogène, que cette hétérogénéité soit syntaxique (langage d'interrogation) ou sémantique. En effet, chaque fournisseur choisit les requêtes qu'il souhaite traiter, et peut vérifier s'il connaît le langage et la sémantique utilisés ou s'il dispose des outils nécessaires pour la traiter avant d'effectuer son choix. Ce n'est donc pas aux agents intermédiaires d'effectuer ces vérifications souvent coûteuses, et l'hétérogénéité n'a donc aucune conséquence directe sur les BaGates. La mise en oeuvre de nouveaux langages ou de nouvelles sémantiques peut donc être réalisée sans aucune modification du système. Il n'en demeure pas moins que le système est plus efficace en cadre homogène car cela évite de nombreuses vérifications, qui même si elles sont réparties, ralentissent le traitement d'une requête.

Le projet ***Bonom*** nous a aussi conduit à nous intéresser à des sujets connexes dont deux ont permis d'obtenir des résultats intéressants.

Développer un prototype jusqu'à le déployer pour expérimentation en situation réelle a nécessité des efforts importants. Cela a constitué une expérience particulièrement intéressante et motivante qui nous a aussi permis d'acquérir des compétences supplémentaires dont une bonne connaissance du développement CORBA. En particulier, l'analyse et le débuggage de notre application répartie nous a amené à réfléchir sur la possibilité d'observer les échanges entre composants distants. Aucun outil ne permettant de répondre à ce besoin, nous avons proposé ***CorbaTrace*** [WW66]. Basé sur les intercepteurs CORBA (norme 2.3) et publié sous licence LGPL, cet outil permet de reconstituer le graphe de séquence décrivant les échanges entre plusieurs objets distants à partir des interceptions réalisées sur chacun de leurs sites, et ce sans accéder au code des objets ainsi surveillés.

Enfin, dans le but d'obtenir un cas d'application concret mais moins général et plus facilement maîtrisable que la recherche d'information sur internet, nous nous sommes intéressés à la gestion des informations présentes dans les formulaires auxquels tout utilisateur est confronté un jour ou l'autre. En effet, que ce soit en intra-entreprise, ou sur Internet dans le cadre de l'e-administration, les formulaires sont de plus en plus présents. La recherche et la saisie des

informations à fournir peut être relativement fastidieuse et quelquefois répétitive. Un outil permettant de gérer les informations personnelles et de partager certaines informations à l'intérieur d'une communauté d'utilisateurs nous a semblé être un cadre d'application intéressant. Dans un premier temps, nous avons travaillé sur la qualification sémantique des informations demandées dans les formulaires et sur un outil permettant à chaque utilisateur de mémoriser les informations qu'il saisit dans les différents formulaires auxquels il est confronté. La caractérisation sémantique permet de réutiliser les informations d'un formulaire à l'autre. L'utilisateur bénéficie alors d'un service de gestion de ses données personnelles qui lui évite en particulier les re-saisies fastidieuses. Ces idées ont été explorées grâce à plusieurs projets d'étudiants menés dans le cadre de la deuxième année du Master ALMA sous le nom de projet ***ASU*** (Agents au Service des Utilisateurs). Nous disposons actuellement d'un prototype permettant à un utilisateur de mémoriser ses données et de les réutiliser d'un formulaire à l'autre avec comme cadre d'application la gestion des missions pour les agents de la Faculté des Sciences et des Techniques de Nantes. La communauté des utilisateurs constitue alors un environnement naturel propice au partage d'informations (non confidentielles), et donc à la recherche d'informations.

2.5 Leçons et Perspectives

Le transfert de technologie du projet Bonom vers la société e-Manation nous a ouvert sur le monde industriel et nous a permis de confronter nos propositions à des besoins réels. Ces besoins étaient orientés "intra-entreprise" pour de grands groupes ou communication entre entreprises sur un même projet. Sans atteindre l'échelle de la recherche d'information sur Internet, les problèmes abordés sont tout de même similaires. L'approche que nous avons proposée avait le soutien de grandes entreprises. En dehors des considérations scientifiques et de développement, s'il n'y avait qu'une seule leçon à retenir, elle concernerait certainement l'importance d'une rédaction méticuleuse des contrats pour ce type d'aventure.

Notre objectif scientifique initial était d'obtenir un système fonctionnant sous les conditions imposées par un contexte largement distribué, ouvert, intégrant des participants désirant conserver un certain contrôle sur leurs ressources. En pouvant sélectionner les requêtes qu'ils

traitent, les fournisseurs disposent d'une capacité de contrôle très importante. L'intérêt de cette approche a été confirmé tant par les industriels s'étant penchés sur le projet que par les validations. Cependant, il résulte de cette possibilité offerte aux fournisseurs que certaines requêtes ne sont pas traitées uniquement pour des raisons personnelles aux fournisseurs. Cela est quelquefois désappointant et il est alors impossible d'appréhender les raisons d'absence de réponse (aucun fournisseur, fournisseurs trop occupés, fournisseurs ne voulant pas traiter la requête). Il est alors difficile de savoir quelle suite à donner. Est-il opportun de ré-émettre la requête ? Et si le résultat est identique, la même question se pose à nouveau. Ce fonctionnement ne répond pas aux attentes des initiateurs de requête. L'autonomie accordée aux fournisseurs s'avère donc trop importante et déséquilibre le système en faveur des fournisseurs. Le besoin d'une solution offrant un meilleur équilibre entre les intérêts des uns et des autres se fait donc sentir. Les chapitres 3 et 4 sont consacrés à ce problème.

Toutes nos expérimentations ont été réalisées dans un contexte distribué, mais sémantiquement homogène. La gestion de l'hétérogénéité sémantique est pourtant un problème difficilement évitable. Cependant, une particularité des solutions présentées dans ce chapitre est qu'elles sont indépendantes de toute considération d'hétérogénéité sémantique. Pour intégrer le système Bonom, un participant doit seulement comprendre la hiérarchie de thèmes, ou plus précisément la partie qui le concerne. Cela est possible, même en cas d'hétérogénéité forte, qu'elle soit syntaxique ou sémantique. Les pages jaunes des télécommunications en sont un bon exemple. Dans la proposition actuelle, la prise en compte de l'hétérogénéité, dont sémantique, est à la charge des participants. Par exemple, un fournisseur ne traite pas une requête qui n'est pas exprimé dans le langage ou la sémantique qu'il utilise, ou alors, il doit s'assurer que les outils lui permettant de comprendre la requête sont opérationnels et à sa disposition. Le problème de l'hétérogénéité sémantique est donc entièrement à la charge des participants et n'impacte pas sur l'organisation ni les BaGates. Il n'en demeure pas moins. Le chapitre 5 de ce document aborde le problème de l'hétérogénéité sémantique.

Les potentialités du projet *ASU* sur la gestion des formulaires, mentionné dans la section précédente, ont conduit à travailler sur un projet industriel avec les sociétés Mandriva et Nexedi. Une application de recherche d'information doit permettre aux utilisateurs d'échanger des in-

formations générales et non personnelles pour renseigner des formulaires (ex. distance entre deux villes, informations générales d'une entreprise, etc), mais aussi des préconisations générales concernant quels formulaires remplir dans un cadre particulier (ex. départ en mission). Cette application tout en fournissant un service indéniable aux utilisateurs ouvre des perspectives intéressantes en recherche. Elle permet de concrétiser la recherche d'information en mode sémantique dans un cadre distribué et ouvert pouvant s'appliquer naturellement à différentes échelles : en entreprise, en mode communautaire ou de manière totalement ouverte sur Internet.

CHAPITRE 3

Une médiation flexible pour l'allocation de requêtes

rt

Résumé : *L'allocation de requêtes est un problème générique qui prend une importance toute particulière dans le cadre des systèmes distribués. L'ouverture d'un système à des participants autonomes dote ce problème d'une nouvelle facette. En effet, une gestion des requêtes inadaptée aux participants peut les inciter à quitter le système. Pour éviter une telle éventualité, nous recherchons une technique d'allocation assurant un certain équilibre entre les intérêts des uns et des autres.*

Ces travaux ont été menés dans le cadre de l'équipe *CID (Connaissances, Informations, Données)* sous la direction d'Henri Briand puis dans le cadre de l'équipe *ATLAS-GDD* sous la direction de Patrick Valduriez, en collaboration avec Sylvie Cazalens.

Ils ont donné lieu aux encadrements académiques suivants :

- [Lem03] Sandra Lemp, "Processus d'allocation équitable de tâches dans un système multi-agent", DEA, 2002-2003.
- [Lem07] Sandra Lemp, "Médiation flexible dans un système pair-à-pair", Doctorat, 2004-2007.

  A poursuivi en post-doctorat à l'Université de La Rochelle.

3.1 Problème et objectifs

L'allocation de requêtes dans le contexte d'un système ouvert, où l'autonomie des participants ne peut qu'être actée, est l'une des problématiques soulevées par le projet Bonom (chapitre 2). Dans le cadre de la recherche d'information sur Internet nous avons constaté que laisser agir les fournisseurs en fonction de leur propre stratégie et sans aucun contrôle peut induire un comportement inadéquat pour les initiateurs de requêtes qui peuvent être privés de réponses. Or, les intérêts des initiateurs de requêtes sont tout aussi importants que ceux des fournisseurs. Un mécanisme d'allocation de requêtes, en introduisant un contrôle sur le système et ses participants, peut éviter ce désagrément. Cependant, pour ne pas être préjudiciable au système, il doit prendre en compte les intérêts propres des différents participants en évitant de favoriser un groupe ou un participant en particulier.

Comme dans bien des cas en informatique, l'allocation de requêtes peut être appréhendée d'un point de vue "donnée" ou "fonctionnel". Par exemple, elle est naturellement abordée d'un point de vue "données" dans les systèmes intégrant plusieurs bases de données [TRV98b, TRV96]. Dans le domaine des intergiciels (MiddleWare) [OMG96], des services [WL15], des systèmes multi-agents [WS00, Syc01], etc il s'agit de trouver sur le réseau une fonctionnalité particulière pour l'utiliser directement ou la composer avec d'autres. Dans ce dernier cas, le point de vue est "fonctionnel".

Quel que soit le point de vue adopté, certains problèmes liés à l'allocation sont similaires. Les points suivants sont présents, partiellement ou en totalité, dans ces différentes approches : *a* - décomposer les requêtes, *b* - déterminer les fournisseurs ayant les compétences adéquates pour répondre à une requête ou une sous-requête, *c* - choisir les fournisseurs à solliciter parmi ceux disposant des compétences requises, et *d* - fusionner/combiner les réponses. Les points *a*, *b*, *d* sont dépendant des applications. Une requête de base de données et une demande de service ne s'analysent pas de la même manière. Le point *c* est le seul point générique commun à toutes les applications. A caractéristiques techniques équivalentes, c'est aussi sur ce point qu'une application peut faire la différence en devenant plus attractive grâce à l'attention qu'elle prête aux intérêts des participants.

Notre objectif est de proposer une technique permettant de choisir des fournisseurs parmi la liste de ceux qui sont compétents. Cette solution doit être suffisamment générique pour s'intégrer aux différents environnements applicatifs, et pour s'adapter à des situations diverses : participants défendant des intérêts individuels ; fournisseurs refusant de communiquer à qui que ce soit les informations considérées comme privées (ex. capacités de calcul, charge…) pourtant utiles pour la gestion du système ; fournisseurs hétérogènes, tant du point de vue de leurs capacités de calcul (temps de traitement) que de leurs données (réponses différentes à une même requête), et pas nécessairement exclusivement dédiés au système considéré. Nombre de ces points sont liés à l'autonomie des participants. Il est cependant difficile de déterminer si l'autonomie est un élément du problème ou si elle constitue une partie de la solution. En effet, l'autonomie est liée à divers aspects dont le premier est individuel. Pour assumer son autonomie, un participant est supposé être capable de déterminer par lui-même ce qui lui convient. Impossible de prétendre à une quelconque autonomie sans cette capacité. Le système peut donc se décharger de certains problèmes sur les participants autonomes. Par exemple, pour un système d'allocation de requêtes :

- Les initiateurs peuvent déterminer par eux-mêmes quels sont les fournisseurs qui leur conviennent.

  C'est particulièrement intéressant en cas d'hétérogénéité des fournisseurs (ou des initiateurs). C'est dans ce cadre que sont utilisés les mécanismes de réputation [Abe01, DdVP+02, JI02, SS01, WW43] sous l'hypothèse supplémentaire que les participants construisent leurs opinions sur des critères compatibles.
- Les initiateurs de requêtes peuvent déterminer par eux-mêmes le nombre de fournisseurs qui doivent la traiter.

  Soumettre une requête simultanément à plusieurs fournisseurs en contrôlant leur nombre présente un intérêt dans deux cas très distincts. La recherche d'informations appartient au cas où les réponses varient naturellement d'un fournisseur à l'autre. L'intérêt d'en interroger plusieurs est alors évident. Le cas où les réponses des fournisseurs sont par hypothèse identiques (ex. résultat d'un calcul déterministe) est très différent. L'intérêt de solliciter plusieurs fournisseurs est alors de confronter leurs résultats pour vérifier que

leurs réponses sont bien identiques. En effet, n'oublions pas que l'autonomie peut aussi avoir pour conséquence l'intégration dans le système de participants malicieux. Il est ainsi possible de détecter les erreurs ou les comportements malicieux. Dans les deux cas, le système peut se reposer sur l'initiateur de la requête pour qu'il détermine les fournisseurs dont il souhaite obtenir des réponses.

- Les fournisseurs peuvent déterminer par eux-mêmes les requêtes qu'ils veulent traiter. Tout comme les initiateurs peuvent différencier les fournisseurs, les fournisseurs peuvent distinguer les requêtes, par leur forme, l'initiateur, l'objet de la recherche, ou tout autre moyen. Au yeux d'un fournisseur certaines requêtes peuvent apparaître plus intéressantes que d'autres. Cet intérêt peut varier en fonction de l'état du fournisseur. Par exemple, il peut être d'autant plus sélectif dans ses choix qu'il est chargé. Le fournisseur est donc le seul à même de déterminer si une requête lui convient ou pas à un instant donné.

Un deuxième aspect de l'autonomie concerne les relations d'un participant avec son environnement. La marge de manœuvre laissée par un le système à un participant détermine sa capacité de s'administrer lui-même et d'influencer le système. Si les requêtes sont allouées par une autorité sans tenir aucun compte des points de vue des différents participants, les participants sont considérés comme des esclaves. Les techniques de médiation actuelles sont basées sur un tel principe. Au mieux, elles tiennent compte d'intérêts présumés des participants (temps de réponse, répartition de charge). C'est là que l'autonomie devient un élément du problème. Si la stratégie de l'allocation ne lui convient pas, un participant autonome par rapport au système peut décider de le quitter. Ce départ a nécessairement une influence sur le système et certains départs sont à éviter car, par effet domino, ils peuvent entraîner le départ de tous les participants. Une approche peut consister à s'appuyer là aussi sur l'autonomie des participants et les laisser négocier l'allocation des requêtes sans intervention d'aucune autorité, avec l'espoir que la solution émergeant aura de meilleures propriétés.

Pour résumer, notre objectif est ici de sélectionner les fournisseurs qui doivent traiter une requête parmi ceux qui sont compétents, en supposant que les fournisseurs conservent leurs informations caractéristiques privées, en préservant au mieux les intérêts des uns et des autres, mais en imposant que toutes les requêtes traitables soient traitées. Toute solution peut être en-

visagée dans la mesure où elle tient compte des points de vue individuels des différents participants dans la prise de décision pour réguler le système en fonction des intérêts de ceux qui le composent.

Les critères classiques d'évaluation d'un processus d'allocation de requêtes concernent en général le temps de réponse perçu par l'initiateur, la répartition de charge entre les différents fournisseurs, le nombre de messages échangés et la charge réseau. Le fait d'intégrer des participants autonomes renouvelle aussi cette problématique et nécessite d'introduire de nouveaux critères : prise en compte des intérêts d'un fournisseur, prise en compte des intérêts d'un initiateur de requête, requêtes traitables effectivement traitées, équilibre de l'allocation entre les fournisseurs et les initiateurs, entre les fournisseurs, entre les initiateurs.

3.2 État de l'art

L'utilisation d'un médiateur dont le rôle est d'allouer les requêtes à traiter permet de créer un lien entre différents participants d'un système distribué, cf. figure 3.1.

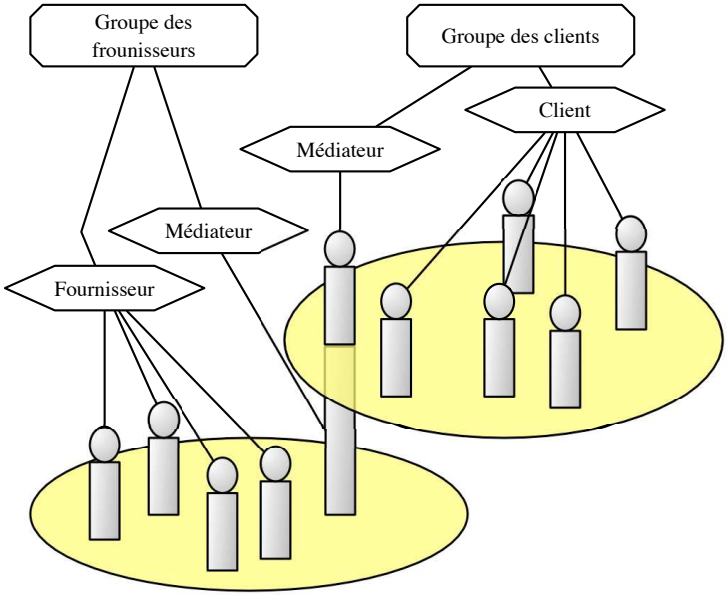


Figure 3.1 – Médiateur représenté avec le modèle Agent/Groupe/Rôle [ONL04].

Des médiateurs sont utilisés dans le domaine des bases de données [TRV98b, WFW] où ils

permettent d'intégrer plusieurs sources (fournisseurs) éventuellement hétérogènes. L'optimisation de critères de répartition de charge ou de temps de réponse [ABKU99, GBGM04, MTS90, RM95, SKS92] a aussi fait appel à des médiateurs. Ils sont aussi étudiés et utilisés dans le domaine des systèmes multi-agents avec une approche plutôt fonctionnelle [DSW97b]. Une distinction est par exemple proposée entre "Matchmaker" et "Broker" [KS01], cf. section 2.2. Dans les deux cas, les initiateurs leur soumettent leurs requêtes. Le premier se contente de leur donner en réponse la liste des fournisseurs ayant les capacités de traiter la requête en laissant le problème du choix aux initiateurs. Le second se charge de ce choix, soumet la requête lui-même aux fournisseurs ainsi sélectionnés et intègre éventuellement les réponses avant de les faire parvenir à l'initiateur. Enfin, les médiateurs sont aussi utilisés dans les domaines des intergiciels et des services où, en étant basés sur des principes similaires ils permettent de découvrir des services (ex. service vendeur [OMG96] de *CORBA* [WPV 62]). Certaines solutions évoquées envisagent l'autonomie des initiateurs (ex. Matchmaker qui suppose qu'ils sont aptes à faire un choix eux même), mais aucune n'offre d'autonomie réelle aux fournisseurs. Dans tous les cas cités précédemment, les principaux problèmes adressés sont la décomposition et planification d'une requête et la détermination des fournisseurs adéquats pour y répondre. Nous concentrons la suite de cet état de l'art sur les travaux concernant l'allocation d'une requête à des fournisseurs parmi ceux ayant les capacités de la traiter.

Une première solution consiste à utiliser des critères système pour effectuer le choix des fournisseurs. Les approches visant un équilibrage de charge [ABKU99, MTS90, GBGM04, RM95, SKS92] (respectivement un temps de réponse minimal [FKN+92, Zho88]), choisissent les fournisseurs les moins chargés (respectivement les plus rapides à répondre). Ces approches supposent que les intérêts des participants se résument aux critères choisis qui sont figés dans le système.

Dans certains cas, il est possible de prendre en compte les préférences en les traitant comme des capacités. *BOINC* [And04, WPV 20], introduit en 2004 comme mise à jour du projet *seti@home* [ACK+02, WPV 21] (initié en 1999) fonctionne sur ce principe. Il permet aux internautes d'apporter bénévolement des ressources de calcul aux projets qu'ils souhaitent soutenir : *seti@home* [WPV 21], projet d'astrophysique dont *BOINC* est issu, dirigé par U.C. Berke-

ley qui étudie des signaux radio pour détecter des anomalies pouvant être le signe d'une vie extraterrestre ; *Docking@home* [WW23], projet de biologie mené par l'Université du Delaware sur l'étude de protéines ; *Climateprediction.net* [WW22], projet en sciences de la terre initié par l'Université d'Oxford qui étudie des modèles météorologiques, etc. Les bénévoles sont en premier lieu attirés par l'intérêt qu'ils portent aux projets gérés par *BOINC*. Chaque volontaire a la possibilité d'exprimer ses préférences en spécifiant le (ou les) projet(s) auxquels il souhaite contribuer par une déclaration statique similaire à une déclaration de capacités. Les projets envoient alors des requêtes (demandes de traitement) aux participants en fonction de leurs besoins et des capacités de chacun. Un participant n'est pas obligé de dédier une machine complète, et ne concède au système que les ressources de calcul qu'il n'utilise pas. En d'autres termes, un volontaire conserve une maîtrise importante sur ses ressources en contrôlant statiquement à quels projets il les dédie, et dynamiquement à quelles périodes il le fait. En dehors de l'aspect statique de la déclaration de préférence, la solution est conceptuellement très proche de ce que nous recherchons, et son succès démontre l'importance de prendre en compte les intérêts des participants.

Une autre approche consiste à laisser les participants communiquer directement entre eux pour négocier, pour faire valoir leurs intérêts, et au final faire émerger une allocation. Les protocoles utilisés alors sont souvent plus coûteux en nombre de messages, mais offrent une grande autonomie aux participants. Le "*Contract Net Protocol*" [Smi80], proposé dans le domaine des systèmes multi-agents en est un bon exemple. Il suppose un cadre coopératif où un initiateur soumet sa requête à tous les participants pouvant y répondre. Il peut pour cela utiliser les services d'un Matchmaker [WS00] ou celui d'un service vendeur [OMG96, WW64] ou d'un service de notification [OMG04, WW63], etc configuré dans cet objectif. En l'absence de solution plus efficace, il peut envoyer sa requête à tous les participants (broadcast, inondation). Les fournisseurs recevant la requête l'analysent, et s'ils le désirent, y répondent en précisant les conditions sous lesquelles ils acceptent de la traiter. Enfin l'initiateur fait son choix parmi les différentes propositions. D'autres protocoles sont envisageables (négociations, redistribution des requêtes, etc) mais ils nécessitent de nombreux échanges et des capacités de raisonnement et d'argumentation particulièrement développées. Qui plus est, ils présentent l'inconvénient que nous cherchons à

éviter : une requête peut ne pas être traitée pour des raisons personnelles aux fournisseurs.

Enfin, la micro-économie fournit aussi des solutions adaptées à l'autonomie des participants. En se basant sur l'hypothèse qu'un acteur est capable de comparer et d'ordonner les différentes situations qui s'offrent à lui, une des bases de la théorie économique est de supposer qu'un individu rationnel essaie de faire des choix le conduisant dans les situations qui lui conviennent le mieux, en d'autres termes qu'il préfère. Cette notion de préférence entre les situations peut être représentée par un pré-ordre total ou une fonction fournissant une valeur cardinale pour chaque situation, nommée *utilité*. Une telle fonction présente l'avantage d'être compatible avec la notion de préférence ainsi qu'avec celle de monnaie. On peut alors dire qu'un acteur économique rationnel s'efforce de maximiser son utilité personnelle [Kre90, MCWG95] dans le respect de la contrainte budgétaire définie par les revenus dont il dispose. Il faut cependant garder à l'esprit que la micro-économie étudie un système en supposant qu'il sera plongé dans l'économie "réelle". Cela signifie que les participants peuvent obtenir des ressources financières en dehors du système étudié. De même, les ressources financières obtenues dans ce système peuvent être utilisées en dehors pour tout autre besoin. Cette "universalité" qui caractérise l'argent réel facilite les échanges commerciaux. Cependant, si c'est une monnaie virtuelle qui est utilisée pour réguler le système informatique (monnaie de singe, jetons, etc), sans conversion possible avec la monnaie réelle, les conditions changent. Il devient alors indispensable de vérifier que l'argent circule bien entre les participants, qu'il n'y a pas de "puits" où l'argent s'accumule, ni de "source" d'où il jaillit sans justification. De même, il faut être attentif à l'importance donnée à l'argent virtuel qui ne peut être équivalent à celui de l'argent réel.

Le problème de l'allocation de requêtes et de tâches se prête bien à un traitement micro-économique. Le couple initiateur de requêtes / fournisseur peut facilement être plaqué sur le couple de rôles économiques classiques acheteur / vendeur pour lesquels de nombreux mécanismes ont été proposés, et étudiés, tant du point de vue individuel (rationalité individuelle, pareto optimalité,...), que d'un point de vue plus global (ex. efficacité, principe de révélation, bien être social...) [MCWG95]. L'usage de ces techniques micro-économiques dans un système informatique pour le réguler peut donc s'appuyer sur une théorie connue et de nombreux résultats.

Dans la mesure où les fournisseurs proposent leurs ressources et les initiateurs leurs requêtes, la mise en correspondance peut se faire de deux manières. Pour la première, et peut être la plus naturelle, ce sont les initiateurs qui achètent les ressources proposées par les fournisseurs (les vendeurs). Par exemple, dans le cas du système Mariposa [SAL+96, SDK+94] qui utilise des mécanismes micro-économiques pour gérer le stockage des données et le traitement des requêtes dans un système distribué, ce sont les initiateurs qui payent pour le traitement de leur requête. Au contraire, pour la seconde, les fournisseurs achètent les requêtes des initiateurs. Cette approche peut sembler moins naturelle que la précédente, mais c'est sur un principe très similaire que fonctionne Google AdWords [WW39]. Ce système propose aux gestionnaires de sites d'information d'acheter, non pas les requêtes directement, mais des mots-clés. Ils obtiennent ainsi des requêtes contenant les mots achetés qui, ils l'espèrent, correspondent à ce qu'ils souhaitent traiter. Les sommes correspondantes ne sont pas versées aux initiateurs de requêtes, mais à l'intermédiaire qu'est Google. Comme le montre cet exemple, l'introduction d'un intermédiaire entre initiateurs de requêtes et fournisseurs d'informations n'impose pas nécessairement qu'il joue un rôle de commissaire priseur. Des liens économiques peuvent être tissés entre lui et l'une des deux parties en laissant l'autre en dehors de la sphère économique (que l'argent utilisé soit réel ou virtuel).

Les ventes aux enchères, dont les propriétés sont bien étudiées en économie et qui fournissent des protocoles clairs et précis, sont adaptées à la régulation de systèmes informatiques. Par exemple, le projet de librairie digitale de l'Université du Michigan, *UMDL* [DMP+99], a exploré très tôt l'utilisation des ventes aux enchères pour traiter les demandes de documents. Cependant, elles permettent aux fournisseurs de ne traiter que les requêtes qui les intéressent. De plus, les ventes aux enchères ne permettent par elles-mêmes aucun contrôle sur le comportement global du système. Notre étude bibliographique [LLCV07, Lem07], a été centrée sur les travaux qui proposent des mécanismes ayant pour objectif explicite la prise en compte des intérêts des participants [DRJ04, GN05, LS03, PRST02, PST04, SAL+96]. Nous avons montré qu'aucun de ces mécanismes ne satisfait toutes les contraintes de nos objectifs [Lem07]. En effet, notre problématique nous confronte à des situations très différentes. D'un côté, plusieurs participants peuvent souhaiter obtenir la même chose. C'est un cas compétitif où le problème

est de les départager. Une approche économique utilise classiquement la quantité d'argent que chacun peut et est prêt à dépenser pour résoudre le problème. D'un autre côté, aucun fournisseur peut ne vouloir traiter la requête. Pour qu'elle soit tout de même traitée, l'argent peut servir à dédommager un participant sollicité contre sa volonté (imposition, réquisition ou expropriation dans la réalité).

3.3 Approche

Pour conserver toute sa généralité à la solution, aucune hypothèse n'est faite sur le domaine d'application ni sur le langage utilisé pour exprimer les requêtes.

Pour imposer les requêtes qu'aucun fournisseur de veut traiter pour des raisons de stratégie individuelle, alors que les ressources suffisantes existent, une autorité est nécessaire. Nous proposons de l'incarner en introduisant un rôle de médiateur. Pour que les agents autonomes l'acceptent, il est important que ce médiateur prenne une dimension sociale en tenant compte des intérêts des uns et des autres et en ne favorisant personne, que ce soit volontairement ou involontairement. Un objectif caractérisant notre approche est donc aussi de tenir compte de manière équilibrée à la fois des intérêts des initiateurs et de celui des fournisseurs.

Les relations entre les initiateurs de requête et le médiateur sont assez simples. Un initiateur souhaite que sa requête soit traitée par les meilleurs fournisseurs. Cette notion de "meilleur" peut être personnelle ou le résultat des expériences de plusieurs participants, i.e. une réputation. Le rôle du médiateur sera alors de faire en sorte qu'une requête soit traitée par des fournisseurs de la meilleure qualité possible en tenant aussi compte des intérêts des fournisseurs. Les liens entre les fournisseurs et le médiateur sont plus complexes. En effet, le rôle du médiateur peut être différent suivant que les fournisseurs sont en compétition pour obtenir une requête ou que aucun ne souhaite la traiter. Dans le premier cas, il joue un rôle d'arbitre, dans le second, il doit imposer la requête à certains fournisseurs. Il est important que cette imposition ne soit pas perçue par les fournisseurs impliqués comme une punition, mais comme un service rendu, "à charge de revanche". Pour régir les liens entre fournisseurs et médiateur, nous avons donc opté pour une approche monétaire à base d'argent virtuel qui permet de rémunérer un fournisseur au-

quel une tâche est imposée. En faisant en sorte qu'en mode compétitif les fournisseurs achètent les requêtes, cette rémunération permet d'augmenter la capacité d'achat du fournisseur réquisitionné et ainsi d'augmenter sa compétitivité par rapport à ses concurrents dans les compétitions futures auxquelles il participera. Une originalité de l'approche est de limiter l'utilisation de l'argent aux relations entre médiateur et fournisseurs. L'avantage de n'avoir pas à se préoccuper de doter les initiateurs de requêtes en argent, ni de définir comment ils en acquièrent au fil du temps.

En résumé, pour déterminer à quels fournisseurs la requête va être allouée, le médiateur doit disposer d'informations sur les intérêts des uns et des autres pour pouvoir en tenir compte.

- L'intérêt de l'initiateur est représenté par des valeurs numériques de "qualité" associées aux différents fournisseurs.
- Les fournisseurs expriment leur intérêt pour une requête via une offre monétaire. S'il souhaite traiter cette requête, l'offre d'un fournisseur est positive, dans la limite de sa capacité de paiement. Au contraire, si un fournisseur ne souhaite pas traiter la requête, il fait une offre négative correspondant au coût de sa réquisition.

Pour une requête donnée, le médiateur intègre les informations précédentes pour classer les fournisseurs. Grâce à un paramètre de la procédure de médiation, il est possible de considérer les intérêts de l'initiateur de la requête au même titre que ceux de fournisseurs ou au contraire de favoriser plus ou moins l'un de ces deux partis. Cela permet d'adapter le comportement de la médiation à différentes situations. Quelle que soit la valeur de ce paramètre, les fournisseurs ne désirant pas traiter la requête (offre monétaire négative) sont systématiquement positionnés en fin de classement pour ne les solliciter qu'en dernier recours. Sont sélectionnés pour traiter la requête les fournisseurs en tête de ce classement. Leur nombre est précisé dans la requête.

Une fois les fournisseurs sélectionnés, le médiateur facture les fournisseurs en les informant du résultat de l'allocation. Si tous les fournisseurs sélectionnés souhaitent traiter la requête, nous sommes dans une situation de compétition. Ces fournisseurs et seulement ceux-là sont facturés en utilisant le principe d'une vente aux enchères Vickrey généralisée [MCWG95, Yok08] qui consiste à calculer les factures en fonction des fournisseurs non sélectionnés de plus haut rang. Cette technique encourage les fournisseurs à révéler leurs offres réelles et à éviter les offres

fantaisistes ou stratégiques. Cette méthode ne permet pas au médiateur de maximiser ses gains, ce qui n'est pas ici un inconvénient dans la mesure où l'argent est virtuel et où contrairement à l'hypothèse habituelle d'une approche économique, l'objectif du médiateur n'est pas de maximiser ses gains, mais de réguler le système (l'argent virtuel n'est qu'un outil qui n'est utilisable que dans ce cadre). Le cas d'une imposition est concrétisé par la sélection d'au moins un fournisseur ayant fait une offre négative. Le médiateur suit alors une politique d'imposition [PST04] dont le principe consiste à dédommager financièrement le fournisseur réquisitionné en prélevant une somme auprès de tous les fournisseurs ayant les capacités à traiter la requête (qu'ils soient ou non sélectionnés). Le dédommagement alloué est inférieur ou égal à celui prélevé. De manière similaire aux enchères Vickrey généralisées, ces deux montants (prélevé ou alloué) sont calculés en fonction des fournisseurs non sélectionnés de plus haut rang, ce qui encourage là aussi les fournisseurs à éviter d'effectuer des offres négatives fantaisistes pour éviter d'être sélectionnés, car cela pourrait les entraîner à payer de fortes sommes. A noter que cette technique ne fait jamais perdre d'argent au médiateur. En recevant une compensation financière alors que ses collègues ont dû verser de l'argent, le fournisseur réquisitionné voit ainsi sa compétitivité augmenter : les capacités de paiement des autres ont été modifiées pour améliorer sa capacité à obtenir les requêtes qu'il souhaite lors d'une future compétition. Il est aussi très important de noter que les montants des factures, que ce soit un mode compétitif ou réquisition, sont calculés non seulement en fonction des offres monétaires effectuées par les fournisseurs mais aussi de leur évaluation par l'initiateur. L'objectif est de favoriser les fournisseurs jugés favorablement par les initiateurs de requêtes. Deux fournisseurs peuvent avoir fait la même offre et pourtant être facturés différemment pour la même requête. Celui qui est jugé le plus favorablement par les initiateurs déboursera moins que l'autre. Ainsi, à capacité financières égales, un fournisseur peut acquérir d'autant plus de requêtes qu'il est apprécié des initiateurs.

Un fournisseur peut signaler qu'il ne souhaite pas traiter une requête en effectuant une offre négative. Cependant, même dans ce cas, il peut être soumis à une réquisition. Dans certaines conditions, par exemple lorsqu'il est en surcharge, un fournisseur peut souhaiter ne plus recevoir de requête pendant un certain temps. Il peut alors se mettre en "pause". Aucune requête ne lui sera proposée tant qu'il sera dans cet état. Il n'est pas pour autant exempté de traitement pour

les requêtes qui lui ont été allouées ni de réponse pour les requêtes qui lui ont été proposées avant qu'il n'entre dans cet état et pour lesquelles il peut encore être réquisitionné. En d'autres termes, cela ne peut être une échappatoire pour éviter de traiter une requête non souhaitée.

Le système utilisant une monnaie virtuelle sans possibilité de conversion, les participants ont donc pour seules ressources financières celles du système, et ne peuvent utiliser leur argent que dans le système. Tout fournisseur intégrant le système pour la première fois est doté d'une somme d'argent fixée. Au départ d'un fournisseur, le médiateur ajuste le volume monétaire en circulation de sorte qu'il soit toujours proportionnel au nombre de fournisseurs effectivement présents. Cependant, le mode de facturation utilisé fait que la classe des fournisseurs ne peut gagner d'argent alors que le médiateur ne peut en perdre. Il en résulte que le médiateur devient une sorte de puits d'où l'argent qu'il obtient ne peut sortir. Pour éviter un blocage par pénurie d'argent, le médiateur redistribue l'argent qu'il obtient en le répartissant à parts égales entre les fournisseurs.

Enfin, cette solution nécessite évidemment des échanges entre les participants. Pour limiter au maximum la charge réseau, nous avons introduit la notion de représentant. Chaque fournisseur dispose d'un représentant situé sur le même site que le médiateur. Ainsi, médiateur et représentants peuvent communiquer sans utiliser le réseau. Un représentant (éventuellement partiellement ou totalement constitué de code mobile) met en œuvre la politique d'offre du fournisseur qu'il représente. Le nombre de messages nécessaires pour une allocation devient alors moins important que pour une vente aux enchères. De plus, le médiateur étant celui qui bat monnaie, aucune communication avec un tiers de confiance n'est nécessaire à la gestion de l'argent. Si les échanges entre un représentant et son fournisseur sont peu fréquents, le nombre de messages est alors comparable à celui nécessaire à une allocation par un Matchmaker ou un Broker.

En résumé, cette solution permet à un initiateur de requête de préciser le nombre de fournisseurs qui vont traiter sa requête, mais aussi d'exprimer son offre sur ces fournisseurs. De plus, toutes les requêtes pour lesquelles le système dispose des ressources nécessaires sont traitées. De leurs côté, les fournisseurs peuvent s'exprimer sur les requêtes qui leurs sont proposées. Pour cela ils ont individuellement le choix des critères (coût, charge, préférence, etc) et de

leur politique. Le médiateur tient compte des opinions exprimées par les uns et par les autres et, lorsque c'est nécessaire, il peut imposer une requête à un fournisseur. Un mécanisme de dédommagement est alors mis en place.

3.4 Validation

Nos validations ont été menées dans deux directions différentes. La première a consisté à étudier les propriétés théoriques de l'approche proposée. La deuxième a consisté à la programmer et l'évaluer dans un simulateur pour la confronter à des solutions différentes.

Nous avons pu montrer que cette procédure est paréto-optimale, c'est-à-dire qu'il n'existe pas d'autre allocation telle qu'au moins un participant soit mieux (en terme d'utilité) et que les autres ne soient pas moins bien. La procédure est aussi "*incentive compatible*", c'est-à-dire qu'elle incite les fournisseurs à faire des offres correspondant à leur estimation réelle, sauf dans le cas où une réquisition est réalisée alors qu'un fournisseur a fait une offre positive. Dans ce cas, le fournisseur ayant fait une offre positive pourrait obtenir un gain supérieur en faisant une offre négative, mais suffisamment importante pour rester sélectionné. Si les fournisseurs ne s'échangent pas leurs offres avant de les transmettre au médiateur, il est très difficile que l'un d'entre eux détecte a priori une telle situation. Enfin, nous avons montré que la propriété de "*rationalité individuelle*" est vérifiée en cas de compétition mais pas en cas de réquisition. En d'autres termes, cela signifie qu'un fournisseur en intégrant le système n'a pas la certitude d'avoir une utilité supérieure à celle qu'il aurait en ne l'intégrant pas. En particulier, lorsqu'une requête lui est imposée son utilité peut être négative, alors que s'il n'avait pas participé, elle serait restée nulle. Pour convaincre un fournisseur de rester dans le système, il est donc important de limiter le nombre de requêtes qui lui sont imposées contre son gré. Toutes les propriétés énoncées jusqu'à présent ont en commun de s'intéresser au résultat de la médiation sur un tour. Nos derniers commentaires sur la rationalité individuelle montrent pourtant que cela est insuffisant. Il est important de savoir si l'inconvénient d'être réquisitionné peut être compensé sur les tours suivants, et donc, si le mécanisme de compensation fonctionne. Nous avons introduit une nouvelle propriété que nous avons appelée "*rationalité individuelle à long terme*". Nous avons

étudié cette propriété dans le cas général, en particulier pour déterminer, sous certaines hypothèses, la taille de mémoire nécessaire pour qu'un fournisseur puisse se rendre compte qu'il est au final gagnant. Mais en pratique, le résultat est très dépendant de l'arrivée des requêtes. En particulier, si aucune requête ne correspond aux attentes d'un fournisseur, quel que soit le nombre de médiations dont il est capable de se rappeler, il est cohérent que sa participation au système soit peu rationnelle.

Pour conduire des validations expérimentales, nous avons programmé un environnement ad hoc avant d'utiliser SimJava. Une première série d'expérimentations a permis de vérifier le comportement de notre proposition sur un tour. Une seconde série a été conduite sur plusieurs tours. Nous avons choisi la répartition de charge comme cadre d'application et paramétré les différents comportements individuels : un initiateur de requête ne s'intéresse qu'au temps de réponse constaté et fait des retours de qualité correspondant, un fournisseur calcule son offre en fonction de sa charge, de ses préférences (bien que tenant compte principalement de sa charge, il peut préférer certaines requêtes à d'autres) et du solde dont il dispose. Notre proposition a été comparée à deux approches : allocation par maximisation de l'utilité globale et allocation par répartition de charge. Dans les deux cas, notre proposition a montré qu'elle permettait d'éviter certains problèmes tels que la famine. En effet, elle évite que les requêtes soient envoyées systématiquement aux mêmes fournisseurs au cours des différentes médiations. Ce comportement offre une chance aux fournisseurs de faible qualité de montrer qu'ils se sont améliorés sans pour autant dégrader gravement les performances du système. Les conséquences de ce comportement sont d'autant plus sensibles que la charge est faible (moins de 50% de la capacité totale du système). Lorsque la charge augmente, quelle que soit la technique considérée, il est nécessaire d'utiliser toutes les ressources du système. Les résultats ont donc tendance à se lisser. Ce dernier point est lui aussi intéressant à noter. Notre proposition ne dégrade pas les résultats en cas de charge importante du système. Plus intéressant encore, sous l'hypothèse que les fournisseurs quittent le système lorsqu'il n'est plus rationnel pour eux d'y rester (i.e. lorsque le système leur impose un trop grand nombre de requêtes qu'ils auraient refusées, par exemple parce que trop chargés, ou travaillant sur des requêtes qui ne les intéressent pas), l'avantage revient à notre proposition. En effet, en sollicitant de manière systématique les fournisseurs les plus efficaces,

l'allocation par répartition de charge finit par les dissatisfaire et donc les conduit à quitter le système. Par contre, notre approche, en tenant compte de leurs intentions, évite leur départ. En maintenant dans le système des fournisseurs utiles elle permet d'obtenir de meilleures performances.

3.5 Contributions

Nous avons montré la possibilité de construire un système où les participants influencent eux-mêmes la technique de d'allocation en s'appropriant en grande partie les critères qu'elle utilise. Cela leur offre un service adaptatif qui tient compte de leurs intérêts sans pour autant leur laisser tout contrôle sur le système. La mise en place d'une autorité de régulation permet entre autre d'imposer le traitement de certaines requêtes qui resteraient non traitées bien que les compétences nécessaires soient présentes dans le système. L'imposition de certaines requêtes est d'autant plus tolérable par les fournisseurs que la procédure d'allocation est en permanence influencée par eux.

Nos travaux [LC03a, LC03b, Lem03, LLC03, LCLV04b, LLCV07, Lem07] ont permis de développer une meilleure vision et une meilleure compréhension des possibilités que l'on peut offrir aux participants d'un système distribué ouvert lors de l'allocation de requêtes. Nous avons proposé une approche originale utilisant les services d'un médiateur personnifiant l'autorité dans le système. Les choix d'allocations réalisés par ce dernier s'établissent uniquement en fonction des intérêts exprimés par les participants. Cela permet par exemple d'éviter les phénomènes de famine (i.e. un fournisseur ne recevant jamais de requête) et plus généralement d'assurer un meilleur comportement social du médiateur. Cette approche bien qu'autoritaire, n'est pas celle du client/serveur classique, car les intérêts des participants sont pris en compte, mais elle ne cède pas non plus tout contrôle aux participants. Elle constitue donc une solution intermédiaire entre une approche client/serveur et une approche où les participants sont entièrement libres de leurs actions.

La généricité de cette approche est particulièrement importante car elle n'est conditionnée par aucune hypothèse sur le domaine d'application, pas plus que sur le langage de représentation

des requêtes. Plus intéressant encore, en fondant ses décisions sur les intérêts exprimés des participants, le médiateur s'adapte naturellement et instantanément à leurs besoins. Par exemple, dans un cadre où les initiateurs de requêtes ne s'intéressent qu'au temps de réponse et où les fournisseurs font particulièrement attention à leur charge, le médiateur offrira un service très similaire à celui d'une solution dédiée à la répartition de charge. Si les initiateurs s'intéressent à la qualité des réponses (et plus seulement au temps de réponse) et les fournisseurs s'intéressent plus au type de requête qu'ils traitent que de leur charge, la répartition de charge n'est plus une réponse adaptée à leurs besoins. Au contraire, la médiation flexible s'adaptera naturellement sans que le médiateur n'ait à changer quoi que ce soit. Cette capacité d'adaptation présente un intérêt majeur pour un système distribué ouvert où les intérêts des participants peuvent évoluer très rapidement (évolution naturelle ou changement de participants).

Pour limiter au maximum les échanges entre les participants et le médiateur, nous avons proposé une architecture particulière où les fournisseurs disposent de représentants auprès du médiateur. Un représentant fait valoir les intérêts de son fournisseur auprès du médiateur. Bien que plus importants qu'en mode client/serveur, les échanges sur le réseau sont ainsi contenus et un peu moins importants que ce qui est nécessaire à la mise en oeuvre d'une vente aux enchères.

Nous avons estimé les potentialités de cette approche en la confrontant au problème réel de la répartition de charge dans un cadre où les initiateurs s'intéressent au temps de réponse, et les fournisseurs font attention à leur charge, mais en prêtant tout de même attention aux requêtes qu'ils traitent. Ces simulations ont permis de montrer que le comportement est bien celui attendu avec des performances techniques très proches de celle d'une solution de répartition de charge mais en apportant une amélioration notable en matière de satisfaction des fournisseurs quant aux requêtes qu'ils ont à traiter.

3.6 Leçons et perspectives

La micro-économie permet de définir des mécanismes d'allocation. Néanmoins, ils sont en général prévus pour être plongés dans l'économie réelle soumises aux lois de la macro-économie. Pour qu'ils puissent avoir un comportement similaire dans un système distribué uti-

lisant de l'argent virtuel comme moyen de régulation, il est indispensable de prendre aussi en compte les aspects macro-économiques, tels que la circulation générale de l'argent au sein du système.

Pour comparer différentes techniques d'allocation (économiques ou non), il est nécessaire de caractériser plus finement les situations auxquelles la médiation est confrontée et leurs difficultés. Il est par exemple facile de réaliser une médiation lorsque les requêtes émises correspondent parfaitement aux attentes des fournisseurs. La comparaison de la pertinence et de l'efficacité d'une médiation nécessite d'approfondir à la fois les notions de satisfaction des participants et d'équité. De manière générale, il faut être attentif aux propriétés économiques de l'allocation réellement significatives dans le cadre choisi, mais ne pas se restreindre aux seules propriétés économiques. En effet, nous avons le sentiment que la prise en compte de l'argent dans la notion d'utilité, comme définie en économie, peut être très éloignée de la satisfaction réelle d'un participant dans la mesure où cet argent est virtuel. Il est donc important de réfléchir à une notion de satisfaction plus générale. De plus, nos retours d'expériences suggèrent de considérer des propriétés caractérisées sur le long terme et pas seulement sur une seule allocation.

Enfin, bien que cette technique ait été étudiée dans un cadre mono-médiateur, le passage à l'échelle semble possible. Deux solutions sont envisageables. La première consiste simplement à introduire dans le système plusieurs médiateurs sans aucun lien entre eux. Ils sont mis en concurrence par les participants qui peuvent passer de l'un à l'autre. De premières simulations ont permis de mettre en évidence certains phénomènes d'auto-organisation tirant partie de cette indépendance entre les médiateurs. Les participants ayant des points en commun se regroupent naturellement autour d'un même médiateur. Une deuxième solution consiste à fédérer les différents médiateurs simplement en leur faisant utiliser la même monnaie virtuelle. Cette solution est très naturelle et tend à donner les mêmes résultats qu'en mode centralisé.

CHAPITRE 4

Modélisation d'un système de médiation ouvert avec participants autonomes

Résumé : *La prise en compte de l'autonomie des participants dans le problème de l'allocation de requêtes ouvre plus de questions que nous ne l'avions initialement envisagé. Nous présentons ici une réflexion plus aboutie concernant la caractérisation du problème. Tout d'abord, nous proposons une approche formelle permettant de caractériser à la fois les participants dans l'environnement du système et le système de médiation lui même, tant du point de vue global que de celui d'un participant. Ces réflexions nous ont alors conduits à proposer une solution de médiation basée sur les notions introduites, et plus particulièrement sur celle de satisfaction des participants.*

Ces travaux ont été menés dans le cadre de l'équipe *ATLAS-GDD* du LINA et de Équipe Projet INRIA *ATLAS*, sous la direction de Patrick Valduriez.

Ils ont donné lieu à l'encadrement académique :

- Jorge-Arnulfo Quiane-Ruiz, "Allocation de requêtes dans des systèmes d'information distribués avec des participants autonomes", Doctorat, 2004-2008.

  Actuellement en poste à l'Université de Saarland - Sarbrüken, Allemagne.

Ils ont été partiellement supportés grâce à la participation aux projets :

- *Respire* [WWW 46], projet ARA Masse de données
- *Grid4All* [WWW 45], Projet Européen STREP.

4.1 Problème et objectifs

Comme pour le chapitre précédent, nous nous intéressons à la problématique de l'allocation de requête en poursuivant l'objectif de définir un processus très général qui s'adapte aux nombreuses situations introduites par l'autonomie des participants.

Les leçons tirées de l'expérience présentée au chapitre précédent nous suggèrent de 1 - mieux caractériser les situations auxquelles le processus d'allocation est confronté, en particulier en termes de complémentarité entre les initiateurs de requêtes et les fournisseurs ; et 2 - mieux définir les critères permettant d'évaluer les différentes solutions comme par exemple la satisfaction des participants sur le long terme, l'équité et l'effort fait par le processus d'allocation en faveur des participants.

Ces différents points doivent permettre de caractériser et de mieux comparer les différentes méthodes dans le cadre des systèmes distribués ouverts. Cela peut aussi conduire à une allocation mieux adaptée.

4.2 Approche proposée

Une partie de l'état de l'art est déjà présentée dans le chapitre précédent. Concernant la caractérisation de participants autonomes dans un système d'allocation de requêtes, à notre connaissance, nous sommes les premiers à explorer ce problème. En ce qui concerne les critères d'évaluation, les références sont présentées au fur et à mesure de la présentation de l'approche.

Nos travaux ont donc porté dans un premier temps sur la caractérisation d'un processus d'allocation et de ses participants. Ces recherches nous ont conduits, dans un deuxième temps, à proposer un nouveau processus d'allocation basé sur les notions introduites.

4.2.1 Caractériser une méthode d'allocation et ses participants

L'autonomie des participants conduit naturellement à considérer que chacun a ses propres ***préférences*** que nous souhaitons prendre en compte dans l'évaluation du processus de médiation. En micro-économie, ces préférences sont souvent représentées par une fonction d'utilité [MCWG95] (cf. chapitre 3). Cependant, nous ne faisons pas l'hypothèse que les participants communiquent au système leurs préférences, serait-ce sous forme d'une valeur d'utilité. Notre hypothèse est plus minimaliste. Les participants ne communiquent que leurs ***intentions***. Cette intention est la réponse à une question fermée qui diffère suivant le rôle des participants. Pour un fournisseur, cette question est : "*voulez-vous traiter cette requête ?*". La question posée à l'initiateur de requête est différente, "*Voulez-vous que votre requête soit traitée par ce fournisseur ?*", mais le principe reste le même. Les intentions résultant de la confrontation de leurs préférences (ce qu'ils souhaitent faire dans l'absolu) avec leur état (ex. très chargé), les participants ne révèlent pas directement leurs préférences. Enfin, l'intervalle de réponse peut être normé ($[-1..1]$), ce qui simplifie les comparaisons.

Chacun des participants, quel que soit son rôle, émet des intentions vis-à-vis des requêtes qui le concernent. Un initiateur exprime son intention de voir sa requête traitée par tel ou tel fournisseur. Pour une requête, un fournisseur émet son intention de la traiter. Les outils nécessaires pour déterminer l'*adéquation* des participants sont là. Cependant, nous avons identifié plusieurs notions d'adéquation. Pour illustrer la première, considérons le cas d'un participant qui n'émet que des intentions négatives sur tout ce qui lui est proposé. Une interprétation naturelle est qu'il ne trouve pas dans le système ce qui lui convient, en d'autres termes, que le ***système est inadéquat*** pour lui. Intuitivement, ***l'adéquation du système à un participant*** se mesure en considérant les intentions émises par ce participant vis-à-vis de ce que le système lui propose. Considérons maintenant un participant qui n'obtient que des intentions négatives de la part des autres participants vis-à-vis de ce qu'il propose. Il n'est pas difficile d'en conclure que ce participant ne présente pas d'intérêt majeur pour son environnement, qu'il est inadéquat par rapport au système. Intuitivement, ***l'adéquation d'un participant au système*** se mesure en ana-

lysant les intentions exprimées par les autres participants vis-à-vis de lui dans le cadre de ce qui a été proposé par le système. Il est intéressant de noter que ces deux notions ne sont pas liées. Un participant peut être adéquat par rapport au système et le système inadéquat pour lui, ou réciproquement. Que l'on considère un initiateur de requêtes ou un fournisseur, l'intuition est identique mais les définitions formelles varient légèrement pour prendre en compte les spécificités de chacun de ces rôles. Considérer ces deux notions permet déjà de clarifier l'objectif que peut avoir une technique d'allocation de requête en présence de participants autonomes. Plus exactement, son objectif n'est pas de conserver systématiquement tous les participants dans le système. Par exemple, le départ d'un participant inadéquat à tous points de vue est parfaitement justifié et la technique d'allocation ne doit pas s'y opposer de manière trop insistante. La gestion des participants inadéquats est donc source de difficulté pour la médiation.

Outre l'adéquation, nous nous intéressons aussi à la satisfaction que retire chaque participant du système. En considérant les intentions émises par les participants et la décision prise lors d'une allocation de requête, il est possible d'évaluer cette satisfaction. Par exemple, un fournisseur peut être supposé d'autant plus satisfait d'obtenir une requête que l'intention de l'obtenir qu'il a exprimée était forte. Cette notion est normalisée pour tous les participants sur l'intervalle $[0..1]$. Pour tenir compte des leçons tirées lors de l'analyse de la médiation flexible et obtenir une évaluation du processus sur le long terme, nous mesurons la satisfaction d'un participant en considérant plusieurs allocations successives. Chaque participant peut avoir une mémoire plus ou moins importante lui permettant de conserver les résultats d'un plus ou moins grand nombre de médiations. Cela devient alors une composante de son "caractère" : un participant ayant peu de mémoire peut être dissastisfait à la moindre contrariété et avoir des réactions assez vives, par exemple en quittant le système. Au contraire un participant avec beaucoup de mémoire a un comportement plus posé. Quelles que soient les différences entre les participants, si le médiateur souhaite prendre en compte leur satisfaction de manière équitable, il doit le faire avec la même taille mémoire pour tous. La satisfaction ainsi obtenue est plus précisément appelée ***satisfaction par rapport au système***. Une deuxième notion de satisfaction peut être considérée. Elle concerne la ***satisfaction du participant vis-à-vis du mécanisme d'allocation***. Cette dernière permet d'évaluer l'effort réalisé par la méthode d'allocation en faveur d'un par-

ticipant en confrontant la satisfaction du participant par rapport au système avec l'adéquation du système par rapport au participant. Une valeur supérieure à 1 dénote une allocation qui acte en faveur des intérêts du participant, au contraire, si cette valeur est inférieure à 1, le participant peut considérer qu'il est 'puni'. Un processus d'allocation peut être qualifié de neutre si le sort qu'il réserve à un participant n'est ni meilleur ni pire que ce qu'il peut attendre en moyenne dans son environnement. Il est important de noter que la manière dont l'effort de la médiation en direction d'un participant est mesuré n'est pas totalement objectif. En effet, cette mesure est réalisée sans tenir compte de l'adéquation du participant par rapport au système. C'est un autre critère, celui d'***efficacité*** qui, en tenant compte des deux notions d'adéquation, permet de mesurer objectivement cet effort. Les participants n'ayant pas accès à leur adéquation vis-à-vis du système, ils doivent se contenter de la satisfaction vis-à-vis de la médiation telle que présentée précédemment et baser leur jugement sur ce critère. La mesure de l'efficacité est donc réservée au processus de médiation ou à un observateur extérieur.

Les notions de satisfaction sont présentées ci-dessus en fonction des intentions exprimées par les participants. Cependant, rien n'empêche chaque participant de baser le calcul de sa satisfaction en utilisant ses préférences en lieu et place de ses intentions exprimées. Cette notion présente alors plus de similarité avec une utilité sur le long terme.

Nous proposons un dernier critère spécifique aux fournisseur pour caractériser la notion de ***famine***. Contrairement aux notions précédentes, la famine s'applique exclusivement aux fournisseurs et elle a une composante temporelle. Intuitivement, un fournisseur est en famine s'il ne reçoit pas assez de requêtes l'intéressant par unité de temps pour justifier sa présence dans le système. Nous proposons donc de caractériser le critère ***famine*** en sommant les intentions affichées sur toutes les requêtes allouées au fournisseur sur une période de temps donnée. Chaque fournisseur a un seuil en dessous duquel il se considère en famine. Cela signifie que le coût associé à sa présence dans le système est plus élevé que l'intérêt qu'il en retire, ce qui peut l'amener à quitter le système.

Dans la mesure où le processus d'allocation peut influer sur le sort des participants, il est important de vérifier qu'il est bien impartial. En particulier, il doit se comporter équitablement envers les participants. Pour cela, nous utilisons la mesure d'équité proposée par [JCH84] qui

est appliquée aux différentes mesures sur les participants (adéquations et satisfactions), ce qui complète les mesures plus classiques (moyenne, équart type, etc).

A noter que sous l'hypothèse où les intentions exprimées ne sont pas rendues publiques, chaque participant ne dispose pas des données nécessaires à l'évaluation de tous les critères proposés. Le tableau 4.1 indique qui peut évaluer quels critères.

propriété	**initiateur**	**fournisseur**	**médiateur**
adéquation d'un participant par rapport au système			✓[2]
adéquation du système par rapport à un participant	✓[1]	✓[1]	✓[2]
satisfaction d'un participant calculée en fonction de ses intentions	✓[1]	✓[1]	✓[2]
satisfaction d'un participant calculée en fonction de ses préférences	✓[1]	✓[1]	
satisfaction d'un participant vis-à-vis du mécanise d'allocation	✓[1]	✓[1]	✓[2]
efficacité du mécanisme d'allocation pour un participant			✓[2]
famine d'un fournisseur		✓[1]	✓[2]
mesures des propriétés du mécanisme d'allocation (moyenne, équité…)			✓

Hypothèses : préférences individuelles entièrement privées ; intentions non révélées en dehors du système d'allocation.

[1] Uniquement individuel : un participant ne peut pas calculer ce critère pour les autres participants.

[2] Pour tout les participants au système.

Table 4.1 – Différents critères et acteurs en mesure de les évaluer.

Dans bien des cas, les mesures proposées offrent des explications au départ d'un participant. Par exemple, un fournisseur peut décider de quitter le système pour cause de mauvaise satisfaction (que ce soit par rapport à ses préférences ou ses intentions), ou parce qu'il considère que le processus d'allocation le punit, ou encore parce qu'il est en situation de famine [QRLV09b, QR08].

4.2.2 *SbQA*: Satisfaction based Query Allocation

Les différentes notions présentées ci-dessus ont servi de base pour proposer un nouveau processus d'allocation de requête dont le principal objectif est la satisfaction des participants. En mémorisant l'historique des allocations sur une certaine période, un médiateur en charge de l'allocation des requêtes dispose des données nécessaires pour calculer la satisfaction de chacun. Pour une requête donnée, après avoir sélectionné les fournisseurs pouvant la traiter, il peut classer ces derniers en fonction de leurs intentions et de celles indiquées par l'initiateur de la requête. C'est sur cette idée qu'est basée la méthode d'allocation ***SbQA*** (pour ***Satisfaction based Query Allocation***).

SbQA classe les fournisseurs en fonction d'un score obtenu en considérant pour chacun 1 - son intention de traiter la requête et 2 - l'intention manifestée par l'initiateur de voir sa requête traitée par lui. Comme pour la médiation flexible, un des critères peut être considéré plus important que l'autre. Ici, c'est le point de vue du moins satisfait[1] qui est privilégié. Son poids est d'autant plus fort que la différence de satisfaction est importante. Si les deux acteurs affichent des satisfactions égales, leurs intentions sont considérées de la même manière. Les scores obtenus permettent de classer les fournisseurs. Ceux ayant obtenu les meilleurs scores sont sélectionnés.

4.3 Validation

Après avoir analysé le coût de ce nouveau processus d'allocation (complexité et nombre de messages échangés), nous avons effectué de nombreuses validations expérimentales. Cela a nécessité la mise en oeuvre de politiques à la fois pour les initiateurs de requête et les fournisseurs.

Pour quantifier son intention de traiter une requête, nous avons proposé qu'un fournisseur tienne compte de ses préférences et de sa charge de manière différente en fonction de son niveau de satisfaction. Lorsqu'il est satisfait, un fournisseur prête surtout attention à sa charge et il est plus enclin à accepter des requêtes correspondant assez peu à ses préférences. Au contraire, lorsqu'il est peu satisfait, il est très attentif à ses préférences et peut par exemple afficher une forte

[1] Satisfaction vis-à-vis du système calculée par rapport aux intentions.

intention pour l'obtention d'une requête qui correspond à ses préférences alors que sa charge le conduirait à la refuser s'il était satisfait. Côté initiateur, la politique mise en oeuvre pour émettre ses intentions tient compte à la fois de ses expériences personnelles et de la réputation des fournisseurs, en privilégiant la première dès qu'elle existe.

Nous avons comparé expérimentalement notre proposition avec une allocation basée sur la capacité des fournisseurs [MTS90, RM95, SKS92]. Elle diffère de notre proposition en cela qu'elle exige des fournisseurs qu'ils communiquent leurs taux d'utilisation et leurs capacités théoriques. Parmi les processus économiques [FNSY96, FYN88, SAL$^+$96] ayant montré une certaine capacité pour la gestion des systèmes distribués, nous avons choisi de comparer notre proposition à celle proposée dans le système Mariposa [SAL$^+$96], qui est, à notre connaissance, le seul système mettant en oeuvre un mécanisme économique pour gérer l'allocation de requêtes et la migration de données dans un système de base de données intégrant plusieurs milliers de sites.

Les résultats des simulations ont permis de montrer que notre proposition garantit un temps de réponse compris entre celui d'une technique centrée sur les performances et celui d'une approche économique. Notons qu'en rendant les participants plus soucieux des performances, les résultats s'améliorent. Concernant la satisfaction des participants, la méthode d'allocation en fonction des capacités se révèle bien moins efficace, ce qui est peu surprenant puisqu'elle ne prend pas ce critère en considération. Du point de vue des fournisseurs, l'approche type Mariposa et notre proposition font à peu près jeu égal, mais notre approche a l'avantage du point de vue des initiateurs. En résumé, comparée à Mariposa, notre proposition permet d'obtenir de meilleures performances (temps de réponse) tout en conservant un bon niveau de satisfaction des fournisseurs et en améliorant celle des initiateurs. Le comportement du système est ainsi mieux équilibré entre les deux groupes de participants. Ces résultats ont été vérifiés que les participants soient captifs ou libres de quitter le système.

Pour les expérimentations où les participants sont supposés pouvoir quitter le système à leur gré, nous avons proposé plusieurs stratégies concernant la prise de décision de départ. Les différents critères pouvant être pris en considération sont une satisfaction trop faible, une situation de famine ou encore, une charge trop importante. Dans tous les cas, les départs observés

sont moins fréquents avec SbQA qu'avec les autres techniques. Cela concorde avec l'observation précédente d'un bon niveau de satisfaction des différents participants en faveur de SbQA. SbQA n'évite cependant pas tous les départs. Une analyse des cas de départ nous a permis de constater que les départs sont pour la plupart liés à une mauvaise adéquation. SbQA n'est donc pas une technique salvatrice pour les participants inadéquats et n'entrave pas leur départ.

Enfin, nous avons comparé SbQA avec la médiation flexible présentée au paragraphe précédent [QRLCV08] en paramétrant la médiation flexible de sorte qu'elle accorde un poids équivalent aux intérêts des fournisseurs et à ceux des initiateurs. Un fournisseur calcule son offre pour traiter une requête à partir du montant d'argent dont il dispose (contrainte de l'approche monétaire) et de son intention de la traiter. Cette intention est calculée en suivant la même stratégie dans les deux approches. Les initiateurs de requêtes mettent en œuvre la même stratégie qu'ils utilisent SbQA ou la médiation flexible. Les résultats ont révélé un léger avantage pour SbQA, mais les comportements de ces deux méthodes sont globalement identiques. SbQA et médiation flexible montrent donc des capacités a peu près équivalentes pour répondre aux besoins des participants autonomes.

4.4 Contributions

Nos contributions portent ici sur deux axes.

Le premier concerne la caractérisation 1 - des participants dans un système (adéquation, satisfaction, famine des fournisseurs), et 2 - du comportement d'un système d'allocation de requête vis-à-vis des participants (efficacité, équité) [QRLV06, QRLV07b, QRLCV07a, QR08, QRLV09b]. Nous sommes convaincus de l'importance d'introduire de telles mesures pour étudier des systèmes intégrant des participants autonomes. Ces mesures permettent de comparer des techniques très différentes (économiques, basées sur la satisfaction, basées sur les capacités, etc.) en prenant en compte les spécificités liées à l'intégration de participants autonomes.

Le deuxième axe concerne la proposition d'une méthode d'allocation des requêtes basée sur les nouvelles notions introduites. Une évaluation expérimentale de cette méthode et une comparaison avec d'autres processus dans différentes situations a permis de montrer ses avan-

tages [QRLV06, QRLV07a, QRLV07b, QRLCV07a, QRLCV07b, QR08, QRLV09b].

Ici aussi notre objectif de développer des systèmes opérationnels nous a conduit à développer un prototype [QRLV08, QRLV09a]. Il est intégré au système d'information sémantique [WWW 59] du projet STREP Grid4All [WWW 45].

4.5 Travaux en cours et perspectives

Nous cherchons à illustrer les potentialités de l'approche par satisfaction en montrant qu'elle permet de répondre à des problématiques existantes. Nous explorons actuellement deux directions.

La première, la plus avancée, concerne le problème de la réplication des requêtes dans un environnement où tous les fournisseurs donnent les mêmes réponses. Cette réplication peut avoir deux objectifs : 1 - palier les pannes de fournisseurs ; et 2 - détecter d'éventuelles réponses fantaisistes de la part de fournisseurs malicieux. Dans le premier cas, l'idée consiste à répliquer la requête sur un nombre suffisant de fournisseurs de sorte que la probabilité d'obtenir une réponse soit suffisamment élevée. Dans le second cas, l'objectif est d'avoir des réponses de différents fournisseurs pour les comparer et ainsi détecter une éventuelle incongruité. Dans les deux cas, le même dilemme se pose. Les considérations sécuritaires tendent à solliciter le plus grand nombre de réplications possible. La charge système et la satisfaction des participants peut au contraire pâtir d'un trop grand nombre de réplications. Une approche par satisfaction apporte ici une solution intéressante et naturelle. Intuitivement, une réplication est souhaitable si cela ne nuit pas à la satisfaction des participants (fournisseurs et initiateurs de requêtes confondus). Dans le cas contraire, imposer une réplication peut introduire plus de problèmes qu'elle n'en résout. Ces travaux sont menés en collaboration avec Jorge-Arnulfo Quiane-Ruiz (Université de Saarland - Sarbrüken, Allemagne).

La deuxième voie que nous explorons aussi avec Jorge-Arnulfo Quiane-Ruiz consiste à utiliser la notion de satisfaction pour gérer des structures de données distribuées similaires aux *DHT (Distributed Hash Tables)*. Ces structures sont actuellement très normatives, ne laissant aucune autonomie aux participants qui la constituent. Cette approche convient parfaitement à

bon nombre d'applications, en particulier lorsque les participants n'ont pas accès aux informations qu'ils stockent. Par contre, lorsque les informations sont publiques, les participants peuvent être plus ou moins intéressés pour stocker telle ou telle information. Une allocation des documents à stocker en fonction des intentions des uns et des autres pourrait être une solution apportant plus de souplesse.

Outre l'application de la notion de satisfaction à certains problèmes, nous travaillons aussi sur le problème du passage à l'échelle de cette solution. Pour cela, il est nécessaire de sortir de l'approche mono-médiateur pour éviter le goulot d'étranglement que représente un médiateur unique. Nous nous sommes pour l'instant plus particulièrement intéressés à l'approche approche économique [QRLCV08]. La monnaie évite aux différents médiateurs de s'échanger les informations à propos des résultats des différentes médiations (qui est satisfait et qui ne l'est pas). D'autres approches méritent d'être étudiées et comparées.

L'étude des stratégies individuelles mises en œuvre par les participants à l'intérieur du système mérite aussi une attention particulière. Nous avons jusqu'ici fait l'hypothèse que les participants sont honnêtes tant dans le calcul de leurs intentions que dans celui de leurs offres. Cependant, sous certaines conditions, un fournisseur a tout intérêt à communiquer une intention très négative (-1) dès qu'il ne souhaite pas traiter la requête. Cette stratégie n'est pas toujours gagnante, mais la question de l'existence de stratégies dominantes, d'équilibres de Nash, etc reste ouverte. Cette problématique mérite donc d'être analysée avec une approche "théorie des jeux".

Les notions de satisfaction et d'adéquation telles que proposées ici sont relatives au système dans son ensemble. Par exemple, un fournisseur peut être considéré comme inadéquat vis-à-vis du système alors qu'un certain nombre d'initiateurs le considèrent particulièrement intéressant. L'inadéquation provient seulement de la faiblesse de ce nombre par rapport au nombre total d'initiateurs. Cependant, il peut être assez grand pour garantir au fournisseur en question une niche lui permettant de travailler et d'être satisfait. En rendant ces propriétés plus spécifiques, par exemple en parlant d'adéquation d'un fournisseur pour un type particulier de requête par rapport à un sous-groupe d'initiateurs, nous pensons qu'il serait possible d'avoir une analyse plus fine d'un système. Les participants et le système de médiation pourrait en tirer partie d'une

meilleure compréhension du fonctionnement du système.

Enfin, l'intégration de participants autonomes devenant un phénomène de plus en plus répendu dans les systèmes distribués, nous pensons qu'il sera de plus en plus difficile de ne pas tenir compte des intérêts individuels des participants qui peuvent devenir de plus en plus attentifs aux conditions dans lesquelles ils intègrent un système. Ils peuvent en particulier souhaiter exercer un contrôle sur les ressources qu'ils apportent au système et ainsi espérer améliorer leur satisfaction en utilisant cette ressource comme levier. Les participants deviennent ainsi des acteurs sociaux tels qu'envisagés dans le cadre de sociologie de l'action organisée. Les modèles et outils développés dans ce cadre [SBAM05] peuvent offrir une solution pour appréhender et analyser les conséquences de cette nouvelle situation. Un axe de nos travaux futurs consiste donc à étudier les aspects sociaux des systèmes distribués ouverts. Nous avons commencé cette étude avec Christophe Sibertin-Blanc de l'Université de Toulouse I.

CHAPITRE 5

Contribution à l'intéropérabilité sémantique

Résumé : *L'autonomie et leur provenance d'horizons différents sont deux caractéristiques fréquentes des participants aux systèmes distribués ouverts. Une seule de ces caractéristiques serait suffisante pour expliquer l'hétérogénéité des participants qui est donc un problème incontournable pour ces systèmes. En présence d'hétérogénéité sémantique, les alignements entre ontologies offrent une certaine interopérabilité en proposant des correspondances entre les éléments d'ontologies différentes. Une requête exprimée en utilisant les concepts d'une ontologie peut alors être transcrite sur une autre ontologie via un alignement. Cependant, l'expressivité est réduite puisque seuls les concepts alignés peuvent être utilisés dans la communication. Notre proposition vise à améliorer l'interopérabilité en permettant un usage efficace de concepts qui n'ont pas été alignés. Cette approche est proposée dans le cadre de la recherche d'information en utilisant des vecteurs sémantiques pour représenter documents et requêtes. Elle contribue à l'autonomie sémantique des participants dans la mesure où elle leur permet d'accroître l'ensemble des concepts qu'ils peuvent utiliser pour échanger.*

Ces travaux ont été menés dans l'équipe *ATLAS-GDD* du LINA, et l'équipe projet INRIA *ATLAS*, sous la direction de Patrick Valduriez en collaboration avec Sylvie Cazalens.

Ils ont été partiellement supportés par les projets :

- *Respire* [WWW 46] projet ARA Masse de données
- *DataRing* [WWW 48] projet ANR DataRing

Encadrements :

- Pôl Uguen, *Expression de focus sur des ontologies pour traiter les demandes des utilisateurs dans un SMA*, DEA co-encadré avec Sylvie Cazalens et Alain Bidault, 2002-2003.
- Anthony Ventresque, *Focus et ontologies dans le cadre de la recherche d'informations*, DEA, 2003-2004.
- Anthony Ventresque, *Espaces vectoriels sémantiques : enrichissement et interprétation de requêtes dans un système d'information distribué et hétérogène*, Doctorat, 2005-2008. ATER à l'Université de Nantes jusqu'au 31 août 2009.

5.1 Problème et état de l'art

Un principe des systèmes distribués ouverts est de fédérer un grand nombre de participants en bénéficiant des ressources qu'ils apportent. Ces participants ne sont donc pas nécessairement créés spécifiquement pour le système qu'ils intègrent et peuvent au contraire avoir une existence préalable ou même indépendante du système lui même. Les systèmes de partage d'informations constituent une illustration parfaite dans la mesure où leur principal intérêt provient de ce qu'ils permettent d'accéder à des informations provenant de diverses sources, informations qui ne sont généralement pas créées spécifiquement pour le système. L'hétérogénéité de représentation fait donc naturellement partie du contexte d'un système distribué ouvert et l'autonomie de représentation est une nécessité pour permettre à chaque participant de mettre en œuvre celle qui lui convient le mieux.

Les modèles que peut utiliser un acteur pour conceptualiser les données varient en fonction de leur forme. Il est possible de classer les données selon trois niveaux : structurées, semi-structurées, et non structurées. Si les données sont structurées, l'usage de l'algèbre relationnelle des bases de données s'impose souvent. Cette approche offre à la fois une représentation des données efficace, un langage d'interrogation puissant (SQL) et des outils fiables, optimisés et particulièrement aboutis à la fois pour le stockage et l'interrogation. Dans le cas des données semi-structurées c'est XML [W3C98] et ses dérivés qui s'imposent actuellement avec des langages d'interrogation comme XQuery [WWW16] ou XPATH [WWW17]. Bien que plus récents que

ceux des bases de données, ils offrent déjà des solutions puissantes. Enfin, les données peuvent être non structurées, ou très faiblement structurées. C'est par exemple le cas des documents textuels. Dans ce cas, des solutions basées sur l'indexation, telles que celles proposées dans le domaine de la recherche d'information [BYRN99, BS08], restent la meilleure solution. Deux niveaux d'indexation existent : lexicale (indexation sur les termes présents dans un document), ou sémantique (indexation sur les concepts). Dans les deux cas, la technique de représentation et de requêtage est très différente des propositions précédentes. Par exemple, documents et requêtes peuvent être représentés en utilisant des vecteurs exprimés dans un espace vectoriel normé à grande dimension (une dimension par terme, respectivement concept, si l'indexation est lexicale, respectivement sémantique). Cet environnement mathématique simple est alors utilisé pour comparer les vecteurs des documents à celui d'une requête.

Il est clair que si différents participants utilisent des techniques de conceptualisation de différents niveaux, l'interopérabilité est extrêmement difficile à obtenir. C'est pourquoi en général, le problème de l'hétérogénéité est abordé en supposant que tous les participants utilisent une conceptualisation du même niveau.

Historiquement, le domaine des bases de données a été l'un des premiers à être confronté au problème de l'hétérogénéité des schémas. Une approche normative demandant à chaque base de données de se conformer au même schéma permet de résoudre ce problème trivialement. Cette approche peut être utilisée pour des systèmes fermés et bien contrôlés (ex. au sein d'une même entreprise regroupant plusieurs sites). Malheureusement, elle n'est pas réaliste, pour un système ouvert intégrant des participants autonomes. L'usage d'un schéma global reste possible, mais sans en imposer l'usage à tous les participants. Il est alors dévolu aux initiateurs qui l'utilisent pour rédiger leurs requêtes. C'est un médiateur qui réalise les opérations nécessaires pour adapter l'exécution en fonction des différentes bases disponibles [Wie92d, TRV98a, ÖV99, ÖV04]. La phase de traduction du schéma général vers ceux des bases locales s'appuie sur des adaptateurs (ou "*wrappers*"). Les réponses obtenues subissent la transformation inverse avant d'être intégrées et envoyées à l'utilisateur. Toute modification du schéma d'une source impose alors de redéfinir l'adaptateur correspondant. Or, ces adaptateurs sont le plus souvent obtenus manuellement, et les modifier est relativement long et coûteux, ce qui est peu propice à une dynamique

importante. Un autre inconvénient est dû à l'utilisation d'un médiateur. Point de passage obligé pour toutes les requêtes et tous les traitements, il constitue un goulot d'étranglement et un point de défaillance particulièrement sensible. Cette approche est efficace tant que le nombre de sources de données n'est pas trop important (quelques dizaines) [NBN99]. Utiliser un médiateur est satisfaisant pour bon nombre d'applications dont le contexte est compatible avec cette limitation (ex. [WWW51]).

Les approches pair-à-pair proposées plus récemment permettent un meilleur passage à l'échelle en évitant à la fois l'usage d'un schéma global et le passage obligé par un médiateur central. On parle alors de "*Peers Data Management Systems*". Tous les pairs sont considérés comme égaux et chacun est libre d'utiliser le schéma de son choix. Un pair peut émettre des requêtes et contribuer au système en apportant données, schémas et mappings. S'il souhaite utiliser un nouveau schéma, pour intégrer le système, un pair doit proposer les mappings nécessaires pour communiquer avec ses voisins. Cette technique de communication directe entre participants est par exemple utilisée par *Piazza* [HIMT03, IHMT03, HIM+04], ou *SomeWhere* [ACG+04, Rou06] qui permettent d'interroger des données structurées ou semi-structurées, ou des bases de connaissances représentées grâce à des ontologies.

Comparé au modèle relationnel, le modèle ontologique est plus riche et plus informatif. Il n'échappe pas au problème de l'hétérogénéité car différents fournisseurs du même domaine peuvent utiliser différentes ontologies au même titre que différents schémas. Cependant sa richesse permet d'envisager la possibilité d'obtenir automatiquement, ou semi-autmatiquement, des alignements entre les éléments de différentes ontologies. Cela représente une avancée très positive pour l'interopérabilité d'un système. De nombreux travaux de recherche sont actuellement en cours dans cette direction [DMDH04, ES04, ES07].

Cependant, quelle que soit la technique utilisée pour l'obtenir (manuelle, automatique, semi-automatique), un alignement entre ontologies ne propose généralement pas de correspondance pour tous les concepts, et les requêtes utilisant des concepts non alignés ne sont pas intégralement "traduites". Cela représente un frein important à l'interopérabilité, et donc à l'autonomie sémantique des participants qui sont contraints d'utiliser des ontologies fortement (totalement) alignables au moins sur les points concernant leurs requêtes pour que la communication soit

efficace.

Nous cherchons à améliorer cette autonomie en allant au delà de ce que permettent les alignements par eux-mêmes. Dans le cadre de la recherche d'information, l'objectif est de rendre possible l'usage pour un participant de concepts qui lui sont propres et qui ne sont pas alignés avec l'ontologie du voisin avec lequel il échange des requêtes.

5.2 Notre proposition : ExSI^{2}D

Notre proposition est prévue pour favoriser l'échange d'information direct entre deux interlocuteurs : l'émetteur de la requête (noté A dans la suite) et le fournisseur (noté B) qui doit la traiter. Elle est donc adaptée à une approche pair-à-pair, ou à un usage dans un système tel que celui présenté au chapitre 2, ou même une approche client/serveur. Elle se situe dans le cadre de la recherche d'informations non structurées et utilise les ontologies comme support. Différents modèles de représentation existent : modèle booléen, modèle vectoriel [BP98, Seb02, SWY75, BBM02, BDJ99], modèle vectoriel sémantique [Woo97, KC92], modèle probabiliste [RWHBG95], graphes conceptuels[Sow76, WW19]. Nous avons choisi d'utiliser des vecteurs sémantiques pour représenter requêtes et documents. Cela est totalement compatible avec l'indexation sémantique [DJ01b, CDJL02] que nous avons utilisée dans le cadre du projet ***B***ONOM (cf. chapitre 2). Nous pensons aussi que cette approche relativement intuitive a un potentiel important.

Le schéma 5.1 présente les différents modules utilisés. On y retrouve les trois modules classiques utilisés en recherche d'informations indépendamment de notre approche : *indexation des documents* par le fournisseur, préalablement à l'interrogation ; *indexation de la requête* par l'initiateur ; et *calcul de pertinence des documents* par le fournisseur. Ce dernier module classe les documents en fonction de leur pertinence par rapport à la requête en utilisant les représentations abstraites de la requête et des documents. Il est important de noter que notre proposition est non intrusive au sens où elle ne nécessite aucune modification des modules existants.

En présence d'hétérogénéité sémantique, c'est-à-dire lorsque les deux interlocuteurs utilisent deux ontologie différentes, Ω_A et Ω_B, la difficulté provient du fait que le module de calcul

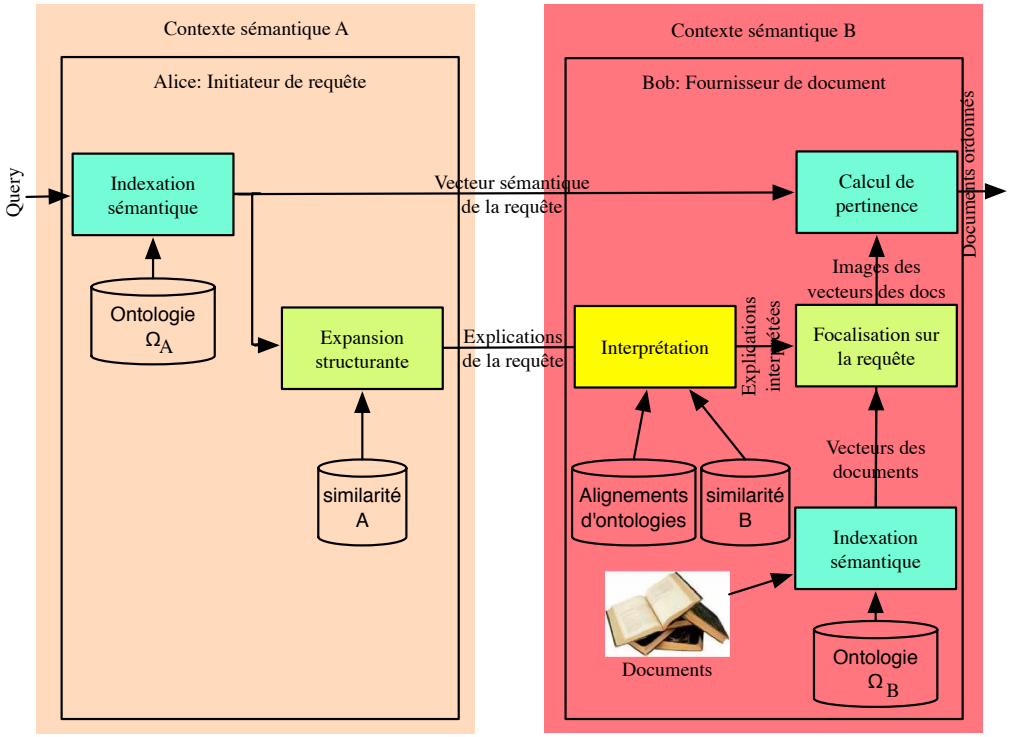


Figure 5.1 – Schéma général de notre proposition.

de la pertinence reçoit des informations à comparer qui ne sont pas exprimées dans le même espace sémantique. En l'état, elle ne sont pas comparables.

Nous avons envisagé la possibilité de construire un nouvel espace de représentation pour comparer les documents issus de contextes sémantiques différents. La fusion des ontologies des deux participants est un candidat naturel. Une fois cet espace obtenu, il ne serait pas difficile d'y représenter documents et requêtes et de les y comparer. Cependant, outre les difficultés liées au processus de fusion d'ontologies (en particulier en présence d'incohérences entre les ontologies) et le coût d'une telle opération, nous n'avons pas été convaincu par la pertinence de cette démarche. Il semble parfaitement légitime de fusionner plusieurs ontologies pour permettre aux participants d'améliorer leurs représentations et d'utiliser le résultat de cette fusion. Il est bien plus difficile de justifier la représentation d'une requête ou d'un document dans une ontologie résultat d'une fusion dont ni l'initiateur de la requête, ni le fournisseur n'ont conscience. La prise en compte de cette nouvelle représentation les aurait vraisemblablement conduits à représenter différemment requête et documents. Ce paradoxe nous a rapidement conduit à abandon-

ner cette voie.

L'usage d'un alignement entre les deux ontologies qui définit des correspondances entre leurs composants respectifs nous est apparu comme étant plus naturelle pour répondre au problème. Il est possible d'exploiter les liens mis en évidence par un alignement pour transformer un vecteur sémantique exprimé sur Ω_A en un autre vecteur sémantique exprimé sur Ω_B. Cependant, cela restreint évidemment la communication sur les parties alignées des ontologies. Cela n'est intéressant ni pour l'initiateur de la requête, qui voit sa capacité d'expression "utile" restreinte à cette "intersection", ni pour le fournisseur qui, bien qu'ayant indexé ses documents sur toute son ontologie, est frappé par le même phénomène.

Notre proposition vise donc à permettre à l'initiateur de la requête et aux fournisseurs de conserver l'usage de concepts de leurs ontologies respectives même s'ils ne sont pas partagés. Nous proposons un processus en trois étapes correspondant aux trois modules visibles sur la figure 5.1 : 1 - l'initiateur explique les concepts qu'il utilise dans sa requête en construisant une "expansion structurante" qui accompagne sa requête, 2 - le fournisseur réalise une "interprétation" de ces explications en les transcrivant dans son contexte ontologique, ce qui lui permet de 3 - adapter les représentations des documents à la requête dans le module de "focalisation". Les nouvelles représentations ainsi obtenues peuvent être comparées à la requête. C'est pourquoi la méthode a été nommée EXSI^{2}D (*Ex*pansion *S*tructurante-*I*nterprétation-*I*mage du *D*ocument, "l'***image***" d'un document étant le résultat de sa focalisation).

L'"***expansion structurante***" contient, pour chaque concept impliqué dans sa requête, une explication sous forme de vecteur qui devra être interprétée par le fournisseur. Intuitivement, l'approche par vecteur sémantique associe une dimension à chaque concept de l'ontologie. L'explication décrit quels sont les concepts assez proches du concept initial pour que, dans le contexte de cette requête, ils puissent être considérés comme faisant partie de la même dimension. C'est en quelque sorte un repli de plusieurs dimensions de l'espace vectoriel pour n'en constituer qu'une seule. Intuitivement, la manière dont une dimension est repliée dépend du degré d'intérêt que présente le concept qui lui est associé avec le concept initial. Pour chaque concept utilisé dans l'explication, une valeur numérique précise donc son degré d'intérêt par rapport au concept initial. Pour arriver à ce résultat, deux fonctions sont utilisées :

- Une ***fonction de similarité*** sim_c quantifie la similarité des concepts de Ω_A par rapport au concept c que l'on explique. Cette similarité peut être obtenue par une approche structurelle [RMBB89, LC98, Bid02], ou par contenu informationnel [Res95, SVH04]. Nous avons opté pour une variante de celle proposée par Bidault [Bid02].
- Une seconde fonction f, appelée ***fonction de propagation***, précise comment l'intérêt de l'initiateur de la requête évolue en fonction de la similarité avec le concept expliqué. Cette fonction infléchit la notion de similarité en positionnant un seuil de pertinence au delà duquel les concepts ne sont plus considérés comme assez similaires pour être associés au concept c, et en pondérant l'intérêt d'un concept en fonction sa similarité avec c. Naturellement, cette fonction est monotone croissante car, plus un concept est similaire à c, plus il est intéressant de l'associer à c. Il est important de noter que la fonction de propagation choisie peut être différente d'un concept à l'autre et d'une requête à l'autre.

L'explication d'un concept c est représentée par un vecteur sémantique appelé "***dimension sémantiquement enrichie***". Elle est obtenue par composition de ces deux fonctions : $\overrightarrow{dse}_c[c'] = f(sim_c(c'))$. Une variante consiste à prendre aussi en compte le poids du concept dans la requête : $\overrightarrow{dse}_c[c'] = \overrightarrow{q}[c] \times f(sim_c(c'))$. C'est cette dernière variante, se rapprochant plus d'une expansion, qui est utilisée dans l'article [VCLV08c] fournit en annexe 6.3. Les résultats obtenus via nos expérimentations sont très similaires d'une version à l'autre car le paramétrage de l'indexation sémantique ne permet pas de différencier ces deux versions. En effet, les pondérations non nulles du vecteur ($\overrightarrow{q}$) représentant la requête q et obtenu par indexation sémantique, sont égales à 1 ou exceptionnellement très voisines.

L'ensemble des explications (ou dimensions sémantiquement enrichies) obtenues, à raison d'une pour chaque concept pondéré de la requête, constitue son "***expansion structurante***". Elle ne modifie donc pas la requête, mais l'accompagne. Cela la différencie d'une expansion "classique" [QF93, Voo94] qui modifie le vecteur initial de la requête en y introduisant toutes les nouvelles pondérations obtenues.

La phase d'***interprétation*** ne concerne pas la requête elle-même, mais uniquement l'expansion structurante (i.e. les explications) qui l'accompagne. Par cette opération, le fournisseur exprime ces explications dans son contexte sémantique. Plus précisément, il détermine si les

concepts qui lui sont spécifiques (non alignés) présentent un intérêt pour la requête, ce qui a soulevé deux points difficiles.

- Le premier est exclusivement relatif au cas où le concept expliqué par une dimension sémantiquement enrichie n'est pas aligné. Pour le prendre tout de même en compte, le fournisseur doit en premier lieu choisir dans son ontologie un concept lui correspondant au mieux. Notre proposition pour résoudre ce problème repose sur l'intuition que l'initiateur de la requête a construit son explication (dimension sémantiquement enrichie) en respectant une propriété simple : plus un concept est similaire au concept expliqué, plus sa pondération devrait être forte. Il semble naturel que le concept choisi pour remplacer celui qui est inconnu respecte lui aussi cette propriété. En d'autres termes, une fois ordonnés en fonction de leur similarité avec le concept choisi, les concepts de la dimension sémantiquement enrichie devraient présenter des valeurs elles aussi en ordre croissant. En quantifiant à quel point un concept s'éloigne de cet idéal, nous mesurons sa capacité à prendre la place du concept initial de la requête, pour au final choisir l'un des plus adéquats.

  En pratique, le nombre de concepts de l'ontologie est bien trop important pour envisager de les vérifier tous pour choisir le meilleur. Nous avons donc introduit une heuristique permettant de ne considérer qu'un petit ensemble de concepts candidats.
- Le deuxième problème apparaît après que le concept correspondant à celui utilisé dans la requête soit "traduit" dans l'ontologie du fournisseur (en utilisant directement l'alignement, ou lorsque ce n'est pas possible, via la méthode résumée au point précédent). Il reste alors à déterminer l'intérêt par rapport à la requête des concepts propres aux fournisseurs (ceux qui ne sont pas alignés). Nous proposons pour cela d'utiliser une fonction d'interprétation, dont l'obtention est très similaire à celle de la fonction utilisée par l'initiateur de la requête pour la construction de la dimension sémantiquement enrichie. Elle est obtenue par composition de la fonction de similarité (du fournisseur) et d'une fonction de pondération. Cette dernière est une fonction affine par morceau qui est définie par les valeurs des concepts pondérés de la dimension sémantiquement enrichie fournie. Les pondérations affectées aux concepts propres dépendent donc du concept central, des valeurs

connues des concepts partagés et de la fonction de similarité utilisée par le fournisseur.

Enfin, par une "***focalisation sur la requête***", le fournisseur modifie la description de ses documents en fonction de l'interprétation des explications fournies qui établit des liens entre les concepts utilisés par l'initiateur dans la requête et ceux utilisés par le fournisseur pour l'indexation. Intuitivement, pour chaque concept de la requête, une dimension est "créée" en repliant en une seule celles des concepts considérés comme proches de ce concept (cf. dimension sémantiquement enrichie interprétée). Les autres dimensions de la requête étant toutes pondérées à zéro, il en résulte que ces modifications de l'espace sont suffisantes pour que représentations de documents et requêtes puissent y être comparées par une méthode usuelle (par exemple, le cosinus). Pour un concept de la requête, la valeur associée à sa dimension est calculée à partir des valeurs du vecteur du document et de l'intérêt exprimé dans l'interprétation de la dimension sémantiquement enrichie. Les valeurs des concepts qui ne sont pas liées à la requête ni à ses dimensions sémantiquement enrichies sont simplement recopiées. Le vecteur sémantique résultant exprimé dans ce nouvel espace de représentation est appelé "l'image" du document.

Grâce à cette approche, même s'ils ne sont pas alignés, des concepts peuvent être pris en compte dans le calcul des réponses, que ce soit pour les initiateurs ou les fournisseurs. Par exemple, il suffit qu'au moins un des concepts utilisés dans la dimension sémantiquement enrichie le décrivant soit aligné pour que le concept d'une requête soit pris en compte dans le calcul des réponses. Cela devrait rendre ce procédé robuste face à l'hétérogénéité. Cependant, plus le nombre de concepts alignés est important, plus la communication est aisée.

L'idée de replier des dimensions a déjà été exploitée pour diminuer la taille de l'espace vectoriel lors de l'indexation des documents (Latent Semantic Indexing [DDL$^+$90b]). Par rapport à notre travail, à la fois l'objectif et la méthode diffèrent : nous ne cherchons pas à diminuer l'espace pour des raisons d'optimisation, et le choix des dimensions à replier est dicté par la requête.

Nous n'entrons pas ici dans le détail des optimisations qu'il est possible de mettre en oeuvre, mais il est bon de noter qu'optimiser ce processus est possible.

5.3 Validation de l'approche

En absence de méthode explicitement concurrente, pour valider notre approche, nous avons pris le cosinus [BS08] comme méthode de référence. Son utilisation est très reconnue dans le domaine de la Recherche d'Informations pour comparer des vecteurs sémantiques. Nous avons aussi comparé nos résultats à ceux obtenus en modifiant la requête par expansion. Nous avons mené des expérimentations en environnement homogène et hétérogène.

Pour réaliser nos expérimentations, nous avons souhaité confronter notre proposition à un environnement réaliste. Pour cela, nous avons utilisé le corpus Cranfield [WWW, WWW] (1400 documents et 225 requêtes) emprunté au domaine de la recherche d'information. Pour la représentation sémantique, nous avons choisi WordNet [WWW, Fel98, MBF+90], particulièrement utilisé en RI. L'indexation sémantique est assurée de manière automatique par la technique RIIO [DJ02b] issue des travaux précédents (cf. chapitre 2). La fonction de similarité utilisée, issue des travaux de Bidault [BFS02, BFS00, Bid02], est une similarité structurelle basée sur l'étude de la relation *is-a* de l'ontologie. L'usage de ces différentes techniques permet de mener des expérimentations sans qu'aucune intervention humaine ne puisse être suspectée d'influencer les résultats.

5.3.1 Validation en environnement sémantiquement homogène

Une première série d'expérimentations a été menée en environnement homogène. Cela peut paraître surprenant pour une technique supposée s'appliquer en environnement hétérogène, mais il est important de vérifier qu'elle est utilisable en environnement homogène et qu'en particulier, elle ne dégrade pas les résultats. Si tel était le cas, son usage serait restreint aux cadres dépassant un certain degré d'hétérogénéité ce qui compliquerait son usage de manière significative.

Concernant l'expansion classique, les résultats obtenus sont au mieux ceux de la méthode de référence. Ils se dégradent rapidement lorsque le nombre de concepts ajoutés devient important.

Pour notre approche, les résultats, sans être spectaculairement différents, sont légèrement plus encourageants. Une dégradation similaire s'observe lorsque le nombre de concepts ajoutés dans les explications dépasse la vingtaine. Mais, avant cette dégradation, il est possible d'obser-

ver une légère amélioration à la fois en précision et en rappel. Bien qu'insuffisante à elle seule pour justifier de l'intérêt de cette approche, cette amélioration est encourageante. De plus, ces résultats permettent de conclure qu'en restant sur la base d'une vingtaine de concepts ajoutés au maximum, cette méthode peut être utilisée sans trop se préoccuper du degré d'hétérogénéité en ce sens qu'en moyenne, elle ne dégrade pas les résultats même en cas d'homogénéité.

5.3.2 Validation en environnement sémantiquement hétérogène.

Pour mener à bien cette validation, nous avons été confronté à un nouveau problème qui s'est révélé difficile. Intuitivement, il se résume simplement : l'hétérogénéité peut avoir différents niveaux, de très forte (voire impossible à résoudre), à triviale à résoudre. Une quantification de l'hétérogénéité présente dans les exemples servant à l'expérimentation semble alors indispensable pour évaluer une telle approche et pouvoir répondre à des interrogations du type "quelle est son efficacité en fonction du degré d'hétérogénéité ?". Plus précisément, deux mesures seraient nécessaires. La première permettrait de mesurer l'hétérogénéité entre deux environnements sémantiques distincts. La seconde devrait permettre de mesurer la difficulté de communication entre deux environnements sémantiques étant donnés les outils à disposition, comme par exemple un alignement entre les ontologies. Cette dernière devrait donc plutôt s'appeler une mesure d'interopérabilité. A noter que cette distinction n'était pas encore réalisée lorsque l'article fournit en annexe a été rédigé. La mesure d'hétérogénéité qui y est présentée correspond plus à ce que nous appelons ici mesure d'interopérabilité. A notre connaissance, il n'existe actuellement pas de telle mesure.

De plus, une autre difficulté est apparue simultanément. La réalisation de tests dans différents milieux hétérogènes nécessite de disposer d'un grand nombre d'ontologies différentes, réalistes, avec plusieurs alignements différents. L'obtention de telles données est un problème suffisamment difficile pour que nous nous soyons tournés vers une autre solution moins réaliste, mais plus réalisable.

Pour palier simultanément les deux problèmes mentionnés ci-dessus, nous avons fait le

choix de tester notre solution en considérant que les deux interlocuteurs disposent de la même ontologie et utilisent la même fonction de similarité. La différence d'interopérabilité entre les deux environnements est entièrement déterminée par l'alignement choisi. Il est alors très simple de maîtriser le niveau d'interopérabilité en supprimant de l'alignement certaines correspondances entre concepts. Les deux cas limites sont : 1 - toutes les correspondances sont présentes, le milieu peut donc être considéré comme homogène ; et 2 - toutes les correspondances sont supprimées, interopérabilité zéro : aucune communication n'est possible. Un milieu offrant 50% d'interopérabilité s'obtient en supprimant 50% des correspondances entre les concepts. Bien qu'un tel environnement ne présente pas toutes les caractéristiques de l'hétérogénéité sémantique, il a l'avantage indéniable de permettre de contrôler simplement le niveau d'interopérabilité et, comme le montrent les résultats d'expérimentations, il est suffisant pour mettre en évidence des différences de comportement entre les approches.

Nous avons donc conduit une série d'expérimentations en utilisant l'environnement sémantique décrit précédemment qui consistait, en partant d'un environnement homogène à le rendre de moins en moins interopérable en supprimant des correspondances entre les concepts (10%, 20%, 30%,...100%) pour arriver à un contexte où l'interopérabilité est nulle. Pour éviter les épi-phénomènes nous avons effectué pour chaque pourcentage considéré une dizaine de mesures sur des environnements différents (alignements supprimés aléatoirement). Par rapport aux autres approches qui perdent en précision et en rappel à peu près linéairement en fonction du nombre d'alignements supprimés, ExSI^{2}D montre une bien meilleure résistance à la perte d'interopérabilité. De plus, l'approche sans interprétation (ExSID) ne montre pas d'amélioration significative. L'apport et l'intérêt de la phase d'interprétation d'ExSI^{2}D sont donc clairement mis en évidence. La figure 5.2 présente des résultats en cours de soumission plus récents que ceux de [VCLV08c] (annexe 6.3).

Une seconde expérimentation a permis de mesurer le comportement des différentes méthodes dans le cas limite où seuls les concepts utilisés dans les requêtes sont non alignés. Les résultats présentés à la figure 5.2 montrent un très net avantage pour notre proposition.

Les résultats obtenus sont donc très bons. Il est cependant important de garder à l'esprit que dans nos expérimentations, les deux participants utilisent la même ontologie et la même fonction

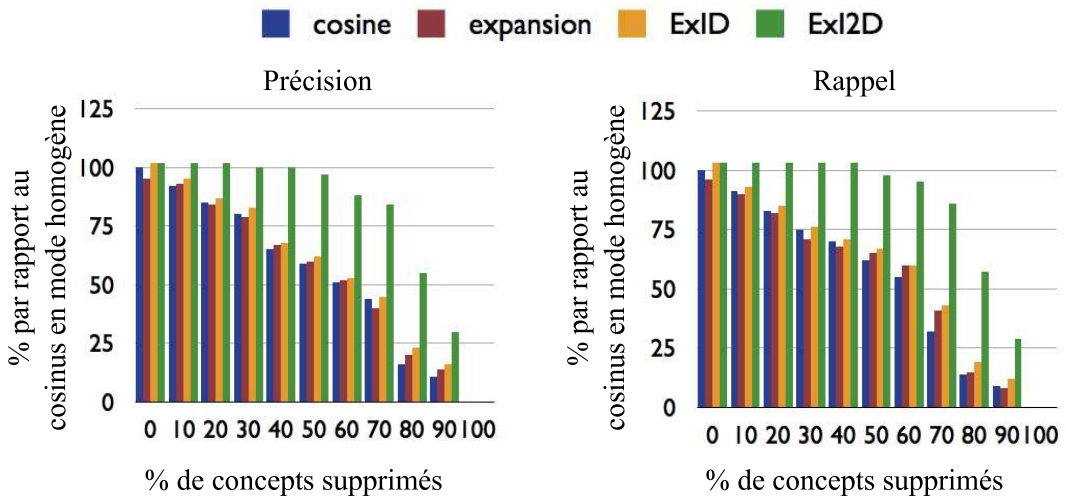


Figure 5.2 – Comportement de différentes méthodes en environnement hétérogène

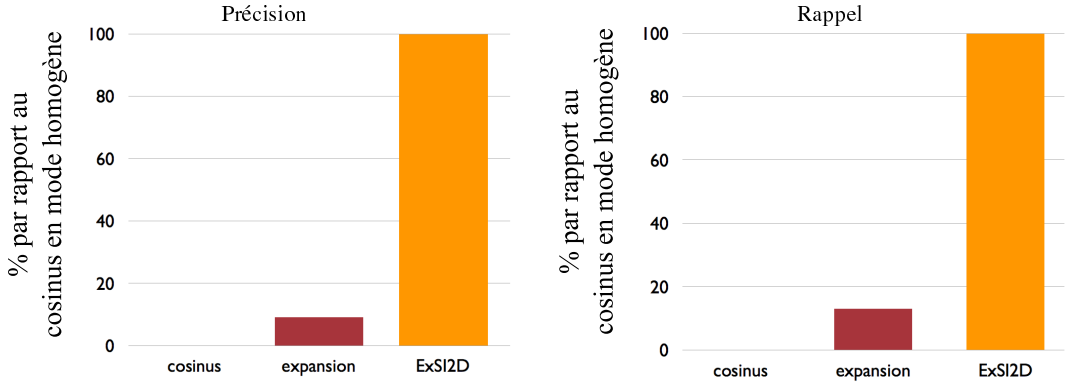


Figure 5.3 – Comportement des différentes méthodes lorsque seuls les concepts utilisés dans les requêtes ne sont pas alignés.

de similarité, ce qui est relativement favorable. Cependant, toutes les méthodes présentées ici sont confrontées à ce même contexte, et à notre connaissance, notre approche est actuellement la seule permettant d'obtenir de tels résultats dans des conditions d'interopérabilité dégradées.

5.4 Contribution et leçons.

Notre principale contribution consiste en une technique originale utilisant les alignements entre ontologies et permettant d'améliorer l'interopérabilité en rendant possible l'utilisation de concepts non alignés [VCLV08d, VCLV08c, VCLV08a, VCLV08b, Ven08, VCLV07, VLC05, Ven04]. Certains points caractérisant cette approche sont intéressants à noter :

- Les modules existants d'indexation et de calcul de pertinence n'ont pas à subir de modification et peuvent être utilisés en l'état.
- L'initiateur de la requête dispose d'un contrôle total sur cette technique. C'est lui qui contrôle entièrement l'explication qu'il fournit et c'est elle qui guide le reste du processus.
- Les participants n'ont pas à communiquer leurs ontologies respectives, si ce n'est pour réaliser l'alignement.
- Concernant le coût de l'approche, le point le plus sensible est le calcul de l'image qui doit être réalisé pour tous les documents. Il reste cependant contenu, dans la mesure où certaines optimisations sont possibles.

Notre proposition favorise l'autonomie sémantique des participants en améliorant l'interopérabilité entre eux. Elle leur offre une plus grande souplesse et une plus grande facilité pour faire évoluer leurs ontologies. En particulier, un participant peut ajouter quelques concepts à son ontologie sans pour autant nécessiter une mise à jour de l'alignement. Dès qu'introduits dans l'ontologie, ces concepts sont immédiatement pris en compte par les processus d'explication/interprétation. De plus, en fonction des explications dont il accompagne la requête, c'est lui qui guide le comportement des outils de recherche utilisés chez les fournisseurs. Il a donc un contrôle accru sur sa requête. Enfin, notre proposition ne nécessite pas la mémorisation des informations échangées qui pourrait être assimilée à un profilage.

Un prototype a été entièrement programmé dans une architecture de services web et doit être prochainement disponible au téléchargement sous licence LGPL.

Nous avons tiré un certain nombre d'enseignements de ces travaux.

Tout d'abord, l'ontologie fait sans conteste partie de l'environnement sémantique d'un participant, mais elle n'en est pas l'unique ingrédient. La fonction de similarité joue elle aussi un rôle qui prend toute son importance en impactant, avec l'alignement, sur l'intéropérabilité. Il n'est pas difficile d'imaginer deux participants partageant exactement la même ontologie, mais utilisant des similarités différentes. L'un peut alors arguer que le concept le plus proche de "chat sauvage" est "félin" car un chat sauvage est un félin (similarité structurelle), alors que l'autre considère que c'est "lynx" qui est le plus proche de "chat sauvage" avec comme argument que "lynx" est l'animal qui présente le plus de similarité avec un chat sauvage. De telles différences peuvent avoir un impact sur l'interopérabilité, en particulier pour la méthode que nous proposons. Une vision plus optimiste permet d'envisager le phénomène inverse : deux participants organisant les concepts de manière différente dans leurs ontologies, peuvent s'accorder en utilisant des similarités compensant les différences.

Ensuite, les valeurs des vecteurs sémantiques se sont révélées difficiles à interpréter. Obtenues par indexation sémantique automatique, la valeur associée à un concept exprime "la représentativité de ce concept pour ce document". Nos intuitions ont souvent été mises en défaut sur ce point. Les valeurs issues de modèles mathématiques plus formels (probabilités, possibilités, etc.) semblent plus faciles à appréhender.

Enfin, une absence de benchmark permettant d'évaluer des solutions en milieu hétérogène avec différents niveaux d'interopérabilité rend les validations difficiles. Les corpus de tests développés dans le cadre de la Recherche d'Information peuvent être utilisés, mais la distribution et l'hétérogénéité sont des notions qui leurs sont étrangères. Ils nécessitent donc une adaptation complémentaire (répartition des documents, des requêtes, hétérogénéité sémantique, etc) pour être utilisés dans ce cadre.

Dernier point à ne pas négliger, une approche de validation basée sur des techniques réalistes comme celle que nous avons conduite ici soulève de nombreux problèmes pratiques et nécessite un temps de développement très important.

5.5 Travaux en cours et perspectives

La définition d'ExSI^{2}D ouvre de nombreuses voies et nous n'évoquons que les principales. Les pistes pour son amélioration sont nombreuses. Son application dans un système distribué avec circulation de la requête de pairs en pairs reste encore à étudier. Il est envisageable d'exploiter ses propriétés pour répondre à des besoins et objectifs différents tel que la personnalisation des réponses. Enfin, elle nous a permis d'appréhender des besoins en termes d'études et de mesures de l'hétérogénéité et de l'intéropérabilité.

5.5.1 Approfondissements autour d'ExSI^{2}D

Dans sa version actuelle, ExSI^{2}D ne prend en compte que les relations d'équivalence entre concepts présentes dans l'alignement considéré. Plusieurs raisons à cela. Ces relations sont certainement parmi les plus sûres trouvées par un alignement. De plus, toutes les techniques d'alignement les proposent. Enfin, ce sont les plus faciles à utiliser. Tirer partie de toutes les correspondances exhibées par un alignement, en fonction de leur degré d'incertitude, permettrait vraisemblablement d'améliorer l'interopérabilité.

Le contexte d'évaluation d'ExSI^{2}D, entre un initiateur de requête et un fournisseur, suggère de pousser plus avant une étude sur l'hétérogénéité et l'intéropérabilité. Des travaux étudient actuellement la distance entre ontologies, en particulier pour déterminer s'il est intéressant de les aligner [Euz08, DE08]. Une étude plus approfondie est nécessaire car cela peut se révéler très proche de ce que nous appelons une mesure de l'hétérogénéité. Nous n'avons cependant pas encore trouvé dans la littérature de travaux pouvant correspondre à une mesure de l'intéropérabilité entre deux participants. ExSI^{2}D nous a déjà appris que ce cette mesure dépend non seulement de l'alignement entre les ontologies, mais aussi des fonctions de similarité utilisées par chaque participant. De futurs travaux devront déterminer si d'autres critères doivent être pris en compte. Une mesure de la difficulté à communiquer entre deux participants permettrait d'évaluer différentes solutions, les comparer plus efficacement, en les confrontant à des cas particuliers dont on pourrait quantifier la difficulté. Une mesure d'interopérabilité peut aussi jouer un rôle important dans un système distribué intégrant de nombreux participants hétérogènes.

Par exemple, dans les réseaux non structurés, l'intérêt de regrouper des participants en fonction de leurs centres d'intérêt dépend de leur capacité de compréhension mutuelle, i.e. de leur degré d'interopérabilité.

5.5.2 ExSI^{2}D dans un système distribué.

Nous nous intéressons plus particulièrement aux réseaux non structurés qui nécessitent la mise en oeuvre d'algorithmes de routage des requêtes et des réponses. La sémantique, et plus particulièrement ExSI^{2}D, peut jouer un rôle non seulement dans le calcul des réponses, mais aussi pour organiser les participants dans le réseau (*semantic overlay networks* [CGM04, ACM05]), pour orienter et guider la requête dans le réseau, et encore pour éventuellement limiter les réponses aux plus intéressantes (*requêtes top-k* [APV06b]). Nous pensons qu'il est possible d'adapter certains algorithmes ou d'en développer de nouveaux pour améliorer l'efficacité et pour obtenir des réponses qualifiées (i.e. que l'on sait caractériser par rapport à l'ensemble des réponses présentes dans le réseau, ex. top-k). Pour que les algorithmes de routage puissent en tirer partie, les informations sémantiques peuvent permettre à un participant de déterminer la pertinence des autres participants par rapport à une requête. Une avancée dans cette direction nécessite d'avoir à disposition une représentation résumant de manière synthétique les informations proposées par un pair ou un groupe de pair. De plus, pour éviter des mises à jours trop lourdes, cette représentation doit éviter un re-calcul complet à chaque modification. Enfin, pour garantir de trouver les meilleurs résultats en évitant les participants non pertinents, elle doit éviter les faux négatifs. De premières avancées ont été réalisées dans cette direction [VCLV09] concernant les propriétés d'une représentation sémantique. Ils ont principalement été obtenus en considérant un milieu homogène, car sans grande surprise, les problèmes les plus difficiles sont liés à l'hétérogénéité.

Que le cadre sémantique soit homogène ou hétérogène, de nombreux travaux partagent le besoin de benchmarks construits de manière rationnelle prenant en compte plusieurs critères comme la structure du réseau (ex. taille, nombre de connexions par pair, organisation), la répartition des informations sur les pairs (ex. aléatoire, thématique, par pair ou groupe de pairs,

taux de réplication, localisation des réplicats), la répartition des requêtes (éloignement des pairs émettant les requêtes par rapport aux données ciblées), etc. Il n'existe à notre connaissance aucun benchmark de ce type. Des travaux sont donc nécessaires pour proposer un environnement de test et de simulation des solutions suffisamment complet et détaillé pour valider et comparer correctement les différentes approches. Les simulations que nous avons menées et que nous continuons à réaliser, en particulier avec le simulateur PeerSim [WJV] nous ont obligé à avoir quelques réflexions sur le sujet. Nous avons très rapidement été amené à programmer un certain nombre de modules additionnels à PeerSim qui nous ont semblé indispensables pour réaliser des expérimentations réalistes, comme par exemple la définition de la charge du réseau et des partipants. Nous sommes maintenant convaincus de la nécessité d'un travail de fond à la fois sur les benchmarks et les outils de simulation. Il est important qu'un tel travail soit mené avec la participation de plusieurs équipes de recherche en prenant en considération des observations provenant des cas réels.

5.5.3 ExSI^{2}D pour la personnalisation des réponses.

Dans l'approche que nous avons proposée, les réponses obtenues à une requête dépendent des explications qui l'accompagnent. Ces explications sont construites en fonction de plusieurs paramètres : contexte sémantique de l'initiateur de la requête et fonctions de propagation utilisées.

Deux initiateurs de requêtes ayant des contextes sémantiques différents (ontologie et similarité sémantique) ne fournissent donc naturellement pas les mêmes explications. En conséquence, ils n'obtiendront généralement pas les mêmes réponses. Intuitivement, les réponses obtenues par un initiateur devraient être d'autant plus précises que ce dernier a fait l'effort d'acquérir et de s'approprier une représentation proche de celle utilisée par les spécialistes (fournisseurs d'informations) au moins en ce qui concerne les concepts impliqués sur la requête.

Même à l'intérieur d'un même contexte sémantique (ontologie et similarité) l'usage de différentes fonctions de propagation peut conduire à obtenir des réponses différentes à une requête.

Dans les deux cas, la variation des résultats dépend exclusivement de l'état et des choix de

l'initiateur de la requête. Il semble donc possible d'affirmer que les réponses sont personnalisées. Ce point nous semble mériter exploration. Il est en particulier nécessaire d'évaluer le bien fondé de l'approche pour une personnalisation, ses avantages et ses limites.

CHAPITRE 6

Conclusion et perspectives

Nous avons présenté dans ce document nos travaux liés à la problématique de l'intégration de participants autonomes dans des systèmes distribués et ouverts avec comme application privilégiée la recherche d'information. Deux axes principaux ont été suivis : l'adaptation aux objectifs individuels et l'autonomie sémantique des participants.

Dans une première approche, nous avons abordé le problème de la recherche d'informations sur internet. Après avoir acté le fait que fournisseurs et initiateurs de requêtes sont des participants autonomes ayant des intérêts individuels, nous avons proposé le système ***Bonom*** (cf. chapitre 2). Son organisation s'articule autour d'une une hiérarchie de thèmes. Chaque thème de la hiérarchie est géré par une communauté de participants. Nous nous appuyons sur les capacités des participants liées à leur autonomie pour demander à chaque initiateur de requête (respectivement, fournisseur) de caractériser lui-même les thèmes auxquels sa requête doit être soumise (respectivement, les thèmes auxquels il contribue). Lorsqu'une requête a atteint un thème la concernant, chaque fournisseur décide lui-même s'il va la traiter. Le modèle n'est donc plus celui du client-serveur où les fournisseurs ont un rôle "d'esclave". Cette approche permet d'obtenir un routage des requêtes et des réponses assez simple et léger à mettre en oeuvre pour permettre un passage à l'échelle.

Ces premiers travaux ont rapidement montré la difficulté à maintenir un équilibre entre les participants. En particulier, en pouvant choisir les requêtes qu'ils traitent, les fournisseurs disposent d'un avantage sur les initiateurs de requête. Cette constatation nous a conduit à proposer une approche où un médiateur incarne l'autorité nécessaire pour administrer le système. Son

rôle social consiste à garantir un certain équilibre entre les participants. Au delà de la recherche d'information, cette approche peut s'appliquer à tout autre domaine utilisant une technique d'allocation (de requête, de tâches, etc). Dans un premier temps, nous avons proposé la ***médiation flexible*** (cf. chapitre 3) qui prend explicitement en compte les intérêts des initiateurs et utilise une approche monétaire pour réguler les liens entre les fournisseurs et le médiateur. Les expérimentations montrent qu'elle s'adapte automatiquement aux variations d'intérêt des participants et prévient les départs.

Nos travaux ont aussi porté sur une modélisation du problème de l'autonomie des participants dans le cadre de l'allocation de requête pour mieux l'appréhender (cf. chapitre 4). Nous avons identifié plusieurs concepts de satisfaction, mais aussi d'adéquation des participants. Des mesures d'équité et d'efficacité des processus de médiation ont aussi été proposées. Ces notions permettent d'analyser plus finement le comportement global d'un système et de caractériser ses participants. A partir de ces réflexions, nous avons proposé un nouveau processus d'allocation qui est exclusivement basé sur les intentions exprimées par les participants et leurs satisfactions individuelles. Les validations expérimentales ont montré les qualités de cette technique générique qui, tout en s'adaptant aux intérêts des participants, préserve les performances du système. Des améliorations sont envisageables, en particulier concernant le passage à l'échelle.

Enfin, nous avons exploré le problème de l'hétérogénéité sémantique, difficilement évitable dans un système intégrant des participants autonomes provenant d'horizons différents. En utilisant les résultats obtenus par un alignement entre ontologies, nous avons proposé une méthode originale qui améliore l'interopérabilité dans le contexte d'une recherche d'information où documents et requêtes sont exprimés par des vecteurs sémantiques (cf. chapitre 5). La solution proposée consiste à demander à l'initiateur d'expliquer les concepts utilisés dans sa requête, explications qui sont alors interprétées par un fournisseur pour adapter ses documents à la requête. Les validations expérimentales ont permis de démontrer l'amélioration importante obtenue. Là encore des améliorations sont envisageables. En particulier, seuls les alignements décrivant des équivalences entre concepts sont actuellement utilisés par notre approche. La prise en compte des autres relations produites par les alignements devrait permettre une avancée supplémentaire.

La suite de ce chapitre présente les perspectives de recherche liées à nos travaux. Nous les

structurons en trois parties complémentaires qui peuvent progresser de concert. Nous présentons dans un premier temps un certain nombre de problématiques existantes qui peuvent être revisitées suite à nos travaux. Ces derniers sont utilisés dans plusieurs applications concrètes que nous décrivons ensuite. Enfin, nous proposons quelques directions pour un effort de modélisation, nécessaire pour mieux comprendre et caractériser les systèmes distribués ouverts, intégrant des participants autonomes.

6.1 Appréhender des problématiques actuelles sous un angle différent

Réplication de requête administrée via la satisfaction. La réplication de requête peut avoir plusieurs objectifs. Elle peut être utilisée dans le cas où les participants fournissent tous la même réponse pour résoudre par anticipation la panne de l'un d'entre eux ou pour obtenir plusieurs réponses afin de détecter un éventuel "menteur" [LSP82]. Un problème consiste alors à déterminer le nombre de fournisseurs à interroger. Nous travaillons sur une solution qui, tout en respectant strictement la méthode d'allocation utilisée, se base sur la satisfaction des participants pour déterminer le nombre de réplicats à réaliser.

Structures de stockage de données distribuées basées sur la satisfaction et la sémantique. Nous travaillons actuellement en collaboration avec l'université de Saarlandes sur une structure de stockage distribuée assez similaire à une DHT [SMK+01, WEW74, RD, WEW75], mais tenant compte des intérêts des participants pour stocker tel ou tel document. Un participant peut déterminer son intérêt vis-à-vis d'un document si celui-ci est qualifié sémantiquement. En utilisant conjointement sémantique et satisfaction l'objectif est d'obtenir une structure de stockage hybride entre cache et DHT, ce qui a priori, ne présente que des avantages pour les participants.

Routage des requêtes dans un système non structuré guidé par la sémantique, et évaluation des performances. Les systèmes pair-à-pair s'imposent comme une technique majeure pour la gestion des données distribuées. En particulier, l'approche non structurée bien que moins efficace pour la recherche d'information présente le double avantage d'être très peu normative, et donc très respectueuse de l'autonomie des participants, mais aussi très résistante aux pannes,

ainsi qu'aux départs et arrivées volontaires des participants. La sémantique joue déjà un rôle pour améliorer les performances de la recherche d'information dans un tel cadre, que ce soit pour le regroupement des participants ou le routage des requêtes [Def07]. Cependant, pour obtenir un ensemble de réponses à une requête qui soit qualifié par rapport à ce qui est disponible sur le réseau (nous nous intéressons en particulier aux meilleures réponses possibles, requêtes top-k), il est nécessaire de solliciter tous les participants du réseau, ce qui n'est pas raisonnable. Nous proposons d'utiliser la sémantique pour router les requêtes, mais aussi pour anticiper les résultats en évitant de faire parvenir une requête à un participant (ou un groupe de participants) qui n'a aucune chance de fournir une réponse entrant dans le top-k. De plus, les systèmes non structurés laissent toute liberté aux participants pour s'organiser. Or la topologie du réseau (ex. : degré de connexion entre les participants), le placement des données (ex. : regroupées sémantiquement), l'origine des requêtes (ex. : proche des données concernées ou très éloignées) sont autant de paramètres qui influencent fortement les performances d'un algorithme. Pour évaluer les différentes solutions de manière efficace il est donc important de disposer d'exemples réels et de benchmarks qu'il faudra construire.

Personnalisation des réponses. Lorsqu'elle est observée du point de vue d'un initiateur de requête, EXSI^{2}D proposée pour améliorer l'interopérabilité en milieu sémantiquement hétérogène (cf. chapitre 5) a une propriété remarquable. En fonction des explications fournies, les réponses obtenues sont différentes. Une différence dans les explications peut provenir d'un degré de précision plus ou moins important, mais aussi d'une compréhension différente des concepts. Deux initiateurs ne partageant pas la même représentation sémantique ne fourniront pas les mêmes explications. Il n'est pas difficile d'observer que cette technique personnalise les réponses en fonction des utilisateurs, ou plus précisément, de leur représentation sémantique et du degré de précision de leurs explications des concepts. La constatation de cette propriété ne suffit pas. Des travaux plus importants sont nécessaires pour évaluer l'intérêt de cette approche par rapport aux solutions existantes. On peut néanmoins d'ores et déjà souligner que cette approche, naturelle, n'est pas basée sur un apprentissage du profil de l'utilisateur par le fournisseur. Ceci peut être un avantage pour préserver la confidentialité.

6.2 Etude d'applications concrètes

Les applications potentielles sont nombreuses. Nous détaillons celles qui vont donner lieu à développement à court terme.

Systèmes communautaires dédiés. Les systèmes communautaires, réseaux sociaux, présentent toutes les caractéristiques de systèmes ouverts intégrant des participants autonomes provenant d'horizons différents. Les besoins d'interopérabilité sémantique sont particulièrement sensibles. Nous travaillons sur l'intéropérabilité sémantique dans un système de gestion de l'environnement composé de participants de différents domaines avec un haut degré de spécialisation (océanographes, hydrologues, géologues, agronomes, etc).

Collaboration inter-entreprises pour le transport. Nous débutons un projet avec les sociétés Euxénis SAS et RISC Solutions d'Assurances concernant l'étude des technologies pair-à-pair pour la collaboration inter-entreprises dans la chaîne logistique du transport. Ce projet, est principalement motivé suite à l'expression d'un besoin d'équilibre global entre les demandeurs et les fournisseurs, équilibre qu'une approche "place de marché" ne permet pas d'atteindre. Nous pensons que la prise en compte de la satisfaction permettra de répondre au besoin.

Gestion des données personnelles. Du point de vue d'un utilisateur, le projet offre la possibilité de mémoriser les informations qu'il communique électroniquement via des formulaires. Ces informations sont qualifiées ontologiquement suivant un usage qui se rapproche du typage ou d'un schéma de base de données. Elles peuvent alors être réutilisées d'un formulaire à l'autre ce qui facilite la saisie. Sans s'opposer aux services Internet qui stockent les données qui leurs sont transmises, notre approche permet à un utilisateur de gérer lui-même ses informations personnelles de manière autonome et transversale à ces services. Cette proposition qui améliore l'autonomie des utilisateurs fait actuellement l'objet d'un projet en cours de finalisation en collaboration avec les sociétés Mandriva et Nexedi (projet labellisé par le pôle de compétitivité SYSTEM@TIC PARIS-REGION).

6.3 Modéliser pour mieux caractériser les systèmes

Les évolutions d'usage actuelles montrent des besoins importants en systèmes distribués ouverts intégrant des participants autonomes qui leurs apportent leurs ressources. Nous sommes convaincus que cette approche est amenée à se développer. Néanmoins, les concepts qui permettent de caractériser un système dans son ensemble, ses participants, ou des situations locales sont encore trop peu nombreux et pas toujours totalement adaptés. La proposition de modélisation de la satisfaction sur le long terme et des diverses notions d'adéquation va en ce sens. Nous décrivons, de manière non exhaustive, un certain nombre de pistes à explorer.

La sémantique pour préciser la notion de satisfaction La notion de satisfaction que nous avons proposée est relative à l'allocation de requêtes. Nous pouvons aussi considérer la satisfaction d'un participant vis-à-vis d'un autre participant. Cette dernière influence un initiateur, respectivement un fournisseur, sur son intention de solliciter un fournisseur, respectivement pour accepter les requêtes d'un initiateur. Mais, pour qu'il soit possible de l'exploiter au maximum, il serait intéressant de qualifier sémantiquement la satisfaction en fonction du contexte dans lequel elle a été acquise. Par exemple, un étudiant peut être satisfait de son enseignant d'informatique dans le cadre d'un cours sur l'architecture des ordinateurs. Bien qu'il le considère comme bon enseignant doit-il le solliciter sur des questions du cours de biologie marine ? A priori, non, car bien qu'il soit enseignant, le domaine sémantique est très différent. De même, bien que l'architecture d'un ordinateur n'ait aucun secret pour cet enseignant, cela ne justifie pas de lui confier un PC pour réparation. La caractérisation précise de la notion de satisfaction nécessite donc de prendre en compte deux notions : le domaine et la prestation. Les deux peuvent être représentés sémantiquement, éventuellement par des ontologies différentes. Les notions de propagation, d'explication et d'interprétation introduites pour améliorer l'interopérabilité sémantique peuvent servir de base à une solution permettant aux participants d'échanger leurs points de vue en milieu hétérogène.

Caractérisation de l'interopérabilité. L'hétérogénéité sémantique est une conséquence directe difficilement évitable de l'intégration de participants autonomes dans des systèmes largement distribués et ouverts. La difficulté consiste alors à rendre sémantiquement interopérables

ces participants. Un premier niveau d'interopérabilité est obtenu au niveau syntaxique grâce à l'usage de normes pour la représentation des ontologies (RDF, OWL,...). Bien qu'indispensable, ce point se révèle insuffisant. Un alignement entre deux ontologies jette les bases nécessaires à la communication. Nous avons pour notre part proposé une méthode complémentaire basée sur l'explication et l'interprétation pour améliorer l'intéropérabilité au delà de ce que permet un alignement. Nos investigations ont mis en lumière l'absence de mesures permettant de qualifier, de quantifier, les notions d'hétérogénéité et d'interopérabilité. Les mesures de similarité entre ontologies peuvent répondre à ce besoin [DE08, Euz08]. De la même manière, la mesure de l'interopérabilité obtenue par la mise en œuvre d'un alignement présenterait l'intérêt de mieux mesurer la difficulté de la communication dans le système résultant. Nous avons aussi mis en évidence que la considération des ontologies et de l'alignement n'est pas toujours suffisante pour juger de l'interopérabilité sémantique. Les fonctions de similarité utilisées par les participants peuvent aussi jouer aussi un rôle important. Peut-être existe-t-il d'autres paramètres. L'absence de mesures combinée au manque de benchmarks rendent les évaluations difficiles. Il est donc souhaitable que des travaux permettent de déboucher sur la définition de mesures pertinentes permettant de quantifier l'hétérogénéité d'une situation particulière ainsi que le degré d'interopérabilité obtenu par les techniques mises en place.

Contrôle distribué des ressources dans un système, modélisation et compréhension. L'intégration de participants autonomes permet de construire des systèmes performants pour un coût relativement faible en profitant des ressources qu'ils apportent. Cependant, un participant autonome peut décider de quitter le système, au même titre qu'il a décidé d'y contribuer. Même si le départ volontaire d'un participant est techniquement supporté (comme s'il s'agissait d'une panne), il convient de l'éviter lorsqu'il n'est pas justifié pour le système. Dans le cadre de l'allocation de requête, nous avons proposé plusieurs solutions dont le principe consiste à satisfaire les participants. Ceci peut être compris comme un moyen d'éviter le départ de participants importants pour le système, sans pour autant leur concéder une autonomie importante. Il n'en demeure pas moins que chaque participant conserve un certain contrôle direct sur les ressources qu'il apporte. Suivant les ressources considérées, cela peut aller de la qualité de service au retrait pur et simple du système. Ce contrôle n'est donc pas sans conséquence sur le système. Pour

appréhender le fonctionnement d'un système distribué ouvert, il nous semble donc important de déterminer précisément quels sont les moyens dont dispose un participant pour contrôler sa ressource, et d'évaluer les conséquences de l'exercice de ce contrôle sur les autres participants et sur le système. A notre connaissance, il n'existe actuellement aucun outil apportant une réponse à ce problème. Les questions sont pourtant nombreuses : "Quels contrôles sont possibles ?", "Quels sont ceux qu'il est primordial de conserver (pour le système, comme pour les participants) ?", "Quels sont ceux qu'il est préférable de régler de manière contractuelle et quels sont ceux qu'il est plus intéressant d'intégrer dans le système informatique comme moyens d'ajustement ?", "Quels sont les avantages que peut tirer un participant (respectivement le système) ?", etc. Nous commençons une collaboration avec C. Sibertin-Blanc pour étudier ce problème en partant du point de vue de la sociologie de l'action organisée et en nous basant sur des modèles et des outils développés dans ce cadre [SBAM05]. Un axe de nos travaux futurs consiste donc à étudier ce que peut apporter une analyse sociologique pour mieux comprendre et spécifier des systèmes distribués ouverts intégrant des participants autonomes.

Bibliographie

[Abe01] Karl Aberer. P-grid: A self-organizing access structure for p2p information systems. In Batini et al. [BGGM01], pages 179–194.

[ABKU99] Y. Azar, A. Z. Broder, A. R. Karlin, and E. Upfal. Balanced Allocations. *SIAM Journal on Computing*, 29(1):180–200, 1999.

[ACG+04] Philippe Adjiman, Philippe Chatalic, François Goasdoue, Marie-Christine Rousset, and Laurent Simon. Somewhere in the semantic web. Technical report, LRI, 2004.

[ACK+02] David P. Anderson, Jeff Cobb, Eric Korpela, Matt Lebofsky, and Dan Werthimer. Seti@home: an experiment in public-resource computing. *Communications of the ACM*, 45(11):56–61, 2002.

[ACM05] Karl Aberer and Philippe Cudré-Mauroux. Semantic overlay networks. In Böhm et al. [BJH+05], page 1367.

[And04] David P. Anderson. Boinc: A system for public-resource computing and storage. In *GRID '04: Proceedings of the Fifth IEEE/ACM International Workshop on Grid Computing*, pages 4–10, Washington, DC, USA, 2004. IEEE Computer Society.

[APV06b] Reza Akbarinia, Esther Pacitti, and Patrick Valduriez. Reducing network traffic in unstructured p2p systems using top-k queries. *Distributed and Parallel Databases*, 19(2-3):67–86, 2006.

[Atl02] Vijayalakshmi Atluri, editor. *Proceedings of the 9th ACM Conference on Computer and Communications Security, CCS 2002, Washington, DC, USA, November 18-22, 2002*. ACM, 2002.

[BBM02] Holger Billhardt, Daniel Borrajo, and Victor Maojo. A context vector model for information retrieval. *JASIST*, 53(3):236–249, 2002.

[BD01] Jean-Pierre Briot and Yves Demazeau. *Princpes et architectures des systèmes multi-agents*. Hermes, 2001.

[BDJ99] Michael W. Berry, Zlatko Drmac, and Elizabeth R. Jessup. Matrices, vector spaces, and information retrieval. *SIAM Rev.*, 41(2):335–362, 1999.

[BFS00] Alain Bidault, Christine Froidevaux, and Brigitte Safar. Repairing queries in a mediator approach. In *Proceedings of the 14th European Conference on Artificial Intelligence, ECAI 2000*, pages 406–410, 2000.

[BFS02] Alain Bidault, Christine Froidevaux, and Brigitte Safar. Proximité; entre requêtes dans un contexte médiateur. In *13 ème congrès Francophone de Reconnaissance des Formes et Intelligence Artificielle, RFIA 2002*, volume 2, pages 653–662, janvier 2002.

[BGGM01] Carlo Batini, Fausto Giunchiglia, Paolo Giorgini, and Massimo Mecella, editors. *Cooperative Information Systems, 9th International Conference, CoopIS 2001, Trento, Italy, September 5-7, 2001, Proceedings*, volume 2172 of *Lecture Notes in Computer Science*. Springer, 2001.

[BHLP07] Omar Boucelma, Mohand-Said Hacid, Thérèse Libourel, and Jean-Marc Petit, editors. *23èmes Journées Bases de Données Avancées, BDA 2007, Marseille, 23-26 Octobre 2007, Actes (Informal Proceedings)*, 2007.

[Bid02] Alain Bidault. *Affinement de requêtes posées à un médiateur*. PhD thesis, University Paris XI, Orsay, Paris, France, july 2002.

[BJH+05] Klemens Böhm, Christian S. Jensen, Laura M. Haas, Martin L. Kersten, Per-Åke Larson, and Beng Chin Ooi, editors. *Proceedings of the 31st International Conference on Very Large Data Bases, Trondheim, Norway, August 30 - September 2, 2005*. ACM, 2005.

[BP98] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1–7):107–117, 1998.

[BPD+06] Olivier Boissier, Julian A. Padget, Virginia Dignum, Gabriela Lindemann, Eric T. Matson, Sascha Ossowski, Jaime Simão Sichman, and Javier Vázquez-Salceda, editors. *Coordination, Organizations, Institutions, and Norms in Multi-Agent Systems, AAMAS 2005 International Workshops on Agents, Norms and Institutions for Regulated Multi-Agent Systems, ANIREM 2005, and Organizations in Multi-Agent Systems, OOOP 2005, Utrecht, The Netherlands, July 25-26, 2005, Revised Selected Papers*, volume 3913 of *Lecture Notes in Computer Science*. Springer, 2006.

[BS08] Mohand Boughanem and Jacques Savoy, editors. *Recherche d'information : état des lieux et perspectives*. Lavoisier, Paris, 2008.

[BYRN99] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retieval*. Addison Wesley, 1999.

[CDJL00c] S. Cazalens, E. Desmontils, C. Jacquin, and P. Lamarre. A web site indexing process for an internet information retrieval agent system. In IEEE Computer Society Press, editor, *International Conference on Web information Systems Engineering (WISE'2000)*, pages 245–249, Hong-Kong, 2000.

[CDJL02] S. Cazalens, E. Desmontils, C. Jacquin, and P. Lamarre. Sources d'informations et de connaissances : de la gestion locale à la recherche distribuée. *RSTI, L'Objet*, 8(4)(47-69), 2002.

[CGM04] Arturo Crespo and Hector Garcia-Molina. Semantic overlay networks for p2p systems. In Moro et al. [MBA05], pages 1–13.

[Cha02] Cédric Champeau. Etude de mécanismes de matchmaking - rapport de dea, 3e année d'ingénieur. Technical report, IRIN - Ecole Polytechnique de l'Université de Nantes, 2002.

[CL92] Gabriella Crocco and Philippe Lamarre. On the connection between non-monotonic inference systems and conditional logics. In *KR*, pages 565–571, 1992.

[CL01a] Sylvie Cazalens and Philippe Lamarre. An organization of internet agents based on a hierarchy of information domains. In Yves Demazeau and Francisco J. Garijo, editors, *Proceedings MAAMAW'01*, 2001.

[CL01b] Sylvie Cazalens and Philippe Lamarre. Organizing internet agents according to a hierarct of information domains. In *Intelligent Agent Technology*, pages 469–473. World Scientific, 2001.

[DBL94] *Proceedings of the Third International Conference on Information and Knowledge Management (CIKM'94), Gaithersburg, Maryland, November 29 - December 2, 1994*. ACM, 1994.

[DBL07] *Actes du XXVème Congrès INFORSID, Perros-Guirec, France, 22 au 25 mai 2007*, 2007.

[DBL09] *Proceedings of the 25th International Conference on Data Engineering, ICDE 2009, March 29 2009 - April 2 2009, Shanghai, China*. IEEE, 2009.

[DDL+90b] Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. Indexing by latent semantic analysis. *JASIS*, 41(6):391–407, 1990.

[DdVP+02] Ernesto Damiani, Sabrina De Capitani di Vimercati, Stefano Paraboschi, Pierangela Samarati, and Fabio Violante. A reputation-based approach for choosing reliable resources in peer-to-peer networks. In Atluri [Atl02], pages 207–216.

[DE08] Jérôme David and Jérôme Euzenat. Comparison between ontology distances (preliminary results). In Sheth et al. [SSD+08], pages 245–260.

[Def07] Bruno Defude. Organisation et routage sémantiques dans les systèmes pair-à-pair. In *INFORSID* [DBL07], pages 12–18.

[DGP08] Anne Doucet, Stéphane Gançarski, and Esther Pacitti, editors. *Proceedings of the 2008 International Workshop on Data Management in Peer-to-Peer Systems, DaMaP 2008, Nantes, France, March 25, 2008*, ACM International Conference Proceeding Series. ACM, 2008.

[DJ01a] E. Desmontils and C. Jacquin. Des ontologies pour indexer un site web. In *actes des journées francophone d'ingénierie des connaissances*, Grenoble, 2001.

[DJ01b] Emmanuel Desmontils and Christine Jacquin. Indexing a web site with a terminology oriented ontology. In *Semantic Web Working Symposium*, pages 549–565, Stanford University, California, USA, 2001.

[DJ02a] Emmanuel Desmontils and Christine Jacquin. *The Emerging Semantic Web*, volume 75 of *Frontiers in Artificial Intelligence and Applications*, chapter Indexing a web site with a terminology oriented ontology., pages 181–197. IOS press, 2002.

[DJ02b] Emmanuel Desmontils and Christine Jaquin. *The Emerging Semantic Web*, chapter Indexing a Web Site with a Terminology Oriented Ontology, pages 181–197. IOS Press, 2002. ISBN 1-58603-255-0.

[DMDH04] AnHai Doan, Jayant Madhavan, Pedro Domingos, and Alon Y. Halevy. Ontology matching: A machine learning approach. In Staab and Studer [SS04], pages 385–404.

[DMP+99] Edmund Durfee, Tracy Mullen, Sunju Park, José Vidal, and Peter Weistein. *Intelligent Information Agents*, chapter Strategic Reasoning and Adaptation in an Information Economy. Springer, 1999.

[DRJ04] Rajdeep Dash, Sarvapali Ramchurn, and Nicholas R. Jennings. Trust-based mechanism design. In *AAMAS-2004 – Proceedings of the Third International Joint Conference on Autonomous Agents and Multi Agent Systems*, 2004.

[DSW97b] Keith Decker, Katia Sycara, and Mike Williamson. Middle-agents for the internet. In *Fifteenth International Joint Conference on Artificial Intelligence (IJCAI'97)*. Morgan Kaufmann, 1997.

[ES04] Marc Ehrig and Steffen Staab. Qom - quick ontology mapping. In *International Semantic Web Conference*, pages 683–697, Hiroshima, Japan, 2004.

[ES07] Jérôme Euzenat and Pavel Shvaiko. *Ontology matching*. Springer-Verlag, Heidelberg (DE), 2007.

[Euz08] Jerome Euzenat. Quelques pistes pour une distance entre ontologies. In *atelier Mesures de Similarité Sémantique, associé à EGC*, 2008.

[FDES98] Dieter Fensel, Stephan Decker, Michael Erdmann, and Rudi Studer. Ontobroker: Or how to enable intelligent access to the www. In *Proceedings of the 11th Banff Knowledge Acquisition for Knowledge-Based System Workshop (KAW'98)*, Banff, Canada, 1998.

[Fel98] Christiane Felbaum. *WordNet : an electronic lexical database*. Bradford Books, March 1998.

[FFMM94] Timothy W. Finin, Richard Fritzson, Donald P. McKay, and Robin McEntire. Kqml as an agent communication language. In *CIKM* [DBL94], pages 456–463.

[FKN+92] Anthony Finkelstein, Jeff Kramer, Bashar Nuseibeh, L. Finkelstein, and Michael Goedicke. Viewpoints: A framework for integrating multiple perspectives in system development. *International Journal of Software Engineering and Knowledge Engineering*, 2(1):31–57, 1992.

[FLM97] T. Finin, Y. Labrou, and J. Mayfield. Kqml as an agent communication language. In J. Bradshaw, editor, *Software Agents*, pages 291–316. MIT Press, 1997.

[FNSY96] D. Ferguson, C. Nikolaou, J. Sairamesh, and Y. Yemini. Economic Models for Allocating Resources in Computer Systems. In S. H. Clearwater, editor, *Market-Based Control: A Paradigm for Distributed Resource Allocation*. World Scientific, 1996.

[FYN88] D. Ferguson, Y. Yemini, and C. Nikolaou. Microeconomic Algorithms for Load Balancing in Distributed computer systems. In *Proceedings of the International Conference on Distributed Computing Systems (ICDCS)*, 1988.

[GBGM04] P. Ganesan, M. Bawa, and H. Garcia-Molina. Online Balancing of Range-Partitioned Data with Applications to Peer-to-Peer Systems. In *Proceedings of the Very Large Data Bases Conference (VLDB)*, 2004.

[GN05] A. Gorobets and B. Nooteboom. Agent based computational model of trust. Technical report, Erasmus Research Institute of Management (ERIM), RSM Erasmus University, January 2005.

[Gro04] Guillaume Grondin. Passage à l'échelle dans le cadre de la recherche d'informations. Technical report, IRIN - Ecole Polytechnique de l'Université de Nantes, 2004.

[GW02] James R. Groff and Paul N. Wienberg. *SQL: The Complete Reference*. Number ISBN:9780072225594. McGraw-Hill Book Companies, second edition, 2002.

[HHL99a] Jeff Heflin, James Hendler, and Sean Luke. Applying ontology to the web: A case study. In *International Work-Conference on Artificial and Natural Neural Networks (IWANN'99)*, 1999.

[HIM+04] Alon Y. Halevy, Zachary G. Ives, Jayant Madhavan, Peter Mork, Dan Suciu, and Igor Tatarinov. The piazza peer data management system. *IEEE Trans. Knowl. Data Eng.*, 16(7):787–798, 2004.

[HIMT03] Alon Y. Halevy, Zachary G. Ives, Peter Mork, and Igor Tatarinov. Piazza: data management infrastructure for semantic web applications. In *WWW '03: Proceedings of the 12th international conference on World Wide Web*, pages 556–567, New York, NY, USA, 2003. ACM.

[IHMT03] Zachary G. Ives, Alon Y. Halevy, Peter Mork, and Igor Tatarinov. Piazza: mediation and integration infrastructure for semantic web data. *Journal of Web Semantics*, 2003.

[JCH84] R. K. Jain, D.-H. Chiu, and W. R. Hawe. A Quantitive Measure of Fairness and Discrimination for Resource Allocation in Shared Computer Systems, DEC-TR-301. Technical report, 1984.

[JI02] Audun Jøsang and Roslan Ismail. The beta reputation system. In *15th Bled Electronic Commerce Conference*, 2002.

[Kao08] Ming-Yang Kao, editor. *Encyclopedia of Algorithms*. Springer, 2008.

[KC92] Robert Krovetz and W. Bruce Croft. Lexical ambiguity and information retrieval. *Information Systems*, 10(2):115–141, 1992.

[KH95a] D. Kuokka and L. Harada. Matchmaking for information agents. In *Proceedings of the International Joint Conferences on Artificial Intelligence (IJCAI)*, 1995.

[Klu99b] Mattias Klusch, editor. *Intelligent Information Agents*. Springer, 1999.

[Kre90] David M. Kreps. *A Course in Microeconomic Theory*. Princeton University Press, 1990.

[KS01] M. Klusch and K. Sycara. *Coordination of Internet Agents: Models, Technologies, and Aplications*, chapter Brokering and Matchmaking for Coordination of Agent Societies: A Survey. Springer-Verlag, 2001.

[Lam91] Philippe Lamarre. S4 as the conditional logic of nonmonotonicity. In *KR*, pages 357–367, 1991.

[Lam92a] Philippe Lamarre. A promenade from monotonicity to non-monotonicity following a theorem prover. In *KR*, pages 572–580, 1992.

[Lam92b] Philippe Lamarre. A tableau like theorem prover for conditional logics (extended abstract). In *TABLEAUX*, pages 52–55, 1992.

[LC98] Claudia Leacok and Martin Chodorow. Combining local context and and wordnet similarity for word sense identification. In Christiane Fellbaum, editor, *WordNet: An electronic lexical database and some of its applications*. The MIT Press, 1998.

[LC03a] Philippe Lamarre and Sylvie Cazalens. Médiation équitable dans un environnement ouvert d'agents compétitifs. In *Modèles Formels de l'Interaction*, 2003.

[LC03b] Philippe Lamarre and Sylvie Cazalens. A procedure for mediating between service requesters and providers. In *Proceedings of the International Conference on Intelligent Agents Technology (IAT 2003)*. IEEE press, 2003.

[LCLV04b] Philippe Lamarre, Sylvie Cazalens, Sandra Lemp, and Patrick Valduriez. A flexible mediation process for large distributed information systems. In Zahir Tari Robert Meersman, editor, *On the Move to Meaningful Internet Systems 2004: CoopIs, DOA, ODBASE*, volume 1 of *LNCS - LNCS3290*. Springer, 2004.

[Lem03] Sandra Lemp. Etude d'un processus de médiation équitable - rapport de dea. Technical report, IRIN-Ecole Polytechnique de l'Université de Nantes, 2003.

[Lem07] Sandra Lemp. *Médiation flexible dans un système pair-à-pair*. PhD thesis, Université de Nantes, 2007.

[LLC03] Philippe Lamarre, Sandra Lemp, and Sylvie Cazalens. Une procédure de médiation équitable pareto optimale. In *RSTI/hors série*, volume JFSMA/2003, pages 283–295, 2003.

[LLCV07] Philippe Lamarre, Sandra Lemp, Sylvie Cazalens, and Patrick Valduriez. A flexible mediation process for large distributed information systems. *Int. J. Cooperative Inf. Syst.*, 16(2):299–332, 2007.

[LS94] Philippe Lamarre and Yoav Shoham. Knowledge, certainty, belief, and conditionalisation (abbreviated version). In *KR*, pages 415–424, 1994.

[LS03] Anton Likhodedov and Tuomas Sandholm. Auction mechanism for optimally trading off revenue and efficiency. In *EC '03: Proceedings of the 4th ACM conference on Electronic commerce*, pages 212–213, New York, NY, USA, 2003. ACM Press.

[LSP82] Leslie Lamport, Robert Shostak, and Marshall Pease. The byzantine generals problem. *ACM Trans. Program. Lang. Syst.*, 4(3):382–401, July 1982.

[MBA05] Gianluca Moro, Sonia Bergamaschi, and Karl Aberer, editors. *Agents and Peer-to-Peer Computing, Third International Workshop, AP2PC 2004, New York, NY, USA, July 19, 2004, Revised and Invited Papers*, volume 3601 of *Lecture Notes in Computer Science*. Springer, 2005.

[MBF+90] Georges A. Miller, Richard Beckwith, Christiane Felbaum, Derek Gross, and Katherine Miller. Introduction to wordnet : an on-line lexical database. *International journal of lexicography*, 3(4):235–244, 1990.

[MCWG95] Andreu Mas-Colell, Michael D. Whinston, and Jerry R. Green. *Microeconomic Theory*. 1995.

[MTS90] R. Mirchandaney, D. F. Towsley, and J. A. Stankovic. Adaptive Load Sharing in Heterogeneous Distributed Systems. *Journal of Parallel and Distributed Computing (JPDC)*, 9(4):331–346, 1990.

[NBN99] M. H. Nodine, W. Bohrer, and A. H. Ngu. Semantic Brokering over Dynamic Heterogeneous Data Sources in InfoSleuth. In *Proceedings of the International Conference on Data Engineering (ICDE)*, 1999.

[OGM04] James Odell, Paolo Giorgini, and Jörg P. Müller, editors. *Agent-Oriented Software Engineering V, 5th International Workshop, AOSE 2004, New York, NY, USA, July 19, 2004, Revised Selected Papers*, volume 3382 of *Lecture Notes in Computer Science*. Springer, 2004.

[OMG96] OMG. *Trading Object Service*, 1996.

[OMG04] OMG. *Event Service*, 2004.

[ONL04] James Odell, Marian H. Nodine, and Renato Levy. A metamodel for agents, roles, and groups. In Odell et al. [OGM04], pages 78–92.

[PRST02] Ryan Porter, Amir Ronen, Yoav Shoham, and Moshe Tennenholtz. Mechanism design with execution uncertainty. In *Proceedings of the 18th conference on Uncertainty in Artificial Intelligence (UAI-02)*, 2002.

[PST04] Ryan Porter, Yoav Shoham, and Moshe Tennenholtz. Fair imposition. *Journal of Economic Theory*, 118(2):209–228, 2004.

[QF93] Y. Qiu and H. P. Frei. Concept based query expansion. In *Research and Development in Information Retrieval, ACM-SIGIR*, pages 160–169, 1993.

[QR08] Jorge-Arnulfo Quiané-Ruiz. *Allocation de requêtes dans des systèmes d'information distribués avec des participants autonomes*. PhD thesis, Université de Nantes, 2008.

[QRLCV07a] Jorge-Arnulfo Quiané-Ruiz, Philippe Lamarre, Sylvie Cazalens, and Patrick Valduriez. Satisfaction balanced mediation. In Silva et al. [SLBY+07], pages 947–950.

[QRLCV07b] Jorge-Arnulfo Quiané-Ruiz, Philippe Lamarre, Sylvie Cazalens, and Patrick Valduriez. A satisfaction balanced query allocation process for distributed information systems. In Boucelma et al. [BHLP07].

[QRLCV08] Jorge-Arnulfo Quiané-Ruiz, Philippe Lamarre, Sylvie Cazalens, and Patrick Valduriez. Managing virtual money for satisfaction and scale up in p2p systems. In Doucet et al. [DGP08], pages 67–74.

[QRLV06] J.-A. Quiané-Ruiz, P. Lamarre, and P. Valduriez. Satisfaction based query load balancing. In *Proceedings of the Cooperative Information Systems Conference (CoopIS)*, 2006.

[QRLV07a] J.-A. Quiané-Ruiz, P. Lamarre, and P. Valduriez. K_nBest - A Balanced Request Allocation Method for Distributed Information Systems. In *Proceedings of the Database Systems for Advanced Applications Conference (DASFAA)*, 2007.

[QRLV07b] J.-A. Quiané-Ruiz, P. Lamarre, and P. Valduriez. SQLB: A Query Allocation Framework for Autonomous Consumers and Providers. In *Proceedings of the Very Large Data Bases Conference (VLDB)*, 2007.

[QRLV08] Jorge-Arnulfo Quiané-Ruiz, Philippe Lamarre, and Patrick Valduriez. Sbqa: Une méthode auto-adaptative pour l'allocation de requêtes. In *Journées Bases de Données Avancées (BDA)*, 2008.

[QRLV09a] Jorge-Arnulfo Quiané-Ruiz, Philippe Lamarre, and Patrick Valduriez. Sbqa: A self-adaptable query allocation process. In *ICDE* [DBL09], pages 1527–1530.

[QRLV09b] Jorge-Arnulfo Quiané-Ruiz, Philippe Lamarre, and Patrick Valduriez. A self-adaptable query allocation framework for distributed information systems. *VLDB J.*, 18(3):649–674, 2009.

[RD] A. Rowstron and P. Druschel. Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems. In *IFIP/ACM International Conference on Distributed Systems Platforms (Middleware)*, pages pp 329–350.

[Res95] Philip Resnik. Using information content to evaluate semantic similarity in a taxonomy. In *IJCAI*, pages 448–453, 1995.

[RM95] E. Rahm and R. Marek. Dynamic multi-resource load balancing in parallel database systems. In *Proceedings of the Very Large Data Bases Conference (VLDB)*, 1995.

[RMBB89] R. Rada, H. Mili, E. Bicknell, and M. Blettner. Development and application of a metric on semantic nets. *IEEE Transaction on Systems, Man, and Cybernetics*, 19(1):17–30, jan–feb 1989.

[Rou06] Marie-Christine Rousset. Somewhere: a scalable p2p infrastructure for querying distributed ontologies. In *CoopIS/DOA/ODBASE*, 2006.

[RWHBG95] Stephen E. Robertson, Steve Walker, Micheline Hancock-Beaulieu, and M. Gatford. Okapi at trec-3. In *Text REtrieval Conference (TREC-3)*, pages 109–126, 1995.

[SAL+96] M. Stonebraker, P. Aoki, W. Litwin, A. Pfeffer, A. Sah, J. Sidell, C. Staelin, and A. Yu. Mariposa: A Wide-Area Distributed Database System. *Journal on Very Large Data Bases (VLDBJ)*, 5(1):48–63, 1996.

[SBAM05] Christophe Sibertin-Blanc, Frédéric Amblard, and M. Mailliard. A coordination framework based on the sociology of organized action. In Boissier et al. [BPD+06], pages 3–17.

[SDK+94] Michael Stonebraker, Robert Devine, Marcel Kornacker, Witold Litwin, Avi Pfeffer, Adam Sah, and Carl Staelin. An economic paradigm for query processing and data migration in mariposa. In *PDIS '94: Proceedings of the third international conference on on Parallel and distributed information systems*, pages 58–68, Los Alamitos, CA, USA, 1994. IEEE Computer Society Press.

[Seb02] Fabrizio Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.

[She99] Onn Shehory. A scalable agent location mechanism. In M. Wooldridge and Y. Lesperance, editors, *Intelligent Agents VI*. Springer, 1999.

[SKS92] N. G. Shivaratri, P. Krueger, and M. Singhal. Load Distributing for Locally Distributed Systems. *IEEE Computer*, 25(12):33–44, 1992.

[SKW99] Katia Sycara, Matthias Klusch, and Seth Widoff. Dynamic service machmaking among agents in open information environments. *ACM SIGMOD Record, Special Issue on Semantic Interoperability in Global Information Systems*, 28(1):47–53, 1999.

[SLBY+07] Mário J. Silva, Alberto H. F. Laender, Ricardo A. Baeza-Yates, Deborah L. McGuinness, Bjørn Olstad, Øystein Haug Olsen, and André O. Falcão, editors. *Proceedings of the Sixteenth ACM Conference on Information and Knowledge Management, CIKM 2007, Lisbon, Portugal, November 6-10, 2007*. ACM, 2007.

[Smi80] R.G. Smith. The contract net protocol: high level communication and control in a distributed problem solver. *IEEE Transactions on Computers*, C29(12):1104–1113, 1980.

[SMK+01] Ion Stoica, Robert Morris, David Karger, M. Frans Kaashoek, and Hari Balakrishnan. Chord: A scalable peer-to-peer lookup service for internet applications. In *Proceedings of the ACM SIGCOMM '01 Conference*, San Diego, California, August 2001.

[Sow76] John F. Sowa. Conceptual graphs for a database interface. *IBM Journal of Research and Development*, 20(4):336–357, 1976.

[SPVG03] Katia P. Sycara, Massimo Paolucci, Martin Van Velsen, and Joseph A. Giampapa. The retsina mas infrastructure. *Autonomous Agents and Multi-Agent Systems*, 7(1-2):29–48, 2003.

[SS01] Jordi Sabater and Carles Sierra. Regret: reputation in gregarious societies. In *Agents*, pages 194–195, 2001.

[SS04] Steffen Staab and Rudi Studer, editors. *Handbook on Ontologies*. International Handbooks on Information Systems. Springer, 2004.

[SSD+08] Amit P. Sheth, Steffen Staab, Mike Dean, Massimo Paolucci, Diana Maynard, Timothy W. Finin, and Krishnaprasad Thirunarayan, editors. *The Semantic Web - ISWC 2008, 7th International Semantic Web Conference, ISWC 2008, Karlsruhe, Germany, October 26-30, 2008. Proceedings*, volume 5318 of *Lecture Notes in Computer Science*. Springer, 2008.

[SVH04] Nuno Seco, Tony Veale, and Jer Hayes. An intrinsic information content metric for semantic similarity in wordnet. In *Proceedings of ECAI'2004, the 16th European Conference on Artificial Intelligence*, 2004.

[SWY75] G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Commun. ACM*, 18(11):613–620, 1975.

[Syc01] Katia P. Sycara. Multi-agent infrastructure, agent discovery, middle agents for web services and interoperation. In *EASSS*, pages 17–49, 2001.

[TRV96] Anthony Tomasic, Louiqa Raschid, and Patrick Valduriez. Scaling heterogeneous databases and the design of disco. In *ICDCS*, pages 449–457, 1996.

[TRV98a] A. Tomasic, L. Raschid, and P. Valduriez. Scaling access to heterogeneous data sources with disco. *IEEE Trans. on Knowledge and Data Engineering*, 10(5), 1998.

[TRV98b] Anthony Tomasic, Louiqa Raschid, and Patrick Valduriez. Scaling access to heterogeneous data sources with disco. *IEEE Trans. Knowl. Data Eng.*, 10(5):808–823, 1998.

[TvS06] Andrew S. Tanenbaum and Maarten van Steen. *Distributed Systems: Principles and Paradigms*. Prentice Hall International, 2nd edition, 2006.

[VCLV07] Anthony Ventresque, Sylvie Cazalens, Philippe Lamarre, and Patrick Valduriez. Query expansion and interpretation to go beyond semantic interoperability. In *OTM Conferences (1)*, pages 870–877, Vilamoura, Portugal, 2007. short paper.

[VCLV08a] Anthony Ventresque, Sylvie Cazalens, Philippe Lamarre, and Patrick Valduriez. Dealing with p2p semantic heterogeneity through query expansion and interpretation. In *Proceedings of the 2008 International Workshop on Data Management in Peer-to-Peer Systems*, pages 3–10, Nantes, France, 2008.

[VCLV08b] Anthony Ventresque, Sylvie Cazalens, Philippe Lamarre, and Patrick Valduriez. Enrichissement sémantique de requête utilisant un ordre sur les concepts. In *Workshop "Similarité Sémantique", associé à EGC'08*, Sophia-Antipolis, France, 2008.

[VCLV08c] Anthony Ventresque, Sylvie Cazalens, Philippe Lamarre, and Patrick Valduriez. Improving interoperability using query interpretation in semantic vector spaces. In *ESWC'08, European Semantic Web Conference*, pages 539–553, 2008. nominee for best paper award (4/51 accepted papers).

[VCLV08d] Anthony Ventresque, Sylvie Cazalens, Philippe Lamarre, and Patrick Valduriez. Query interpretation to help peers understand each others in semantically heterogeneous systems. In *BDA*, 2008.

[VCLV09] Anthony Ventresque, Sylvie Cazalens, Philippe Lamarre, and Patrick Valduriez. Représentation optimiste de contenu dans les systèmes p2p. In *25èmes Journées Bases de Données Avancées, BDA 2009, Namur (Belgique), 20-23 Octobre 2009, Actes (Informal Proceedings)*, 2009.

[Ven04] Anthony Ventresque. Focus et ontologie pour la recherche d'information. Mémoire de DEA d'informatique, Université de Nantes, France, 2004.

[Ven08] Anthony Ventresque. *Espaces vectoriels sémantiques : enrichissement et interprétation de requêtes dans un système d'information distribué et hétérogène*. PhD thesis, Université de Nantes, 2008.

[VLC05] Anthony Ventresque, Philippe Lamarre, and Sylvie Cazalens. échange d'information grâce a des caractérisations sémantiques. In *Modèles Formels de l'Interaction*, Caen, France, mai 2005.

[Voo94] E. M. Voorhees. Query expansion using lexical-semantic relations. In *Research and Development on Information Retrieval - ACM-SIGIR*, pages 61–70, Dublin, 1994.

[W3C98] W3C. Extensible markup language(xml) 1.0. http://www.w3.org/TR/REC-XML, February 1998. W3C Recommendation Reference: REC-xml-19980210.

[Wie92c] Gio Wiederhold. Mediators in the architecture of future information systems. *Computer*, 25(3):38–49, 1992.

[Wie92d] Gio Wiederhold. Mediators in the architecture of future information systems. *IEEE Computer*, pages 38–49, March 1992.

[Woo97] W. Woods. Conceptual indexing: A better way to organize knowledge. Technical report, Sun Microsystems Laboratories, April 1997.

[WS00] H. Chi Wong and Katia Sycara. A taxonomy of middle-agents for the internet. In *Fourth International Conference on MultiAgent Systems (ICMAS 2000)*, pages 465–466, July 2000.

[Yok08] Makoto Yokoo. Generalized vickrey auction. In Kao [Kao08].

[Zho88] Songnian Zhou. A trace-driven simulation study of dynamic load balancing. *IEEE Trans. Software Eng.*, 14(9):1327–1341, 1988.

[ÖV99] Tamer M. Özsu and Patrick Valduriez. *Principles of Distributed Database Systems*. Prentice Hall, 2nd edition, 1999.

[ÖV04] Tamer Özsu and Patrick Valduriez. *Handbook of Computer Science and Engineering*, chapter Distributed and Parallel Database Systems. CRC Press, 2nd edition, 2004.

Références hypertextes

[WSA] *Web Services Architecture, W3C Working Group Note 11 February 2004.*
http://www.w3.org/TR/ws-arch/

[XQU] *XQuery 1.0: An XML Query Language. W3C Recommendation 23 January 2007.*
http://www.w3.org/TR/xquery/

[XPA] *XML Path Language (XPath), W3C Recommendation 16 November 1999.*
http://www.w3.org/TR/xpath

[CGH] *Conceptual Graphs home page.*
http://conceptualgraphs.org/

[BOI] *BOINC : logiciel ouvert de calcul bénévole et de calcul distribué.*
http://boinc.berkeley.edu/

[SET] *SETI@home.*
http://setiathome.ssl.berkeley.e%du/

[CLI] climateprediction.net *investigates the approximations that have to be made in state-of-the-art climate models.*
http://climateprediction.net/

[DOC] *Docking@Home aims to further knowledge of the atomic details of protein-ligand interactions and, by doing so, will search for insights into the discovery of novel pharmaceuticals.*
http://docking.cis.udel.edu/

[WOR] *WordNet : a lexical database for the English language.*
http://wordnet.princeton.edu/

[ELL] *Ellen Voorhees defends Cranfield (TREC) evaluation.*
http://www.searchenginecaffe.com%/2008/04/ellen-voorhees-defends-cranfield-trec.html

[WWW 39] *Service Google Adword*.
http://adwords.google.com/

[WWW 40] *Les pages cachées google*.
http://www.googleguide.com/cache%d_pages.html

[WWW 43] *The eBay System*.
http://business.ebay.com/

[WWW 45] *The Grid4All STREP project*, 2006-2009.
http://grid4all.elibel.tm.fr/

[WWW 46] *Projet ARA Masse de données Respire*, 2006-2008.
http://respire.lip6.fr/

[WWW 47] Orbacus. *ORB Corba ORBACUS*.
http://www.ooc.com/ob

[WWW 48] *Projet ANR DataRing (programme Future Networks and Services (VERSO))*, 2009-2011.
http://www.lina.univ-nantes.fr/p%rojets/DataRing/

[WWW 51] *Comparateur de prix*.
http://www.kelkoo.fr/

[WWW 52] *The Open Agent Architecture*.
http://www.ai.sri.com/~oaa/main.%html

[WWW 53] *Fondation Védiorbis pour la recherche et l'emploi*.
http://www.fdf.org/La-Fondation-%de-France/Fonds-et-fondations-sous-egide/Toutes-les-fondations/Vediorbis-pour-%la-recherche-et-l-emploi

[WWW 54] *Fondation de France*.
http://www.fdf.org/

[WWW 55] *Société e-Manation (rachetée par lingway)*.

.....http://www.e-manation.com/

[WWW56] *KQML as an agent communication language.*

.....http://www.cs.umbc.edu/kqml/

[WWW57] FIPA. *Agent Communication Language*, 1997.

.....http://www.fipa.org/spec/index.h%tml

[WWW58] *Foundation for Intelligent Physical Agents*, 1997.

.....http://www.fipa.org/spec/index.h%tml

[WWW59] *Semantic Information System (a semantic registry of ressources and services for the Grid4All Projet).*

.....http://icsd-ai-lab.aegean.gr:808%0/grid4all_sis/

[WWW60] *Internet des Objets, Internet du Futur. Conférence de la Présidence Française de l'Union Européenne, 2008.*

.....http://www.internet2008.eu/

[WWW62] *CORBA , bject Management Group.*

.....http://www.corba.org/

[WWW63] *Notificaiton Service, Object Management Group.*

.....http://www.omg.org/technology/do%cuments/formal/notification_service.htm

[WWW64] *Trading Service, Object Management Group.*

....http://www.omg.org/technology/do%cuments/formal/trading_object_service.htm

[WWW66] *Outil d'interception et de trace des communications entre objets CORBA distants.*

.....http://corbatrace.sourceforge.ne%t/

[WWW70] *The InfoSleuth agent system.*

.....http://www.argreenhouse.com/Info%Sleuth/

[WWW71] *RETSINA, open multi-agent system.*

.....http://www.cs.cmu.edu/~softagent%s/retsina.html

[WWW72] *A cranfield corpus repository*.

.....http://www.dcs.gla.ac.uk/idom/ir%_resources/test_collections/cran/

[WWW73] *PeerSim: A Peer-to-Peer Simulator*.

.....http://peersim.sourceforge.net/

[WWW74] *The Chord project*.

.....http://pdos.csail.mit.edu/chord/%

[WWW75] *Pastry : a substrate for peer-to-peer applications*.

.....http://www.freepastry.org/

Table des figures

Table des matières

Annexes

Annexe 1 - Revue l'Objet 2000

[CDJL02] Sylvie Cazalens, Emmanuel Desmontils, Christine Jacquin et Philippe Lamarre. *Sources d'informations et de connaissances : de la gestion locale à la recherche distribuée*, RSTI L'Objet, 8(4)(47-69), 2002.

De la gestion locale à la recherche distribuée dans des sources d'informations et de connaissances

Sylvie Cazalens — Emmanuel Desmontils — Christine Jacquin

Philippe Lamarre

IRIN - Institut de Recherche en Informatique de Nantes
2, rue de la Houssinière BP 92208, F-44322 Nantes Cedex 3
(Cazalens, Desmontils, Jacquin, Lamarre)@irin.univ-nantes.fr

RÉSUMÉ. *Cet article présente une vue générale du projet Bonom qui a pour cadre la recherche et l'exploitation d'informations et de connaissances dans un environnement hautement distribué, hétérogène, évolutif, où les sources peuvent être très nombreuses et potentiellement concurrentes. Pour exploiter la connaissance au niveau d'une source, le projet propose la construction et l'exploitation locale d'un index sémantique structuré basé sur l'utilisation d'ontologies. Pour traiter la recherche dans l'ensemble des sources, le projet adopte une vue orientée "agents". Des agents intermédiaires spécifiques assurent une médiation entre des agents personnels et des agents sites représentant les sources d'informations et de connaissances. Les agents sont situés par rapport à une hiérarchie de thèmes.*

ABSTRACT. *This paper gives a general overview of the Bonom project. The project deals with searching for information and knowledge within a distributed, evolving framework where information sources are heterogeneous, numerous and possibly competing. In order to exploit knowledge within an information source, a structured semantic index is built locally, using ontologies. For searching all the sources, an agent oriented view of the problem is adopted. Specific middle agents mediate between personal agents and site agents which represent the information and knowledge sources. The agents are situated with respect to a hierarchy of topics.*

MOTS-CLÉS : *Approche multi-agent, recherche d'informations et de connaissances distribuées, web sémantique, Traitement Automatique du Langage Naturel (TALN).*

KEYWORDS: *Multi-Agent Approach, Searching for Distributed Information and Knowledge, Semantic Web, Natural Language Automatic Treatment.*

1. Introduction

Dans le contexte de l'intranet d'une grande entreprise ou dans celui d'internet, les informations sont naturellement distribuées. Le problème de l'obtention d'informations pertinentes par un utilisateur, et l'inadéquation des moteurs de recherche classiques ont été largement soulignés dans la littérature ces dernières années.

Ce manque de qualité peut trouver plusieurs explications. L'une d'elles est que la plupart des recherches effectuées par mots-clés sont réalisées sur une base principalement syntaxique. Or, plusieurs projets [FEN 98a, HEF 99, HEN 00] ont montré l'utilité de prendre en compte la sémantique des documents pour mieux répondre à la demande d'un utilisateur. En effet, les raisonnements visant à déterminer l'ensemble des informations pertinentes peuvent être menés à un niveau sémantique. De ce fait, les réponses fournies à l'utilisateur peuvent comporter aussi bien des données ou documents bruts, que des informations sémantiques supplémentaires comme leur contexte d'utilisation, l'ontologie utilisée etc. L'utilisateur obtient ainsi plus de connaissances.

Le manque de qualité a aussi pour origine la centralisation de l'indexation. Comme le nombre de pages va sans cesse croissant, leur indexation par un moteur de recherche centralisé est de moins en moins fréquente. En conséquence, les informations retournées ne sont pas forcément à jour. De plus, un moteur généraliste peut difficilement proposer une indexation et une connaissance d'un site aussi fine que ce que pourrait faire un moteur spécifique à ce site. C'est pourquoi de plus en plus d'entreprises choisissent de ne plus autoriser l'indexation de leur site et proposent leur propre moteur de recherche. Ceci a aussi pour conséquence intéressante qu'elles maîtrisent d'autant mieux leur politique de communication vis-à-vis des utilisateurs. Pour l'utilisateur, la contrepartie peut être que les informations risquent d'être plus difficiles d'accès.

Ces deux aspects suggèrent une vision du processus de recherche où :

– les sources d'informations ne sont plus passives au sens où elles ne subissent plus un processus d'indexation extérieur, mais deviennent actives : elles disposent d'un certain nombre de mécanismes propres qui leur permettent de gérer et exploiter la connaissance présente dans leurs documents, de répondre de manière pertinente aux requêtes d'utilisateurs qui leur parviennent, d'exprimer leur compétences, de prendre en compte des retours d'utilisateurs, etc ;

– l'exploitation de ces sources distribuées passe par des processus permettant de déterminer les sources qui sont les plus pertinentes vis-à-vis de la requête d'un utilisateur, d'assurer qu'elles traitent effectivement la requête et que les réponses parviennent, sous forme synthétique ou non, à l'utilisateur. Le problème est a priori difficile, car cette approche autorise une grande hétérogénéité des sources. Les processus utilisés peuvent être distribués entre utilisateurs et sources d'informations ou faire intervenir d'autres entités logicielles délocalisées, communicantes, organisées, etc.

Cette vision sert de base au projet *Bonom*[1] développé au sein du thème Ingénierie des Connaissances de l'IRIN et dont cet article présente une vue générale. Bonom a pour objectif la recherche et l'exploitation d'informations et de connaissances dans un environnement hautement distribué, hétérogène et évolutif et où les sources peuvent être très nombreuses. Sa spécificité réside à plusieurs niveaux. En ce qui concerne l'exploitation de la connaissance au niveau d'une source, le projet propose un certain nombre de mécanismes basés sur l'utilisation d'ontologies : indexation semi-automatique de pages web permettant d'associer chaque page aux concepts de l'ontologie, vérification de l'adéquation d'un ensemble de pages à une ontologie, reformulation de requêtes d'utilisateurs en se referrant aux concepts de l'ontologie... Ces mécanismes viennent en complément d'autres méthodes qui utilisent des annotations manuelles pour expliciter la sémantique. Pour traiter la recherche dans l'ensemble des sources, le projet adopte une vue orientée « agent ». Les sources sont vues d'un point de vue extérieur comme *fournisseurs* d'informations et de connaissances, appelés *agents site*. Chaque utilisateur (ou groupe d'utilisateurs) dispose d'un *agent personnel* qui est vu de l'extérieur comme un *demandeur* d'informations et de connaissances. Entre agents personnels et agents sites, des *agents intermédiaires* assurent une sorte de médiation pour permettre aux requêtes des utilisateurs de parvenir aux agents sites les plus pertinents. Le système se distingue d'autres travaux [NOD 99, KUO 95, SYC 99] utilisant des agents intermédiaires sur les points suivants : (a) agents intermédiaires et agents sites sont situés par rapport à une hiérarchie de domaines informationnels (ou thèmes) ; ils sont organisés en groupes relevant chacun d'un thème ; (b) le nombre d'agents intermédiaires peut croître lorsque le nombre de sources croit, sans lourdes conséquences sur la complexité des communications entre agents ; (c) le fonctionnement des agents intermédiaires ne relève pas des classes d'agents intermédiaires habituellement utilisées.

Pour parvenir au système final, le projet s'est organisé suivant plusieurs axes de recherche, dont deux ont été privilégiés dans un premier temps. D'une part, il fallu travailler l'organisation générale des agents et la circulation des requêtes. La spécification de cette organisation, tant dans ses aspects statiques que dynamiques [CAZ 01] nous a permis de fixer l'ossature générale d'une organisation Bonom. Ce travail sert de référence à la fois pour mener des études théoriques sur l'organisation (preuves, calculs, simulations...) et pour implémenter les différents prototypes. D'autre part, puisque le projet privilégie l'autonomie des agents sites en ce qui concerne leur politique de communication, il nous a paru important d'explorer de nouveau outils pour améliorer la gestion des sites. L'indexation semi-automatique basée sur l'utilisation d'ontologies en est un exemple [DES 01]. Les résultats obtenus dans ces deux axes, qui sont présentés de manière relativement indépendante dans cet article, permettent d'envisager la réalisation d'un prototype « commercial » dans le cadre d'un transfert de technologie en cours.

1. *Bonom* n'est pas un acronyme, juste l'usage commun des membres qui interviennent dans le projet.

L'article est organisé comme suit. La section 2 concerne l'explicitation de connaissances dans des pages web. Elle décrit en particulier comment est obtenu un index sémantique structuré. La section 3 expose deux utilisations possibles de cet index : pour vérifier l'adéquation entre un site et une ontologie et pour traiter des requêtes adressées au site indexé. Ces deux sections illustrent donc des mécanismes qui permettent d'exploiter la connaissance au niveau d'une source. Les sections suivantes concernent la recherche d'informations et de connaissances dans les sources distribuées. La section 4 traite de l'approche orientée « agent » qui peut être utilisée pour résoudre ce problème. La section 5 concerne l'organisation d'agents Bonom. Elle présente d'abord les principes de base, propose ensuite un exemple de traitement d'une requête, puis décrit l'implémentation du prototype réalisé pour tester la validité de l'organisation. Enfin nous concluons et dégageons quelques perspectives du projet Bonom.

2. Explicitation de la connaissance dans les pages web

2.1. *Différentes approches*

De nombreux travaux en ingénierie des connaissances et plus généralement dans la communauté du web sémantique s'intéressent à doter les informations disponibles sur le web d'annotations sémantiques pour ensuite les exploiter à des fins de recherche d'informations. Ces communautés s'intéressent à définir des langages de représentation des connaissances (dans ce contexte, les connaissances sont des ontologies) afin d'annoter des pages web.

Divers langages de représentation d'ontologies préexistent : Ontolingua [GRU 93] basé sur KIF [GEN 92], les logiques de descriptions comme Loom [NAP 97], les *Frame* logiques [KIF 89]... Ces langages sont tous basés sur la logique des prédicats du premier ordre mais ils ont des pouvoirs d'expressivité différents et ne sont pas complètement adaptés pour représenter des connaissances issues du web. A l'heure actuelle, XML devient le langage standard pour échanger des données sur le web et donc naturellement, ce formalisme tend à être utilisé dans le cadre de représentation d'ontologies. Dans ce contexte, différents langages basés sur XML ou/et sur des formalismes comme les logiques de descriptions ont été mis au point. SHOE [HEF 99], OML (Ontology Markup Language) [KEN 99], OIL (Ontology interchange Language) [FEN 00] qui se base sur RDF et sur les RDF Schemas [W3C 00], en sont des exemples.

Dernièrement, le langage DAML+OIL [HEN 00] a été développé conjointement par des groupes de chercheurs américains et européens. Ce langage permet de manipuler des taxonomies et des relations logiques entre entités plus complexes qu'avec les autres formalismes. Ces différents langages ou systèmes sont utilisés afin d'apposer des annotations sémantiques au sein du code HTML des pages et sont la plupart du temps effectuées manuellement par un spécialiste (KA2 [FEN 98a], SHOE [HEF 99], DARPA DAML program [HEN 00]). Ces données annotées sémantiquement sont ensuite exploitées par un moteur de recherche spécifique (par exemple, Ontobroker dans

le projet KA2). Ces moteurs se basent donc sur les annotations des pages et sur les règles d'inférence inhérentes à l'ontologie pour retrouver des informations pertinentes dans les documents. Les informations renvoyées sont donc de ce fait très précises et de granularité très fine. La contrepartie majeure à ces approches est la difficulté de modifier des pages ou d'en créer d'autres. En effet, ces mises à jour demandent de renouveler l'annotation des pages. De plus le passage à l'échelle paraît quelque peu utopique. Il faudrait un consensus global où toutes les personnes qui déposent une page sur le web accepteraient d'annoter leurs pages *via* une ontologie. Ce qui reste un travail fastidieux relevant des compétences d'un spécialiste [HEF 99].

Une autre communauté qui est celle du domaine du KDD (Knowledge Discovery in Databases) s'intéresse à extraire des connaissances des sites web. [FEL 95] appliquent des techniques de KDD sur les mots-clés qui sont attachés aux documents et qui sont alors considérés comme des attributs. Des calculs statistiques permettent de découvrir des règles d'association et des patrons intéressants. D'autres chercheurs [LIN 98] utilisent des termes extraits automatiquement des documents pour les caractériser et trouver des associations qui les relient aux documents. D'autres approches utilisent des techniques de KDD après avoir exploité des techniques d'extraction d'informations qui transforment l'information contenue dans les textes en des informations structurées dans une base de données [COW 96]. D'autres approches [LOH 00] allient des techniques de TALN (Traitements Automatiques des Langues) à des techniques de type KDD pour extraire automatiquement des informations à partir de documents. Ils ne travaillent plus en utilisant des mots-clés comme attribut mais en travaillant à partir de concepts qui sont extraits *via* l'utilisation d'un *thesaurus*. L'approche des derniers auteurs semblent la plus intéressante car ils ne travaillent plus au niveau des simples mots-clés mais au niveau des concepts qui sont inclus dans les pages. [MAT 00] soutiennent d'ailleurs que pour une extraction efficace de connaissances, une connaissance *a priori* sur le domaine (par exemple des ontologies) est essentielle.

Dans la section suivante, nous présentons notre approche permettant de repérer la connaissance dans les pages web et d'indexer un site à partir de cette connaissance. La connaissance *a priori* sur le domaine que nous prenons en compte, est une ontologie orientée terminologie [DES 01], où les étiquettes relatives à chaque concept ont été désambiguïsées. L'indexation s'effectue à partir des concepts inclus dans les pages qui sont repérés à l'aide de techniques de traitement automatique des langues.

2.2. *Indexation sémantique structurée*

Notre objectif est de construire un index structuré des pages d'un site web en fonction d'un domaine de connaissance. La structure est donnée par une ontologie de ce domaine. Il est à noter que pour l'heure, nous avons principalement travaillé sur des pages HTML, mais que notre étude peut s'étendre à d'autres sources d'information. Le processus d'indexation comporte les phases suivantes (figure 1) :

1) d'abord, pour chacune des pages est constitué un index à plat des termes, avec leur fréquence respective (nombre d'occurrences) pondérée par les marqueurs HTML qui leurs sont relatifs. Il est à noter que les poids associés aux marqueurs ont été définis de manière expérimentale ;

2) ensuite, un *thesaurus* permet de déterminer tous les concepts candidats associés aux termes précédemment acquis. Dans notre expérimentation, nous utilisons le *thesaurus* WordNet [MIL 90] ;

3) pour chaque concept candidat, le calcul d'un coefficient de représentativité permet d'évaluer sa représentativité dans la page étudiée. Ce calcul s'appuie sur la fréquence pondérée et sur une mesure de similarité [WU 94] entre concepts. Cette mesure permet aussi de déterminer en contexte le sens le plus probable d'un terme. Ainsi, un concept sera d'autant plus important dans une page qu'il sera fortement reliés aux autres concepts de cette page. Cette évaluation permet de relativiser la fréquence pondérée. Elle accentue l'importance des concepts fortement reliés aux autres et diminue celle des concepts plus ou moins isolés (même s'ils ont une fréquence importante) ;

4) parmi tous les concepts retenus à la phase précédente, un filtre est réalisé *via* l'ontologie et la représentativité des concepts. Il permet de sélectionner les concepts présents dans l'ontologie et dont la représentativité dépasse un certain seuil. Ceci permet ensuite de construire l'index structuré en associant la page concernée aux concepts de l'ontologie qu'elle contient.

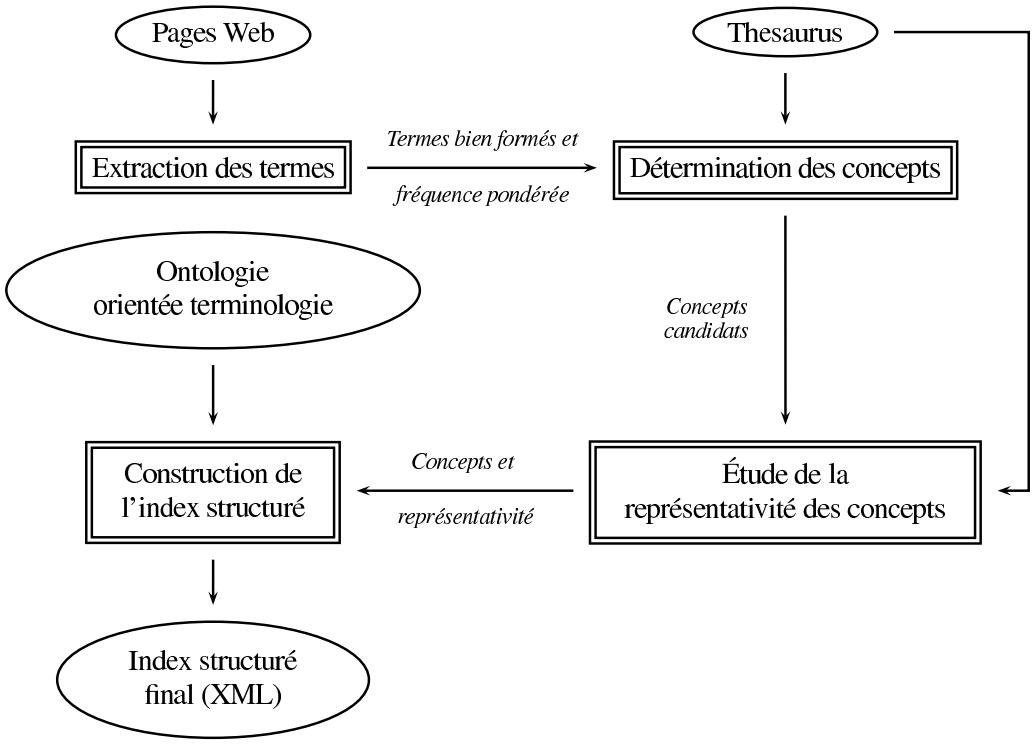


Figure 1. *Le processus d'indexation*

Dans notre approche, l'index est construit indépendemment des pages du site et est stocké sous un format XML. En cela, elle diffère notablement des approches basées sur l'annotation de pages web. En effet, notre processus est semi-automatique et il permet d'avoir facilement un point de vue global sur le site. Il permet aussi d'indexer des sites dont le code source des pages ne peut être modifié. Nous ne le considérons pas comme totalement automatique, car des ajustements peuvent être effectués par l'utilisateur en fin de processus. La contrepartie à cette automatisation est, bien évidemment, une moins bonne précision du traitement.

Par rapport aux techniques de type KDD comme [LOH 00], nous travaillons aussi au niveau des concepts et non plus au niveau de simples mots-clés.Mais nous avons pris l'option d'avoir des traitements linguistiques beaucoup plus fins et surtout nous privilégions une connaissance *a priori* sur le domaine (une ou des ontologies du domaine). [LOH 00] utilisent de la connaissance *a priori* sur le domaine (un *thesaurus*) exclusivement pour extraire les concepts des pages. Dans notre approche, les concepts sont aussi extraits des pages à l'aide d'un *thesaurus*, mais l'indexation proprement dite s'appuie aussi sur une ontologie du domaine.

Le projet qui se rapproche le plus de notre étude est le projet CHIMERE [SEG 00]. Ce projet s'attache à extraire des informations de formulaires à partir d'une ontologie du domaine et de traitements linguistiques. Ces formulaires se composent de zones de saisie et de données textuelles. Mais les données textuelles sont très réduites. Les techniques linguistiques employées dans ce contexte ne peuvent s'adapter à notre cas d'étude qui traite des pages web comportant principalement de grandes zones de données textuelles, souvent très peu structurées.

3. Exploitation de l'index sémantique structuré

3.1. *Évaluation de l'adéquation entre un site et une ontologie*

Afin d'évaluer l'adéquation entre l'ontologie et le site web cinq coefficients sont calculés. Lorsque plusieurs ontologies sont susceptibles d'être associées au site, la meilleure peut être choisie eu égard à cette évaluation. Les quatre premiers coefficients (normalisés entre 0 et 1) définissent :

– la proportion de concepts présents directement dans les pages (Degré d'Indexation Direct ou DID) ;

– la proportion de concepts présents indirectement dans les pages (Degré d'Indexation Indirecte ou DII) qui est calculée en tenant compte de la relation générique/spécifique et du DID ;

– la proportion de pages concernées par l'ontologie (Degré de Couverture de l'Ontologie ou DCO),

– la représentativité moyenne des concepts sélectionnés (RMC).

Actuellement, les coefficients (DID, DII, DCO et RMC) sont évalués pour différents seuils concernant le coefficient de représentativité (de 0 et 1 par pas de 0,02). Ensuite, pour chacun de ces coefficients est calculée sa moyenne pondérée (le poids étant la valeur du seuil). Cette pondération permet de donner plus de poids aux concepts les plus représentatifs des pages. Une ontologie représentative d'un site possède des coefficients proches de 1. Notons toutefois que cette évaluation dépend aussi du *thesaurus* utilisé puisqu'elle dépend des relations entre concepts. Finalement, l'évaluation globale de l'indexation (le DAOS : Degré d'Adéquation Ontologie Site) est une combinaison linéaire de ces moyennes pondérées. Pour l'instant, les coefficients sont évalués de manière expérimentale.

Après avoir analysé manuellement un échantillon représentatif des résultats d'indexation, pour les différents seuils testés, l'expérience montre qu'une valeur de 0.3 pour le coefficient de représentativité donne de bons résultats. En dessous de ce seuil, trop de concepts peu représentatifs du contenu sont conservés. Pour ce seuil, la discrimination des concepts est relativement efficace (sachant qu'elle est d'autant plus efficace que les pages sont de plus grande dimension).

La figure 2 présente les résultats de l'analyse de l'indexation d'un site[2] de 1315 pages HTML. La figure 3 présente un extrait de l'index structuré pour le seuil 0.3. Ce site concerne le département « Computer Science » de l'université de Washington. Il a été choisi parce qu'à priori intéressant par rapport à l'ontologie. Cependant, le degré d'adéquation de ce site par rapport à l'ontologie (DAOS) n'est pas très élevé (46). Ceci peut s'expliquer par le fait que l'ontologie utilisée (c'est une adaptation de celle du projet SHOE qui comporte 79 concepts), n'est pas exhaustive du point de vue du domaine qu'elle devrait recouvrir. Par exemple, le site étudié comporte de nombreuses pages personnelles qui sont rarement indexées *via* l'ontologie.

Ces évaluations du site permettent donc de caractériser l'adéquation entre le site et l'ontologie. Suite à cela, plusieurs comportements sont envisageables :

1) les coefficients étant corrects, l'indexation est conservée et exploitée ;

2) les coefficients ne sont pas satisfaisants :

a) les pages qui ne conviennent pas sont supprimées (en particulier avec un DCO et/ou un RMC faibles),

b) l'ontologie utilisée est ajustée (en particulier avec un DID faible),

c) le processus est relancé avec une nouvelle ontologie (les valeurs de l'ensemble des coefficients sont très faibles).

3.2. *Traitement de requêtes concernant le site indexé*

Les moteurs courants font des indexations à partir de termes. Les requêtes sont souvent des listes de mots-clés connectés par des opérateurs. Ces opérateurs sont les

2. « http ://www.cs.washington.edu/ »

opérateurs logiques (et, ou, non, et non, +, -...) et parfois l'opérateur proche (distance entre deux termes en nombre de termes). Quand il y a utilisation de linguistique, c'est le plus souvent pour seulement l'analyse de la requête, c'est-à-dire pour avoir la possibilité de formuler des requêtes en langage naturel.

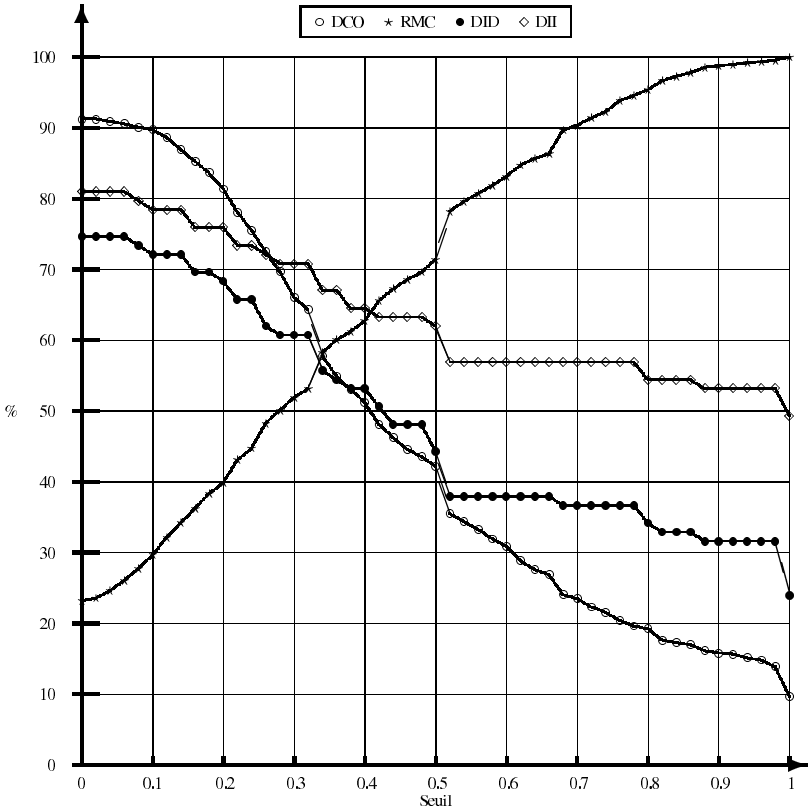


Figure 2. *Exemple de résultats de l'analyse d'une indexation*

Dans notre cadre, il est possible d'utiliser l'ontologie orientée terminologie et l'index structuré pour optimiser le traitement d'une requête. Une fois l'indexation structurée terminée et validée, l'index produit permet alors de traiter des requêtes sur le site non pas au niveau terminologique mais au niveau conceptuel. Cette amélioration passe d'abord par une reformulation de la requête en terme de concepts puis, ensuite par une exploitation des opérateurs utilisés avec un point de vue sémantique et finalement par la construction de réponses plus riches sur le plan de l'évaluation de leur qualité.

Un premier bénéfice à utiliser des ontologies concerne donc la reformulation des requêtes. Naturellement, l'utilisateur d'un moteur propose un ensemble de termes connectés par des opérateurs logiques qui sont souvent ambigus. Aussi, nous proposons un système permettant de reformuler la requête en remplacant les termes par les concepts qu'ils représentent. Cette reformulation s'effectue sur la base de(s) l'ontologie(s) du site. En effet, à un terme donné peuvent correspondre plusieurs concepts. La recherche du concept associé à un terme donné s'effectue d'abord en recherchant, dans les ontologies, les concepts candidats et, ensuite, en étudiant les autres concepts de la

requête et les connecteurs utilisés. Finalement, s'il reste encore plusieurs concepts candidats, un retour à l'utilisateur permet d'effectuer le bon choix. Dans le cas où des termes ne peuvent être associés à des concepts dans les ontologies, ils sont considérés comme invalides vis-à-vis du site et, suivant les opérateurs utilisés, soit ils sont supprimés, soit la requête n'a pas de réponse.

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<!DOCTYPE ontology SYSTEM "http://.../onto.dtd">
<ontology id="university-ont" version="3.0" description="">
  <def-category name="University" short="university"
                isa="EducationOrganization">
    <sense name="University" no="3" origin="wn" convenience="1.0">
      <synset>university#3</synset>
      <page name="http://www.cs.washington.edu/news/recent/latest10.html"
            frequence="0.4" representativeness="0.85"/>
      <page name="http://www.cs.washington.edu/news/"
            frequence="0.37" representativeness="0.85"/>
      <page name="http://www.cs.washington.edu/commercialization/policyprop.html"
            frequence="0.3" representativeness="0.75"/>
      <page name="http://www.cs.washington.edu/homes/ghulten/"
            frequence="0.38" representativeness="0.85"/>
      <page name="http://www.cs.washington.edu/news/chan.html"
            frequence="0.86" representativeness="0.85"/>
      <page name="http://www.cs.washington.edu/homes/lazowska/"
            frequence="1.0" representativeness="0.0"/>
      <page name="http://www.cs.washington.edu/homes/tiwary/"
            frequence="0.5" representativeness="0.55"/>...
    </sense>
  </def-category>
  <def-category name="Department" short="university department"
                isa="EducationOrganization">
    <sense name="Department" no="1" origin="wn" convenience="1.0">
      <synset>department#1,section#11</synset>
      <page name="http://www.cs.washington.edu/education/courses/590m/"
            frequence="0.4" representativeness="0.65"/>
      <page name="http://www.cs.washington.edu/leadership/"
            frequence="0.5" representativeness="0.35"/>
      <page name="http://www.cs.washington.edu/homes/lazowska/chair/summer.support.html"
            frequence="0.67" representativeness="0.75"/>
      <page name="http://www.cs.washington.edu/education/courses/590b/"
            frequence="0.4" representativeness="0.65"/>
      <page name="http://www.cs.washington.edu/info/public/"
            frequence="1.0" representativeness="0.75"/>
      <page name="http://www.cs.washington.edu/education/courses/590zpl/"
            frequence="0.4" representativeness="0.65"/>
      <page name="http://www.cs.washington.edu/education/courses/510/"
            frequence="0.4" representativeness="0.65"/>
      <page name="http://www.cs.washington.edu/education/courses/490ap/"
            frequence="0.33" representativeness="0.65"/>
      <page name="http://www.cs.washington.edu/homes/carlson/"
            frequence="0.33" representativeness="0.35"/>
      <page name="http://www.cs.washington.edu/"
            frequence="0.33" representativeness="0.65"/>...
    </sense>
  </def-category>...
</ontology>
```

Figure 3. *Extrait de l'index structuré*

Une fois la requête reformulée sous forme conceptuelle, elle est exploitée pour rechercher sur le site les pages répondant à cette requête. L'indexation structurée permet d'améliorer l'interprétation des opérateurs utilisés. Si, pour l'instant, les opérateurs **et** et **ou** restent inchangés par rapport aux outils classiques, le passage au niveau conceptuel fait que la sémantique des opérateurs **non** et **proche** évoluent assez fortement. Le **non** porte maintenant sur un concept. Il faut donc aussi ajouter à ce concept tous ceux qui lui sont plus spécifiques (en relation plus ou moins directe avec lui selon la relation « ISA »). Ainsi, toutes les pages contenant ce concept ou un concept plus spécifique seront rejetées. L'opérateur **proche** ne porte plus sur une distance en nombre de mots comme dans les moteurs classiques mais une distance entre concepts selon la mesure

de similarité [WU 94] utilisée pour le calcul de la représentativité. La proximité de deux concepts est donc fonction du nombre minimal de relations « ISA » à emprunter pour aller de l'un à l'autre. Ainsi, dans notre cadre, l'opérateur proche devient un opérateur unaire et permet d'ajouter tous les concepts proches sémantiquement du concept spécifié.

4. Recherche dans des sources distribuées : une approche orientée agent

Que l'on s'intéresse à l'intranet d'une grande entreprise ou au cas d'internet, les sources d'informations et de connaissances sont, à l'évidence, hétérogènes et largement distribuées. Le problème de leur exploitation par des utilisateurs, eux aussi distribués, en est rendu d'autant plus complexe. Sur la base des travaux mentionnés et décrits dans les sections précédentes, nous faisons l'hypothèse que ces sources sont actives, au sens où elles disposent de mécanismes leur permettant de gérer et d'exploiter la connaissance présente dans leurs documents, de répondre de manière pertinente à des requêtes d'utilisateurs, d'exprimer leurs compétences, etc. Il nous a donc semblé naturel d'adopter une vue orientée agent.

De manière générale, il est possible de distinguer deux catégories d'agents [DEC 97] : les *fournisseurs* (*providers*) et les *demandeurs* (*requesters*). Un *fournisseur* est un agent qui propose un service, comme la recherche d'informations et de connaissances dans la source qu'il représente. Un *demandeur* représente un (ou des) utilisateur(s) et émet des requêtes. Les fournisseurs traitent les requêtes qui leur parviennent. Le problème est que très souvent, un demandeur ne peut déterminer directement par lui-même quels sont les fournisseurs les mieux à même de traiter sa requête. La nécessité d'un mécanisme de liaison apparaît donc. Ce problème, parfois appelé « problème de connexion » [DEC 97], admet plusieurs solutions.

On peut penser, par exemple, qu'un agent qui ne peut traiter une requête localement, utilise ses accointances pour la soumettre à d'autres agents. Une étude théorique de ce processus, qui peut être récursif, est menée dans [SHE 99]. Le nombre d'accointances considéré est de quatre agents et les calculs montrent de bons résultats en moyenne. Cependant, nous ne lui connaissons pas de mise en œuvre pratique.

Une méthode maintenant largement répandue consiste à utiliser un agent *intermédiaire*. Celui-ci facilite le processus de recherche entre demandeurs et fournisseurs. Dans le domaine des bases de données distribuées, le concept de *médiateur* a été introduit au début des années 90 [WIE 92]. Il suppose la présence d'un schéma de bases de données global intégrant les schémas de chaque source. Dans le domaine des agents intelligents, des éléments en vue d'une classification des types d'agents intermédiaires sont proposés dans [DEC 97, WON 00]. Aucune terminologie ne semble adoptée actuellement. Le terme de *facilitateur* est utilisé dans plusieurs travaux (par exemple [FIN 94, SRI , Ret]) pour désigner de manière générale un agent qui coordonne les interactions avec d'autres agents. Leurs propriétés peuvent cependant varier

sensiblement d'un système à l'autre. Les deux catégories d'agents les plus connues sont peut-être les *broker* et les *matchmakers*. Leur principe commun [KLU 99] est que les fournisseurs doivent déclarer toutes leurs capacités au *broker* (respectivement *matchmaker*) qui les enregistre. Lorsqu'un demandeur envoie une requête au *broker* (resp. *matchmaker*), ce dernier calcule les fournisseurs les plus appropriés en utilisant sa vue globale des capacités des différents fournisseurs. Le *broker* se distingue du *matchmaker* en interrogeant directement les fournisseurs qu'il a sélectionnés et en collectant leurs réponses, qu'il fait lui-même parvenir au demandeur. Ainsi le demandeur peut rester anonyme et le *broker* peut gérer la charge des fournisseurs. Au contraire, le *matchmaker* envoie au demandeur la liste des fournisseurs avec leurs capacités, lui laissant le soin d'effectuer ses propres choix et de gérer la communication avec les fournisseurs. Dans les deux cas, les déclarations de capacités et l'appariement entre les requêtes peuvent être aussi bien syntaxiques que sémantiques. Par exemple, dans [SYC 99, NOD 99], il est possible de spécifier l'ontologie utilisée.

Chacune de ces solutions présente des avantages et il est nécessaire de les confronter avec les caractéristiques de l'application visée pour faire un choix.

Les approches de type médiateur supposent un degré d'hétérogénéité suffisamment faible pour permettre la constitution d'un schéma général. Les réalisations relevant d'une approche utilisant un *broker* ou un *matchmaker* [NOD 99, BOU 00] supportent en général un degré d'hétérogénéité des fournisseurs plus important, en particulier grâce à l'utilisation d'un langage de déclaration des capacités [SYC 99]. La distribution et la quantité des agents impose de considérer la possibilité d'un passage à l'échelle des solutions étudiées. La centralisation par un agent intermédiaire unique n'étant pas satisfaisante [NOD 99], l'utilisation de plusieurs de ces agents s'envisage naturellement. Plusieurs techniques peuvent être envisagées : distribution, fédération, hiérarchie...sans qu'aucune ne se dégage nettement. Par exemple, comment gérer la déclaration d'un fournisseur à de multiples *brokers* pour éviter sa surcharge éventuelle ? D'un autre côté, si un fournisseur ne se déclare qu'à un seul agent intermédiaire, il faut assurer que les agents intermédiaires puissent dialoguer de façon à prendre en compte le fait qu'un agent intermédiaire se déclare à un autre pour obtenir des réponses à une requête envoyée au premier.

De plus, des facteurs autres que des considérations techniques, comme la politique générale et la stratégie d'une société, peuvent intervenir. Par exemple, un fournisseur utilisant un *broker* devra accepter de lui déléguer la gestion de sa charge et la sélection des requêtes qu'il devra traiter. Dans une situation concurrentielle, la sensibilité de ces points s'exacerbe. La répartition des requêtes entre différents concurrents prend toute sa dimension stratégique. Une entente concernant le langage commun de description des connaissances ou des capacités ainsi que la gestion de l'intermédiaire devient nécessaire. Cela suppose un degré de coopération et de confiance qui peut être difficile à atteindre. Les projets consistant à développer des systèmes d'agents au sein d'une entreprise peuvent être considérés comme travaillant dans un cadre non concurrentiel (ou faiblement) et où un certain degré d'uniformisation et de maîtrise peut être atteint. Par exemple CoMMA, travaillant dans un tel cadre, suppose «... *que la communauté*

de l'entreprise partage une certaine vue du monde qui permet de penser qu'un consensus ontologique est possible…» [GAN 00]. C'est pourquoi une approche utilisant des agents intermédiaires de type *broker* ou *matchmaker* semble adaptée.

Étant donné le cadre dans lequel nous nous sommes placés, nous devons considérer des agents hétérogènes, nombreux, distribués et évolutifs. L'évolution du système, ainsi que celle des agents, n'est maîtrisée par aucune autorité centrale. Des agents peuvent apparaître ou disparaître dynamiquement, leurs capacités évoluer indépendamment les uns des autres. Aucun système proposé que nous avons confronté à cette réalité ne nous a donné entière satisfaction. C'est pourquoi nous nous sommes intéressés à l'étude d'un système pouvant répondre de manière plus satisfaisante à ce problème. La section suivante détaille les spécificités de notre approche.

5. Bonom : des agents organisés suivant une hiérarchie de thèmes

5.1. *Principes de base*

Même en présence d'un environnement fortement concurrentiel, il est envisageable d'obtenir un consensus minimal concernant au moins une classification des différents fournisseurs[3] en fonction de leur métiers, de leurs domaines d'expertise et des services qu'ils proposent. Les pages jaunes de France Télécom en sont une illustration. Nous supposons donc l'existence d'une hiérarchie de domaines informationnels, ou thèmes, par rapport à laquelle les agents autres que les agents personnels peuvent être situés[4]. Un agent fournisseur peut, bien entendu, se déclarer compétent sur différents domaines.

Nous utilisons aussi des agents intermédiaires mais ceux-ci, s'ils conservent leur rôle par rapport aux demandeurs, deviennent des serveurs de requêtes vis-à-vis des fournisseurs. Ceci les distingue des *broker*s et *matchmakers*, c'est pourquoi nous utilisons le terme spécifique de *BaGate*[5]. A chaque domaine informationnel peuvent être associés plusieurs BaGates. Si aucun *BaGate* n'est associé à un domaine, ce dernier est considéré comme inactif.

Dans ce schéma, les agents site ont un rôle plus actif qu'en présence d'un *broker* par exemple. C'est en effet à eux qu'incombe la démarche d'approvisionnement en requêtes auprès des *BaGates* des domaines dans lesquels ils sont situés. Ils entretiennent donc une relation de « clientèle » avec ces BaGates. Un agent site peut choisir d'être client de tel ou tel *BaGate* pour des raisons qui lui sont propres (proximité géographique, proximité réseau…). Il peut aussi choisir d'être client de plusieurs *BaGates*

3. Dans Bonom, les fournisseurs s'appellent aussi des *agents sites*. Les demandeurs sont appelés *agents personnels*. Nous utiliserons indifféremment les termes fournisseur ou agent site de même que demandeur ou agent personnel.
4. Les agents personnels sont donc considérés comme extérieurs à l'organisation.
5. *BaGate* est la contraction des deux termes anglais « Bag » et « Gate » qui font référence à deux fonctions assurées par ces agents : une porte d'entrée pour un domaine et un dépot de requêtes.

associés à un même thème pour améliorer la robustesse (en cas de panne de l'un d'eux) ou pour obtenir un maximum de requêtes. Soulignons le changement de point de vue vis-à-vis de la notion de client. D'un point de vue externe à l'organisation d'agents, un agent personnel est client de l'organisation car il est demandeur de réponses. D'un point de vue interne, ce sont les *BaGates* et les agents sites qui sont clients de *BaGates* (lorsqu'ils leur demandent des requêtes). Un fournisseur spécifie aux *BaGates* auprès desquels il se fournit, les requêtes qu'il désire, tant d'un point de vue quantitatif que qualitatif. Ainsi, il gère entièrement sa politique et sa stratégie de communication. Il peut choisir d'accepter toutes les requêtes générales sur le thème qui le concerne en se servant auprès d'un seul *BaGate* ou bien de ne traiter que les requêtes relatives à un point précis en souscrivant auprès de tous les *BaGates* du thème ou bien de combiner plusieurs demandes.

La tâche d'appariement entre *requêtes d'utilisateurs* et *demandes de requêtes d'utilisateurs* est effectuée par les *BaGates*. Cette tâche est relativement similaire à celle réalisée par un *broker* ou *matchmaker*. Comme cela a déjà été expliqué, elle comporte des aspects syntaxiques et sémantiques. Par exemple, un agent site peut demander à traiter des requêtes exprimées dans le langage XML-QL et concernant la marée noire en Bretagne.

Un tel schéma donne un poids important aux agents fournisseurs : ce sont eux qui font la démarche de s'associer à tel ou tel thème. Ce sont aussi eux qui choisissent la forme syntaxique et sémantique des requêtes qu'ils vont traiter ainsi que leur nombre. Ils peuvent à tout instant changer de politique ou même se retirer du système. L'introduction de mécanismes de modération est donc indispensable pour éviter les abus. Ces mécanismes interviennent principalement lors de l'introduction d'un fournisseur sur un thème, ou plus précisément lors de l'initialisation de son processus de clientèle avec un *BaGate* . En effet, le *BaGate* a toute latitude pour accepter ou refuser la candidature d'un fournisseur. Parmi les raisons pouvant motiver un refus, on peut citer des raisons locales au BaGate, comme un trop grand nombre de clients déjà inscrits, mais aussi des raisons plus sémantiques. En effet, en utilisant les techniques d'analyse sémantique du contenu présentées dans la section 2, un BaGate peut vérifier l'adéquation d'un fournisseur par rapport à un thème. De plus, les utilisateurs peuvent l'avoir évalué[6]. Ces évaluations sont autant d'informations dont il faut tenir compte dans cette démarche de modération. Il est important de préciser que l'acceptation d'un fournisseur en tant que client par un *BaGate* n'est pas définitive. La limite dans le temps de cette relation est fixée par le BaGate. Le fournisseur peut arrêter cette relation ou au contraire effectuer une demande de renouvellement quand il le juge nécessaire. Dans ce dernier cas, ce sera l'occasion pour le *BaGate* de vérifier la pertinence de ce fournisseur.

6. Ce processus de feedback est géré par l'agent de l'utilisateur avec un processus manuel, par renseignement d'un formulaire d'évaluation à la demande de l'utilisateur, ou automatique, ou semi-automatique, par observation de l'usage fait par l'utilisateur des résultats renvoyés (consultation, enregistrement en local, mise à la corbeille…)

Enfin, les *BaGates* référant à un même thème maintiennent entre eux une relation d'accointance. Cela leur permet en particulier de se concerter lors de la candidature d'un fournisseur, mais aussi de faire circuler certaines requêtes d'utilisateurs. Un *BaGate* ne pouvant apparier une requête avec les demandes de ses clients peut ainsi la faire suivre à l'un de ses frères (i.e. un *BaGate* du même thème). L'ensemble des agents d'un même thème potentiellement atteignable par ce processus constitue ce que nous appelons un *groupe*. Le schéma de la figure 4 présente un exemple de liens entre des groupes d'agents et la hiérarchie de thèmes.

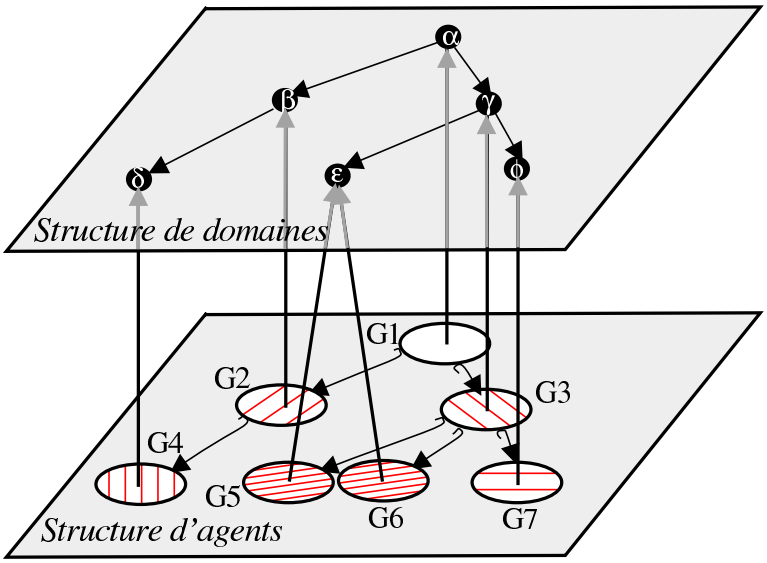


Figure 4. *Liens entre groupes d'agents et hiérarchie de thèmes*

La figure 5 propose une vision incluant la représentation de groupes mais aussi d'agents site et de BaGates. Le fait que les *BaGates* d'un même groupe apparaissent sur une même ellipse symbolise la relation d'accointance. La relation *client* n'est pas représentée sur ce schéma. Le terme *d'ouvrier*, plus général que celui d'agent site est utilisé car d'autres types d'agents peuvent intervenir dans le processus et faire partie de groupes. Par exemple, on pourrait introduire des agents reformulateurs spécialistes de la reformulation de requêtes pour un thème précis.

Enfin, soulignons que l'organisation que nous présentons ici en quelques points, a été étudiée de manière formelle tant dans ses aspects statiques que dynamiques. En effet, la caractérisation précise de ce qu'est un groupe et son lien avec un thème n'est pas forcément aisée. On ne peut supposer sur internet que les introductions d'agents se font de manière synchrone. Ainsi, par exemple, il se peut que deux groupes relevant d'un même thème se forment en parallèle. Le lecteur intéressé par ces aspects pourra se reporter à [CAZ 01] où sont également décrites les opérations d'introduction de nouveaux agents et certains protocoles. La sous-section suivante illustre comment les

agents distribués et structurés suivant l'organisation Bonom, intéragissent pour faire parvenir une requête à des agents sites pertinents.

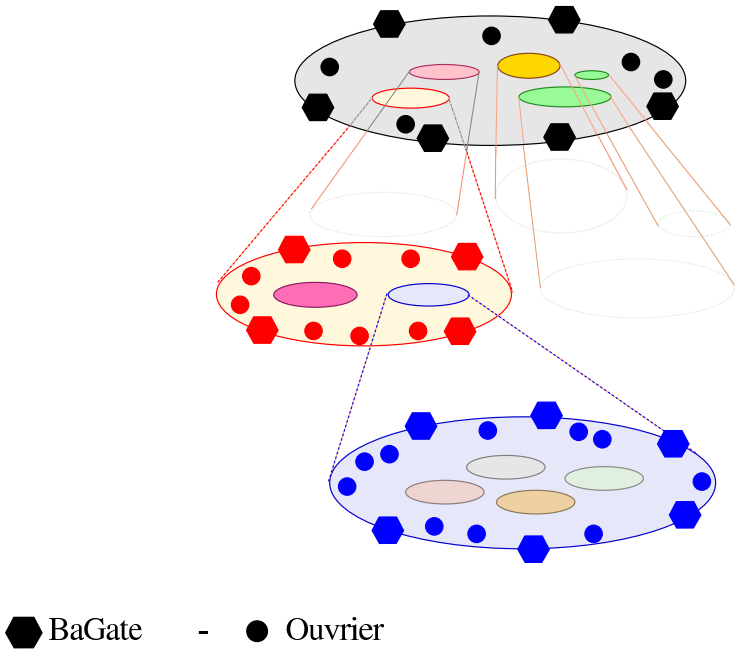


Figure 5. *Un exemple d'organisation d'agents en thèmes et sous-thèmes*

5.2. *Exemple d'interactions pour la prise en charge d'une requête*

Une requête émise par un utilisateur *via* son agent personnel, comporte plusieurs parties dont :

– le texte de la requête (par exemple « Algèbre linéaire »),

– le langage d'expression de la requête (par exemple « booléen »),

– l'ontologie de référence si besoin,

– les thèmes auxquels l'utilisateur destine la requête,

– si l'utilisateur souhaite que la requête soit aussi traitée par des agents de thèmes plus précis, de combien de niveau de thèmes il souhaite voir descendre la requête,

– s'il y a des sous-thèmes dans lesquels il ne souhaite pas que la requête soit traitée,

– le nombre d'agents pouvant traiter la requête pour un thème donné,

– le nombre maximal de réponses par agents, etc.

Les deux derniers points cités permettent à l'utilisateur de minimiser le nombre de réponses à une requête donnée. En effet, l'expérience montre qu'il est assez rare

qu'un utilisateur souhaite obtenir toutes les réponses à une requête. En général, seules les premières réponses sont lues et éventuellement utilisées pour affiner la requête. Il est donc inutile de solliciter tous les agents pour qu'ils renvoient toutes les réponses possibles (sauf demande explicite). La figure 6 présente un exemple de formulation de requête. Dans cette figure, l'utilisateur souhaite envoyer sa requête aux agents du thème `/enseignement` et des sous-thèmes (arrêt à la profondeur trois) exception faite du thème `/enseignement/primaire`. Le thème ciblé `/enseignement` est encadré. Les thèmes potentiellement atteints sont soulignés. Les thèmes interdits sont rayés. Entre deux et cinq agents par thème atteint doivent être sollicités et chacun de ces agents devra renvoyer au maximum quatre réponses.

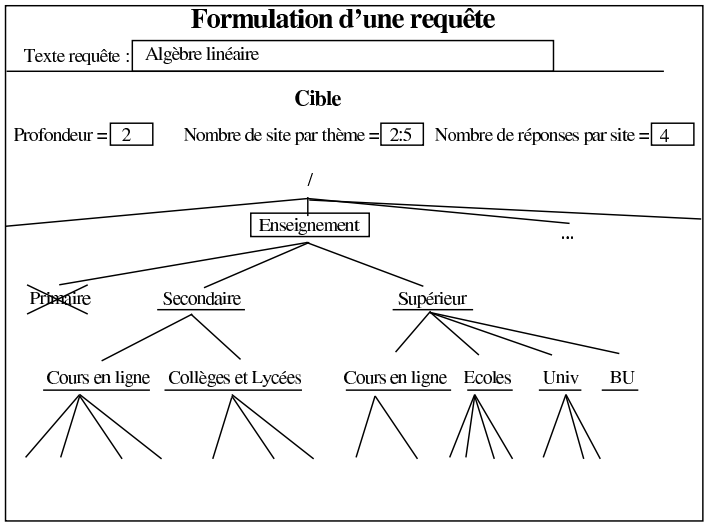


Figure 6. *Exemple de formulation d'une requête*

Une fois les différentes informations relatives à la requête renseignées par l'utilisateur, cette dernière est expédiée en direction de la société d'agents. En fonction des connaissances dont il dispose sur cette société, l'agent personnel peut expédier la requête soit aux *BaGates* du thème le plus général[7] qui se chargeront de faire circuler cette requête jusqu'à des *BaGates* des thèmes ciblés, soit directement à ces derniers. Le deuxième cas est plus rapide et plus économique mais il nécessite des connaissances préalablement acquises.

Les mécanismes permettant de faire circuler une requête en direction des thèmes ciblés sont simples. Prenons l'exemple de la figure 7. Un *BaGate* du thème `/enseignement` recevant une requête ciblée sur le thème `/enseignement/supérieur/université` essaiera de la faire suivre à l'un de ses clients du thème `/enseignement/supérieur`[8]. S'il n'a aucun *BaGate* de ce thème

7. Tout agent du système connaît, *via* un fichier de configuration, les services de nommages (services de pages blanches) où sont déclarés ces agents.
8. Un tel client ne peut être qu'un *BaGate* du thème `/enseignement/supérieur`

parmi ses clients, il sollicite l'un de ses frères (i.e. un *BaGate* du même thème que lui). Ce dernier effectue le même traitement. Ce processus se répète jusqu'à réussite ou épuisement de tous les *BaGates* du thème `/enseignement`.

Un *BaGate* recevant une requête ciblée sur son thème effectue les appariements entre cette requête et les demandes de ses clients. Il l'expédie ensuite à ceux pour lesquels l'appariement est le meilleur [9]. Deux cas peuvent alors se présenter : soit le *BaGate* dispose de suffisamment de clients adéquats à qui expédier la requête, soit il ne peut atteindre le minimum fixé par l'utilisateur. Dans ce dernier cas, il fait suivre la requête à l'un de ses frères pour qu'il sollicite d'autres agents. Ce processus peut lui aussi se répéter jusqu'à satisfaction de la requête de l'utilisateur ou l'épuisement des *BaGates* du thème.

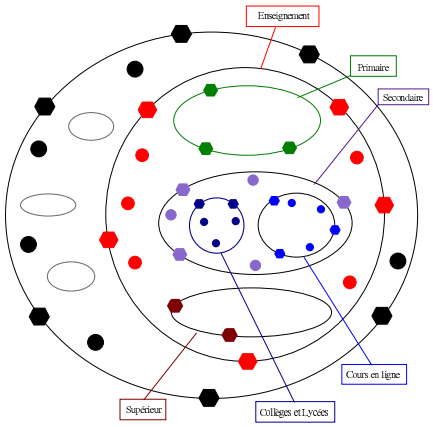
Les agents sites ayant obtenu la requête la traitent et calculent leurs réponses. Reste alors à faire parvenir ces réponses à l'agent personnel. Les réponses vont en fait suivre un chemin inverse à celui de la requête et passer de *BaGate* en *BaGate* jusqu'à revenir à l'agent de l'utilisateur. Cependant, tous les *BaGates* ayant participé à la descente de la requête ne sont pas concernés. Au maximum un *BaGate* par thème sera sollicité pour ce travail et ce seulement pour les thèmes que la requête n'a pas simplement traversés mais au sein desquels elle a été distribuée à plusieurs agents. Dans un tel thème, c'est le premier *BaGate* atteint par la requête qui assume ce rôle[10] appelé *concentration* qui consiste à regrouper les réponses des différents sous-traitants pour les faire suivre. Le mode de transmission est asynchrone. C'est-à-dire que le concentrateur n'attend pas d'avoir toutes les réponses de tous les sous-traitants pour expédier les réponses. Ils renvoie périodiquement les résultats obtenus (s'il y en a) de sorte à ce que l'utilisateur n'ait pas à attendre la fin du traitement pour avoir les premières réponses.

5.3. *Faisabilité de l'organisation : réalisation d'un prototype*

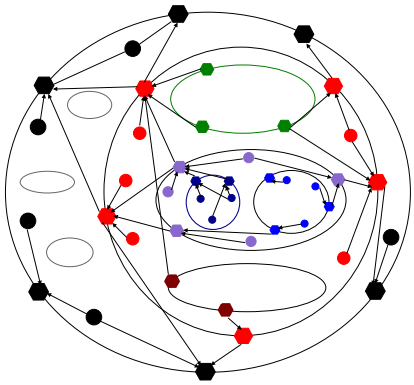
De façon à tester ce type d'organisation et les protocoles associés, nous avons réalisé un prototype en utilisant Java [Sun] (version actuellement utilisée 1.2.3) et CORBA [Obj a]. L'ORB utilisé est Orbacus [Obj b] (version actuellement utilisée 3.3.4). Ces choix se justifient par une volonté d'obtenir une portabilité et une ouverture maximale. D'autres solutions technologiques, comme SOAP, sont envisageables dans la mesure où elles permettent de répondre à ces critères.

9. L'utilisateur peut aussi avoir interdit le traitement de la requête par certains agents, information que le *BaGate* doit prendre en compte à ce niveau. Pour des raisons de clarté, nous avons fait le choix de ne pas présenter toutes les options dans cet article.

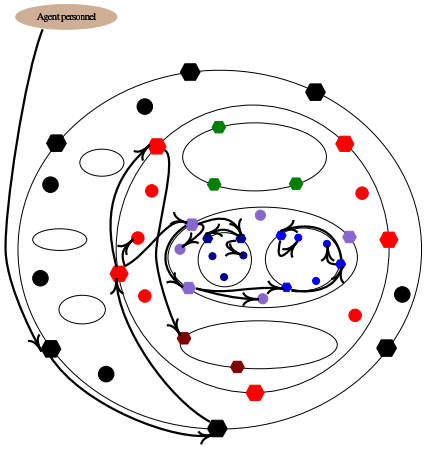
10. Sous l'hypothèse d'une répartition uniforme de l'arrivée de requêtes sur les différents *BaGate* d'un même thème, cela est suffisant. Une gestion de la charge et une répartition de ce rôle entre les différents *BaGates* est d'ores et déjà envisagée pour les cas où cette hypothèse ne serait pas valable.



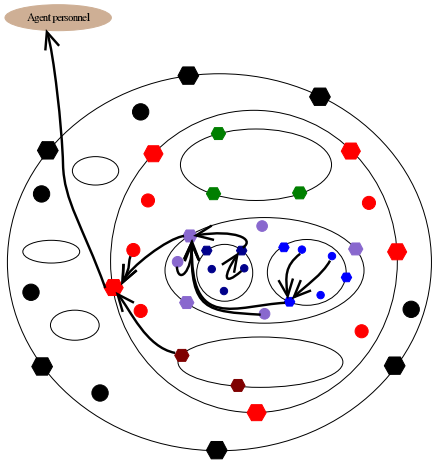
Groupes d'agents correspondant aux thèmes de la figure 6. Pour des raisons de lisibilité, tous les thèmes ne sont pas représentés de manière complète.



Relations *client* entre les différents agents.



Expédition de la requête présentée à la figure 6. Cette requête circule dans la société pour atteindre des agents du thème `/enseignement` et ses sous-thèmes exception faite du sous thème `/enseignement/primaire` qui a été exclu lors de la rédaction de la requête.



Cheminement des réponses expédiées par les agents.

Figure 7. *Exemple de circulation d'une requête et de remontée des réponses*

Actuellement, les agents sites et personnels sont simulés et il n'est pas nécessaire d'utiliser un langage de communication agent standard tel que KQML ou FIPA-ACL. Les *BaGates* sont implémentés, exception faite des méthodes de modération. Des groupes de thèmes différents peuvent être déployés sur plusieurs machines distantes. Des scénarios simples d'introduction et de suppression d'agents ont été testés avec succès. Structures et protocoles implantés sont ceux décrits dans [CAZ 01]. En ce qui concerne les requêtes, elles atteignent les agents sites et les réponses reviennent convenablement aux utilisateurs. Des prototypes plus sophistiqués d'agents sites et d'agents personnels sont en cours de développement.

6. Conclusion

Dans le domaine de la recherche d'informations ou de connaissances distribuées sur un intranet ou sur internet, le projet Bonom se particularise par un certain nombre de points.

En premier lieu, la vue orientée agent repose sur l'hypothèse que les sources d'informations ou de connaissances peuvent devenir actives dès lors qu'elles disposent d'un certain nombre de mécanismes leur permettant de gérer et exploiter la connaissance, répondre de manière pertinente à des requêtes, d'exprimer leurs compétences, etc. Les techniques diffèrent, en particulier selon le type d'informations manipulées. Parmi elles, Bonom propose une méthode semi-automatique pour expliciter la connaissance dans des pages web par construction d'un index sémantique structuré. Il montre aussi comment prendre en compte cette connaissance pour vérifier l'adéquation entre un site et une ontologie et pour améliorer le traitement d'une requête. Ces techniques sont particulièrement adaptées lorsqu'il est difficile d'envisager une annotation manuelle des pages comme dans CoMMA[GAN 00] ou Ontobroker[FEN 98b].

L'organisation d'agents proposée suppose que les sources sont représentées par des agents sites. Cela revient à adopter une vue externe vis-à-vis des sources et a pour conséquence de ne pas supposer telle ou telle technique d'explicitation de la connaissance. L'hypothèse fondamentale est que les agents sites sont capables d'exprimer les requêtes qui conviennent aux formalismes, raisonnements, calculs effectués par la source et d'en faire la demande aux agents intermédiaires *BaGates*. Ainsi, l'organisation Bonom peut inclure, *via* les agents qui les représentent, des sources dont les pages sont annotées manuellement, des sources où l'on utilise une indexation sémantique structurée, des bases de connaissances, etc. Une autre conséquence est la totale maîtrise par l'agent site de la politique de communication de la source, de la gestion de la charge, etc.

L'organisation des agents suivant une hiérarchie de thèmes suppose un consensus préalable sur cette hiérarchie. En ce sens, elle demande beaucoup moins d'harmonisation, voire d'uniformisation, que les approches où la définition d'un schéma global ou, dans une moindre mesure, celle d'un langage de description des capacités, est requise. C'est pourquoi, elle se présente comme complémentaire de celles-ci, à la fois parce

qu'elle peut être plus adaptée dans des cadres fortement concurrentiels ou lorsque les agents sites sont très nombreux, mais aussi parce qu'elle peut, par exemple, inclure un agent représentant un ensemble de sources qui ont décidé de se fédérer entre elles ; ces sources peuvent utiliser, par exemple, un broker pour se faire représenter.

Le projet dans son ensemble comprend de nombreuses parties et requiert une certaine interdisciplinarité. Cet article, qui n'a présenté que les grandes lignes du projet et détaillé seulement certains points en est déjà un exemple. D'autres aspects doivent aussi être pris en compte : reformulation des requêtes, intégration et visualisation des résultats au niveau de l'agent personnel (en exploitant toutes les données disponibles notamment les ontologies). Par exemple, il est envisageable d'utiliser une technique de graphe hyperbolique pour visualiser les réponses en fonction des thèmes ou des concepts d'une ontologie [HAS 00].

7. Bibliographie

[BOU 00] BOULANGER D., DUBOIS G., « Coordination système multi-agents/objets pour la coopération de systèmes d'information », Séminaire "Systèmes distribués et connaissances", INRIA Sophia Antipolis, Novembre 2000.

[CAZ 01] CAZALENS S., LAMARRE P., « An organization of Internet agents based on a hierarchy of information domains », DEMAZEAU Y., GARIJO F. J., Eds., *Proceedings MAA-MAW*, 2001.

[COW 96] COWIE J., LEHNERT W., « Information extraction », *Communications of the ACM*, vol. 39, 1996.

[DEC 97] DECKER K., SYCARA K., WILLIAMSON M., « Middle-Agents for the Internet », *Fifteenth International Joint Conference on Artificial Intelligence (IJCAI'97)*, Morgan Kaufmann, 1997.

[DES 01] DESMONTILS E., JACQUIN C., « Des ontologies pour indexer un site Web », *actes des journées francophone d'ingénierie des connaissances*, Grenoble, 2001.

[FEL 95] FELDMAN R., DAGAN I., « Knowledge discovery in textual databases (KDT) », *First international conference on knowledge discovery (KDD'95)*, Montreal, 1995.

[FEN 98a] FENSEL D., DECKER S., ERDMANN M., STUDER R., « Ontobroker : Or How to Enable Intelligent Access to the WWW », *the 11th Banff Knowledge Acquisition for Knowledge-Based System Workshop (KAW'98)*, Banff, Canada, 1998.

[FEN 98b] FENSEL D., DECKER S., ERDMANN M., STUDER R., « Ontobroker : Or How to Enable Intelligent Access to the WWW », *Proceedings of the 11th Banff Knowledge Acquisition for Knowledge-Based System Workshop (KAW'98)*, Banff, Canada, 1998.

[FEN 00] FENSEL D., HORROCKS I., HARMELEN F. V., DECKER S., ERDMANN M., KLEIN M., « OIL in a Nutshell », *proceedings of the European Knowledge Acquisition Conference (EKAW 2000)*, 2000.

[FIN 94] FININ T., FRITZSON R., MCKAY D., MCENTIRE R., « KQML as an Agent Communication Language », *Third International Conference on Information and Knowledge Management (CIKM'94)*, ACM Press, Nov. 94.

[GAN 00] GANDON F., DIENG R., GIBOIN A., « CoMMA : Une approche distribuée de la mémoire organisationnelle », Séminaire "Systèmes distribués et connaissances", INRIA Sophia Antipolis, Novembre 2000.

[GEN 92] GENESERETH M., FIKES R., « Knowledge Interchange Format », technical report, 1992, Computer science department, Stanford university.

[GRU 93] GRUBER T., « A Translation Approach to Portable Ontology Specification », *Knowledge Acquisition*, vol. 5, 1993, p. 199–220.

[HAS 00] HASCOËT M., « Navigation and interaction within graphical bookmarks », Séminaire "Systèmes distribués et connaissances", INRIA Sophia Antipolis, Novembre 2000.

[HEF 99] HEFLIN J., HENDLER J., LUKE S., « Applying Ontology to the Web : A Case Study », *International Work-Conference on Artificial and Natural Neural Networks (IWANN)*, 1999.

[HEN 00] HENDLER J., MCGUINESS D., « The DARPA Agent Markup Language », *IEEE Intelligent systems, Trends and Controversies*, , 2000, p. 6-7.

[KEN 99] KENT E., « Conceptual Knowledge Markup Language : The Central Core », *Proceedings of the Twelfth Workshop on Knowledge Acquisition, Modeling and Management (KAW'99)*, Banff Alberta, 1999.

[KIF 89] KIFFER M., LAUSEN G., « F-Logic : a higher-order language for reasoning about objects, inheritance and scheme », *proceedings of ACM Sigmod international conference on management of data*, Portland, Oregon, 1989.

[KLU 99] KLUSCH M., Ed., *Intelligent Information Agents*, Springer, 1999.

[KUO 95] KUOKKA D., HARADA L., « Matchmaking for Information Agents », *Fourteenth International Joint Conference on Artificial Intelligence (IJCAI'95)*, Morgan Kaufmann, 1995.

[LIN 98] LIN S., AL, « Extracting classification knowledge of internet documents with mining term associations : a semantic approach », *International acm-sigir conference on research and development in information retrieval (SIGIR-98)*, 1998.

[LOH 00] LOH S., WIVES L., DE OLIVEIRA J. P. M., « Concept-based knowledge discovery in texts extracted from the web », *journal SIGKDD explorations*, vol. 2, n° 1, 2000, p. 29-39.

[MAT 00] MATTOX D., SELIGMAN L., SMITH K., « Rapper : a wrapper generator with linguistic knowledge », *ACM workshop on information and data management*, 2000.

[MIL 90] MILLER G. A., « Wordnet : An Online Lexical Database », *International journal of lexicography*, vol. 3, n° 4, 1990, p. 235–312.

[NAP 97] NAPOLI A., « Une introduction aux logiques de descriptions », rapport de recherche, 1997, INRIA.

[NOD 99] NODINE M., BOHRER W., NGU A. H. H., « Semantic Brokering over Dynamic Heterogeneous Data sources in InfoSleuth », *International Conference on Data Engineering (ICDE)*, 1999.

[Obj a] OBJECT MANAGEMENT GROUP, « CORBA », http ://www.corba.org/.

[Obj b] OBJECT ORIENTED CONCEPTS (IONA), « Orbacus », http ://www.ooc.com/ob.

[Ret] RETICULAR SYSTEMS INC., http ://www.agentbuilder.com/Documentation/PMail/.

[SEG 00] SEGRET M.-S., POMPIDOR P., HÉRIN D., « Extraction et Intégration d'Informations Semi-Structurées dans les pages Web - Projet Chimère », *IC'2000*, 2000, p. 277–288.

[SHE 99] SHEHORY O., « A Scalable Agent Location Mechanism », WOOLDRIDGE M., LESPERANCE Y., Eds., *Intelligent Agents VI*, Springer, 1999.

[SRI] SRI INTERNATIONAL, « The Open Agent Architecture », `http ://www.ai.sri.com/~oaa/main.html`.

[Sun] SUN MICROSYSTEMS, « Java », http ://www.java.sun.com/.

[SYC 99] SYCARA K., KLUSCH M., WIDOFF S., « Dynamic Service Machmaking Among Agents in Open Information Environments », *ACM SIGMOD Record, Special Issue on Semantic Interoperability in Global Information Systems*, vol. 28, n° 1, 1999, p. 47-53.

[W3C 00] W3C, Resource Description Framework (RDF) Schema Specification 1.0 W3C Recommendation, March 2000.

[WIE 92] WIEDERHOLD G., « Mediators in the Architecture of Future Information Systems », *IEEE Computer*, vol. 25(3), 1992, p. 38-49.

[WON 00] WONG H. C., SYCARA K., « A Taxonomy of Middle-agents for the Internet », *Fourth International Conference on MultiAgent Systems (ICMAS 2000)*, July 2000, p. 465-466.

[WU 94] WU Z., PALMER M., « Verb semantics and lexical selection », *the 32nd annual meeting of the association for computational linguistics*, Las Cruses, New Mexico, 1994.

Annexe 2 - Int. J. Cooperative Inf. Syst. 2007

[LLCV07] Philippe Lamarre, Sandra Lemp, Sylvie Cazalens and Patrick Valduriez, *A Flexible Mediation Process for Large Distributed Information Systems* International Journal of Cooperative Information Systems 16(2):299-332, 2007.

International Journal of Cooperative Information Systems

A FLEXIBLE MEDIATION PROCESS FOR LARGE DISTRIBUTED INFORMATION SYSTEMS

PHILIPPE LAMARRE, SANDRA LEMP, SYLVIE CAZALENS

LINA
2 rue de la Houssiniere
BP92208
44322 Nantes Cedex 3
France
Firstname.Lastname@univ-nantes.fr

PATRICK VALDURIEZ

INRIA and LINA
2 rue de la Houssiniere
BP92208
44322 Nantes Cedex 3
France
Patrick.Valduriez@inria.fr

We consider distributed information systems that are open, dynamic and provide access to large numbers of distributed, heterogeneous, autonomous information sources. Most of the work in data mediator systems has dealt with the problem of finding relevant information providers for a request. However, finding relevant requests for information providers is another important side of the mediation problem which has not received much attention. In this paper, we address these two sides of the problem with a flexible mediation process. Once the qualified information providers are identified, our process allows them to express their interest in a request via a bidding mechanism. It also requires to set up a requisition policy, because a request must always be answered if there are qualified providers. This work does not concern pure market mechanisms because we counter-balance the providers' bids by considering their quality wrt a request. We validate our process on a set of simulations in the context of load balancing, which is a good indicator of the system's overall performance. The results show that the mediation process provides a very good long-run regulation of the system, in particular when providers can leave the system. However, load balancing is not the natural application of the flexible mediation and additional testing is required to show the generality of the approach to non-depletable resources.

Keywords: distributed information system, flexible mediation, economic approach, load balancing.

1. Introduction

We consider distributed information systems that are open, dynamic and provide access to large numbers of distributed, heterogeneous, autonomous information sources. Information requesters and providers may come in or leave the system at any time, because of technical reasons or of their own choice. Entrance may be motivated by some expected benefits while exit may result from disappointment. On the one hand, one can estimate that a requester satisfaction is a function of the quality of the answers it gets. On the other hand, the reasons for a provider's disapointment are more diverse. It may be for example because it never gets interesting requests, that is requests it would prefer treating, while it is often solicited for uninteresting ones. Thus, it is important for the flexibility of the system to preserve the highest diversity by avoiding the leave of requesters or providers.

In this context, most of the work in data mediator systems has dealt with the problem of finding relevant information providers for a request[36]. In such cases the main objective is the user's satisfaction. However, finding interesting relevant requests for information providers is another important side of the mediation problem which has not received much attention. In that case, the providers' satisfaction should also be considered.

This paper proposes the definition of a mediation process with the following characteristics. First, it selects the providers according to their qualities (indication of the users satisfaction in choosing one or the other) and their bids (indication of their interest in treating the request). Combining both parameters leads to balance between providers and users interests. Second, it may happen that no provider wants to treat a given request, even those which are able to do so. In such a case, the request is imposed to some providers, even if this leads to their temporary dissatisfaction.

The overall goal is to define a mediation mechanism that considers both the users' and information providers' long run satisfaction and ensures a kind of stability in the system. In the context of open system with autonomous databases, stability means that users and information providers do not always leave the system because of dissatisfaction (because they never get good answers, never get interesting requests).

The main contribution of this paper is the definition and validation of a mediation process, called *Flexible Mediation* which takes into account the above considerations. It can be used each time the providers represent competing companies which have to participate in the common effort of providing the requester with the required number of providers.

We validated the process behaviour through simulation in the context of load balancing. Indeed, although flexible mediation allocates requests with both the users' and providers' long run satisfaction in mind, it would be of poor interest if it would present major performance degradation, in particular in terms of load balancing. This is why the flexible mediation is confronted to an algorithm which

always chooses the least loaded provider. We show that performance degradation is acceptable and understandable. Of course, additional testing is required to verify the generality of the approach to non-depletable resources such as information services. To our knowledge, there is no work which combines both qualities of the providers and their bids and also introduces a requisition process, with the same very good long-run regulation of the system.

The paper is organized as follows. Section 2 describes different motivating scenarios which help illustrate the problem. In Section 3, we make precise the objectives of the paper. Section 4 is devoted to the overall architecture of the system and the mediator's main modules. Section 5 describes the mediation process. It also illustrates the mathematical model with short series of mediations. Section 6 describes an extensive experimental validation based on simulation in the context of load balancing. Section 7 discusses related works, in particular economic approaches to mediation. Section 8 concludes.

2. Motivating scenarios

We consider a distributed system gathering thousands of information providers in the healthcare field. There may be medical doctors, pharmaceutical companies, pharmacies, hospitals, universities...

Let us consider a requester who has problems with mosquitoes and wants information about insect bites, associated diseases and repellent lotions. Because the providers may have different data, the requester wants several of them to answer. Because there are many providers, the requester does not want all of them to answer. Let us assume that the requester wants ten of them. Here, there is no notion of *correct* answer. The requester just wants to consult different information providers because they may have different data, viewpoints or experiences, while limiting the number of answering providers to avoid an information flooding. The request is sent to a mediator which job is to find the ten most *relevant* providers. In our work relevance is based on two types of parameters: *quality* and *bid* with the intuitions below.

In this example, we assume that all the providers are able to treat the given request [a]. However, some providers may perform the request better than others, for example because they have more or different data, more experience... This idea is captured by the notion of *quality of a provider with respect to a given request*. It reflects an evaluation of how well the provider is expected to treat the request. The way this evaluation is conducted and evolves is out of the scope of the paper. For example, it may be based on a reputation mechanism or on a regular benchmarking or both. In our scenario, the tropical diseases department of the University of London may get a higher quality than the consulting room of general practitionners in Berlin: the former's specialty is closely linked to the request and their answers

[a]Otherwise, a matchmaking mechanism based on a service description subscription can be used.

generally have very good feedbacks. The latter are not specialists of the problem, and because of their geographical situation, they have little experience about it. Notice that poor quality with respect to a request should not be "punished". The problem arises when a provider always gets very low quality in the system. A choice might be to exclude it physically from the system. Another option is to manage this problem through the mediation mechanism itself by never giving requests to such a provider.

We also assume that providers are more interested in treating some requests than others. This assumption is justified by the fact that the providers act on behalf of companies which, in a competitive environment, have their own public relations policy, with their own priorities. For example, consider a provider for a pharmaceutical company which wants to promote its newest insect repellent. So it is more interested in treating the requests which are linked in some way to mosquitoes, insect bites and so on than requests about other problems or other drugs. In this work, we assume that each provider expresses how much it is interested in treating a given request by a bid (a real number). For example, the pharmaceutical company would bid 20 on requests about mosquitoes and insect bites, while the tropical diseases department of the University of London would bid less (15 for example) because it currently wants to avoid treating broad requests. In this work, a negative bid means that the provider does not want to treat the request. This may be because the request is very far from its current concerns: the previous company would bid -10 on a request about influenza or even less if it is overloaded.

Given a request and the corresponding bids and qualities of the providers, several mediation processes are possible. The following scenarios highlight different difficulties in defining a satisfying mediation process. To keep it simple, we consider only four providers named p_1, p_2, p_3 an p_4 which are all able to treat some incoming request r. Table 1 gives the provider quality with respect to r.

Table 1. The providers' qualities.

p_1	p_2	p_3	p_4
12	12	6	8

2.1. *Scenario 1: the limit of a simple direct auction*

The required number of providers is 2. Bids are shown in Table 2 where all the providers are interested in treating the request. Provider p_3 is the most interested, maybe because the request matches its public relations policy. A simple way to treat the problem is to allocate the request to the most interested providers, thus considering the bids only. This comes to use a direct auction mechanism [31,23]. We illustrate the limit of such a mechanism with a Vickrey auction without loss of

generality.

Table 2. The providers' bids in scenario 1.

p_1	p_2	p_3	p_4
12	10	19	5

Because they have the highest bids, providers p_1 and p_3 get the request. Each of them pays provider p_2's bid: 10. A first obvious problem is that the mechanism may select providers with very poor quality. This is the case for p_3 which is selected because of its high bid although it has the lowest quality. A second problem is that this mechanism only makes sense when all the bids are positive.

2.2. *Scenario 2: imposition with compensation*

Now, the required number of providers is 3. Table 3 shows that providers p_2 and p_4 do not want to treat the request. To keep things simple, the mediation process still considers bids only.

Table 3. The providers' bids in scenario 2.

p_1	p_2	p_3	p_4
12	-5	19	-7

Because of the negative bids, there are two options. Either the process only selects the providers that want to treat the request (i.e. p_1 and p_3), or it also imposes the request to p_2 in order to come up to the requested number of providers (remember that all the providers are able to treat the request). Our choice is to impose p_2 thus making the user's requirements prevail over the providers' preferences. This ensures that a request never ends up with no answer even if all the providers bid negatively.

The next question is "who pays and how much?". Using a Vickrey like auction with negative bids would lead to give money to p_1, p_2, and p_3: 7 to each. Even if it makes sense to compensate p_2 which is imposed, it is totally unintuitive to give money to p_1 and p_3. Hence a solution is to make p_1, p_3 and p_4 give money to p_2 because they have been satisfied by the mediation contrary to p_2 which has been imposed [22]. This money transfer among the four providers would increase p_2's chances to get the requests it wants against p_1, p_3, and p_4, in the next mediations (because p_2 has more money which comes from the three other providers).

2.3. *Scenario 3: balance between bids and qualities*

This scenario illustrates the problem of balancing between quality and bid in the mediation process. Table 4 shows the provider bids. The qualities are still given by Table 1. The required number of providers is 2.

Table 4. The providers' bids in scenario 3.

p_1	p_2	p_3	p_4
15	20	20	15

Providers p_2 et p_3 have the same highest bid value. However, p_3's quality is much lower. Thus, p_2 should be prefered. p_1 and p_2 have the same highest quality, but p_1 bid is lower. Obviously, p_2 is the best. The problem is to precisely define how to order the other providers while balancing between bids and qualities. Indeed, the second one may be p_1 if the process makes quality prevail; it may be p_3 if bid prevails. A point is to find a good balance between both criterias, so that it has a good long run behaviour. Another point which is not illustrated here is to include the quality parameter in the imposition with compensation case.

2.4. *Scenario 4: long run behaviour*

This scenario considers two series of successive mediations. It illustrates a medium quality provider's external viewpoint based on whether the process globally fulfills its wants or, on the contrary, imposes it too much. For each request, we indicate the provider's bid, the result of the mediation (*yes* if it gets the request, *no* if it does not get it), and the assessment: the symbol '+' is used when the provider gets a request it wants or when it is not imposed a request it does not want; a '−' is used when the provider is imposed a request.

Table 5. A satisfying series of five mediations.

r_1	r_2	r_3	r_4	r_5
30	20	-8	15	-10
yes	yes	yes	yes	no
+	+	-	+	+

In Table 5, in the course of the five mediations, the provider always gets the requests it wants and it is only imposed once, for request r_3. So, if it made an assessment after the five mediations, the provider would conclude that the mediation

process is globally satisfying, and it will go on with it. If it were the case for the majority of the providers, the system long-run behaviour would be stable.

Table 6. A dissatisfying series of five mediations.

r_1	r_2	r_3	r_4	r_5
30	-10	-8	15	-10
no	yes	yes	no	no
-	-	-	-	+

On the contrary, Table 6 shows that the provider is imposed twice and does not get the requests it wants. Hence, the assessment after the five mediations is negative. If this should continue for a while, the provider might just leave the mediator or the system. Unstability may appear when there are a lot of dissatisfied providers leaving the system.

3. Objectives - Focus of the paper

The previous scenarios focus on the mediation process itself. In a real world application, additional processes may be required. First, query planning processes may be needed. This problem is adressed in different ways[36,35]. Thus, we can indifferently assume that query planning is ensured by the mediator, or by the requesters or by any external module, without loss of generality. Second, the providers advertise their capabilities at the mediator which must support matchmaking techniques in order to match a given request with the providers able to treat it. Several matchmaking algorithms have been proposed [6,27,20,34] and they can be re-used here. Third, the mediator must evaluate how well the providers might perform a given request, under the form of a positive number, called *quality*. This aspect is related to reputation acquisition and several solutions have been proposed[19]. This is a broad domain and quality acquisition is out of the scope of this paper. Notice that, in order to validate the mediation process we have used some basic acquisition mechanisms.

This paper focuses on three main problems. The first one is the definition of a realistic architecture for the global mediation system. Indeed, many systems record the description of the providers's informative capacities at some specific sites, for example through subscription to yellow pages. This works fine because, although they may change, the capacities are rather static. On the contrary, bids are very dynamic. Thus we have considered an architecture which uses providers' representatives at the mediator's site, to avoid heavy traffic between the providers and the mediators.

The second problem is the definition of the mediation process itself. Given a request, bids and qualities, the problem is to define which providers to select and

what they have to pay while ensuring a global long-run regulation of the system. The intuition is that there must be a kind of balance between bids and qualities, resulting in a balance between requesters and providers. But there must be also a balance between the different providers. This is why we use the term *mediation* (and thus mediator) with the meaning of the Merriam-Webster dictionary: *intervention between conflicting parties to promote reconciliation, settlement, or compromise*. To our knowledge, this problem has never been adressed before in its whole generality. There has been much work based on pure economics dealing with bids only or considering imposition and fair pricing only. Recently, some work has considered the introduction of a trust parameter[3], but imposition is not considered.

The third problem is the validation of the process. We have provided the definition of our flexible mediation process with a very simple preliminary validation[14]. We extend this to a thorough validation in the context of load balancing. The advantage of flexible mediation is to allocate requests with both the users and providers long-run satisfaction in mind. But this interest would be lost if it introduced performance degradation, in particular in terms of load balancing. Thus, we will challenge our flexible mediation with an algorithm which always chooses the least loaded providers. The objective is to show that the performance degradation is acceptable and understandable. This also makes it possible to illustrate the long run behaviour of the flexible mediation. Of course, additional testing is required to verify the generality of the approach to non-depletable resources such as information services.

4. Mediation system architecture

The global system architecture is described in Figure 1, with a single mediator which processes the requests. Let us stress that the money we use is *virtual*. We could either talk of tokens or any other term indicating a mechanism to regulate the system. Notice also that only the mediator manages money. It may regularly redistribute it if necessary. We consider k requesters and m providers, which advertise their capabilities. These two numbers vary over time.

The use of *provider representatives* is important in order to avoid significant network traffic. Indeed, request, bid and bill are exchanged between the mediator and each representative which are both located on the same computer. The counterpart of this choice is that each provider has to regularly inform its representative of its *preferences* on the kind of requests it would like to get. If the number of requests is important, this choice makes the number of exchanged messages decrease.

The mediator uses a registration room, because at any time, it must be able to welcome a new provider and/or accept a provider resignation. These changes are taken into account after the current mediation. When a new provider advertises its capabilities, its application is studied. If it is accepted, the registration room updates the capabilities database (*Cap*) which gathers all the registered providers' advertisements. Then it welcomes the provider's representative. With this approach, the

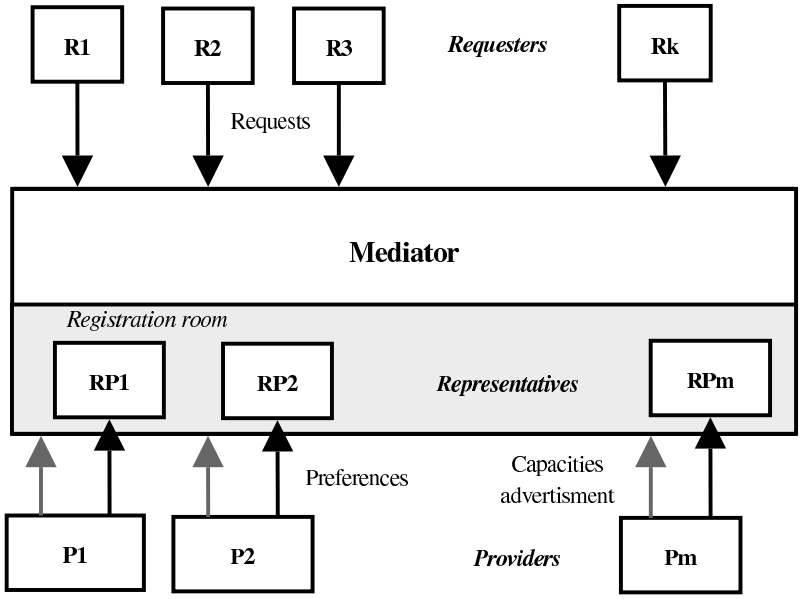


Figure 1. Mediation system architecture.

provider must regularly update its preferences at its representative. When a provider deregisters (or after a long period of inactivity), the representative is removed and the capabilities database is updated.

Quey processing does not appear in Figure 1. In fact, as for querying the providers, different options exist, depending on the model of mediation that is needed[6,32]. Thus, the querying and answers composition modules are placed on the requester side or on the mediator.

We represent the mediator's inner architecture in Figure 2. We focus on the selection of providers relevant to a given request where n providers are required. We do not mention some additional modules like those in charge of query planning or payment, which are less central. The way the quality and the providers' strategies are computed depend on the application. This is why we do not detail the nature of feed-backs nor the kind of information in the qualities database.

Each incoming request is first submitted to the matchmaking module, which uses the capabilities database to match the request with the providers capabilities. It computes a set of N providers which are able to treat the request. Then the quality evaluation module and the bidding module can be run in parallel. A qualities database (Qal) gathers feed-backs from providers or other mediators (feed-backs may come in at any time) as well as results from the mediator's own evaluation of providers (from benchmarks or analysis of answers). Given the incoming request, the quality evaluation module uses the qualities database, computes a quality for each of the N providers and gives back a quality vector of positive real numbers ($\vec{Q}$). The bidding module is in charge of collecting the bids from the N provider

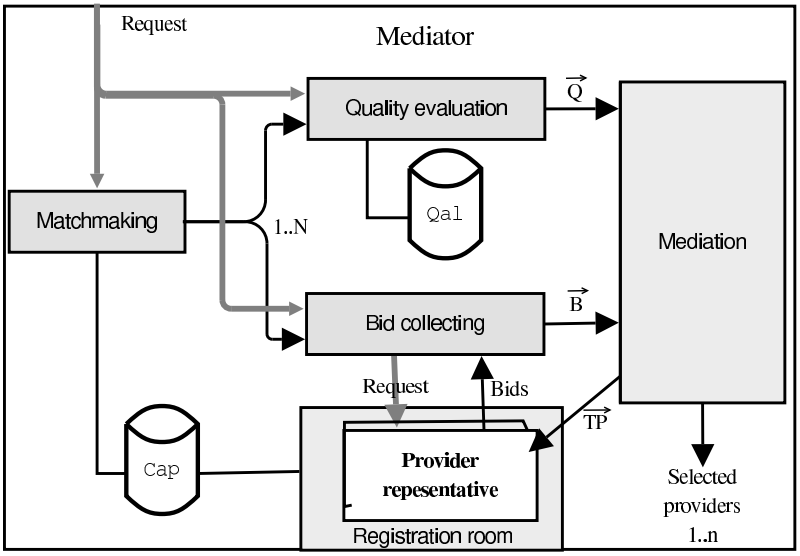


Figure 2. Mediator's architecture.

representatives. It sends them the requests, waits for the bids until a given deadline and returns a bid vector of N real numbers ($\vec{B}$).

The mediation module uses a two step process. The first step selects the n required providers among the N possible ones. The second step determines the invoicing of each of the N providers ($\vec{TP}$). Both steps use quality and bid vectors. A bill is sent to each representative. This procedure is the core of the mediator and is detailed in the next sections.

5. Mediation process

In this section, we describe our mediation process. We focus on the case where, from the mediation point of view, any given request can be viewed as a single "unit" of work called *task*. A task includes a query together with additional information like the sender, the required number of providers (noted n) or some meta-data which characterize the query. Notice that this information may be used by the representatives to determine their bids.

We assume that the matchmaking step has generated a number N of providers which are able to treat the request, named $1..N$ for convenience. The quality of those providers is represented by a vector $\vec{Q}[i]$ ($i \in [1..N]$) taking its values in $\mathbb{R}^+$. Similarly, the vector $\vec{B}[i]$ ($i \in [1..N]$) represents the providers' bids for the request and its values are in $\mathbb{R}$. A provider bids positively when it wants a given request, and it bids negatively when it does not want to treat it. For a positive bid, the higher it is, the more the provider is willing to be selected for the request. For a negative bid, the lower it is, the less the provider wants to treat the request. We assume that the values of the quality function are comparable but not necessarily

bounded. The same assumption holds for the bids.

The algorithm in Figure 3 shows the main steps of the mediation process. The ranking of the providers (vector $\vec{R}$) is based on the notion of *level* (vector $\vec{L}$). In the invoicing step, the total amount $\vec{TP}[j]$ due by a provider is the sum of the partial amounts $\vec{PP}[i,j]$ due to the selection of providers. The details of the different notions and calculations are given in the following sections and illustrated in Table 7.

```
{ IN  : [1..N], Q⃗, B⃗, n }
{ OUT : selection , TP⃗ }
begin
  for k ← 1 to N do compute R⃗[k];{ Rank the providers }
  selection ← R⃗[1..min(n, N)]; { Select the n best ones }
                          { Invoicing }
  for j in [1..N] do
     { compute j's total amount due in this mediation }
     TP⃗[j] ← 0;
     for i in selection do
        { j's partial amount due to i's selection }
        compute PP⃗[i,j];
        TP⃗[j] ← TP⃗[j] + PP⃗[i,j]
end
```

Figure 3. Mediation algorithm.

5.1. *Selection of the providers*

Definition 5.1. Vector of providers' levels.

$$\forall i \in [1..N], \vec{L}[i] = \begin{cases} (\vec{B}[i]+\varepsilon)^{\omega} \times (\vec{Q}[i]+\varepsilon)^{1-\omega} & \text{if } \vec{B}[i] \geq 0 \\ -(-\vec{B}[i]+\varepsilon)^{\omega} \times (\vec{Q}[i]+\varepsilon)^{\omega-1} & \text{otherwise.} \end{cases}$$

with $\omega \in [0..1]$ and $\varepsilon > 0$.

Intuitively, two different notions must be considered: quality and bid. Whatever their values are, no one should be neglected. Hence a weighted sum is not appropriate. Moreover, the increase of the value of one or the other parameter should increase the level. This is why a product is used. Parameter ω ensures a balance between a provider's quality and bid. It reflects the relative importance that the mediator gives to the providers' quality or bid. In particular, if $\omega = 0$ (respectively 1) the mediator only takes into account the quality (respectively the bid) of a provider. Notice that in all our simulations, up to now, we have considered that ω is fixed by a human administrator. Parameter ε, usually set to 1, prevents the level from lowering down to 0 when the bid (resp. quality) is equal to 0 whatever the

quality (resp. bid) is. In Table 7, influence of the quality can be seen by comparing p_3 and p_{10} for example. Their bids are close, but p_{10} gets a higher level because its quality is greater. Conversely, the difference between p_4 and p_5 is obtained by the values of the bids. The level induces a natural ordering:

Definition 5.2. Providers ordering.
Let r be a request. Relation $<_r$, is defined by : $\forall (i,j) \in [1..N]^2$, $i <_r j$ iff

(1) $\vec{L}[i] < \vec{L}[j]$, or

(2) $\vec{L}[i] = \vec{L}[j]$ and $i < j$

Relation $\leq_r$, obtained from $<_r$ where equality represents syntactical equality of names, is a total order on the set of N providers[12]. It always places the providers that want to treat the request before those which do not want to.
Notation. For technical reasons, we introduce the notation $\vec{R}[k]$ which represents the k^{th} provider according to the $\leq_r$ order. Intuitively, $\vec{R}[1]$ is the best provider according to ordering $\leq_r$, $\vec{R}[2]$ the second, and so on up to $\vec{R}[N]$ which is the last. The selection step selects the n best providers, i.e. from $\vec{R}[1]$ to $\vec{R}[n]$, also noted $\vec{R}[1..n]$. If there are less than n providers, all of them are selected. The complexity of the selection step is $O(N \log_2(N))$ [12].

Table 7 shows the rank obtained by the ten providers. If request r asks for three providers ($n = 3$), p_2, p_1, p_{10} are selected (selection s_1). If $n = 8$, all the providers with a positive bid are selected as well as p_6 (selection s_2) even if its negative bid reflects that it does not want to treat the request. We say that p_6 is imposed the request.

	$\vec{Q}$	$\vec{B}$	$\vec{L}$	$\vec{R}$	s_1	$\vec{TP}$	s_2	$\vec{TP}$
p_1	8	2	4.655	2	*	1.201	*	0.485
p_2	2	10	6.542	1	*	3.579	*	0.485
p_3	3	2	3.366	5		0.0	*	0.485
p_4	1	5	3.866	4		0.0	*	0.485
p_5	1	1	1.999	6		0.0	*	0.485
p_6	10	-3	-0.880	8		0.0	*	-4.231
p_7	8	-4	-1.091	9		0.0		0.485
p_8	20	-8	-1.106	10		0.0		0.485
p_9	0	1	1.516	7		0.0	*	0.485
p_{10}	10	1	3.955	3	*	0.926	*	0.485

Table 7. Two examples of selection with $\omega = 0.6$: s_1 ($n = 3$) and s_2 ($n = 8$)

5.2. *Invoicing*

In usual auction mechanisms, invoicing is based on the comparison of the bids only. Here, the task is more complicated by the fact that each bid is balanced by a quality. Hence we cannot directly compare the bids. This is why we introduce the notion of *theoretical bid* of a provider.

5.2.1. *Theoretical bid.*

It represents the bid that the provider should make in order to get a given level l. We do not consider the same question for the quality. Indeed, the provider cannot change its quality as it does for bids.

Proposition 5.1. *Let r be a request and let $i \in [1..N]$ be a provider. If the mediator uses a selection strategy such that $\omega \neq 0$ then, the theoretical bid with respect to r that i should make to reach level l is :*
$\vec{B}^{Th}(i,l) = \alpha\ max(((\alpha \times l)^{\frac{1}{\omega}}(\vec{Q}[i]+\varepsilon)^{\frac{\alpha(\omega-1)}{\omega}} - \varepsilon), 0)$
where $\alpha = 1$ if $l \geq 0$, and $\alpha = -1$ otherwise.

This formula is the result of the resolution of the equation $\vec{L}[i] = l$, $\vec{Q}[i]$, ω, and ε being fixed. According to this proposition, with the data from Table 7, provider p_2 has to bid 3.579 in order to obtain the same level as p_4. Conversely, in order to come to p_2's level, p_4 must increase its bid up to 13.414. Notice that the theoretical bid grows as a function of l (all other parameters being fixed). The definition of theoretical bid enables to specify the invoicing. Two cases are considered: competition and requisition. In this latter case, the cost of requisition is shared between *all the providers which are able to treat the request* (including those which have not been selected[22]). In other words, when some provider is requisitioned, all the others pay to support its effort. If more than one provider are requested, the total amount due by a provider j ($\vec{TP}$ [j]) is the result of the addition of each requisition cost. To reflect this, we introduce the notation $\vec{PP}[i,j]$ (j's partial invoice corresponding to the selection of provider i). To obtain an homogeneous notation, we use it in case of a requisition and a competition though it is not useful in the latter case.

5.2.2. *Partial invoice in the competitive case.*

In a competitive situation, a selected provider has made a positive bid. Competition is effective when more than n providers have done so. The calculation of the invoicing is carried out by comparing a selected provider with the best one which has not been selected. However, the amount of the invoice is not computed only from the bids. Instead, we consider the level of the providers which takes both offer and quality into account. Therefore, the partial invoicing of a selected provider corresponds to the bid which it should make to get the same level as the best unselected provider (theoretical bid). Note also that only selected providers have to pay something.

Definition 5.3. The partial invoicing of a provider $j \in [1..N]$ concerning the selection of $i \in [1..N]$ in the case $\vec{B}[i] \geq 0$, is:

$$\vec{PP}[i,j] = \begin{cases} \vec{B}^{Th}(j, \vec{L}[\vec{R}[n+1]]) & \text{if } n < N \text{ and } i = j \text{ and } \vec{B}[\vec{R}[n+1]] \geq 0 \\ 0 & \text{else} \end{cases}$$

5.2.3. *Partial invoice in the requisition case.*

The situation is a requisition when at least one provider, having quoted negative, is selected. The idea is to distribute the cost of the requisition on all the providers able to answer the request (and not only on those selected).

Definition 5.4. The partial invoicing of a provider $j \in [1..N]$ concerning the selection of $i \in [1..N]$ in the case $\vec{B}[i] < 0$ is:

$$\vec{PP}[i,j] = \begin{cases} \frac{-\vec{B}^{Th}(i,\vec{L}[\vec{R}[min(n+2,N)]])}{N} & \text{if } i \neq j \\ \vec{B}^{Th}(i, \vec{L}[\vec{R}[min(n+1,N)]]) - \frac{\vec{B}^{Th}(i,\vec{L}[\vec{R}[min(n+2,N)]])}{N} & \text{else} \end{cases}$$

In the first line, the provider, which is not selected, is required to support the selected one. In the second line, the amount allocated to the requisitioned provider is computed. Even if requisitioned, the provider is asked a given amount. The fact that some provider p_a is imposed a request r is supported by all the providers.

5.2.4. *Global invoicing.*

The total amount owed by each provider is obtained by adding the partial bills related to each selected provider. Of course, in the following, if i is not selected, $\vec{PP}[i,j] = 0$.

Definition 5.5. The invoicing of every provider $j \in [1..N]$ is defined by:

$$\vec{TP}[j] = \sum_{i \in [1..min(N,n)]} \vec{PP}[i,j]$$

The complexity of this calculation is $\Theta(N \times min(N,n))$.

Notice that in a competitive case, the provider never pays more than its own bid. In Table 7, selection s_1 corresponds to a competitive case. The selected providers p_1, p_2, p_{10} are the only ones to pay something. Selection s_2 corresponds to a requisition case (p_6). In that case, all the (ten) providers support the financial effort. Under the same conditions, had n be equal to 7, all the providers wanting the request would have gotten it and none of them would have been requisitioned. There would have been no invoicing because this is neither a competition nor a requisition.

5.3. *Illustration of the Flexible Mediation behaviour*

This section gives some examples to illustrate the behaviour of Flexible Mediation over short sequences of 5 mediations. The next section on validation shows the long run behaviour of the process.

To make it simple, we consider five providers, p_1, p_2, p_3, p_4, and p_5, which have the same quality (9) and the same initial amount of money (100). The required number of providers is always 2.

5.3.1. *Sequence of five competitive mediations.*

All the providers bid the same on the first request (10). Their bids decrease while they spend their money. Avoiding all computation details, Table 8 shows who gets the requests. As we can see, things are rather fair since, after the five mediations, all of them have obtained 2 requests. The starvation problem is avoided.

Table 8. Requests allocations in a sequence of competitive mediations.

	1	2	3	4	5
p_1	*		*		
p_2	*			*	
p_3		*			*
p_4		*			*
p_5			*	*	

But, due to the invoicing, after these five mediations, they do not exactly end with the same amount of money (Table 9).

Table 9. Providers balance after a sequence of competitive mediations.

p_1	p_2	p_3	p_4	p_5
81	81	81.8	81.8	82

5.3.2. *An imposition followed by four competitions.*

For the first incoming request, all the bids have the same negative value (-10). For the following requests, they bid positively according to the amount of money they own, as in the previous example.

Table 10 shows that p_1 and p_2 are imposed the first request. The money they obtain in compensation through the invoicing makes it possible for them to obtain the two following requests they want before the other providers start to obtain any. Considering their final amount of money (see Table 11), if there were another similar run, it would be p_5 and p_2 which would win the competition. Hence satisfaction of the providers is met.

Table 10. An imposition followed by 4 competitions: requests allocations.

	1	2	3	4	5
p_1	*	*	*		*
p_2	*	*	*		
p_3				*	
p_4				*	
p_5					*

Table 11. Providers balance after the imposition and the 4 competitions

p_1	p_2	p_3	p_4	p_5
78.12	86.8	86.4	86.4	87.32

5.3.3. *Sequence of five impositions.*

If the providers bids are the same for each mediation, the imposed providers are always p_1 and p_2. A turn over can be obtained in two different ways. Fist, the choice can be randomized in case of equal level instead of using the alphabetic order. A second solution is to make the bid decrease when the number of previous impositions increases. The results presented in Table 12 use the second method. Each provider is imposed two requests. However, they do not end with exactly the same money balance (see Table 13). After these impositions, their amount of money is quite similar to what it was at the begining (100 each).

Table 12. Requests allocations in a sequence of five impositions.

	1	2	3	4	5
p_1	*		*		
p_2	*			*	
p_3		*		*	
p_4		*			*
p_5			*		*

6. Experimental validation

In this section, we provide an experimental validation of our flexible mediation (FM) in the context of load balancing which is a good indication of the system's overall

Table 13. Providers balance after a sequence of five impositions.

p_1	p_2	p_3	p_4	p_5
99	99	99	100	101

performance. Recall that the advantage of flexible mediation should not be at the expense of performance. This is why, load balancing is a first stage of performance evaluation. We confront the flexible mediation algorithm to an algorithm (called LL) which always chooses the least loaded providers. The objective is to show that the performance degradation is acceptable and understandable. In parallel, we check the long run behaviour of the flexible mediation. Of course, additional testing is required to verify the generalizability of the approach to non-depletable resources such as information services.

In the following, we first present the validation's assumptions for FM and LL which are specific to load balancing. Then, we conduct a set of experiments when the system load varies. Finally, we study the process behavior when parameter ω evolves. Each set of experiments is conducted both when providers cannot leave the system and when they can.

6.1. *General parameters of the experiments*

In the flexible mediation process, the mediator plays the role of a central bank. It gives out a virtual amount of money to the providers' representatives at the time of subscription; the providers themselves know nothing about this money. Each representative manages its money in order to carry out the needs of its providers in the best possible way. When a provider deregisters from the mediator, the corresponding amount of money is progressively withdrawn from the system by the bank. Thus, the total amount of money which circulates in the system is proportional to the number of registered providers. After some mediations, even if the aim of a mediator is not to make money, it would get back the money which it has itself put in circulation. Thus, it regularly redistributes this profit equitably to all the representatives to avoid blocking the system. Indeed, representatives may become unable to make a positive bid because of lack of money. Notice that the amount of money that each provider can own is limited[b].

We have restricted our experiments to one requester, one mediator and many providers. The number of providers varies from 10 to 1000 according to the values given in Table 14.

Time has a discrete representation in terms of time units. Requester, mediator and providers are all synchronized with respect to this time. For each provider, a given request represents some workload, called execution cost of the request,

[b]This is just to avoid a provider which does nothing, and so does not spend its money, to capitalize all the money.

number of providers	10	100	250	500	750	1000

Table 14. Variation of the number of providers

expressed in some unit (this can be CPU, disk space...). Each provider has a processing capacity which represents the number of units which it is able to treat per time unit. Each provider has a load tolerance threshold about the work to be done and stored in a FIFO structure. In the experiments, this threshold is set to 100%.

The LL process aims at obtaining the smallest response time. It does not use the representatives' bids. Its principle is to allocate a request to the n providers that could evaluate it in the fastest way taking into account their current load.

In FM, a representative bids for a request according to the current load of its provider as well as to the load tolerance threshold fixed by its provider. When a provider load is null, its representative will bid positively at least once. When its load becomes strictly higher than its load tolerance threshold, it will bid negatively: if the mediator wants the provider to treat a request, it has to requisition it. Moreover, the process estimates providers' qualities in function of their current loads.

During any allocation process, there is a calculation time in order to determine the selected providers. The complexity of the computation itself is the same (in $\Theta(N \log_2(N))$) for the two processes. However, the times necessary to get data for this computation are different. In FM, bids are locally obtained from the representatives while in LL, the providers' current loads are incured via an exchange of messages. Thus, in LL, getting all the data is longer than in FM. To obtain comparable results, we do not take into account this calculation time in the final response time. We only focus on the response time between requester and providers.

For each process, for each simulation, the requester sends a minimum of 18000 requests sequentially. A request asks for one provider at least. It can ask for up to 25 % of providers that are present in the system. Requests are assigned in a FIFO manner, on the one hand by the mediator and on the other hand by providers. We apply the same set of requests (with the same properties) to the two processes in order to obtain comparable results.

As we consider an open system, providers can decide to deregister from the mediator if they are not satisfied by the requests they receive. In order to express the providers' (dis)satisfaction with an allocated request, we define a utility function U and a satisfaction threshold. U is negative when a provider obtained the request while it is overloaded; otherwise it is positive. We sum utilities for a same provider. When this result is under the satisfaction threshold, the provider deregisters from the mediator. As there is only one mediator in the system, this provider tries to register again regularly.

6.2. *Variation of response time in function of system utilization*

In FM, the providers bid according to their preferences: low loaded providers bid positively while high loaded ones bid negatively. Hence, we want to study the behaviour of the mediation process when load increases.

6.2.1. *Experiments setup*

The system capacity is defined as the sum of all the providers' processing capacities. We define the system's theoretical utilization (called utilization in the following) during a time unit as a ratio: the sum of the costs of all the requests sent during the time unit divided by the system capacity. The system real load depends on the allocation process that impacts the response time that we measure.

Different utilization intervals have been defined (Table 15). For a given interval, we compute the number of requests which the requester must inject in the system by time unit according to the number of present providers.

Theoretical utilization (in %)	0-25	25-50	50-75	75-100

Table 15. Theoretical utilization intervals used for the experiments

For each simulation, we fix a number of providers and an utilization interval. Finally, the value of parameter ω is fixed to 0.5.

We present the experiments only in the case where providers never leave the system during a simulation. When they sometimes do, the results are similar to those presented in this section, except when utilization is between 75% and 100%. This difference appears also in Section 6.3 and will be discussed there.

6.2.2. *Results*

We have noted a stability of response time during all simulations. This is why we only represent the average response time of each number of providers in Figure 4. The x-axis shows the different utilization intervals that we applied.

When utilization is between 0% and 50%, we note a constant response time for the two processes: an average of 3.53 time units for FM and 3.0 for LL. When few requests are sent during a time unit, the mediator of LL only uses the fastest providers, i.e. those that can reply in three time units at most. The FM process is slightly slower but favours the work for all.

When utilization raises slightly (between 50% and 75%), the response time of LL also increases up to 3.20 time units whereas the time of FM decreases to 3.48 time units. Indeed, the mediator of LL is now compelled to use providers that are slower because the number of requests to treat is greater and all the fastest providers are overloaded. This increase of load has a weak impact on the behaviour of FM because it already used to allocate requests to slower providers.

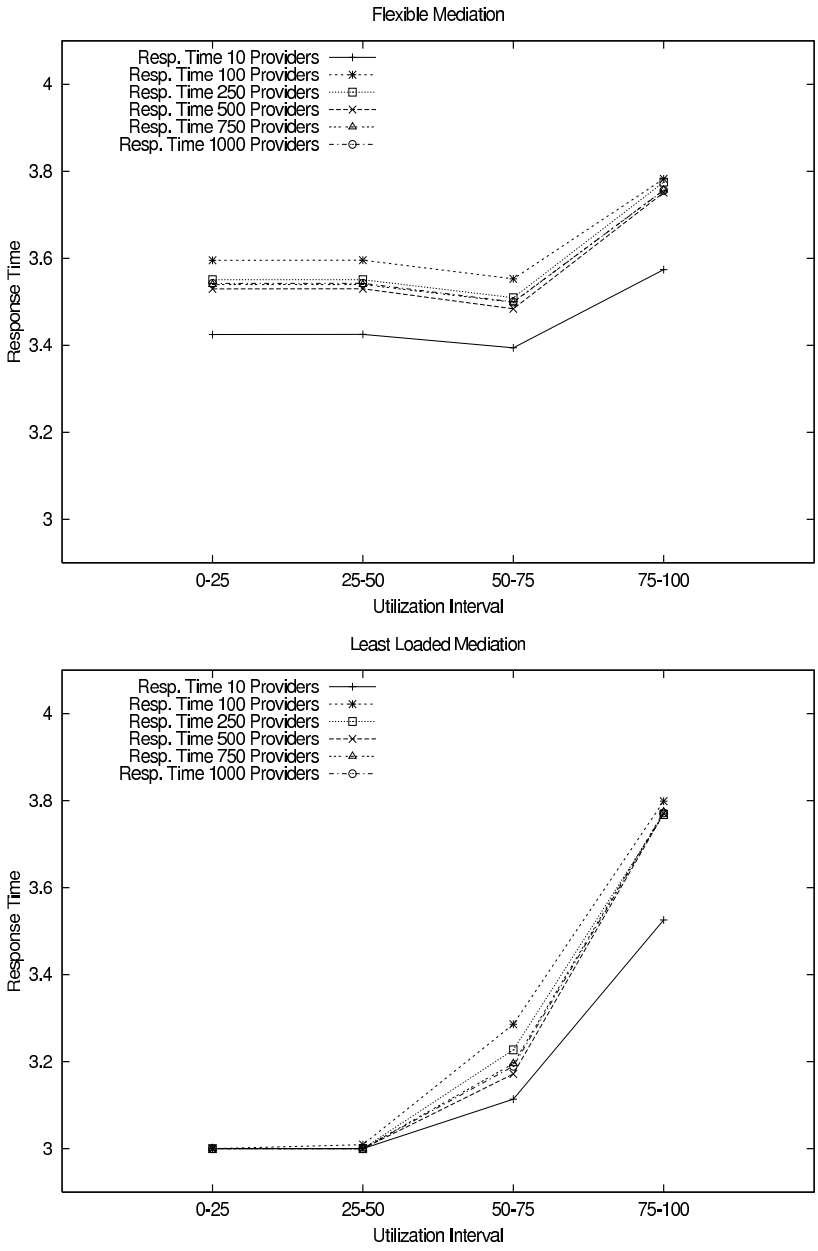


Figure 4. Variation of response time versus system utilization

Finally, when the utilization is between 75% and 100%, the two processes have a similar response time, near to 3.73 time units. This behaviour is consistent because, at this level of load, all the providers (even the slowests) are overloaded.

6.3. *Variation of response time in function of parameter ω*

In FM, favouring the quality of the response corresponds to ω near to 0. On the contrary, when ω is close to 1, the process favours the providers bids. Hence, our aim is to study the influence of ω in the context of load balancing.

The LL process has a constant behaviour for all values of ω because parameter ω is specific to FM. In Figures 5, 6, 7 and 8, for each value of ω, we consider six cases, corresponding to different numbers of providers (from 10 to 1000). The six corresponding response times are indicated for the FM process. For?the LL process, for each value of ω, we average the results of the six response times (central square in the caption). We also quote the minimal and maximal response time.

6.3.1. *Experiments setup*

Parameter ω influences the interest that the requester and the providers can have to participate actively in the system. We test FM with ω taking the following values:

ω	0	0.00001	0.25	0.5	0.75	1

Table 16. Variation of parameter ω

The results are presented for the four utilization intervals. To obtain consistent results established on a same base, the set of requests is the same for all the experiments. During a simulation, there are several possible combinaisons of parameters: a fixed utilization interval, a fixed number of providers and one value for ω.

6.3.2. *When providers never leave the system*

As in Section 6.2, we represent the average response time. On the x-axis, we note different values that parameter ω takes during the simulations.

In Figures 5 and 6, when ω is null, the results are similar for the two processes. These results are conform with our expectations because in this case, we only take into account the quality of providers and bids do not interfere. We also notice that the curves paces are similar for the different numbers of providers.

In FM when utilization is between 0% and 50% (Figures 5a and 5b), the curves have similar shapes. As ω gets closer to 1, we take into account providers' bids more and more while ignoring their qualities more and more. So, the slowest providers making greater bids (in order to compensate their bad qualities) can be selected. Finally, their bids become negative because they reach their load tolerance threshold. This avoids poor response time with acceptable limits for the requester.

In Figure 6a (utilization is between 50% and 75%), the response time increases until ω reaches 0.5 (state where opinions of requester and providers have the same weight) followed by a stabilization. Indeed, as the load increases, the FM mediator must use more providers. At first, the slowest providers can obtain requests thanks

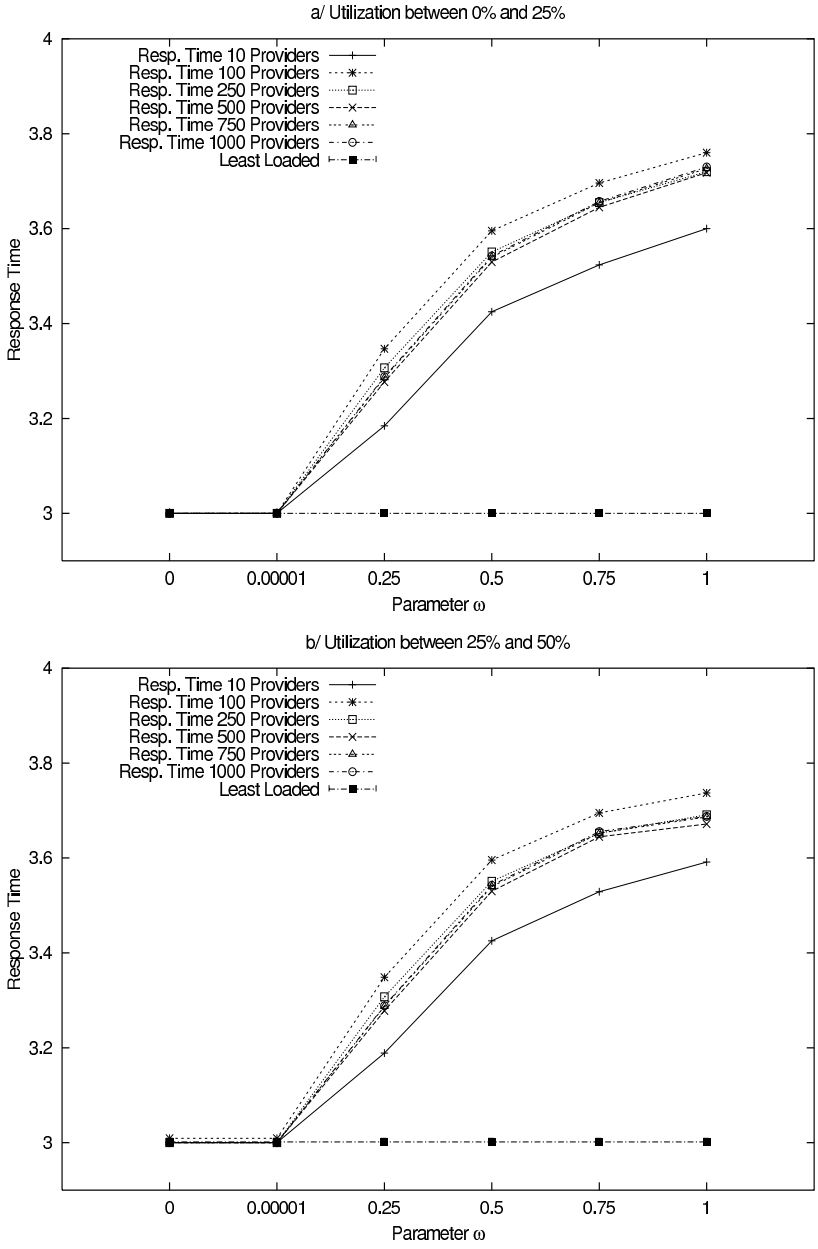


Figure 5. Variation of response time versus parameter ω without providers' departure

to their bids but they become overloaded whereas the fastest can manage more requests and bid positively for a longer time. Compared to Figure 5a, the fastest make the average response time decrease because they participate more in the process.

For highest utilization (Figure 6b), the response time of LL increases a lot. Results of the two processes are the same up to the value of 0.75 for ω (on average 3.80 time units). In FM when ω is maximal, the quality is no more considered.

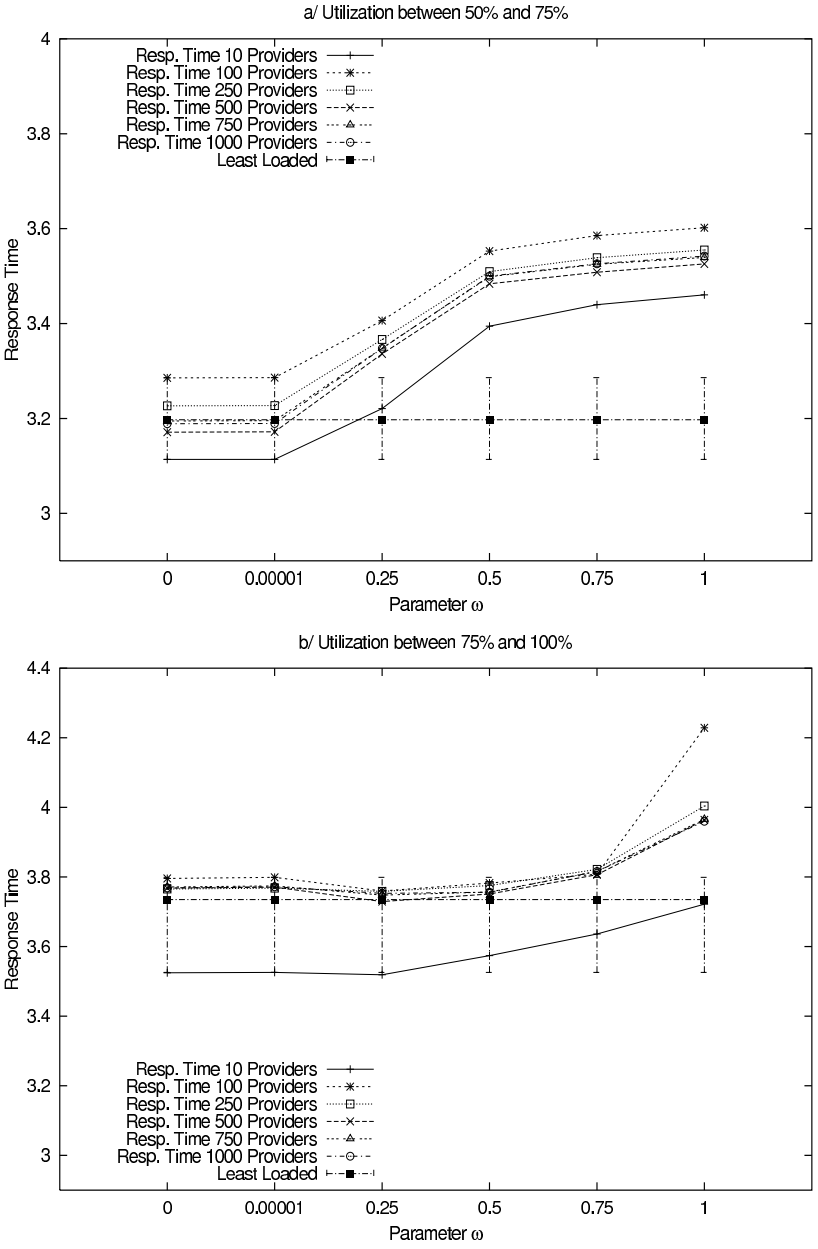


Figure 6. Variation of response time versus parameter ω without providers' departure

Thus, the slowest providers can obtain requests via their bids. We also note a little decrease when ω increases from 0 to 0.25. This is because when ω is null, bids do not interfere.

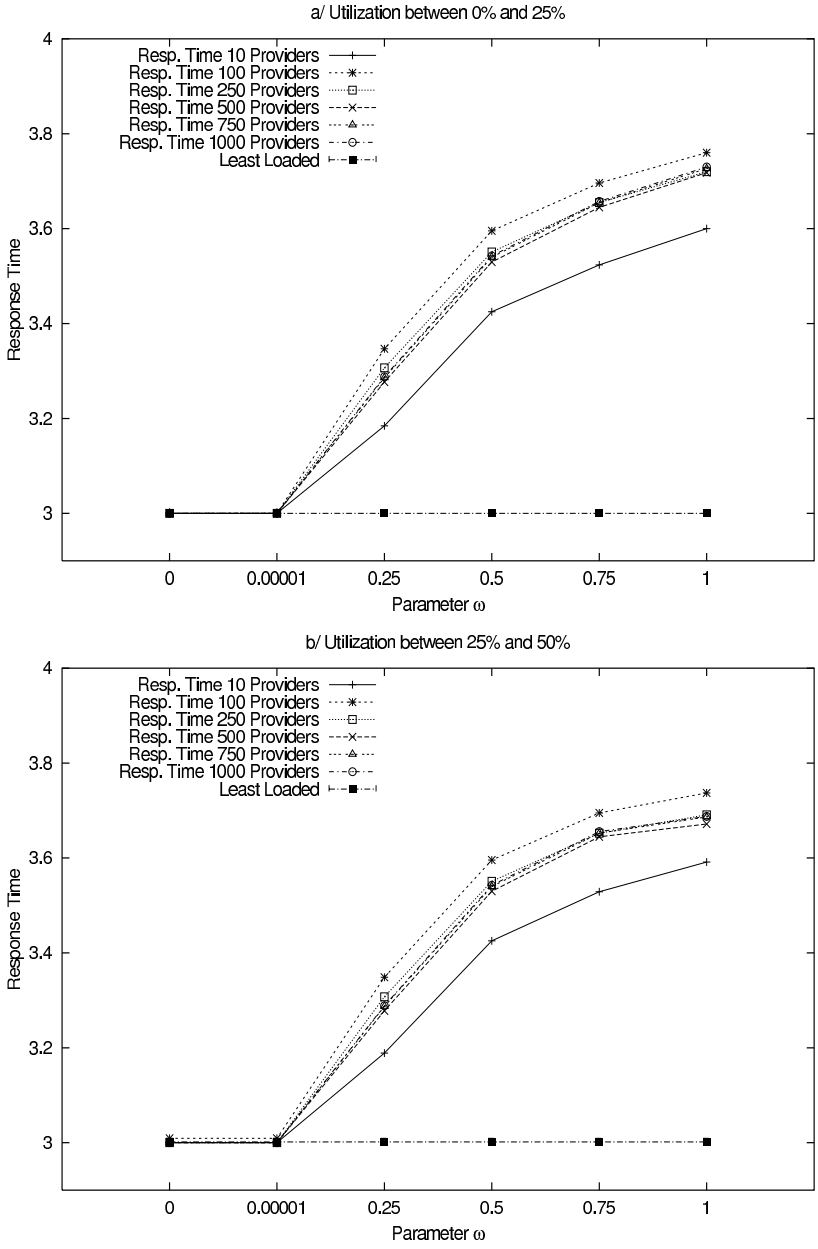


Figure 7. Variation of response time versus parameter ω with providers' departure

6.3.3. *When providers sometimes leave the system*

For a provider, the leave from the system is due to a long disatisfaction. This appears when the provider is imposed (see Section 6.1), i.e. when it is overloaded in a long time. So, the results are similar to those of Section 6.3.2 as far as utilization is between 0% and 75% (Figures 7a,b and 8a), i.e. when the load is not high.

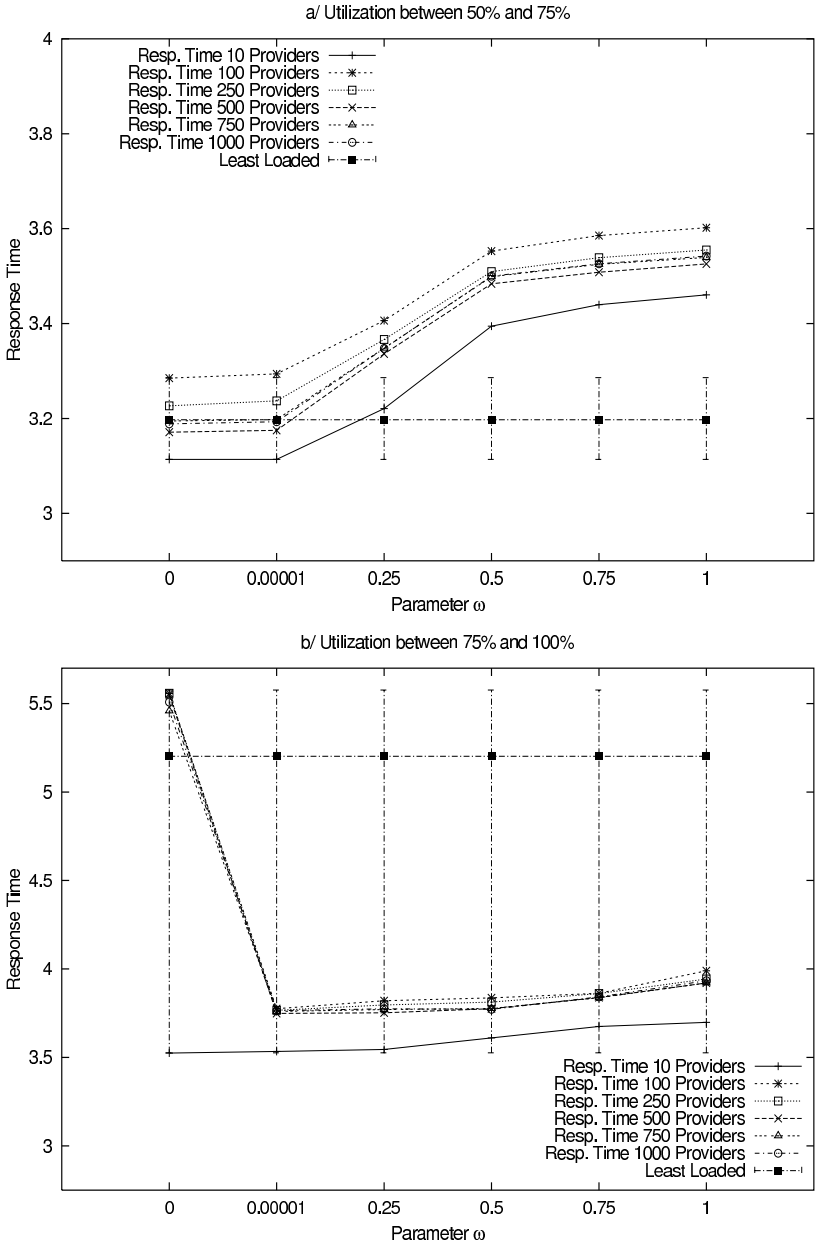


Figure 8. Variation of response time versus parameter ω with providers' departure

The importance of the providers departure appears when utilization is over 75% (Figure 8b). Indeed, the LL process carries on charging the fastest providers in a first time. Thus, these latter cancel their subscription and the mediator must choose slower providers until some faster ones register again. In the case of FM process, this phenomenon does not occur. The fastest providers bid negatively when they are overloaded and the process charges slower providers. So, the number of providers

which deregister is very low.

In Figure 8b, an important difference appears in FM results when ω has the values 0 and 0.00001. In the first case, FM has the same behaviour as LL and does not take into account providers bids whereas in the second case, bids interfere in the mediation. When ω is equal to 0.00001, a representative bidding negatively indicates that it is overloaded and thus, the mediator selects an other provider. It is not the case when ω has the value 0.

6.4. *Conclusions*

When utilization is low, we note that the results of LL are better than those of FM. However, when utilization increases, the LL's results deteriorate significantly. In the case of FM, they also detoriate. However, when utilization is high, the behavior of FM is the same as LL's.

In FM , the value of parameter ω has a great influence on the results. When ω is close to 0 (i.e. when quality prevails) and whatever the load, FM behaves just as LL. When ω is greater, the results of FM deteriorate owing to the important recognition of bids in comparison of qualities. This increase of response time is the counterpart of the mediator's policy which chooses to allow all the providers to work (even if they are slower).

The providers departure also plays an important role when utilization is high. We observe that FM's results are better than those of LL. Indeed, in LL, fastest providers are saturated during the first mediations whereas the FM process allocates requests to more providers, even to slowest ones. Thus, in LL, the fastest providers are dissatisfied faster and they deregister faster. Hence, the FM mediator has more providers at its disposal.

7. Related work

The problem that we address is very general. The field of distributed databases uses the term *query allocation* which just appears as a subproblem of query processing[10]. In multi-agent applications, the problem is often referred to as task allocation, where the goal is to find a plan to have the subtasks allocated and executed. In fact, request or task allocation is addressed in many domains ranging from distributed database systems to multi-agent systems, P2P data systems, networking systems, etc. The assumptions and techniques often differ depending on the context and do not result in the same system characteristics.

We first draw a picture of related and sometimes complementary techniques which do not adopt an economic approach. Then we focus on the economic ones.

Data mediators rely on distributed database technology[35,36] to allow users to transparently query different data sources that are typically "wrapped" to provide an uniform interface to a mediator[30]. A mediator decomposes a user query into queries for the different data sources and integrates the results, much like a distributed database system. To work properly, data mediators require a global

schema, typically relational or XML, to be designed over all data sources[28]. However, maintaining a global schema is difficult when source schemas change frequently or heterogeneity increases. Our solution does not require a common global schema. Furthermore, data sources are not passive since they can bid for requests provided by the mediator.

In the field of multi-agent systems (MAS), the Contract Net Protocol (CNP)[25] is often mentioned as a way to allocate tasks. Some agent A that wants a task to be completed by another agent sends a call for proposal to its accointances. Those agents which want to complete the task reply by giving the conditions of execution. Then agent A compares the offers and chooses the best agent according to its own criteria and informs the agent that has been selected. At first sight, this protocol does not seem very far from an auction protocol. However, the CNP is meant to be used in a cooperative context (the agents are not self-interested). Also, it is generally assumed a rather small number of agents, and a detailed description of the conditions of execution which is not the case in our work.

Several types of matchmaking based facilitators have been defined[11,6,20,32]. The matchmaking algorithms find the providers which are able to treat a given request by matching their capabilities advertisements with the given request. Languages to advertise capabilities have been defined[27]. Matchmaking algorithms are efficient but the number of selected providers may remain too large. Recently, some works have investigating the possibility of reducing it by using a notion of quality[34] or word of mouth[2]. The former work clearly suggests to first perform classical matchmaking, and then to refine the obtained selection. This viewpoint is quite similar with our. The facilitator in[34] uses track records about each provider. The records are obtained using benchmarks and users' feed-back. Our model uses a simpler representation of quality, which is just represented as a number. It can be obtained in a similar way. Our proposal strongly differs from these works because the mediation process uses not only the providers quality but also their bids for requests, thus allowing them a more active participation in the allocation process.

More generally, the notion of quality that we use is related to that of trust and reputation. As for computing reputation or trust, several rather succeeded works exist. They can be found in several surveys[19,16]. Trust and reputation are used in many works, in particular in conjunction with other parameters in economic models.

7.1. *Economic approaches*

Over the last twenty years, an increasing number of works have considered using the principles and models of Micro-Economy[18] in the field of Computer Science.

Auctions are widely recognized as a way to manage negotiation among participants. They have probably been among the first models to be studied and used because of their online use to sell material goods to people. Several kinds of auction mechanisms exist[31,23]. For example, the generalized Vickrey auction selects the n best bidders who pay the price offered by the $(n+1)^{th}$ best bidder. In the purely

competitive case our work looks like this generalized Vickrey auction, but we push generalization further because we take into account the quality factor via ranking and theoretical bid. The flexible mediation comes back to a generalized Vickrey auction when all the bids are positive, $\omega = 1$ (i.e. does not take quality into account), and $\varepsilon = 0$.

Multi-attribute auctions[5,29] are another kind of generalization, which help finding goods suppliers, without considering requisition. The basic idea is that a good is not only qualified by a price but several other attributes such as quality. Obviously, in that case quality is attached to an item, while it is attached to the provider in our work. The technical consequence is that price and quality do not evolve the same way at all (for example, in multi-attribute auctions, the price increases if quality increases) leading to different formulas.

When used on-line for selling some goods to people or organizations, auctions are immersed in a robust pre-existent total economic system (the human one). This is not the case when applying economic principles and models to the design of computer systems where all the entities are non-human. Hence a designer must carefully think of the impact of the proposed model on the global long-run regulation of the system. This is a reason why a transposition from pure Economy to Computer Science is not always obvious. Other reasons are the computers bounded computationnal power, decentralization, dynamicity and openness[4]...

Despite these difficulties and probably because they seem appropriate whenever autonomous entities have to interact[8], economic models have been used for many different purposes. For example, The University of Michigan Digital Library (UMDL) project[7] has explored the use of auctions to treat requests for documents (without using any notion of quality nor of requisition). Many recent works explore the use of market based approaches for the management of resources in P2P or Grid environment[1]. For example, trust and incentive-based mechanisms can ensure that peers forward the requests they get, thus addressing the non-cooperation problem[33,17].

In this context of plentiful production, our bibliographic study has particularly focused on six works (the only ones to our knowledge) which seem close to ours, either because of the objectives (request or task allocation) or because of the technical implementation (same kind of parameters for example)[26,15,21,22,3,9]. In all these works, requesters and providers can be identified, although they may not be called so. Some works clearly use a kind of facilitator (called the *center* for example). In other works, a requester can become a center, temporarily or not.

Mariposa[26] pioneered the use of a market approach for data mediators (then called distributed data managers). It uses an economical model for allocating queries to data sources based on a bidding process. A data source temporarily becomes a broker by receiving a request of a user and the budget that this user grants to it. The broker parses the request in subrequests and prepares query execution plans according to data movement, parallel execution.... So, it sends subrequests to potential providers (obtained via a service of yellow pages...) The providers reply

with cost and delay to make the request and the scratch date of their proposals. If a subrequest does not have a response, the total request will not be realised. Otherwise, the broker chooses the best set of providers respecting the fixed budget within the best possible delay. However, the mediation procedure of Mariposa[26] is limited. It does not take into account providers' quality nor trust and some queries may not get processed although relevant data sources exist.

The mechanism proposed by Likhodedov and Al[15] aims at maximizing both the buyers' utilities and the seller's utility. This is difficult because these two goals are contradictory. To cope with this difficulty, the buyers' viewpoint is adopted with a constraint corresponding to the seller's viewpoint. The main principle is the following : the seller puts on sale several units of the same item. The buyers can buy at most one unit each. The seller is not forced to sell all its units, in particular when buyers proposals are below its reservation price. The seller allocates the units according to its utility (and thus its reservation price). Each buyer must pay the amount which the item would have been worth to it if it had submitted its lowest possible winning bid.

Porter and Al[21] focus on fault-tolerant mechanism design in a task allocation context. The kind of failure that is considered is when an agent encounters a problem beyond its control during task execution. For each type of task, an agent has an execution cost and a probability of success to complete it. A center manages the task allocation. Several tasks may be allocated at the same time, but the problem is kept non-combinatorial. The center asks each agent that it declares its type (expressed as a vector of success probability and a vector of costs). Task allocation maximizes the system welfare. Each task is allocated to an agent. After some time, the center pays the agents which have completed their tasks and it claims a compensation for the agents which have not succeeded in completing the task. With this system, some tasks may not be completed.

Imposition[24] occurs any time a participant is forced (required) to perform a task that it does not want to. The basic idea of fair imposition[24,22] is that all the participants must support the imposed one. The problem is tackled from a purely economical viewpoint, each participant sending their cost to perform the task. Fairness is obtained because the invoicing asks all participants to pay the same amount and gives a compensation to the imposed one. In our flexible mediation, the requisition case generalizes the fair imposition mechanism, with the notion of quality and to n selected participants. It comes back to fair imposition[12,13] when $n = 1$ (only one selected provider), $\omega = 1$ (don't take quality into account), and $\varepsilon = 0$ (removing the technical parameter).

Dash and Al[3] revisit traditional mechanism design when augmented with the notion of trust, in a task allocation context. In the system, the roles of the agents may be users or/and providers. Agent i's trust in agent j depends on i's perception of j's probability of success (POS) in completing a given task, but also on the POS the other agents attribute to j. In the course of allocations, the evaluation of trust is refined because there are more and more interactions and thus more

and more reliable POS. Given some agent i, that another agent completes a task is evaluated to some given value by i, whereas i has a cost to complete this task itself. Every agent sends all the tasks it wants to allocate to the auctionneer. It also announces its POS vector and its trust calculation function. For all agents, the center evaluates their trust on other agents and sends the set of tasks to be allocated. In response, each agent sends its costs and values for the proposed tasks. So, the center determines the efficient allocation by maximizing the value of the allocation. At last, it calculates the payment of each agent.

The paper by Gorobets and Al[9] does not lie in the same field as the previous ones, but rather in Agent Based Computational Economics[c]. Its aims is to "investigate under what conditions trust can be viable" in the standard Transaction Cost Economics. In the system, the roles of the agents may be buyers or suppliers. A buyer can make rather than buy if it estimates that it is more profitable for it (there is a tolerance threshold). Transactions occur on the basis of long run relations between the agents. Each agent establishes a ranking of all its possible alternatives. The agent's scoring takes into account the profit expected from the transaction (through product selling) and trust (expressed as a probability of possible defection of the suppliers). Technically, a parameter enables to support profit more than trust and vice-versa. Each buyer sends a given number of requests to its most preferred suppliers. Each supplier is free to accept or reject a request, according to its most preferred buyers. If one of its request is rejected, the buyer sends it to less preferred suppliers according to its ordering. This is repeated until the request is accepted or the tolerance threshold is reached. In this latter case, the buyer carries out the request itself. After this matching phase, the selected suppliers produce and deliver for their buyers. Finally, all buyers sell their products on the final-goods market. Profit is shared equally with their supplier, if they have one.

From a user's viewpoint, in our approach, users can send a feedback about the providers that are proposed to them. In this way, they are more active with respect to the mechanism. This feature is also present in some of these works[9,3]. But, as far as we can tell, they is no such possibility in the others[15,26]. There, the users just send their requests and must be satisfied with the answers they get. Notice that fair imposition[22] does not use any notion of quality nor trust. So the question does not arise.

The works also differ on the number of providers that a user can ask for, and on the number that it gets back. Only a single provider can be asked for in four works[21,9,3,22]. Our flexible mediation enables a user to ask for several ones. This is also the case in the work by Likhodedov and Al[15]. We cannot tell whether it is possible in Mariposa[26]. Whether the number of suppliers that can be indicated is 1 or n, another question is the number of providers the users really get. Only the flexible mediation and two other mechanisms[9,22] ensure that the request will be

[c]Agent-based computational economics (ACE) is the computational study of economic processes modeled as dynamic systems of interacting agents.

treated by the specified number of providers (as far as there are enough providers in the system).

Let us consider the providers' viewpoint. They must indicate the cost associated to the request in three works[21,3,22]. Bids are used by the flexible mediation and by Likhodedov and Al's mechanism[15]. No bids nor costs are used in the work by Gorobets and Al[9], but trust and estimation of profit. To our mind, in order to express preferences on requests, using bids is a larger and more active means than costs. Indeed, the cost does not enable a provider to express its interest in a request whereas additional criteria may be taken into account via a bid (financial statement, current workload, own preference for given types of requests...)

Some mechanisms make it possible for the providers to tell the center that they do not want to treat the request that is proposed[26,9]. This is also true in the flexible mediation. In Mariposa[26], it seems that the providers can not even treat an allocated request. On the contrary, once a request is allocated, the providers in [21,9,22] must treat it. This is also the case in the flexible mediation, which is rather reassuring for the user.

8. Conclusion and future work

In this paper, we addressed the problem of mediation in large distributed information systems, considering that it does not only consist in finding relevant information providers for requests but also in finding relevant requests for providers. Our work brings several contributions:

First, we proposed a mediation system architecture where the mediator maintains databases about the providers capabilities and qualities, collects the providers bids for each request and uses a mediation module to select the required number of providers in a balanced way. Providers' representatives were used to reduce network load due to bidding.

Second, we defined the flexible mediation and detailed both the selection and invoicing steps. The originality of this process is to take into account both the providers interest and qualities while ensuring that every request is satisfied as far as enough providers with the required capabilities are present. To our knowledge there is no work which combines both qualities and bids and also introduces a requisition process, with the same very long-run regulation of the system showed by the first experiments.

Finally, validation has been conducted in the context of load balancing. We stress on the fact that the flexible mediation advantage is to allocate requests with both the users and providers long-run satisfaction in mind. However, it is important to check whether the flexible mediation does not present major degradations from a performance viewpoint, in particular considering load balancing. This is why, load balancing appeared as a first necessary stage of evaluation. From this viewpoint, the results are quite good. In parallel, these tests helped us to check the long run behaviour of the flexible mediation. The process shows more flexibility because it

avoids some providers to monopolize the requests, gives medium quality providers the opportunity to get some requests and thus, gives them some chance to improve their quality score. This is why the process can adapt faster to changes in the providers' behaviour and ensure a very good long-run regulation of the system.

In the future, at least three directions should be considered.

First, this paper shows that we need the definition of new theoretical properties of a process. Indeed, it is generally of use to check whether properties which are generally valued in the field of Economy are verified. For example, Dash and Al[3] report about the incentive compatibility of their mechanism, and the individual rationnality of the providers in this mechanism. Our approach remains very careful on this subject. To our mind, these properties may not be useful, nor welcome depending on the required effects, particularly for the long term behaviour of the system. For example, we are convinced that in our case, we are not looking for the providers' individual rationnality. Indeed, the providers may be abstracted to self-interested entities which bid in order to obtain the requests they want. However, because every request must be treated, some providers will have to treat it *even if this is not in their immediate interest*. Thus the mechanism we are looking for should make the self-interested providers stay in the system because they are satisfied with it *most of the time*. And because they are satisfied most of the time, it is possible to sometimes impose a request to them. Despite some impositions, with medium or long term, they have an interest to stay in the system. In other words, we are looking for a special kind of rationnality, one which is made for providers which are not myopic nor amnesic. In some way, the definition of this property should consider the providers past events and future hopes.

Second, additional testing is required to verify the generalizability of the approach to non-depletable resources such as information services. Once more, additional meaningful simulations may require the definition of new measures that would estimate the requesters and providers satisfaction in the system. Here too, the problem is to characterize the long run behaviour of the process. Hence, we should define a kind of satisfaction average over several mediations.

Third, we should confront the mediation with a practical application, in which we can specify the obtention of the quality and the providers' strategies. We also plan to extend the mediation system architecture to several mediators, with auto-specialization according to requesters' feed-backs, thus forming communities of providers and requesters sharing the same interests.

Bibliography

1. P. Antoniadis, C. Courcoubetis, and R. Mason. Comparing economic incentives in peer-to-peer networks. *Computer Networks*, 46:133–146, 2004.
2. S. N. Chihiro Ono and al. Truth-based facilitator: Handling word-of-mouth trust for agent-based e-commerce. *Electronic Commerce Research*, 3, 2003.
3. R. Dash, S. Ramchurn, and N. R. Jennings. Trust-based mechanism design. In *AAMAS-2004 – Proceedings of the Third International Joint Conference on Au-*

tonomous Agents and Multi Agent Systems, 2004.
4. R. K. Dash, N. R. Jennings, and D. C. Parkes. Computational mechanism design : a call to arms. *IEEE Intelligent Systems*, 3, 2003.
5. E. David, R. Azoulay-Schwartz, and S. Kraus. Protocols and strategies for automated multi-attribute auctions. In *First International Conference on Autonomous Agents and Multiagent Systems (AAMAS'02)*. ACM Press, 2002.
6. K. Decker, K. Sycara, and M. Williamson. Middle-agents for the internet. In *Fifteenth International Joint Conference on Artificial Intelligence (IJCAI'97)*. Morgan Kaufmann, 1997.
7. E. Durfee, T. Mullen, S. Park, J. Vidal, and P. Weistein. *Intelligent Information Agents*, chapter Strategic Reasoning and Adaptation in an Information Economy. Springer, 1999.
8. D. F. Ferguson, C. Nikolaou, J. Sairamesh, and Y. Yemini. *Market-Based Control : a Paradigm for Distributed Resource Allocation*, chapter Economic Models for Allocation REsources in Computer Systems. World Scientific Publishing, 1996.
9. A. Gorobets and B. Nooteboom. Agent based computational model of trust. Technical report, Erasmus Research Institute of Management (ERIM), RSM Erasmus University, Jan. 2005.
10. D. Kossmann. The state of the art in distributed query processing. *ACM Computing Surveys*, 32(4):422–469, 2000.
11. D. Kuokka and L. Harada. Matchmaking for information agents. In *Fourteenth International Joint Conference on Artificial Intelligence (IJCAI'95)*. Morgan Kaufmann, 1995.
12. P. Lamarre and S. Cazalens. Médiation équitable dans un environnement ouvert d'agents compétitifs. In *Modèles Formels de l'Interaction*, 2003.
13. P. Lamarre and S. Cazalens. A procedure for mediating between service requesters and providers. In *Proceedings of the International Conference on Intelligent Agents Technology (IAT 2003)*. IEEE press, 2003.
14. P. Lamarre, S. Cazalens, S. Lemp, and P. Valduriez. A flexible mediation process for large distributed information systems. In Z. T. Robert Meersman, editor, *On the Move to Meaningful Internet Systems 2004: CoopIs, DOA, ODBASE*, volume 1 of *LNCS - LNCS3290*. Springer, 2004.
15. A. Likhodedov and T. Sandholm. Auction mechanism for optimally trading off revenue and efficiency. In *EC '03: Proceedings of the 4th ACM conference on Electronic commerce*, pages 212–213, New York, NY, USA, 2003. ACM Press.
16. S. Marti. *Trust and reputation in Peer-to-Peer Networks*. PhD thesis, Stanford University, 2005.
17. S. Marti and H. Garcia-Molina. Quantifying agent strategies under reputation. In *Proceedings of the 5th IEEE International Conference on Peer-to-Peer Computing*, 2005.
18. A. Mas-Colell, M. D. Whinston, and J. R. Green, editors. *Microeconomic Theory*. Oxford University Press Press, 1995.
19. L. Mui, A. Halberstadt, and M. Mohtashemi. Notions of reputation in multi-agents systems : A review. In *First International Conference on Autonomous Agents and Multiagent Systems (AAMAS'02)*. ACM Press, 2002.
20. M. Nodine, W. Bohrer, and A. H. H. Ngu. Semantic brokering over dynamic heterogeneous data sources in infosleuth. In *International Conference on Data Engineering (ICDE)*, 1999.
21. R. Porter, A. Ronen, Y. Shoham, and M. Tennenholtz. Mechanism design with execution uncertainty. In *Proceedings of the 18th conference on Uncertainty in Artificial*

Intelligence (UAI-02), 2002.
22. R. Porter, Y. Shoham, and M. Tennenholtz. Fair imposition. *Journal of Economic Theory*, 118(2):209–228, 2004.
23. T. W. Sandholm. *Multiagent Systems, a modern approach to Distributed Artificial Intelligence*, chapter Distributed Rational Decision Making. The MIT Press, 2001.
24. Y. Shoham and M. Tennenholtz. Fair imposition. In *Seventeenth International Joint Conference on Artificial Intelligence (IJCAI'01)*. Morgan Kaufmann, 2001.
25. R. Smith. The contract net protocol: high level communication and control in a distributed problem solver. *IEEE Transactions on Computers*, C29(12):1104–1113, 1980.
26. M. Stonebraker, P. M. Aoki, R. Devine, W. Litwin, and M. A. Olson. Mariposa: a new architecture for distributed data. In *IEEE Int. Conf. on Data Engineering*, 1994.
27. K. Sycara, M. Klusch, and S. Widoff. Dynamic service machmaking among agents in open information environments. *ACM SIGMOD Record, Special Issue on Semantic Interoperability in Global Information Systems*, 28(1):47–53, 1999.
28. A. Tomasic, L. Raschid, and P. Valduriez. Scaling access to heterogeneous data sources with disco. *IEEE Trans. on Knowledge and Data Engineering*, 10(5), 1998.
29. N. Vulkan and N. R. Jennings. Efficient mechanisms for the supply of services in multi-agent environments. *Decision Support Systems*, 28:5–19, 2000.
30. G. Wiederhold. Mediators in the architecture of future information systems. *IEEE Computer*, 25(3), 1992.
31. E. Wolfstetter. Auctions : an introduction. *Journal of Economic Surveys*, 10(4):367–420, 1996.
32. H. C. Wong and K. Sycara. A taxonomy of middle-agents for the internet. In *Fourth International Conference on MultiAgent Systems (ICMAS 2000)*, pages 465–466, July 2000.
33. B. Yang, S. Kamvar, and H. Garcia-Molina. Addressing thte non-cooperation problem in competitive p2p systems. In *Proceedings of the 1rst Workshop on Peer-toPeer and Economics*, 2003.
34. Z. Zhang and C. Zhang. An improvement to matchmaking algorithms for middle agents. In *First International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'02)*. ACM Press, 2002.
35. T. Özsu and P. Valduriez. *Principles of Distributed Database Systems*. Prentice Hall, 2nd edition, 1999.
36. T. Özsu and P. Valduriez. *Handbook of Computer Science and Engineering*, chapter Distributed and Parallel Database Systems. CRC Press, 2nd edition, 2004.

Annexe 3 - VLDB Journal 2009

[QRLV09b] Jorge-Arnulfo Quiané-Ruiz, Philippe Lamarre and Patrick Valduriez, *A self-adaptable query allocation framework for distributed information systems*, VLDB Journal, 18(3):649-674, 2009.

manuscript No.
(will be inserted by the editor)

A Self-Adaptable Query Allocation Framework for Distributed Information Systems

Jorge-Arnulfo Quiané-Ruiz · Philippe Lamarre · Patrick Valduriez

Abstract In large-scale distributed information systems, where participants are autonomous and have special interests for some queries, query allocation is a challenge. Much work in this context has focused on distributing queries among providers in a way that maximizes overall performance (typically throughput and response time). However, preserving the participants' interests is also important. In this paper, we make the following contributions. First, we provide a model to define the participants' perception of the system regarding their interests and propose measures to evaluate the quality of query allocation methods. Then, we propose a framework for query allocation called *Satisfaction-based Query Load Balancing* (*SQLB*, for short), which dynamically trades consumers' interests for providers' interests based on their *satisfaction*. Finally, we compare *SQLB*, through experimentation, with two important baseline query allocation methods, namely *Capacity based* and *Mariposa-like*. The results demonstrate that *SQLB* yields high efficiency while satisfying the participants' interests and significantly outperforms the baseline methods.

Keywords distributed information systems, query allocation, query load balancing, satisfaction

Work partially funded by ARA "Massive Data" of the French ministry of research (Respire project) and the European Strep Grid4All project.

J.-A. Quiané-Ruiz · P. Lamarre · P. Valduriez
Atlas group, INRIA and LINA – Université de Nantes
2 rue de la Houssinière
44322 Nantes, France

J.-A. Quiané-Ruiz
E-mail: Jorge.Quiane@univ-nantes.fr
P. Lamarre
E-mail: Philippe.Lamarre@univ-nantes.fr
P. Valduriez
E-mail: Patrick.Valduriez@inria.fr

1 Introduction

We consider distributed information systems with a mediator that allows consumers to access information providers through queries [23,36]. Consumers and providers (which we refer to participants) are autonomous in the sense that they are free to leave the mediator at any time and do not depend on anyone to do so. In the context of a single mediator, leaving the mediator is equivalent to depart from the system, but it could be that, in a multi-mediator system, a participant registers to another competing mediator.

Providers can be heterogeneous in terms of capacity and data. Heterogeneous capacity means that some providers are more powerful than others and can treat more queries per time unit. Data heterogeneity means that providers provide different data and thus produce different results for the same query. Providers declare their *capabilities* for performing queries to the mediator. Then, the main function of the mediator is to allocate each incoming query to the providers that can satisfy it. Much work in this context has focused on distributing the query load among the providers in a way that maximizes overall performance (typically throughput and response time), i.e. *query load balancing* (*qlb*) [6,12,24, 35,40]. Nevertheless, participants usually have certain expectations with respect to the mediator, which are not only performance-related (see Example 1). Such expectations mainly reflect their *preferences* to allocate and perform queries. Consumers' preferences may represent e.g. their interests towards providers (based on reputation for example), preferred providers, or quality of service. Providers' preferences may represent, for example, their topics of interests, relationships with other participants, or strategies.

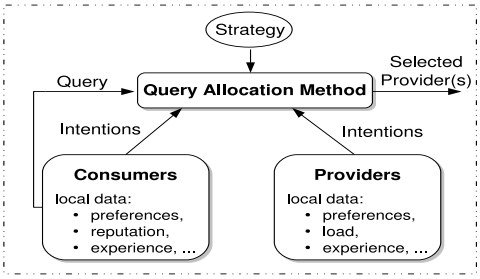


Fig. 1 Overview of Query Allocation.

Example 1 Consider a provider that represents a courier company. During the promotion of its new international shipping service, the provider is more interested in treating queries related to international shipments rather than national ones. Once the advertising campaign is over, its preferences may change. Similarly, consumers expect the system to provide them with information that best fits their preferences.

In this context, because of participants' autonomy, *dissatisfaction* is a problem since it may lead participants to leave the mediator. Thus, it is important to have a query allocation strategy that balances queries such that participants are satisfied. Participant's *satisfaction* means that the query allocation method meets its expectations. To make this possible, the participants' *preferences* must be taken into consideration when balancing queries. However, preferences are usually considered as private data by participants (e.g. in an e-commerce scenario, enterprises do not reveal their business strategies). In addition, preferences are static data, i.e. long-run, while the desire of a participant to allocate and perform queries may depend on its context and thus is more dynamic, i.e. short-run. For instance, in Example 1, even if the provider (the courier company) prefers to perform queries related to international shipments during its advertising campaign, it is possible that, at some time, it may not desire to perform such queries because of other local reasons, e.g. by *overload*. Thus, participants are required to express their desire to allocate and perform queries via their *intention*, which may stem e.g. from combining their preferences and other local consideration such as *load* and *reputation* (see Figure 1).

In such distributed information systems, query allocation is a challenge for several reasons.

- There is no definition of satisfaction to reflect how well the system meets the participants' expectations in the long-run. Economic approaches consider *utility* and *individual rationality*, but utility is not normalized and is commonly related to economic notions (e.g. money), and individual rationality does not capture long-run aspects.
- Participants' expectations may be contradictory among them as well as with respect to the system performance.
- The query allocation process should be adaptable to applications and self-adaptable to changes in the participants' expectations because such expectations usually change in the course of time.
- Unlike several economic models [9,10,42], queries must be always treated whenever possible (if there exits at least one provider to perform it) even if providers do not desire to deal with them. This is because consumers that do not get results may become dissatisfied and thus simply leave the system, which may hurt providers as well.
- Participants' departures may have consequences on the functionalities provided by the system. The providers' departure may mean the loss of important system capabilities and the consumers' departure is a loss of queries for providers.

To our knowledge, this problem has not been addressed completely before. Thus, our first objective is to propose a model that provides a satisfaction notion to reflect how well the mediator meets the participants' expectations in the long-run. Then, our second objective is to propose a query allocation framework that considers both satisfaction and intentions of participants.

1.1 Motivating Example

Consider a public e-marketplace where thousands of companies can share information and do business (such as ebay-business [1] and freightquote [3]). Here, business is understood in a very general sense, not necessarily involving money. Each site, which represents a company, preserves its preferences to allocate and perform queries. To scale up and be attractive over time, an e-marketplace should *(i)* protect, in the long-run, the participants' intentions for doing business, *(ii)* allow consumers to quickly obtain results, and *(iii)* allocate queries so that providers should have the same possibilities for doing business, i.e. to avoid *starvation* [11].

Consider a simple scenario where a company (*eWine*), which desires to ship wine from France to USA, requests the mediator for companies providing international shipping services, such as freightquote [3]. Here, a query is a call for proposals that providers have to answer in order to provide their services. Suppose that *eWine*, to make its final choice, desires to receive proposals from the two best providers that meet its intentions. Similarly, providers desire to participate only

Table 1 Providers for *eWine*'s query.

Providers	Prov.'s Int.	Cons.'s Int.	Avail. Cap.
p_1	Yes	No	0.85
p_2	No	Yes	0.57
p_3	Yes	No	0.22
p_4	No	Yes	0.15
p_5	Yes	Yes	0

in those negotiations that involve queries meeting their intentions. In this scenario, the mediator must perform several tasks.

First, it needs to identify the sites that are able to deal with *eWine*'s query, i.e. to find the relevant providers. There is a large body of work on matchmaking, see e.g. [17,20], so we do not focus on this problem in this paper.

Second, the mediator should obtain *eWine*'s intentions to deal with such providers and the providers' intention to deal with *eWine*'s query[1]. This can be done following the architecture proposed in [18]. Assume that the resulting list contains, for simplicity, only 5 providers: $p_1, \ldots, p_5$. Table 1 shows these providers with their intention to perform the query and *eWine*'s intention to deal with each of them. To better illustrate the query allocation problem in these environments, we also show in Table 1 the providers' *available capacity*. However, it is not always possible to know this information since providers may consider it as private.

Suppose, then, that p_5 is *overloaded*, i.e. has no more resources for doing business, and that p_2 and p_4 do not intend to deal with *eWine*'s query (notice that this does not means they can refuse it) because e.g. p_2 is more interested in its new shipping service to the Asian continent (such as in Example 1) and p_3 has bad experience with *eWine*. Also, assume that *eWine* does not intend to deal with p_1 nor p_3 since it does not trust them e.g. because of their reputation.

Finally, the mediator needs to select the two most available providers, such that *eWine*'s and providers' intentions be respected. To the best of our knowledge, no existing e-marketplace is able to do so. In fact, current *qlb* methods, whose aim is to select the most available providers, also fail in such scenarios since neither p_2 intends to deal with the query nor p_1 is of *eWine*'s interest. Thus, allocating the query to these providers dissatisfies p_2 and *eWine* in such a query allocation. And, whether this occurs several times may cause their departure from the system. The only satisfactory option, regarding the participants' intention, is p_5. But, allocating the query to it may considerably hurt response time, which dissatisfies *eWine* with a poor response time and p_5 by overloading it. Again, whether this occurs several times may cause their departure from the system. Furthermore, *eWine* desires to receive two different proposals.

So, *what should the mediator do in the above scenario? Should it consider the consumer's intention? the providers' intention? or the providers' available capacity?* In this paper, we address this question so that a query allocation method can decide on the fly what to do according to the participants' status.

[1] For simplicity, we assume in this example that the intentions values are binary.

1.2 Contributions and Organization

The rest of this paper is organized as follows. After defining the problem in Section 2, we present the main contributions of this paper:

- We propose a new model to characterize the participants' expectations in the long-run, which allows evaluating a system from a satisfaction point of view. This model facilitates the design and evaluation of *qlb* methods when confronted to autonomous participants (Section 3).
- We define the properties that allow evaluating the quality of *qlb* methods and propose measures to do so (Section 4).
- We propose *Satisfaction-based Query Load Balancing* (*SQLB*, in short), a flexible framework with *self-adapting* algorithms to allocate queries while considering both *qlb* and participants' intentions. *SQLB* affords consumers the flexibility to trade their preferences for the providers' reputation and providers the flexibility to trade their preferences for their utilization. It also allows the mediator to trade consumers' intentions for providers' intentions. Furthermore, *SQLB* affords the mediator the flexibility to adapt the query allocation process to the application by varying several parameters (Section 5).
- We demonstrate, through experimental validation, that *SQLB* significantly outperforms baseline methods, the *Capacity based* and *Mariposa-like* methods, and yields significant performance benefits. We demonstrate the self-adaptability of *SQLB* to participants' expectations and its adaptability to different kinds of application. We also show that applying the proposed measures over the provided model allows the prediction of possible departures of participants (Section 6).

Then, we survey related work in Section 7 and conclude the paper in Section 8.

This paper is an extended version of [32] with the following added value. We present new global character-

istics that allow evaluating the query allocation method in a more objective way (Section 3.3) and discuss in Section 3.4 the two possible levels of satisfaction that a participant can have. We also define in Section 5.3.2 a strategy that allows the mediator to adapt the query allocation process to applications independently of the way in which participants compute their intentions. Furthermore, we analyze the *SQLB* communication cost in Section 5.3.4. Finally, we run new experimentations to demonstrate the adaptability of *SQLB* to the participants' expectations (Section 6.3.3) as well as to validate the proposed strategy (Section 6.3.4).

2 Problem Definition

We consider a system consisting of a mediator m, of a set of consumers C, and of a set of providers P. These sets are not necessary disjoint, an entity may play more than one role. Queries are formulated in a format abstracted as a triple $q = < c, d, n >$ such that $q.c \in C$ is the identifier of the consumer that has issued the query, $q.d$ is the description of the task to be done, and $q.n \in \mathbb{N}^*$ is the number of providers to which the consumer wishes to allocate its query. Parameter $q.d$ is intended to be used within a matchmaking procedure to find the set of providers that are able to treat q, denoted by set P_q. As noted earlier, such techniques are out of the scope of this paper and thus we assume there exists one in the system, e.g. [17,20], that is sound and complete: it does not return false positive nor false negatives. We use N_q for denoting $||P_q||$, or simply N when there is no ambiguity on q.

Consumers send their queries to mediator m that allocates each incoming query q to $q.n$ providers in P_q. We only consider the arrival of *feasible queries*, that is those queries in which there exists at least one provider, which is able to perform them, in the system. For the sake of simplicity we only use, throughout this paper, the term "query" to denote a feasible query. Query allocation of some query q among the providers in P_q is a vector $\overrightarrow{All\overline{oc}}$ of length N, or $\overrightarrow{All\overline{oc}}_q$ and N_q if there is an ambiguity on q, such that,

$$\forall p \in P_q,\ \overrightarrow{All\overline{oc}}[p] = \begin{cases} 1 & \text{if } p \text{ gets } q \\ 0 & \text{otherwise} \end{cases}$$

As we assume that queries should be treated if possible, this leads to $\sum_{p \in P_q} \overrightarrow{All\overline{oc}}[p] = min(q.n, N)$. In the following, the set of providers such that $\overrightarrow{All\overline{oc}}[p] = 1$ is noted $\widehat{P_q}$. Notice that, without any loss of generality, in some cases, e.g. when consumers pay services with real money, query allocation just means that providers are selected for participating in a negotiation process with consumers. Providers have a finite *capacity* to perform queries, denoted by function $cap > 0$. The capacity of a provider denotes the number of computational units that it can have. Thus, the *utilization* of a provider p at time t, denoted by function $\mathcal{U}_t(p)$, is defined as p's load with regards to its capacity.

A consumer $c \in C$ is free to express its intention $ci_c(q,p)$ for allocating its query q to each provider $p \in P_q$, which are stored in vector $\overrightarrow{CI}_q$. Similarly, a provider $p \in P_q$ is free to express its intention $pi_p(q)$ for performing a query q. Values of participants' intention are in the interval $[-1..1]$. A positive value means that a provider (resp. a consumer) intends to perform (allocate) a query, while a negative value means that a provider (a consumer) does not intend to perform (allocate) a query[2]. A null value, i.e. a 0 value, denotes a participant's indifference. It is up to a participant to compute its own intentions by combining different local and external criteria (e.g. utilization, preferences, response time, reputation, past experience, etc.). The way in which a participant computes its intentions is considered as private information and is not revealed to other participants.

In these environments, where participants are autonomous, it is crucial that a query allocation method considers participants' intentions in order to preserve the total system capacity, i.e. the aggregate capacity of all providers (e.g. in terms of computational or physical resources). To summarize, we can state the query allocation problem as follows.

Problem Statement. Given a mediator dealing with autonomous participants, the problem we address is computing and using participants' intentions to perform query allocation at the mediator such that response time, system capacity, and participants' satisfaction are ensured.

3 Pariticpants' Characterization

We define, in this section, a model that allows comparing query allocation methods having different approaches to regulate the system, such as economic and *qlb* methods. We are interested in two characteristics of participants that show how they perceive the system in which they interact.

The first one is *adequation*. From a general point of view, two kinds of adequation could be considered:

- the system adequation to a participant, e.g. a system where a provider (respectively consumer) can-

[2] It is worth remembering that this does not means it can refuse to perform (resp. allocate) the query.

not find any query (resp. provider) it desires is considered inadequate to such a participant, and
- the participant's adequation to the system, e.g. a provider (respectively consumer) that no consumer wants to deal with (resp. issuing queries that no provider intends to treat) is considered inadequate to the system.

Let us illustrate both adequation notions via an example. Consider the case of the courier company of the Example 1, which is interested in its new international shipping service. A market place may be adequate to such a courier company because many consumers are interested in sending products abroad. But the courier company may be not adequate to the market place because many consumers do not want to deal with it.

Both adequation notions are needed to evaluate whether it is possible for a participant to reach its goals in the system. A participant cannot know what the other participants think about it, except if it has a global knowledge of the system. Therefore, we consider the participant's adequation to the system as a global characteristic.

The second characteristic is *satisfaction*. As for adequation, two kinds of satisfaction could be considered:

- the satisfaction of a participant with what it gets from the system, e.g. a provider (respectively consumer) that receives queries (resp. results from the providers) it does not want is not satisfied, and
- the participant's satisfaction with the job that the query allocation method does, e.g. a provider (respectively consumer) that performs queries (resp. results from the providers) it does not want is not satisfied with the query allocation method whether there exist queries (resp. providers) of its interests that it does not get.

To illustrate both satisfactions, consider again the case of the courier company of the Example 1. This courier company may be dissatisfied, in a market place, because consumers are rarely interested in doing their shipments abroad and thus almost all queries it performs are requests for national shipments. Nevertheless, it is possible that this courier company is satisfied with the query allocation method because the only incoming queries requesting for international shipments are allocated to it.

Both satisfaction notions may have a deep impact on the system, because participants may decide whether to stay or to leave the system based on them. In addition to the two kinds of adequation and satisfaction, we are interested in two other global characteristics: *Allocation Efficiency w.r.t. a Consumer* and *Allocation Efficiency w.r.t. a Provider*. In the following, we define all previous notions with regards to what a participant can observe in Sections 3.1 and 3.2. We then define the previous global notions in Section 3.3, which are only observable by the mediator.

It is worth noting that, because of autonomy, preserving the participants' intentions is quite important so that they have some interest in staying in the system. At first glance, the system should satisfy participants in each interaction with them. However, this is simply not possible in reality, considering that a query is generally not allocated to all relevant providers. Furthermore, it is not because a single query allocation penalizes a participant's intention that it decides to leave the system. A participant generally considers some past queries to measure its happiness in the system and to evaluate if it should leave the system. A way to achieve this is to make a regular assessment over all their past interactions with the system, but participants have a limited memory capacity. Thus, they regularly assess only their k last interactions with the system. This is why we define the characteristics of participants over the k last interactions. Clearly, the k value may be different on each participant depending on its memory capacity. For simplicity, we assume they all use the same value of k.

Let us make two other general remarks. First, the participant's characteristics may evolve with time, but for the sake of simplicity we do not introduce time in our notations. Second, the following presentation can be expressed with respect to participants' intentions (dynamic data) or with respect to their preferences (static data). However, applying the following characterization to intentions and preferences yields to different results, because the intentions of participants consider their context (such as their strategy and utilization) and their preferences do not. While in almost all information systems preferences tend to be private information, intentions tend to be public. Since we only intend to observe the system behavior, we develop the following definitions for intentions.

3.1 Local Consumer Characterization

Our characterization considers only the information that a consumer can obtain from the system. This characterization needs to use the memory of each consumer $c \in C$, which is denoted by set IQ_c^k. Intuitively, the characteristics we present in this section are useful to answer the following questions:

- "How well do the expectations of a consumer correspond to the providers that were able to deal with its last queries?" – *System-Consumer Adequation* –,

– "How far do the providers that have dealt with the last queries of a consumer meet its expectations?" – *Consumer Satisfaction* – , and
– "Does the query allocation method do a good job for a consumer?" – *Consumer Allocation Satisfaction* –.

3.1.1 Adequation

The system adequation to a consumer characterizes the perception that the consumer has from the system. For example, in our motivating example of Section 1.1, *eWine* considers the mediator as interesting (i.e. adequate), in such a query allocation, because it advertises providers that *eWine* considers interesting: p_2, p_4, and p_5. Formally, the system adequation w.r.t. a consumer $c \in C$ and concerning a query q, denoted by $\delta_{sca}(c, q)$, is defined as the average of c's intentions towards set P_q (Equation 1). Its values are in the interval [0..1].

$$\delta_{sca}(c,q) = \frac{1}{N_q} \cdot \sum_{p \in P_q} \left((\overrightarrow{CI}_q[p] + 1) \Big/ 2 \right) \quad (1)$$

We thus define the system adequation to a consumer as the average over the adequation values concerning its k last queries.

Definition 1 System-Consumer Adequation

$$\delta_{sca}(c) = \frac{1}{||IQ_c^k||} \cdot \sum_{q \in IQ_c^k} \delta_{sca}(c,q)$$

Its values are between 0 and 1. The closer the value to 1, the more a consumer considers the system as adequate.

3.1.2 Satisfaction

This notion evaluates whether a mediator is allocating the queries of a consumer to the providers from which it wants to get results. To define the consumer's satisfaction over its k last issued queries, we first define the satisfaction of a consumer concerning the allocation of a given query. The average of intentions expressed by a consumer to the providers that performed its query is an intuitive technique to define such a notion. Nevertheless, a simple average does not take into account the fact that a consumer may desire different results. Let us illustrate this using our motivating example. Assume that the mediator allocates *eWine*'s query only to p_2, to which *eWine* has an intention of 1, but it was requiring two providers. A simple average would not take this into account. This is why the following equation takes this point into account using n instead of $||\widehat{P_q}||$.

$$\delta_s(c,q) = \frac{1}{n} \cdot \sum_{p \in \widehat{P_q}} \left((\overrightarrow{CI}_q[p] + 1) \Big/ 2 \right) \quad (2)$$

where n stands for $q.n$. The $\delta_s(c, q)$ values are in the interval [0..1]. The satisfaction of a consumer is then defined as the average over its obtained satisfactions concerning its k last queries. Its values are between 0 and 1. The closer the satisfaction to 1, the more the consumer is satisfied.

Definition 2 Consumer Satisfaction

$$\delta_s(c) = \frac{1}{||IQ_c^k||} \cdot \sum_{q \in IQ_c^k} \delta_s(c,q)$$

Since this notion of satisfaction does not consider the context, it does not allow to evaluate the efforts made by the query allocation method to satisfy a consumer. Let us illustrate this by means of our motivating example. Assume that *eWine* has an intention of 1, 0.9, and 0.7 for allocating its query to p_2, p_4, and p_5, respectively. Now, suppose that the mediator allocates the query to p_4. Such a query allocation corresponds to *eWine*'s high intentions, so *eWine* is satisfied. However, there is still a provider to which its intention is higher (p_2). The *Consumer Allocation Satisfaction* notion, denoted by $\delta_{as}(c)$, allows to evaluate how well the query allocation method works for a consumer. Its values are in the interval [0..∞].

Definition 3 Consumer Allocation Satisfaction

$$\delta_{as}(c) = \frac{1}{||IQ_c^k||} \cdot \sum_{q \in IQ_c^k} \frac{\delta_s(c,q)}{\delta_{sca}(c,q)}$$

If the obtained value is greater than 1, the consumer can conclude that the query allocation method acts to its favor. However, if the value is smaller than 1, the query allocation method dissatisfies the consumer. Finally, a value equal to 1 means that the query allocation method is neutral.

3.2 Local Provider Characterization

This section is devoted to the possible characterization of a provider according to the information that it can obtain from the system. To this end, a provider $p \in P$ tracks its expressed intentions for performing the k last proposed queries (allocated to it or not) into vector $\overrightarrow{PPI}_p$. We denote the k last proposed queries to p by set PQ_p^k. Intuitively, this characterization is useful to answer the following questions:

– "How well do the expectations of a provider correspond to the last queries that the mediator has proposed to it?" – *System-Provider Adequation* – ,
– "How well do the last queries that a provider has treated meet its expectations?" – *Provider Satisfaction* – , and

– "Does the query allocation method do a good job for a provider?" – *Provider Allocation Satisfaction* –.

3.2.1 Adequation

The system adequation w.r.t. a provider evaluates if the system corresponds to the expectations of a provider. Considering our motivating example, one can consider the mediator as adequate to p_1, p_3, and p_5, because $eWine$'s query is of their interest. However, it is difficult to conclude by considering only one query. An average over the k last interactions is more informative. Thus, we define the adequation of the system w.r.t. a provider $p \in P$, $\delta_a(p)$, as the average of p's shown intentions towards set PQ_p^k.

Definition 4 System-Provider Adequation

$$\delta_{spa}(p) = \begin{cases} \frac{1}{||PQ_p^k||} \cdot \sum_{q \in PQ_p^k} \left((\overrightarrow{PPI}_p[q] + 1) \big/ 2 \right) \\ 0 \qquad if\ PQ_p^k = \emptyset \end{cases}$$

The values that this adequation can take are in the interval [0..1]. The closer the value is to 1, the greater the adequation of the system to a provider is.

3.2.2 Satisfaction

Conversely to the adequation notion, the satisfaction of a provider only depends on the queries that it performs and is independent of the other queries that have been proposed to it. To illustrate this notion, suppose that in our motivating example, the mediator allocates $eWine$' query to p_2. In such a query allocation, p_2 is not satisfied since it did not intend to perform the query. Nonetheless, considering a query allocation alone is not very meaningful for a provider. What is more important for a provider is to be globally satisfied with the queries it performs. Thus, we formally define the satisfaction of a provider $p \in P$ in Definition 5. Set SQ_p^k ($SQ_p^k \subseteq PQ_p^k$) denotes the set of queries that provider p performed among the set of proposed queries (PQ_p^k). The $\delta_s(p)$ values are between 0 and 1. The closer the value to 1, the greater the satisfaction of a provider.

Definition 5 Provider Satisfaction

$$\delta_s(p) = \begin{cases} \frac{1}{||SQ_p^k||} \cdot \sum_{q \in SQ_p^k} \left((\overrightarrow{PPI}_p[q] + 1) \big/ 2 \right) \\ 0 \qquad if\ SQ_p^k = \emptyset \end{cases}$$

The satisfaction notion evaluates whether the system is giving queries to a provider according to its (those of the provider) expectations so that it fulfills its objectives. So, as for consumers, a provider is simply not satisfied when it does not get what it expects. Here again, there are different reasons for this. First, it may be because the system does not have interesting resources, i.e. the system has a low adequation w.r.t. the provider. Second, the query allocation method may go against the provider's intention. The latter is measured by the *allocation satisfaction* notion. In other words, by means of this notion a provider can evaluate how well the query allocation method works for it. Conversely to a consumer that always receives results at each interaction, a provider is not allocated all the proposed queries. So the formal definition is a little different. We formally define the allocation satisfaction notion of a provider $p \in P$, denoted by $\delta_{as}(p)$, as the ratio of its Satisfaction to its *system-provider adequation*. Its values are between 0 and ∞.

Definition 6 Provider Allocation Satisfaction

$$\delta_{as}(p) = \frac{\delta_s(p)}{\delta_{spa}(p)}$$

If the allocation satisfaction of a provider p is greater than 1, the query allocation method works well for p (from the point of view of p). If the value is smaller than 1, the closer it is to zero, the more p is dissatisfied with the query allocation method. Finally, a value equal to 1 means the query allocation method is neutral.

3.3 Global Characterization

Conversely to Sections 3.1 and 3.2 that evaluate the query allocation method regarding what a participant perceives from the system, this section allows evaluating the query allocation method from a general point of view. For example, it is possible that a consumer (respectively a provider) is not satisfied with the job the query allocation is doing because providers (resp. consumers) generally do not want to deal with its queries (resp. do not want to get results from it). This global characterization considers this point. The goal of these characteristics is to answer the following questions:

– "How well do the last queries of a consumer correspond to the expectations of the providers that were able to deal with?" – *Consumer-System Adequation* – ,
– "How well does a provider correspond to the consumer's expectations?" – *Provider-System Adequation* – , and
– "How well does the query allocation method perform w.r.t. a consumer or a provider?" – *Allocation Efficiency w.r.t. a Consumer* – and – *Allocation Efficiency w.r.t. a Provider* – , respectively.

The consumer's adequation to the system evaluates how much providers are interested in the queries of a consumer. Going back to our motivating example, we can say that *eWine* is adequate to the system since great part of providers desire to treat its query. According to this intuition, the adequation of a consumer c to the system concerning its interaction with the system for allocating its query q, noted $\delta_{csa}(c,q)$, is defined as the average of the intentions shown by set P_q towards its query q (Equation 3). Its values are between 0 and 1. Vector $\overrightarrow{PI}_q$ denotes the P_q's intentions to perform q.

$$\delta_{csa}(c,q) = \frac{1}{||P_q||} \cdot \sum_{p \in P_q} \Big((\overrightarrow{PI}_q[p] + 1) \big/ 2 \Big) \quad (3)$$

Thus, we define the consumer's adequation to the system as the average over the δ_{csa} values obtained in its k last queries. Its values are between 0 and 1. The closer the value to 1, the greater the adequation of a consumer to the system.

Definition 7 Consumer-System Adequation

$$\delta_{csa}(c) = \frac{1}{||IQ_c^k||} \cdot \sum_{q \in IQ_c^k} \delta_{csa}(c,q)$$

Having formally defined the *consumer-system adequation* global notion, the *query allocation efficiency w.r.t. a consumer* $c \in C$, $\delta_{ae}(c)$, is then defined as in Definition 8. Its values are between 0 and ∞.

Definition 8 Allocation Efficiency w.r.t. a Consumer

$$\delta_{ae}(c) = \frac{1}{||IQ_c^k||} \cdot \sum_{q \in IQ_c^k} \frac{\delta_s(c,q)}{\delta_{sca}(c,q) \cdot \delta_{csa}(c,q)}$$

On the one hand, as for the allocation satisfaction notion, the query allocation efficiency w.r.t. a consumer allows to evaluate the job done by the query allocation method for a consumer. This evaluation is objective since it considers the consumer's adequation to the system in addition to the system's adequation to the consumer. On the other hand, the *query allocation efficiency w.r.t. a provider* objectively evaluates (since it also considers the provider's adequation to the system) how much the query allocation method strives to give interesting queries to providers. To define this latter global notion, as for a consumer, we need to know how much a provider is adequate to the system.

The adequation of a provider to the system allows to evaluate if consumers are interested in interacting with it. To illustrate the *Provider-System Adequation*, we use again our motivating example. One may consider p_1 and p_3 as inadequate to the system (with regards to what they can perceive) since *eWine* does not want to deal with. Nevertheless, the most important is to evaluate that interaction over set PQ_p^k of queries. So, we formally define the adequation of a provider $p \in P$ to the system over the last k proposed queries in Definition 9. Its values are in the interval [0..1]. The closer the value to 1, the greater the adequation of a provider to the system.

Definition 9 Provider-System Adequation

$$\delta_{psa}(p) = \begin{cases} \frac{1}{||PQ_p^k||} \cdot \sum_{q \in PQ_p^k} \Big((\overrightarrow{CI}_q[p] + 1) \big/ 2 \Big) \\ 0 \qquad\qquad if\ PQ_p^k = \emptyset \end{cases}$$

We then define the efficiency of the query allocation w.r.t. a provider $p \in P$, denoted by the function $\delta_{ae}(p)$, as the ratio of its satisfaction to the product of its system-provider adequation by its provider-system adequation. Its values are in [0..∞].

Definition 10 Allocation Efficiency w.r.t. a Provider

$$\delta_{ae}(p) = \frac{\delta_s(p)}{\delta_{spa}(p) \cdot \delta_{psa}(p)}$$

If the efficiency value of the query allocation with regards to a participant (consumer or provider) is greater than 1, the query allocation method does a good job for the participant (considering its adequation to the system). If the value is smaller than 1, the efficiency of the query allocation is not good. In the case the value is 1, the query allocation method is neutral to a participant.

3.4 Discussion

The proposed model can be applied with different purposes. First, to evaluate how well a query allocation method satisfies the participants' expectations. Second, to try to explain the reasons of the participants' departures from the system. For example, to know if they are leaving the system because *(i)* they are dissatisfied with the queries they perform, *(ii)* they are dissatisfied with the mediator's job, or *(iii)* the system is inadequate to them. To do so, one has to apply measures, which reflect a global behavior, over all concepts of the model: adequation, satisfaction, and allocation efficiency (see Section 4). Third, to design new self-adaptable query allocation methods that meet the participants' expectations in the long-run (see Section 5).

As noted earlier, even if the model can be applied to the preferences and intentions of participants, the interpretation of results is not the same. Thus, two different levels of satisfaction exist: at the preferences' and intentions' level. On the one hand, the satisfaction at the preferences' level reflects the happiness of a participant

with what it is doing in the system. On the other hand, it is with the satisfaction at the intentions' level that a participant evaluates if the mediator generally gives to it the queries it asks for. Thus, a participant can know if it is properly computing its intentions by evaluating both satisfactions. For instance, a participant can observe that its expressed intentions do not allow it to be satisfied at its preferences' level even if the mediator does a good job for it and then it is satisfied at its intentions' level.

Moreover, notice that several possibilities to compute participants' satisfaction may exist. For example, participants' satisfaction may decrease with the time or consider the number of received queries. However, to explore and explain all the possibilities to compute participants' satisfaction is well beyond the scope of this paper. In fact, this could be the subject of a full paper. We thus report this to future work.

As final remark, reputation does not directly appear, but it is clear that it has a major role to play in the manner that participants work out their intentions. Thus, it is taken into account as much as participants consider it important.

4 System Measures

The measures we use are the same for consumers and providers, and can be used to evaluate the δ_{sca}, δ_{csa}, δ_s, δ_{as}, δ_{ae}, and $\mathcal{U}_t$ values of a participant. Thus, for simplicity, the g function denotes one of these functions and S denotes either a set of consumers or providers, i.e. $S \subseteq C$ or $S \subseteq P$. To better evaluate the quality of a query allocation method for balancing queries, one should reflect:

- the effort that a query allocation method does for either maximizing or minimizing a set S of g values – *efficiency* – ,
- any change in a set S of g values – *sensitivity* – , and
- the distance from the minimal value to the maximal one in a set S of g values – *balance* – .

A well-known measure that reflects the efficiency of a query allocation method is the *mean* μ function. Because participants' characteristics (see Section 3) are additive values and may take zero values, we utilize the arithmetic mean to obtain this representative number (Equation 4).

$$\mu(g,S) = \frac{1}{||S||} \cdot \sum_{s \in S} g(s) \tag{4}$$

However, the mean measure might be severely affected by extreme values. Thus, we must reflect the g values' fluctuations in S, i.e. the sensitivity of a query allocation method. In other words, we evaluate how fair a query allocation method is w.r.t. a set S of g values. An appropriate measure to do so is the *fairness index* f proposed in [14] (defined in Equation 5). Its values are between 0 and 1.

$$f(g,S) = \frac{\left(\sum_{s \in S} g(s)\right)^2}{||S||\left(\sum_{s \in S} g(s)^2\right)} \tag{5}$$

Intuitively, the greater the fairness value of a set S of g values, the fairer the query allocation process with respect to such values. To illustrate the sensitivity property, suppose that there exist two competitive mediators m and m' in our motivating example. Assume, then, that the set of providers registered to m and m' are $P = \{p_1, p_2, p_3\}$ and $P' = \{p'_1, p'_2, p'_3\}$, respectively. Now, consider that the satisfaction of such providers are $\delta_s(p_1) = 0.2$, $\delta_s(p_2) = 1$, $\delta_s(p_3) = 0.6$, $\delta_s(p'_1) = 1$, $\delta_s(p'_2) = 0.7$, and $\delta_s(p'_3) = 0.9$. Reflecting the sensitivity of both mediators w.r.t. satisfaction (0.77 and 0.97 for m and m' respectively), we can observe that companies have almost the same chances of doing business in m', which is not the case in m.

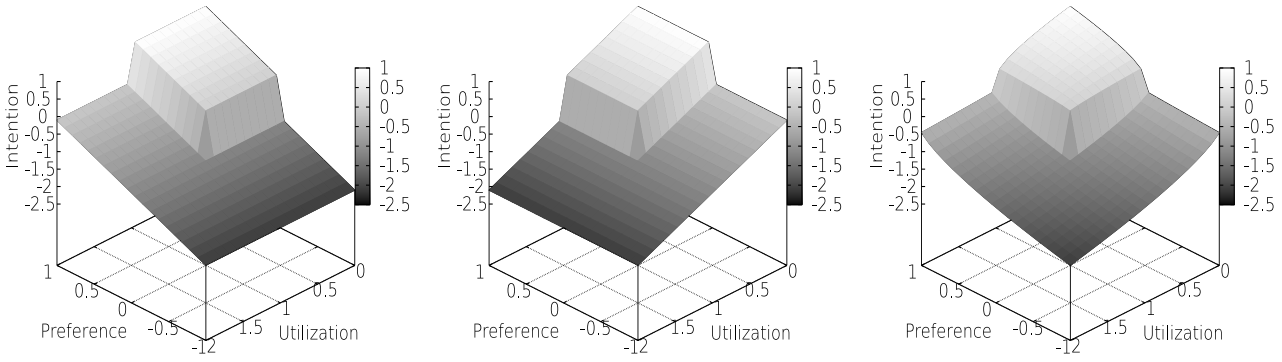
Finally, a traditional measure that reflects the ensured balance by a query allocation method is the *Min-Max* ratio. The Min-Max ratio σ is defined in Equation 6 (where $c_0 > 0$ is some fixed constant). Its values are between 0 and 1. The greater the balance value of a set S of g values, the better the balance of such values. The Min-Max ratio is useful to know whether there exists a great different between the most satisfied entity $s \in S$ and the less satisfied entity $s' \in S$ (with $s \neq s'$), and then, one can evaluate if this is because of the query allocation method or the entity's adequation.

$$\sigma(g,S) = \frac{\min_{s \in S} g(s) + c_0}{\max_{s' \in S} g(s') + c_0} \tag{6}$$

The above three measures are complementary to evaluate the global behavior of the system, and the use of only one of them may cause the loss of some important information.

5 The SQLB Framework

We now present *SQLB*, a flexible framework for balancing queries while considering the participants' intentions. A salient feature of *SQLB* is that it affords consumers the flexibility to trade their preferences for the providers' reputation (Section 5.1) and providers



(a) when provider's *satisfaction* is 0 (b) when provider's *satisfaction* is 1 (c) when provider's *satisfaction* is 0.5

Fig. 2 Tradeoff between *preference* and *utilization* for getting providers' *intention*.

the flexibility to trade their preferences for their utilization (Section 5.2). Then, a mediator allocates queries in accordance to the intentions and satisfaction of participants (Section 5.3). In this way, *SQLB* continuously adapts to changes in participants' expectations and workload. Without any loss of generality, participants may differently obtain their intentions.

5.1 Consumer's Side

When a consumer is required by the mediator to give its intention for allocating its query q to a given provider p, it computes its intention based on its preferences towards p and p's reputation. The idea is that a consumer makes a balance between its preferences for allocating queries and the providers' reputation, in accordance to its past experience with providers. For example, if a consumer does not have any past experience with a provider p, it pays more attention to the reputation of p. A consumer may base its preferences on different criterias, such as quality of service, response times or price of services. Hence, several ways to compute preferences exist. Dealing with the way in which a consumer obtains its preferences is beyond the scope of this paper.

We formally define the intention of a consumer $c \in C$ to allocate its query q to a given provider $p \in P_q$ as in Definition 11. Function $prf_c(q,p)$ gives c's preference (which may denote e.g. some interest to *quality of service* or response time) for allocating q to p, and function $rep(p)$ gives the reputation of p. Values of both functions (prf and rep) are in the interval $[-1..1]$.

Definition 11 Consumer's Intention

$$ci_c(q,p) = \begin{cases} prf_c(q,p)^{\upsilon} \times rep(p)^{1-\upsilon} & if\ prf_c(q,p) > 0 \\ & \wedge\ rep(p) > 0 \\ -\Big((1 - prf_c(q,p) + \epsilon)^{\upsilon} \times (1 - rep(p) + \epsilon)^{1-\upsilon}\Big) & else \end{cases}$$

Parameter $\epsilon > 0$, usually set to 1, prevents the consumer's intention from taking zero values when the consumer's preference or provider's reputation values are equal to 1. Parameter $\upsilon \in [0..1]$ ensures a balance between the consumer's preferences and the providers' reputation. In particular, if $\upsilon = 1$ (resp. 0) the consumer only takes into account its preferences (resp. the provider's reputation) to allocate its query. So, if a consumer has enough experience with a given provider p, it sets $\upsilon > 0.5$, or else it sets $\upsilon < 0.5$. When $\upsilon = 0.5$, it means that a consumer gives the same importance to its preferences and the provider's reputation.

5.2 Provider's Side

The provider's intention to perform a given query is based on its preferences for performing such a query and its current utilization. Nonetheless, the question that arises is: *what is more important for a provider, its preferences or its utilization?* We propose to balance, on the fly, the preferences and utilization of a provider according to its satisfaction. Intuitively, on the one hand, if a provider is satisfied, it can then accept sometimes queries that do not meet its expectations. On the other hand, if a provider is dissatisfied, it does not pay so much attention to its utilization and focuses on its preferences so as to obtain queries that meet its expectations. To do so, the satisfaction it uses to make the balance has to be based on its preferences and not on its intentions. Thus, the satisfaction definition of Section 3.2.2 has to be adapted to the preferences of a provider by using its preferences instead of its intentions. As for a consumer, a provider may compute its preferences either by considering its context or independently of its context. For example, a provider may no longer desire to perform some kind of queries when it is overutilized and another provider may always have the same preferences for queries no matter its utilization. In fact, several strategies can be adopted by a provider to compute its preferences. However, how a provider im-

plements its preference's function, prf, is out of scope of this paper. We just assume that providers' preferences are in the interval $[-1..1]$.

We define the intention of a provider $p \in P_q$ to deal with a given query q as in Definition 12. Parameter $\epsilon > 0$, usually set to 1, prevents the intention of a provider from taking 0 values when its preference is equal to 1 whatever its utilization is.

Definition 12 Provider's Intention

$$pi_p(q) = \begin{cases} prf_p(q)^{1-\delta_s(p)} \times (1 - \mathcal{U}_t(p))^{\delta_s(p)}, & if\ prf_p(q) > 0 \\ & \wedge \mathcal{U}_t(p) < 1 \\ -\Big((1 - prf_p(q) + \epsilon)^{1-\delta_s(p)} \times (\mathcal{U}_t(p) + \epsilon)^{\delta_s(p)}\Big) & else \end{cases}$$

Figure 2 illustrates the behavior that function pi takes for different provider's satisfaction values. We can observe in Figure 2(a) that when a provider is not satisfied at all, its utilization has no importance for it and its preferences denote its intentions. In contrast, when a provider is completely satisfied, its utilization denotes its intentions (see Figure 2(b)). In the case that a provider has a satisfaction of 0.5 (Figure 2(c)), we observe that its preferences and utilization have the same importance for it. Moreover, we can observe in Figure 2 that a provider shows positive intentions, whatever its satisfaction is, only when it is not overutilized and queries are of its interests. This helps satisfying providers while keeping good response times.

5.3 Mediator's Side

So far, we assumed that a matchmaking technique has found the set of providers that are able to deal with a query q, denoted by set P_q. Therefore, we only focus on the allocation of q among set P_q. Given a query q, $SQLB$ allows the mediator to trade consumers' intentions for providers' intentions according to their satisfaction (Section 5.3.1). Furthermore, $SQLB$ affords the mediator the flexibility to regulate the system w.r.t. some predefined function and adapt the query allocation process to the application by varying several parameters (Section 5.3.2). In Section 5.3.3, we describe the query allocation process and analyze, in Section 5.3.4, the number of messages that the mediator transfers over the network to perform q.

5.3.1 Scoring and ranking providers

A natural way to perform query allocation is to allocate queries in a consumer-centric fashion, such as several e-commerce applications do. This leads to take into account the consumers' intentions only, which may seems correct at first glance. However, doing so may severely penalize providers' intentions and hence it may cause their departure from the mediator, which implies a loss of capacity and functionality of the system but also a loss of revenues for the mediator when it is paid by providers after each transaction (e.g. in ebay sellers pay a percent of the transactions they conclude). Respectively, if a mediator only considers the providers' intentions when allocating queries, consumers may quit the mediator by dissatisfaction, which in turn may cause the departure of providers. This is why we decide to balance consumers' and providers' intentions with the aim that both of them be satisfied.

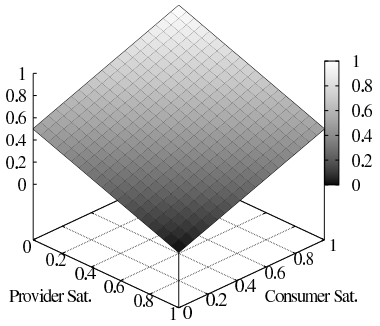


Fig. 3 The values that parameter ω can take.

Thus, given a query q, a provider is scored by considering both its intention for performing q and $q.c$'s intention for allocating q to it. That is, the *score* of a provider $p \in P_q$ regarding a given query q is defined as the balance between the $q.c$'s and p's intentions (see Definition 13).

Definition 13 Provider's Score

$$scr_q(p) = \begin{cases} (\overrightarrow{PI}_q[p])^{\omega} (\overrightarrow{CI}_q[p])^{1-\omega} & if\ \overrightarrow{PI}_q[p] > 0 \wedge \\ & \wedge \overrightarrow{CI}_q[p] > 0 \\ -\Big((1 - \overrightarrow{PI}_q[p] + \epsilon)^{\omega} (1 - \overrightarrow{CI}_q[p] + \epsilon)^{1-\omega}\Big) & else \end{cases}$$

Vector $\overrightarrow{PI}_q[p]$ denotes P_q's intentions to perform q. Parameter $\epsilon > 0$, usually set to 1, prevents the provider's score from taking 0 values when the consumer or provider's intention is equal to 1. Parameter $\omega \in [0..1]$ ensures a balance between the consumer's intention for allocating its query and the provider's intention for performing such a query. In other words, it reflects the importance that the query allocation method gives to the consumer and providers' intentions. To guarantee equity at all levels, such a balance should be done in accordance to the consumer and providers' satisfaction. That is, if the consumer is more satisfied than the provider, then the query allocation method should pay more attention to the provider's intentions. Thus, we compute the ω value as in Equation 7. Conversely to provider's intention, the query allocation module has

not access to private information. Thus, the satisfaction it uses must be based on the intentions.

$$\omega = \Big(\big(\delta_s(c) - \delta_s(p)\big) + 1\Big) \Big/ 2 \quad (7)$$

Figure 3 illustrates the tradeoff between the consumer and provider' intention for obtaining the ω value. One can also set ω's value according to the kind of application. For instance, if providers are cooperative (i.e. not *selfish*) and the most important is to ensure the quality of results, one can set ω near or equal to 0. Finally, providers are ranked from the best to the worst scored, the $\overrightarrow{R}_q$ vector. Intuitively, $\overrightarrow{R}_q[1]$ is the best scored provider to deal with q, $\overrightarrow{R}_q[2]$ the second, and so on up to $\overrightarrow{R}_q[N]$ which is the worst. As a result, if $q.n <= N$ the $q.n$ best ranked providers are selected, or else all the N providers are selected.

5.3.2 Regulating the system

The mediator can proceed to allocate queries by considering only the providers' ranking based on their score ($\overrightarrow{R}$), which affords participants to take the control of the query allocation process. However, the mediator may have certain objectives or goals that it aims to achieve. It is possible that the mediator wants to regulate the system regarding some predefined function τ, e.g. to ensure short response times to consumers. To allow this, we assume that the mediator uses the K_nBest strategy that we proposed in [31]. K_nBest is inspired by the *two random choices (TRC)* paradigm [25,6]. The idea is that, given a query q, the mediator selects a set K_n of k_n providers that either maximize or minimize function τ from set K, where set K is a random selection of k' providers from set P_q of providers[3]. Then, it allocates q to the $q.n$ best ranked providers among set K_n of providers. We explain further the query allocation principle in Section 5.3.3. We assume, without any loss of generality, that function τ denotes function $\mathcal{U}$, which means that the mediator strives to regulate the system with respect to providers' utilization (i.e. to perform *qlb*).

Theorem 1 summarizes the K_nBest's properties that bound its behavior.

Theorem 1 *Given a query q, the behavior of a query allocation method using K_nBest is bounded by the following properties,*

(i) if $k' = 2q.n \wedge k_n = q.n$, K_nBest has a TRC behavior.

(ii) if $k' = N_q \wedge k_n = q.n$, K_nBest has a Capacity based behavior.

(iii) if $k' = N_q \wedge k_n = k'$, K_nBest has an Intention based behavior.

Proof Say a query allocation method qa implements the K_nBest strategy. The following is the same for any value that parameter $q.n$ can take.

Consider that qa sets $k' = 2q.n \wedge k_n = q.n$. In this case, qa allocates a query q to the less utilized provider $p \in P$ among a set of $2q.n$ random selected providers from P_q. This leads to satisfy the below equation,

$$\forall p \in \widehat{P_q}, \nexists p' \in K \backslash \widehat{P_q} : \mathcal{U}_{(p')} < \mathcal{U}_{(p)}$$

which is also ensured by a query allocation method using a TRC process. This proves property *(i)*.

Now, consider that qa sets $k' = N_q \wedge k_n = q.n$. In this case, qa allocates an incoming query q to the less utilized providers in set P_q, which is also the objective of a *Capacity based* method. Thus, both qa and *Capacity based* ensure the following equation,

$$\forall p \in \widehat{P_q}, \nexists p' \in P_q \backslash \widehat{P_q} : \mathcal{U}_{(p')} < \mathcal{U}_{(p)}$$

which proves property *(ii)*.

Finally, consider that qa sets $k' = N_q \wedge k_n = k'$. Doing so, an incoming query q is allocated by qa to a set $\widehat{P_q}$ such that,

$$\forall p \in \widehat{P_q}, \nexists p' \in P_q \backslash \widehat{P_q} : scr_q(p') > scr_q(p)$$

Thus, the only thing that is considered by qa is the participants' intentions and thus it will have an *Intention based* behavior. In other words, the mediator has no control to regulate the system. We call this way to operate the *intention based* approach. This proves property *(iii)*. □

The great advantage of using K_nBest is that it allows the mediator to adapt the query allocation process to the application by varying several parameters. To illustrate this, consider the following examples. First, if providers and incoming queries are homogeneous, the mediator can take a TRC behavior (which has been proved to operate well in homogeneous distributed systems [25]) when allocating queries by setting parameters of K_nBest as in property *(i)*. Second example, consider that providers and incoming queries are heterogeneous and that the most important is to perform *qlb* with no consideration for participants' *intentions*. In this case, the mediator can allocate queries following a *Capacity based* behavior, by setting parameters of

[3] We can indifferently assume that k' and k_n values are predefined by the administrator or defined on the fly by the mediator.

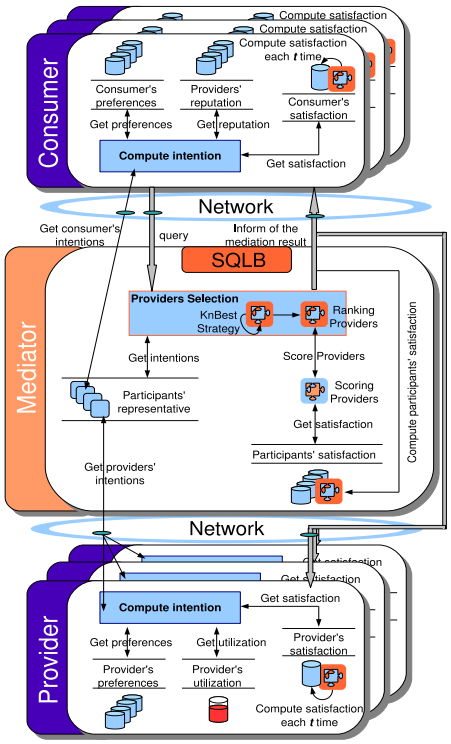


Fig. 4 SQLB system architecture.

$K_n Best$ as in property *(ii)*. Finally, consider that participants are autonomous and there is no other objective in the system than satisfying participants, the mediator can then allocate queries based only on the participants' *intentions* by setting parameters of $K_n Best$ as in property *(iii)*.

In the rest of this paper, as we focus on heterogeneous distributed information systems, we assume, for the sake of simplicity, that k' is always equal to N (i.e. we discard the random selection phase).

5.3.3 Query allocation principle

We now describe how the mediator allocates queries. Figure 4 illustrates the general *SQLB* system architecture and Algorithm 1 shows the main steps of the query allocation process. Given a query q and a set P_q of providers that are able to perform q, the mediator first asks for $q.c$'s intention for allocating q to each provider $p \in P_q$ (line 2 of Algorithm 1). In parallel, it also asks for P_q's utilization (with the assumption that function τ denotes function $\mathcal{U}$) and intention for performing q (lines 3 and 4). Then, it waits for this information from both $q.c$ and set P_q or for a given *timeout* (line 5). Once such vectors $\overrightarrow{CI}_q$, $\overrightarrow{U}$, and $\overrightarrow{PI}_q$ are computed (where $\overrightarrow{U}$ stores the utilization of each provider in P_q), the mediator selects the k_n less utilized providers, denoted by set K_n, from set P_q (line 6). This selection phase can

Algorithm 1: QueryAllocation

```
   Input  : q, k_n, P_q
   Output: All⃗oc_q
1  begin
       // Consumer's intentions
2      fork ask for q.c's intentions;
       // Providers' intention
3      foreach p ∈ P_q do
4          fork ask for p's utilization and intention w.r.t. q ;
5      waituntil CI⃗_q, U⃗, and PI⃗_q be calculated or timeout;
       // qlb regulation
6      K_n ← select k_n less utilized providers from set P_q ;
       // Scoring and ranking providers
7      foreach p ∈ K_n do
8          compute p's score concerning CI⃗_q[p] & PI⃗_q[p] ;
9      rank set K_n of providers regarding scr_p(q), R⃗_q ;
       // Query Allocation
10     for i = 1 to min(n, k_n) do All⃗oc[R⃗_q[i]] ← 1 ;
11     for j = min(n, k_n) + 1 to N do All⃗oc[R⃗_q[j]] ← 0 ;
12 end
```

be solved using a sorting algorithm, so, in the worst case, its complexity is $O(N \log_2(N))$. Next, the mediator computes the score of each provider $p \in K_n$ by making a balance between $q.c$'s and p's intentions (line 7 and 8) and computes the ranking of providers in K_n (line 9), whose complexity is $O(k_n \log_2(k_n))$ in the worst case. Finally, the mediator allocates q to the $q.n$ best scored providers in set K_n and sends the mediation result to all P_q providers (lines 10 and 11). Notice that in the case that $q.n \geq k_n$, the mediator thus allocates q to all k_n providers. Indeed, Algorithm 1 can be optimized, but our goal is to show the steps involved in the query allocation process.

5.3.4 Communication Cost

We analyze communication cost in terms of number of messages that the mediator should transfer over the network to perform a query. The communication cost is given by the following theorem.

Theorem 2 *The total number of transferred messages by the mediator to perform a query is* $3(N+1)+n$.

Proof As we saw in the previous section, given any incoming query q, the mediator transfers $mssg_0 = 2N+2$ messages over the network to ask the consumer's intentions and the utilization and intention of providers in set P_q. Then, it selects the k_n least utilized providers in set P_q and allocates q to the $q.n$ best scored providers in set K_n. After this, the mediator informs all providers in set P_q of the mediation result and waits for results from the $q.n$ selected providers. This implies to exchange $mssg_1 = N + n$ messages among the mediator

and participants, where n stands for $q.n$. Finally, the mediator transfers $mssg_2 = 1$ messages to give results to $q.c$. Thus, the total number of messages transferred over the network by the mediator to perform a query is $mssg_0 + mssg_1 + mssg_2 = 3(N+1) + n$. □

We can further reduce the number of messages by using participants' *representatives* [18] or by introducing again the random selection phase (see Section 5.3.2). However, the problem of reducing communication cost is orthogonal to the problem we address in this paper.

5.4 Discussion

We pointed out in Sections 5.1 and 5.2 that there exist several ways a participant can compute its preferences. To the best of our knowledge, there is no work that proposes a comparison study of these different preference functions and hence it is still an open problem. We believe that such a study may be quite interesting to allow a participant knowing which strategy it can adopt to compute its preferences. Similarly, several manners to compute the consumers' and providers' intentions exist. This is also an open problem that should be explored so as to identify the best ways for a participant to adapt their intentions to their context and application. Improving on these functions is not the focus of our work. Instead, our framework is designed so it can leverage any existing preference and intention function.

Moreover, the score function of a query allocation method is usually based on specific demands, which are given by the application challenges that one wants to solve. Thus, a large number of specific query allocation methods with different behaviors may exist. For example, the score function of a *qlb* method is designed for those applications whose goal is to ensure good system performance. However, when the behavior of a query allocation method is specific to an application, it cannot be applied elsewhere, and worse, it cannot perform in environments where participants change their interests on the fly.

Therefore, we proposed a score function that makes no assumption about either the kind of application nor the way in which a participant obtains its preferences. It just allocates queries based on the participants' intentions. But, we are aware that sometimes a mediator, or even the system administrator, is required to satisfy some constraint, e.g. to ensure a specific *Quality of Service*, no matter what the participants prefer. This is why we also proposed a strategy that allows the query allocation method to regulate the system with regards to a given function. As a result, conversely to specific query allocation methods, *SQLB* is quite general, self-adaptable to the interests of participants, and adaptable to the application. This allows *SQLB* to perform in many kinds of environments and to perform as well as any specific query allocation method by tuning its parameters or if participants desire so.

We assumed a mono-mediator system, i.e. a system that contains only one mediator to allocate queries. Clearly, a mediator may become a single point of failure and a performance bottleneck and thus one may desire to have more than one mediator in the system to allocate queries. In this case, *SQLB* does not scale well because it considers current participants' satisfaction, which a mediator can no longer compute itself as it also depends on the query allocations made by other mediators. Hence, when allocating a query, a mediator should keep informed all other mediators of the mediation result to update participants' satisfaction. This tends to significantly increase the network traffic. A way to avoid such a traffic overhead between mediators is that providers express their interest for queries through "monetary" bids so that mediators no longer consider the providers' satisfaction but only their bids. This requires introducing some "virtual" money to be used by providers and mediators. In this context, we are currently doing some work to show how to adapt *SQLB* to use virtual money and demonstrate that such an economic version can easily scale up to several mediators. However, a further discussion on this subject is not the focus of this paper (see [33,34]).

Finally, in large-scale distributed information systems participants may fail, usually because of network failures. Nevertheless, we do not deal with fault-tolerance issues in this paper since the main focus of this work is to evaluate *SQLB* from a satisfaction point of view. To make *SQLB* fault-tolerant is the focus of one of our forthcoming work. Generally speaking, the idea is to enable a mediator to set, in a predictive way, the number of replicas that should be created for an incoming query based on participants' satisfaction.

6 Experimental Validation

Our experimental validation has three main objectives: *(i)* to evaluate how well query allocation methods operate, *(ii)* to analyze if *SQLB* satisfies participants while ensures good *qlb* because it is not obvious that when adding new criteria a query allocation method still gives good results for the initial criteria, and *(iii)* to study how well our measures capture query allocation methods' operation. To do so, we carry out four kinds of evaluations. First, we evaluate the general query allocation process as well as the computed measures. Second,

we evaluate the impact of participants' *autonomy* on performance. Third, we evaluate the self-adaptability of *SQLB* to participants' expectations. Finally, we analyze the effects of varying the values of k_n parameter, i.e. we evaluate the *SQLB*'s adaptability to different kinds of applications.

6.1 Setup

We built a Java-based simulator and simulate a *mono-mediator* distributed information system, which follows the mediation system architecture presented in [18]. For all the query allocation methods we tested, the following configuration (Table 2) is the same and the only change is the way in which each method allocates the queries to providers. Before defining our experimental setup let us say that the definition of a synthetic workload for environments where participants are autonomous and have special interests towards queries is an open problem. *Pieper et al.* [29] discuss the need of benchmarks for scenario-oriented cases, which are similar to the case we consider, but this remains an open problem. Another possibility to validate our results is to consider real-world data over long periods of time. However, even if we had (we don't) the resources to obtain real-world data, the validation would get biased towards the specific applications. Therefore, in our experiments, we decided to generate a very general workload that can be applied for different applications and environments in order to thoroughly validate our results.

Participants work out their satisfaction, adequation, and allocation satisfaction as presented in Section 3. We initialize them with a satisfaction value of 0.5, which evolves with their last 200 issued queries and 500 queries that have passed through providers. That is, the size of k is 200 for consumers and 500 for providers. The number of consumers and providers is 200 and 400 respectively, with only one mediator allocating all the incoming queries. We assign sufficient resources to the mediator so that it does not cause bottlenecks in the system. We assume that consumers and providers compute their intentions as defined in Sections 5.1 and 5.2, respectively. For simplicity, we set $\upsilon = 1$, i.e. the consumers' preferences denote their intentions.

To simulate high heterogeneity of the consumers' preferences for allocating their queries to providers, we divide the set of providers into three classes according to the interest of consumers: to those that consumers have *high* interest (60% of providers), *medium* interest (30% of providers), and *low* interest (10% of providers). Consumers randomly obtain their preferences between .34 and 1 for *high*-interest providers, between $-.54$ and .34 for *medium*-interest providers, and between -1 and $-.54$ for *low*-interest providers. On the other side, to simulate high heterogeneity of the providers' preferences towards the incoming queries, we also create three classes of providers: those that have *high* adaptation (35% of providers), *medium* adaptation (60% of providers), and *low* adaptation(5% of providers). Here, adaptation stands for the *system-provider adequation* notion we defined in Section 3.2.1. Providers randomly obtain their preferences between $-.2$ and 1 (*high*-adaptation), between $-.6$ and .6 (*medium*-adaptation) or between -1 and .2 (*low*-adaptation). More sophisticated mechanisms for obtaining such preferences can be applied (for example using the *Rush* language [37]), but this is beyond the scope of this paper and orthogonal to the problem we address here. Without any loss of generality, the participants' expectations, in the long run, are static in our simulations. We assume this to evaluate the query allocation methods in a long-run trend, but our model allows expectations to be dynamic.

Table 2 Simulation parameters.

Parameter	Definition	Value
nbConsumers	Number of consumers	200
nbProviders	Number of providers	400
nbMediators	Number of mediators	1
qDistribution	Query arrival distribution	Poisson
iniSatisfaction	Initial satisfaction	0.5
conSatSize	k last issued queries	200
proSatSize	k last treated queries	500
nbRepeat	Repetition of simulations	10

We set the providers' capacity heterogeneity following the results presented in [39]. We generate around 10% of providers with *low*-capacity, 60% with *medium*, and 30% with *high*. The *high*-capacity providers are 3 times more powerful than *medium*-capacity and still 7 times more powerful than *low*-capacity providers. We generate two classes of queries that consume, respectively, 130 and 150 treatment units at the *high*-capacity providers. *High*-capacity providers perform both classes of queries in almost 1.3 and 1.5 seconds, respectively. We consider in our experiments, without any loss of generality, that providers offer computational services to consumers. Thus, inspired from [12], we assume that providers compute their utilization as in Equation 8. Set Q_p denotes the set of queries that have been allocated to p but have not already been treated, i.e. the pending queries at p. Function $cost_p(q)$ represents the computational resources that a query $q \in Q_p$ consumes at provider p.

$$\mathcal{U}_t(p) = \frac{\sum_{q \in Q_p} cost_p(q)}{cap(p)} \quad (8)$$

We do not consider the random selection phase because we consider heterogeneous distributed systems. In other words, we assume in all our experimentations that k' is equal to N. We assume that queries arrive to the system in a *Poisson* distribution, as found in dynamic autonomous environments [22]. Since our main focus is to study the way in which queries are allocated, we do not consider in this paper the bandwidth problem and assume that all participants have the same network capacities. Finally, for the sake of simplicity, we assume that consumers only ask for one informational answer (i.e. $n = 1$) and all the providers in the system are able to perform all the incoming queries.

6.2 Baseline Methods

We briefly justify, in this section, the choice of the algorithms to which we compare *SQLB*.

6.2.1 Capacity based method

In distributed information systems, there are two well-known approaches to balance queries across providers: *Load Based* and *Capacity based* methods. We discard *Load Based* [12,6] methods since, unlike *Capacity based*, they inherently assume that providers and queries are homogeneous. In *Capacity based* [24,35,40] methods, one common approach is to allocate each query q to providers that have the highest available capacity (i.e. the least utilized) among set P_q of providers. *Capacity based* has been shown to be better than *Load Based* in heterogeneous distributed information systems. Thus, we use *Capacity based* in our simulations. Note that *Capacity based* does not take into account the consumers nor providers' intentions.

6.2.2 Economical method

Economical models have been shown to provide efficient query allocation in heterogeneous systems [9,10,42]. Mariposa [42] is one of the most important approaches to allocate queries in autonomous environments. In this approach, all the incoming queries are processed by a *broker* site that requests providers for *bids*. Providers bid for obtaining queries based on a local bulletin board and then the *broker* selects the set of bids that has an aggregate price and delay under a bid curve provided by the consumer. In Mariposa, providers modify their bids with their current load (i.e. bid $\times$ load) in order to ensure *qlb*. Since Mariposa has shown good results, we implemented a *Mariposa-like* method to compare it with *SQLB*. In our *Mariposa-like* implementation we assume that consumers are only interested in the price for getting results. Note that different economical methods may lead to different performance results than those presented here.

6.3 Results

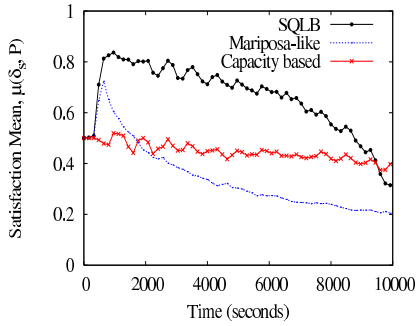
We start, in Section 6.3.1, by evaluating the quality of the three query allocation methods with regards to satisfaction and *qlb*. In Section 6.3.2, we evaluate how well these methods deal with the possible participants' departure by *dissatisfaction*, *starvation*, or *overutilization*. Then, in Section 6.3.3, we show the self-adaptability of *SQLB* to participants' expectations. In these three first sections, we assume that $k_n = k'$, i.e. set K_n denotes set P_q considering that $k' = N$. Finally, in Section 6.3.4, we study the adaptability of *SQLB* to the kind of application by varying parameter k_n.

6.3.1 Quality results without autonomy

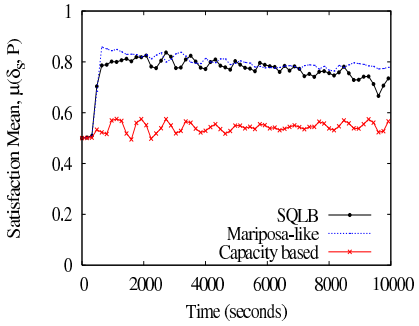
If participants are autonomous, they may leave the system by dissatisfaction, starvation, or overutilization. Nevertheless, the choice of such departure's thresholds is very subjective and may depend on several external factors. Thus, for these first experiments, we consider *captive* participants, i.e. they are not allowed to leave the system. To measure the quality of the three methods, we apply the measures defined in Section 4. We ran a series of experiments where each one starts with a workload of 30% that uniformly increases up to 100% of the total system capacity.

We first analyze the providers results. Figure 5(a) shows the satisfaction mean ensured by the three methods. The satisfaction used in this measurement is based on the providers' intentions, i.e. what the mediator can see. We observe in these results that providers are more satisfied with *SQLB* than with the two others. As the workload increases, providers' satisfaction decreases because their intentions decrease as they are loaded (just because utilization becomes the most important for them). Thus, *SQLB* cannot satisfy the providers' intentions for high workloads since their adequation (based on intentions) is low. *Capacity based* and *Mariposa-like* do not satisfy the providers' intentions from the beginning, simply because they allocate queries based on other criteria, which do not exactly meet intention.

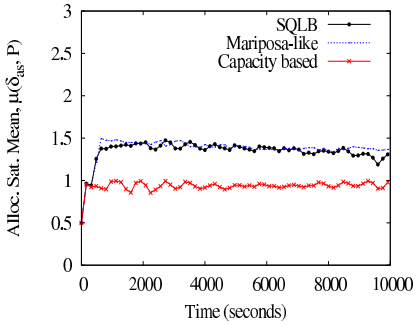
Nonetheless, this does not reflect what providers really feel with respect to their preferences. To show this, we need to measure the mean ensured by the three methods concerning the providers' satisfaction based on their preferences. Although we can measure



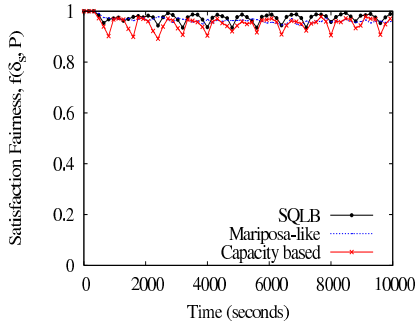
(a) Providers' satisfaction mean based on intentions.



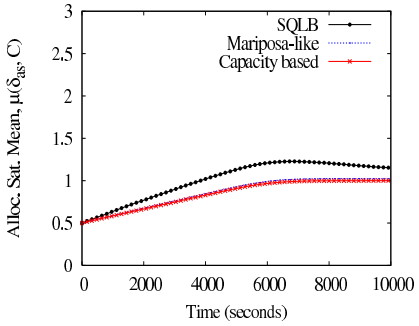
(b) Providers' satisfaction mean based on preferences.



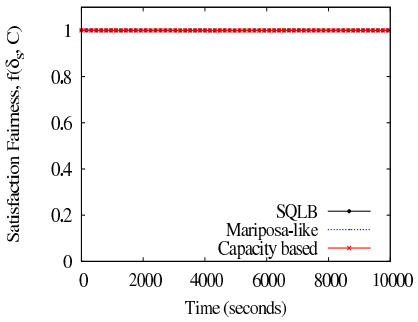
(c) Providers' allocation satisfaction.



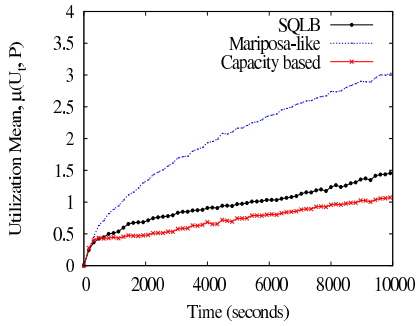
(d) Provider satisfaction fairness.



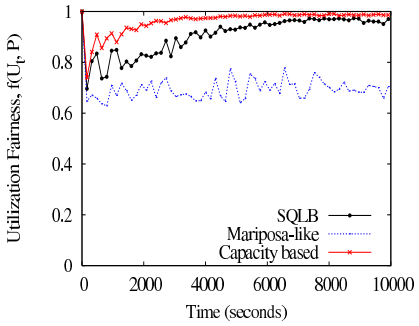
(e) Consumers' allocation satisfaction.



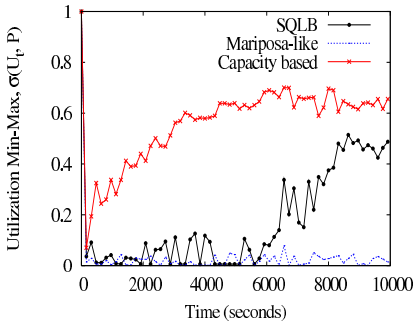
(f) Consumer satisfaction fairness.



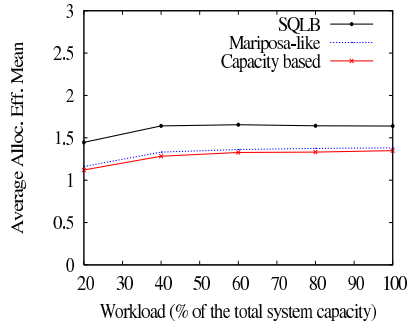
(g) Query load mean.



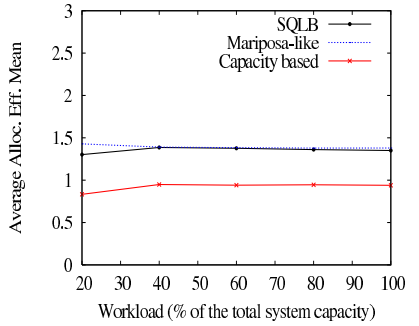
(h) Query load fairness.



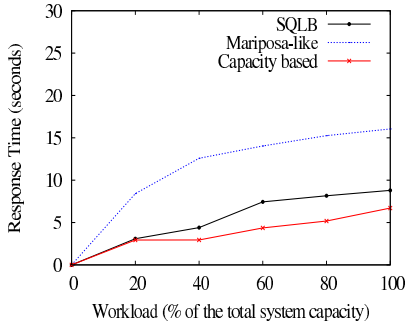
(i) Query load min-max.



(j) Allocation efficiency w.r.t. Consumers.



(k) Allocation efficiency w.r.t. Providers.



(l) Response times.

Fig. 5 Results with *captive* participants. (a)-(i): quality results for a *workload* range from 30 to 100% of the total system capacity, (j)-(k): allocation efficiency results for different workloads, and (l): ensured response times for different workloads.

such a satisfaction in our simulations, this is not always possible since such preferences are usually considered as private. Figure 5(b) shows the results of these measurements. We observe that *SQLB* has the same performance as *Mariposa-like* even if it considers the consumers' intentions. When the workload is close to 100%, the providers' satisfaction slightly decreases with *SQLB*. As noted earlier, this is because providers pay more attention to their utilization for obtaining their intentions, thus their preferences are less considered by the *SQLB* method.

It is worth noting that, as expected, *Capacity based* is the only one among these three methods that penalizes the providers. This is clear in Figure 5(c), which illustrates the mean ensured by these three methods with respect to the providers' allocation satisfaction. We observe that providers are not satisfied with *Capacity based* having, in general, allocation satisfaction values under 1. Then, based on these results, we can predict that when providers will be free to leave the system, *Capacity based* will suffer from serious problems with providers' departures by dissatisfaction reasons. Figure 5(d) illustrates the satisfaction fairness ensured by the three methods. We see that they guarantee almost the same satisfaction fairness. However, as seen in the previous results, this does not mean that providers are satisfied with all three methods.

Now, let us analyze the consumer results. Figure 5(e) illustrates the allocation satisfaction mean concerning the consumers' intentions. We observe that while *SQLB* is the only one to satisfy consumers, the two others are neutral to consumers (mean values equal to 1). These results allows us to predict that *Capacity based* and *Mariposa-like* may suffer from consumer's departures while *SQLB* does not. The *SQLB*'s mean decreases for high workloads because of providers. Remember that providers' satisfaction decrease because they take care of their utilization. So, *SQLB* pays more attention to providers' satisfaction than to consumers' satisfaction. Nonetheless, consumers are never penalized! Conversely to providers, we can observe in Figure 5(f) that consumers' satisfaction fairness has less variations because they are not in direct competition to allocate queries.

Concerning *qlb*, as expected, *Capacity based* better balances the queries among providers than *SQLB* and *Mariposa-like* (see Figure 5(g)). We can observe that *SQLB* performs well, while *Mariposa-like* has serious problems to balance queries. Thus, *Mariposa-like* may lose providers by starvation or overutilization reasons. Figure 5(h) shows that *SQLB* has some difficulties to be fair (w.r.t. *qlb*) for workloads under 40%. In contrast, when the workload increases, *SQLB* pays more attention to *qlb* and becomes fairer. This is clearly illustrated in Figure 5(i), which shows the results about the utilization Min-Max. The reason that *SQLB* performs better for high workloads is that providers become overutilized and thus they take much more care with their utilization, which is not the case for low workloads. These *qlb* results demonstrate the high adaptability of *SQLB* to the variations in the workloads.

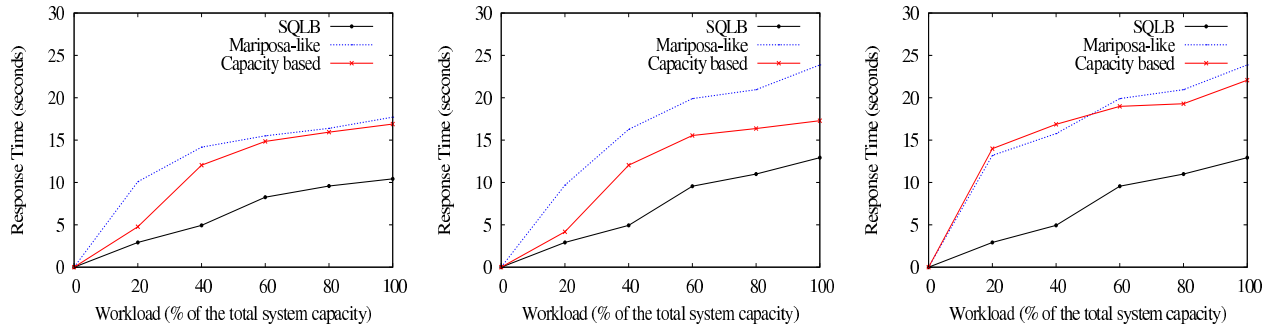
Figures 5(j) and 5(k) illustrate the allocation efficiency with respect to consumers and providers for different workloads. These results clearly illustrate the superiority of *SQLB* over *Capacity based* and *Mariposa-like* since we can observe, *(i)* on the one hand, that *SQLB* significantly outperforms *Capacity based* in both cases; and *(ii)* on the other hand, that *SQLB* and *Mariposa-like* have the same allocation efficiency w.r.t. providers, but *SQLB* significantly outperforms *Mariposa-like* in the consumers' case, which demonstrates the equity at both levels of *SQLB*.

Finally, Figure 5(l) shows the ensured response times in these environments (with captive participants). As is conventional, response time is defined as the elapsed time from the moment that a query q is issued to the moment that $q.c$ receives the response of q. As expected, the *Capacity based* method outperforms the two others. However, even if *SQLB* takes into account the participants' intentions, it only degrades performance by a factor of 1.4 in average while *Mariposa-like* does so by a factor of 3!

All above results show that *Capacity based* may severely suffer from providers' departures by dissatisfaction, while *Mariposa-like* may also suffer from providers' departures by query starvation or overutilization. Furthermore, above results demonstrate the *SQLB*'s self-adaptability to changes in the participants' satisfaction and to the workload. This feature makes our proposal highly suitable for autonomous environments. Furthermore, as concluding remark, we can say that even if not designed for environments where participants are captive, *SQLB* ensures quite good response times and pays attention to the quality of results and queries that consumers and providers get from the system, respectively.

6.3.2 Dealing with autonomy

To validate our measurements and intuitions of Section 6.3.1, we also ran several experimental simulations where participants are given the *autonomy* to leave the system. Our main goal, in this section, is to study the reasons by which providers leave the system and evaluate the impact on performance. We evaluate the ensured response times by the three methods in au-



(a) Providers may leave by *dissatisfaction*. (b) Providers may leave by *dissatisfaction* or *starvation*. (c) Providers may leave by *dissatisfaction*, *starvation*, or *overutilization*.

Fig. 6 Impact on performance of providers' departures.

tonomous environments and compare it with those of the captive environments (see Figure 5(l)).

To do so, we have to set the thresholds under, or over, which a participant decides to leave the system. To avoid any suspicion on the choice of such thresholds, we assume that participants support high degrees of dissatisfaction, starvation, and overutilization. Thus, a consumer leaves the system, by dissatisfaction, if its satisfaction is smaller than its adequation, i.e. the allocation method penalizes it. A provider leaves the system *(i)* by dissatisfaction, if its satisfaction value is 0.15 smaller than its adequation, *(ii)* by starvation, if its utilization is smaller than 20% of its optimal utilization, and *(iii)* by overutilization, if its utilization is greater than 220% of its optimal utilization. With a workload of 80% of the total system capacity, the optimal utilization of a provider is 0.8.

We ran a first series of experiments with different workloads where providers are allowed to leave the system by dissatisfaction only (see Figure 6(a)). We can see that our approach outperforms both *Capacity based* and *Mariposa-like* because it better satisfies providers than *Capacity based*, and better ensures *qlb* in the system than *Mariposa-like*. Recall that in previous section we note that *Mariposa-like* tends to overutilize some providers (those that are the most adapted to the incoming queries). This is why, even if *Mariposa-like* better satisfies providers than *Capacity based* (see Figure 5(b)), it ensures higher response times than *Capacity based*.

A second series of experiments allows providers to leave the system by dissatisfaction or starvation. A provider might quit the system by starvation e.g. when it simply does not obtain the queries that it needs to survive. Figure 6(b) illustrates these results. We observe again that *SQLB* significantly outperforms the other two methods for all workloads and that its performance is almost the same than last series of experiments, which means that *SQLB* generally does not suffer from starvation departures. Furthermore, we can see that *Capacity based* better performs than *Mariposa-like* because it better balances the query load than *Mariposa-like*. As previous series of experiments, this is because *Capacity based* ensures a better *qlb* in the system.

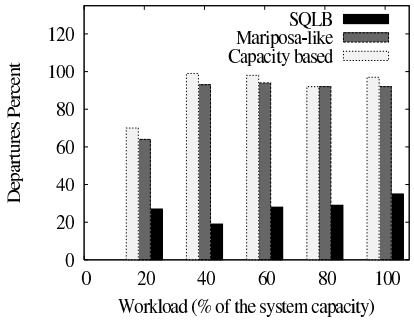


Fig. 7 Providers' departures.

Also, we run a series of experiments where providers are allowed to leave the system by dissatisfaction, starvation, or overutilization. A provider may quit the system by overutilization if this implies for example a loss of business for it, e.g. when overutilization deteriorates the quality of service provided by a provider and consumers are interested in good quality of services. This results are illustrated by Figure 6(c). We observe that while *SQLB* and *Mariposa-like* degrade their performance only by a factor of 1.4 in average (w.r.t. Figure 5(l)), *Capacity based* does it by a factor of 3.5! Figure 7 shows the number of provider's departures with the three methods. We observe that, except for a workload of 20%, *Capacity based* and *Mariposa-like* lose almost all the providers for all workloads. Note that *SQLB* only loses 28% of providers in average! This demonstrates the high efficiency of *SQLB* in autonomous environments.

Table 3 Provider's departures reasons for a workload of 80% of the total system capacity.

		SQLB				*Capacity based*				*Mariposa-like*			
		low	med	high	total	low	med	high	total	low	med	high	total
Dissat.	Cons. Interest to Prov.	1%	5%	13%		5%	16%	31%		1%	7%	11%	
	Providers' Adequation	2%	9%	8%	19%	3%	34%	15%	52%	0%	15%	4%	19%
	Providers' Capacity	13%	6%	0%		13%	30%	9%		5%	12%	2%	
		low	med	high	total	low	med	high	total	low	med	high	total
Starv.	Cons. Interest to Prov.	0%	0%	4%		0%	0%	0%		0%	2%	6%	
	Providers' Adequation	4%	0%	0%	4%	0%	0%	0%	0%	3%	3%	2%	8%
	Providers' Capacity	2%	2%	0%		0%	0%	0%		3%	5%	0%	
		low	med	high	total	low	med	high	total	low	med	high	total
Overuti.	Cons. Interest to Prov.	0%	0%	6%		0%	0%	38%		0%	0%	65%	
	Providers' Adequation	0%	3%	3%	6%	3%	8%	27%	38%	1%	15%	49%	65%
	Providers' Capacity	1%	4%	1%		0%	18%	20%		0%	30%	35%	

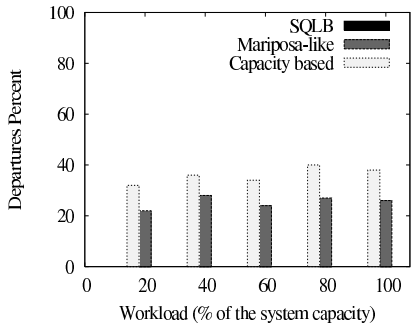


Fig. 8 Consumers' departures.

We show, in Table 3, an analysis of providers' reasons to leave the system when the workload is 80%. We observe that, as predicted in Section 6.3.1, providers leave the system with *Capacity based* because of dissatisfaction, while they do so because of overutilization with *Mariposa-like*. Furthermore, the providers that decide to leave in both methods are mainly those that are the most adapted to incoming queries and that consumers desire the most. With *SQLB*, providers leave the system by dissatisfaction, but such providers are mainly those that are *low*-capacity. In fact, we can see that *SQLB* mainly maintains the *high*-interest, *high*-adaptation, and *high*-capacity providers in the system.

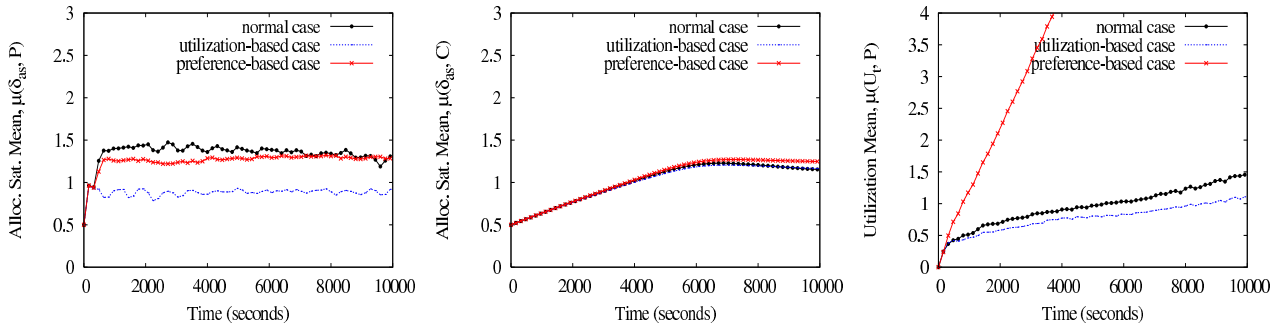
Finally, Figure 8 shows the consumers' departure by dissatisfaction with these three methods. Again, *SQLB* is a clear winner with no consumer's departures! Note that, the consumer's departures have also a direct impact on performance since the less the incoming queries, the less the chances for satisfying providers.

6.3.3 Adaptability to participants' interests

Our objective in this section is to study how well *SQLB* adapts to different participants' *intentions*. With this in mind, we consider again captive environments such as in Section 6.3.1. For simplicity, we evaluate in this paper providers with two different intentions: those that are only interested in their preferences (the *preference-based case*), i.e. the providers' preferences denote their intentions, and those that are only interested in their load (the *utilization-based case*), i.e. providers compute their intentions based on their utilization. Consumers work out their intentions regarding the providers' capacity to perform queries, such as in previous sections. We compare results of *SQLB* in both cases with those obtained in the *normal case*, i.e. when providers make a balance between their preferences and utilization to compute their intentions, such as in Section 6.3.1.

Figure 9 shows the results of these experiments with a workload range from 30 to 100% of the total system capacity. We can observe in Figures 9(a) and 9(b) that the results are strongly related to the participants' expectations. We can observe in Figure 9(a) that, as expected, providers are more satisfied in the *preference-based case* than in the *utilization-based case*. But, contrary to the expected, providers are less satisfied in the *preference-based case* than in the *normal case*. During our experimentations, we observed that those providers with *high*-adaptation tend to monopolize the queries, which causes dissatisfaction to the *medium* and *low*-adaptation providers. This phenomenon does not occur in the *normal case* because *SQLB* also considers the providers' utilization. This is why providers are in average less satisfied in the *preference-based case* than in the *normal case*. However, since in the *normal case* providers pay more attention to their utilization as the workload increases, providers have the same degree of satisfaction, for high workloads, in both *preference-based* and *normal cases*.

In Figure 9(b), we observe that consumers have the same degree of satisfaction in the three cases, but we can observe, in the *preference-based case*, a very small gain for high workloads. This is because for high workloads, providers give more importance to their *utilization* in both *utilization-based* and normal cases. Hence,



(a) Providers' allocation satisfaction. (b) Consumers' allocation satisfaction. (c) Query load mean.

Fig. 9 Quality results for a workload range from 30 to 100% of the total system capacity when participants are captive and for three kinds of providers: *(i)* when they are interested only in their preferences (the *preference-based case*), *(ii)* when they are just interested in their utilization (the *utilization-based case*), and *(iii)* when their utilization is as important as their preferences (the *normal case*).

for high workloads, the query allocation pays more attention to providers and thus the consumers' satisfaction decreases in these both cases.

Now, concerning *qlb*, *SQLB* performs well in the *utilization-based* and *normal cases* while, in the *preference-based case*, *SQLB* significantly degrades the providers' utilization because providers have no consideration for *qlb*. On the other side, observe that, in the *utilization-based case*, *SQLB* follows the behavior of the *Capacity based* approach (see Figures 9(a) and 9(c)) with regards to the providers' results, but it is much better from a consumer point of view.

All above results allow us to conclude that *SQLB* allows participants to obtain from the system what they want and not what the system considers relevant for them. In other words, our results demonstrate that *SQLB* ensures good levels of satisfaction as far as the system is adequate to participants and *vice versa*. Thus, if the participants correctly work out their intentions, *SQLB* allows them to reach their expectations.

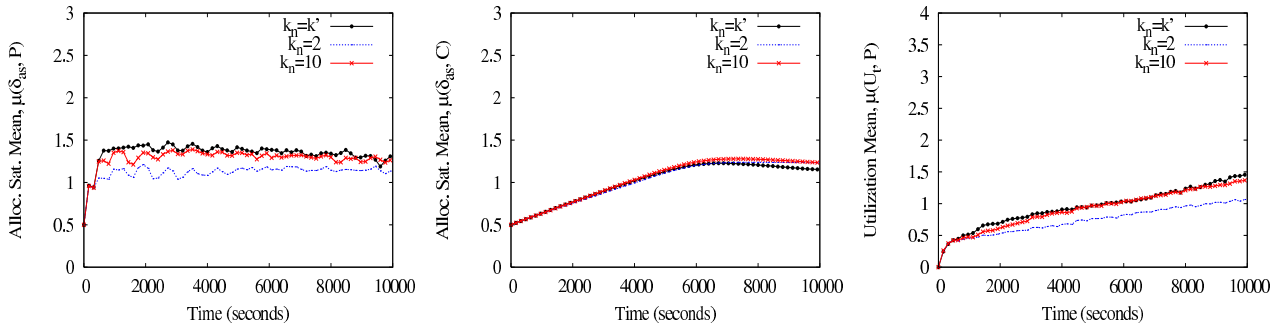
6.3.4 Query load balance control

We finally discuss how to adapt *SQLB* to different applications by varying parameter k_n. To better illustrate the effects of varying parameter k_n (i.e. the regulation of the system concerning *qlb*), we consider two kinds of providers: those that do not have any consideration for their *utilization* when they compute their intentions (the *preference-based case*), and those that make a balance of their *preferences* and their utilization to compute their intentions (the *normal case*).

For simplicity, we consider only two different applications in this work: *(i)* one where ensuring the performance of the system is mandatory such as in distributed databases and *(ii)* other where participants' satisfaction is mandatory and some level of system's performance is desired such as in *e-commerce* scenarios. For the first kind of application, the mediator should perform *qlb* while guaranteeing interesting results and queries to participants because of their autonomy. For the second kind of application, the mediator's priority is to satisfy providers while ensuring an acceptable system performance. To do so, for the first kind of application, the mediator sets parameter $k_n = 2$ and it sets $k_n = 10$ for the second one.

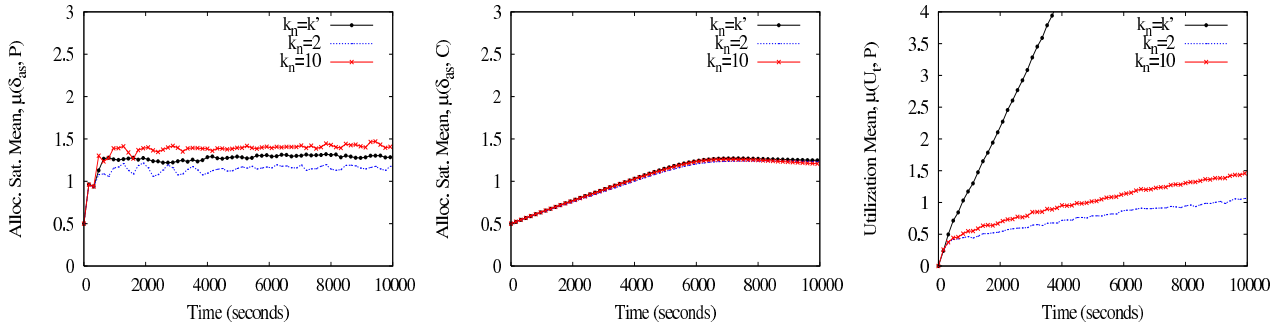
To clearly see the impact of parameter k_n, we compare both results (i.e. when $k_n = 2$ and $k_n = 10$) with the case where the mediator has no control to regulate the system (i.e. when $k_n = k'$). Notice that the previous sections assumed that $k_n = k'$, thus the results of *SQLB* in the *normal* and *preference-based cases* that we present in this section are the same as those we presented in Sections 6.3.1 and 6.3.3, respectively. We present them again as references for both two other k_n sizes ($k_n = 2$ and $k_n = 10$).

We can see in Figures 10(a) and 11(a) that for all three different k_n values and both *normal* and *preference-based cases*, providers are generally satisfied with the job done by our approach, which is not obvious in applications when *qlb* is the most important (e.g. the $k_n = 2$ case). Notice that providers are more satisfied in the *normal case* as the k_n value increases (see Figure 10(a)), but this is not the case for providers in the *preference-based case* when $k_n = k'$ and $k_n = 10$ (see Figure 11(a)). This is because, as noted in the previous section, the *high*-adaptation providers tends to monopolize the queries when they compute their intentions based only on their preferences, i.e. the *preference-based case*. But, when the mediator regulates the system with respect to *qlb*, it better distributes queries among providers and thus avoids, in the *preference-based case*, the query starvation in the less adapted providers (i.e. in the providers with *medium* and *low*-adaptation). Of



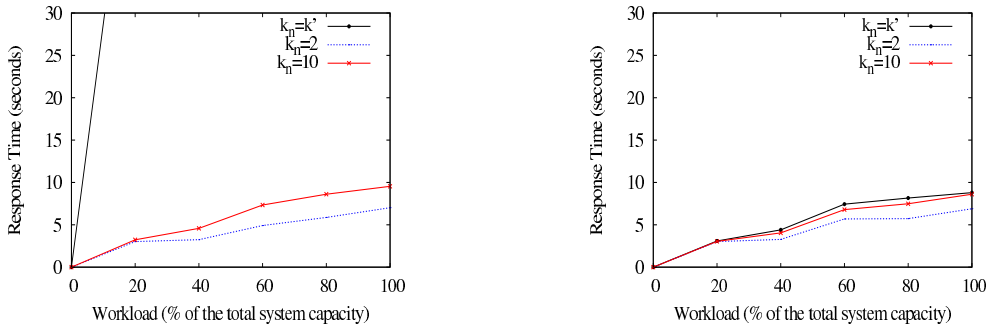
(a) Providers' allocation satisfaction mean. (b) Consumers' allocation satisfaction mean. (c) Query load mean.

Fig. 10 Quality results for a *workload* range from 30 to 100% of the total system capacity when participants are *captive* and providers compute their intentions based on their preferences and utilization (the *normal case*).



(a) Providers' allocation satisfaction mean. (b) Consumers' allocation satisfaction mean. (c) Query load mean.

Fig. 11 Quality results for a *workload* range from 30 to 100% of the total system capacity when participants are *captive* and providers compute their intentions based on their preferences (the *preference-based case*).



(a) When providers' preferences denote their intentions (the *preference-based case*).

(b) When providers make a balance between their preferences and utilization to compute their intentions (the *normal case*).

Fig. 12 Performance results with captive participants.

course, when k_n takes small values the providers are less satisfied (which is the case of $k_n = 2$) because the mediator pays less attention to the providers' intentions. In these cases, however, even if the objective is the same for both, *SQLB* performs much better than the *Capacity based* approach because it satisfies both consumers and providers (see Figures 5(c) and 5(e) for *Capacity based*). In fact, we can observe in Figures 10(b) and 11(b) that the regulation of the system has almost no impact on the consumers, which are equally satisfied for all k_n values.

Concerning *qlb*, we can see in Figures 10(c) and 11(c) that the mediator can ensure good *qlb* even if providers do not have any consideration to their *utilization*. Obviously, the smaller the k_n value, the better the ensured *qlb* in the system. In these results, it is worth noting that, even when ensuring participants' satisfaction is the most important in an application (when $k_n = k'$), the way in which *SQLB* computes the providers' *score* allows it to ensure an acceptable *qlb* in the system as far as providers take care of their load, e.g. in the *normal case* (see Figure 10(c)). This is not the case for the *preference-based case*, when $k_n = k'$, even if providers' preferences are the same (see Figure 11(c)). But, by setting small k_n values, *SQLB* can ensure short response times for consumers in both *cases*, no matter how providers compute their intentions.

The ensured response times with different k_n values are shown by Figures 12(a) and 12(b). We can observe that, as expected, the mediator can ensure good response times, even if providers are not interested in, by playing with parameter k_n (the $k_n = 2$ and $k_n = 10$ results). This is not the case when the mediator does not regulate the system and providers do not care about the system performance (the $k_n = k'$ results for the *preference-based case*).

The results in this section demonstrate that with small k_n values, one can adapt *SQLB* to applications where the mediator needs to regulate the system w.r.t. a given predefined function (*qlb* in this work) without mattering how participants compute their intentions. With high k_n values, one can adapt *SQLB* to applications where the mediator has to meet the participants' expectations.

7 Related Work

The query allocation problem, which appears as a subproblem of query processing [16], is very general and is addressed in many domains such as distributed databases, networking systems, grid systems, and multi-agent systems. The assumptions and techniques to allocate queries often differ depending on the context and the system goals. To the best of our knowledge, the problem of allocating queries by considering *qlb* and the participants' intentions has not received much attention and is still an open field. In the remainder of this section, we discuss five main approaches related to our query allocation framework: economics, data mediators, multi-agent systems, load balancing approaches, and web services. Notice that the scope of this paper goes well beyond related work by characterizing the participants' expectations in the long-run, proposing measures to analyze them and new algorithms to exploit them.

7.1 Economics

Economics is a social science concerned mainly with description and analysis of the production, distribution, and consumption of goods and services. It is subdivided into microeconomics, which studies how individuals make decisions to allocate limited resources, and macroeconomics, which studies aggregated indicators to understand how the whole economy functions. We are interested in the former in this work. In the following, we first discuss some microeconomic properties that are closely related to the satisfaction notion we proposed in this paper. Then, we present some microeconomics-based approaches to allocate queries.

7.1.1 Theory

In microeconomics, one describes participants' preferences by means of a *utility* function. A utility function assigns a numerical value to each element of a set of choices, ranking such elements according to the participants' preferences [21]. That is, for each query (good or service) a participant computes its *marginal utility* of participating in the allocation of such a query. Notice that, in our case, the participants' intentions represent somehow their marginal utility. Then, a participant computes its *total utility* gained in a given set of queries by adding its marginal utility gained in each query. In other words, as the satisfaction notion, the total utility is an abstract concept that measures the happiness or gratification of participants by consuming or performing queries. Furthermore, the total utility as well as the satisfaction makes no assumption about the way in which participants compute their marginal utility function and intention function, respectively. This is because both marginal utility and intention functions depend on applications and participants. We go beyond this by proposing a way in which participants can compute their intentions. For all this, total utility is clearly related to the notion of satisfaction we presented in this paper, but the satisfaction notion differs from the total utility in three ways. First, the satisfaction is bounded by 0 and 1 and normalized while the total utility is neither bounded nor normalized. Therefore, one can easily compare the satisfaction of participants. Second, while total utility generally considers all the queries that a participant consumed or performed, satisfaction only considers the k last queries. This is very useful when participants have a limited capacity. Finally, total utility is generally reduced to monetary

concerns only, which is not the case for the satisfaction notion.

Moreover, most economic properties (e.g. pareto-optimality, nash equilibrium, and individual rationality) focus on only one interaction. As a result, most of the economic approaches [9], which are based on one of these economic properties, look for the happiness of participants in solely one query allocation and not in the long-run. In contrast, the satisfaction notion we proposed represents the happiness of participants in several interactions, i.e. in the long-run. Also, in the field of distributed rational decision making [38], participants are assumed to be *individually rational*: the utility of any participant in the process is no less than the utility it would have by not participating. This is not relevant in environments where participants may have the interest that the system be efficient and hence, in some query allocations, they may be interested in participating in some query allocations even if this means to lose sometimes. Furthermore, it is not relevant in cooperative contexts where some participants may be imposed, which implies having a lower utility in participating. Therefore, the satisfaction notion is still relevant because it is a long-run notion.

7.1.2 Approaches

Economic approaches can claim to take into account the participants' intentions and have been shown to provide efficient query allocation in heterogeneous systems [10, 42]. A survey of economic models for various aspects of distributed system is presented in [9].

Mariposa [42] is one of the first systems to deal with the query allocation problem in distributed information systems using a *bidding* process. In Mariposa, all the incoming queries are processed by a *broker site* that requests providers for *bids*. Providers bid for acquiring queries based on a local bulletin board. Then, the broker site selects a set of bids that has an aggregate price and delay under a bid curve provided by the consumer. Mariposa ensures a crude form of load balancing by modifying the providers' bid with the providers' load. Nevertheless, our experimentations show that, in some cases, providers suffer from overutilization. Besides, queries may not be treated even if providers exist in the system. This leads to a certain domination of the providers' intentions over the consumers' intentions.

In [28], the authors focus on the optimization algorithms for *buying* and *selling* query answers, and the negotiation strategy. Their query trading algorithm runs iteratively, progressively selecting the best execution plan. At each iteration, the buyer sends requests for bids, for a set of queries, and sellers reply with offers (bids) for dealing with them. Then, the buyer finds the best possible execution plan based on the offers it received. These actions are iterated until either the found execution plan is not better than the plan found in the previous iteration or the set of queries has not been modified (i.e. there is no new subqueries). This approach uses some kind of bargaining between the buyer and the sellers, but with different queries at each iteration. However, this way of dealing with subqueries optimization is orthogonal to our proposal and one may combine them to improve performance.

In [18], the authors propose an economic *flexible mediation* approach that allocates queries by taking into account the providers' *quality* (given by consumers) and the providers' bids. In contrast to our approach, the authors inherently assume that participants are captive. In addition, their proposed economic model is complementary to our proposal and one can combine them to obtain an economic version of *SQLB*, by computing bids with respect to intentions [33,34].

7.2 Data Mediators

Over the last years, data mediator systems [44] have been accepted as a viable approach for integrating heterogeneous and distributed providers. Data mediators allow consumers to query different providers that are typically wrapped to provide an uniform interface to a mediator. Two of the most prominent approaches are TSIMMIS [13] and Information Manifold [19]. In data mediator systems, the mediator allocates queries to providers and integrates results for consumers, much like distributed database systems [27]. Nevertheless, data mediators require some global information such as global schemas [43], which is difficult to maintain in dynamic systems because source schemas change frequently. *SQLB* does not require any global knowledge, but it does not address the integration problem.

7.3 Multi-agent Systems

In multi-agent systems, the *Contract Net Protocol (CNP)* [41] is often mentioned as a way to allocate queries. However, *CNP* is a simple protocol and there is no control to regulate the system. Besides, it is generally assumed a rather small number of participants and a detailed description of the conditions of execution, which is not our case. Several approaches of middle-agents have been defined [8,17,26] and a survey can be found in [15]. A classical goal of middle-agents is to find the providers that are able to deal with a given query by matching providers' capabilities advertisements with

the given query. All these works are efficient but the number of selected providers may remain too large.

Thus, some works have investigated the possibility of reducing the list of selected providers. For example, *Z. Zhang and C. Zhang* [45] propose to perform classical matchmaking and then refine the result list of providers by considering the providers' quality. Nonetheless, the participants' intentions are not considered by the providers selection procedure, which does not allow a participant to have an active participation in the selection process.

D. Bernstein et al. [7] propose an adaptive approach to allocate queries, in file sharing-systems, based on the machine learning methodology. In this approach, a consumer can perform partial downloads from providers before finally settling on one. This approach allows the consumers to improve response times by aborting bad download attempts until an acceptable provider is discovered. However, the authors inherently assume that consumers are only interested in response times and providers have no interests to perform queries.

7.4 Load Balancing Approach

In the context of large-scale and heterogeneous distributed systems, most of the work on query allocation has mainly dealt with the problem of balancing queries among providers. There exist two well-known approaches to balance queries across providers: *Load Based* and *Capacity based* methods. *Load Based* methods [12,6] are not suitable for heterogeneous systems since they inherently assume that providers and queries are homogeneous. *Capacity based* methods [24, 35,40] allocate queries in accordance to the providers' utilization, i.e. they allocate each incoming query to providers with the most available capacity. Nevertheless, all these approaches have no consideration for the participants' intentions.

In [30] and [31], we propose a method and strategy, respectively, to balance queries among providers by considering providers' intentions and satisfaction, but no notion of intentions nor satisfaction of consumers is considered.

7.5 Web Services

To locate and select services, web services (the providers) have to describe properly all their proposed services [5]. Once services have been properly described, these descriptions are made available, via a service directory (the registry), to those interested in using them (the consumers). These directories can be hosted and managed by a trusted entity (centralized approach) or each provider can host and manage them (peer-to-peer approach). When a consumer has located the providers providing the service it desires, it then selects the provider that it wants. Then, the consumer selects the provider with respect to its interests, e.g. with the highest score among the providers in the registry. However, providers cannot express their intentions to perform queries and are considered as captive values.

8 Conclusion

We considered distributed information systems where participants are *autonomous* to leave the system at will. In this context, it is crucial to consider the participants' *intentions* to allocate and perform queries so that their expectations, response times, and system capacity are ensured. We presented, in this paper, a general and complete solution to allocate queries among providers by considering the participants' intentions and *query load balancing* (*qlb*). Our work carried out fourth main contributions.

First, we characterized the participants' expectations in a new model, which allows to evaluate a system from a satisfaction point of view. The definitions that we proposed are original, considering the long-run notions of *adequation* and *satisfaction*. They are independent of the way participants compute their intentions and how the mediator considers them. This model facilitates the design and evaluation of new query allocation methods for these environments. The proposed model is general, and thus, can be used for any distributed systems architecture.

Second, we proposed three different measures to evaluate the quality of *qlb* methods:

- The *mean* measure reflects the effort that a query allocation method does for equally either maximizing or minimizing a given set of values.
- The *fairness* measure evaluates how fair a query allocation method is.
- The *balance* measure measures the Min-Max values.

We proved that using these proposed measures together, one can predict possible consumers' and providers' departures from the system.

Third, we presented the *SQLB* framework for balancing queries in these environments. The originality of *SQLB* is to perform all query demand while satisfying participants' expectations. *SQLB* strongly differs from the related work in several ways:

- It allows providers to trade their preferences for their utilization while keeping their strategic information private.

- It affords consumers the flexibility to trade their preferences for providers' reputation.
- It allows trading consumers' intentions for providers' intentions.
- It strives to balance queries at runtime via the participants' satisfaction, thus reducing *starvation*.
- It affords the mediator to regulate the system with respect to some predefined function and can adapt the query allocation process to the kind of application.
- It can ensure good levels of satisfaction as far as the system is adequate to participants and *vice versa*, which allows participants to reach their expectations in the system whether they correctly work out their intentions.

Fourth, we evaluated and compared *SQLB* with two baseline query allocation methods (*Capacity based* and *Mariposa-like*), in two kinds of environments: *captive* and *autonomous*. We showed through experimentation that, by considering together the *qlb* and satisfaction of participants, *SQLB* significantly outperforms both baseline methods. We observed that participants are, in general, very satisfied with *SQLB* and *Mariposa-like*, which is not the case for *Capacity based* that suffers from several providers' departures due to dissatisfaction. However, *Mariposa-like* has serious problems for balancing queries correctly. On the one hand, we showed that, unlike the baseline methods, *SQLB* maintains the *high*-interest, *high*-adaptation, and *high*-capacity providers in the system. On the other hand, the results show that while baseline methods lose more than 20% of consumers (for all workloads), *SQLB* has no consumer's departures! We showed the self-adaptability of *SQLB* to the expectations and satisfaction of participants. We also discussed its adaptability to different kinds of applications. All these results demonstrate that *SQLB* can scale up with autonomous participants, while *Capacity based* and *Mariposa-like* cannot.

As future work, we plan to address fault-tolerant issues so that *SQLB* ensures good system performance even in the presence of participants' failures. We also plan to study the impact, on performance and participants' satisfaction, of introducing the random providers selection phase (discussed in Section 5.3.2). In our experimentations, we also noted that self-organizing phenomenons occur. When the system is composed of several mediators and the participants have divergent interests, the participants that share the same interests gather on the same mediator. We wish to explore this phenomenon that we not observe with the other approaches. Finally, the problem addressed in this paper as that addressed by economic approaches is to regulate a distributed information system while satisfying participants. We then plan to develop an economic version of *SQLB*, based on [18], to scale up to several mediators and to analyze in detail the contributions of the use of money for allocating queries.

References

1. The eBay System. http://business.ebay.com.
2. The Freenet Project. http://freenetproject.org.
3. The Freightquote System. http://www.Freightquote.com.
4. The Gnutella Project. http://www.gnutella.com.
5. Alonso G., Casati F., Kuno H., and Machiraju V.: Web Services: Concepts, Architecture, and Applications. Springer-Verlag, (2004).
6. Azar Y., Broder A. Z., Karlin A. R., and Upfal E.: Balanced Allocations. SIAM Journal on Computing. 29(1), 180–200 (1999).
7. Bernstein D. S., Feng Z., Levin B. N., and Zilberstein S.: Adaptive Peer Selection. In Proceedings of the International Workshop on Peer-to-Peer Systems, IPTPS, (2003).
8. Decker K., Sycara K., and Williamson M.: Middle-Agents for the Internet. In Proceedings of the International Joint Conferences on Artificial Intelligence, IJCAI, (1997).
9. Ferguson D., Nikolaou C., Sairamesh J., and Yemini Y.: Economic Models for Allocating Resources in Computer Systems. In: S. H. Clearwater (ed) Market-Based Control: A Paradigm for Distributed Resource Allocation, World Scientific, (1996).
10. Ferguson D., Yemini Y., and Nikolaou C.: Microeconomic Algorithms for Load Balancing in Distributed Computer Systems. In Proceedings of the International Conference on Distributed Computing Systems, ICDCS, (1988).
11. Fong T., Fowler D., and Swatman P.: Success and Failure Factors for Implementing Effective Electronic Markets. Journal of Electronic Commerce and Business Media. 8(1), 45–47 (1998).
12. Ganesan P., Bawa M., and Garcia-Molina H.: Online Balancing of Range-Partitioned Data with Applications to Peer-to-Peer Systems. In Proceedings of the Very Large Data Bases Conference, VLDB, (2004).
13. Garcia-Molina H., Papakonstantinou Y., Quass D., Rajaraman A., Sagiv Y., Ullman J. D., Vassalos V., and Widom J.: The TSIMMIS Approach to Mediation: Data Models and Languages. Journal of Intelligent Information Systems. 8(2), 117–132 (1997).
14. Jain R. K., Chiu D.-H., and Hawe W. R.: A Quantitive Measure of Fairness and Discrimination for Resource Allocation in Shared Computer Systems. Digital Equipment Corporation, Technical Report. DEC-TR-301 (1984).
15. Klusch M. and Sycara K.: Brokering and Matchmaking for Coordination of Agent Societies: A Survey. In: Coordination of Internet Agents: Models, Technologies, and Aplications, Springer-Verlag, (2001).
16. Kossmann D.: The State of the Art in Distributed Query Processing. ACM Computing Surveys. 32(4), 422–46 (2000).
17. Kuokka D. and Harada L.: Matchmaking for Information Agents. In Proceedings of the International Joint Conferences on Artificial Intelligence, IJCAI, (1995).
18. Lamarre P., Cazalens S., Lemp S., and Valduriez P.: A Flexible Mediation Process for Large Distributed Information Systems. In Proceedings of the Cooperative Information Systems Conference, CoopIS, (2004).

19. Levy A. Y., Rajaraman A., and Ordille J.J.: Querying Heterogeneous Information sources Using Source Descriptions. In Proceedings of the Very Large Data Bases Conference, VLDB, (1996).
20. Li L. and Horrocks I.: A Software Framework for Matchmaking Based on Semantic Web Technology. In Proceedings of the World Wide Web Conference, WWW, (2003).
21. Mas-Colell A., Whinston M., and Green J.: Microeconomic Theory. Oxford University Press, (1995).
22. Markatos E. P.: Tracing a Large-Scale Peer to Peer System: An Hour in the Life of Gnutella. In Proceedings of the IEEE/ACM International Symposium on Cluster Computing and the Grid (2002).
23. Miller R.: Special Issue on Integration Management of the IEEE Data Engineering Bulletin. 25(3), (2002).
24. Mirchandaney R., Towsley D. F., and Stankovic J. A.: Adaptive Load Sharing in Heterogeneous Distributed Systems. Journal of Parallel and Distributed Computing, JPDC. 9(4), 331–346 (1990).
25. Mitzenmacher M.: The Power of Two Choices in Randomized Load Balancing. PhD. Thesis, UC Berkeley (1996).
26. Nodine M. H., Bohrer W., and Ngu A. H.: Semantic Brokering over Dynamic Heterogeneous Data Sources in InfoSleuth. In Proceedings of the International Conference on Data Engineering, ICDE, (1999).
27. Özsu T. and Valduriez P.: Principles of Distributed Database Systems, Second Edition. Prentice-Hall, (1999).
28. Pentaris F. and Ioannidis Y.: Query Optimization in Distributed Networks of Autonomous Database Systems. ACM Transactions on Database Systems, TODS. 31(2), 537–583 (2006).
29. Pieper S., Paul J., and Schulte M.: A New Era of Performance Evaluation. IEEE Computer. 40(9), 23–30 (2007).
30. Quiané-Ruiz J.-A., Lamarre P., and Valduriez P.: Satisfaction Based Query Load Balancing. In Proceedings of the Cooperative Information Systems Conference, CoopIS, (2006).
31. Quiané-Ruiz J.-A., Lamarre P., and Valduriez P.: K_nBest - A Balanced Request Allocation Method for Distributed Information Systems. In Proceedings of the Database Systems for Advanced Applications Conference, DASFAA, (2007).
32. Quiané-Ruiz J.-A., Lamarre P., and Valduriez P.: SQLB: A Query Allocation Framework for Autonomous Consumers and Providers. In Proceedings of the Very Large Data Bases Conference, VLDB, (2007).
33. Quiané-Ruiz J.-A., Lamarre P., Sylvie Cazalens, and Valduriez P.: Satisfaction Balanced Mediation. In Proceedings of the Conference on Information and Knowledge Management, CIKM, (2007).
34. Quiané-Ruiz J.-A., Lamarre P., Sylvie Cazalens, and Valduriez P.: Managing Virtual Money for Satisfaction and Scale Up in P2P Systems. In Proceedings of the EDBT Workshop on Data Management in Peer-to-Peer Systems, DAMAP, (2008).
35. Rahm E. and Marek R.: Dynamic Multi-Resource Load Balancing in Parallel Database Systems. In Proceedings of the Very Large Data Bases Conference, VLDB, (1995)
36. Roth M. and Schwarz P.: Don't Scrap It! Wrap It! A Wrapper Architecture for Legacy Data Sources. In Proceedings of the Very Large Data Bases Conference, VLDB, (1997).
37. Sah A., Blow J., and Dennis B.: An Introduction to the Rush Language. In Proceedings of the TCL Workshop (1994).
38. Sandholm T. W.: Distributed Rational Decision Making. In: G. Weiss (ed) Multiagent Systems, a Modern Approach to Distributed Artificial Intelligence, The MIT Press, (2001).
39. Saroiu S., Gummadi P. K., and Gribble S. D.: A Measurement Study of Peer-to-Peer File Sharing Systems. In Proceedings of the Multimedia Computing and Networking Conference, MMCN, (2002).
40. Shivaratri N. G., Krueger P., and Singhal M.: Load Distributing for Locally Distributed Systems. IEEE Computer. 25(12), 33–44 (1992).
41. Smith R. G.: The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver. IEEE Transactions on Computers. C-29(12), 1104–1113 (1981).
42. Stonebraker M., Aoki P., Litwin W., Pfeffer A., Sah A., Sidell J., Staelin C., and Yu A.: Mariposa: A Wide-Area Distributed Database System. Journal on Very Large Data Bases, VLDBJ. 5(1), 48–63 (1996).
43. Tomasic A., Raschid L., and Valduriez P.: Scaling Access to Heterogeneous Data Sources with DISCO. IEEE Transanctions on Knowledge and Data Engineering, TKDE. 10(5), 808–823 (1998).
44. Wiederhold G.: Mediators in the Architecture of Future Information Systems. IEEE Computer. 25(3), 38–49 (1992).
45. Zhang Z. and Zhang C.: An Improvement to Matchmaking Algorithms for Middle Agents. In Proceedings of the Autonomous Agents and Multiagent Systems Conference, AAMAS, (2002).

Annexe 4 - European Semantic Web Conference 2008

[VCLV08c] Anthony Ventresque, Sylvie Cazalens, Philipp Lamarre and Patrick Valduriez, *Improving Interoperability Using Query Interpretation in Semantic Vector Spaces*, European Semantic Web Conference, pp539-553, 2008.

nominee for best paper award.

Improving Interoperability Using Query Interpretation in Semantic Vector Spaces

Anthony Ventresque[1], Sylvie Cazalens[1], Philippe Lamarre[1], and Patrick Valduriez[2]

[1]LINA, University of Nantes
FirstName.LastName@univ-nantes.fr
[2]INRIA and LINA, University of Nantes
Patrick.Valduriez@inria.fr

Abstract. In semantic web applications where query initiators and information providers do not necessarily share the same ontology, semantic interoperability generally relies on ontology matching or schema mappings. Information exchange is then not only enabled by the established correspondences (the "shared" parts of the ontologies) but, in some sense, limited to them. Then, how the "unshared" parts can also contribute to and improve information exchange ? In this paper, we address this question by considering a system where documents and queries are represented by semantic vectors. We propose a specific query expansion step at the query initiator's side and a query interpretation step at the document provider's. Through these steps, unshared concepts contribute to evaluate the relevance of documents wrt. a given query. Our experiments show an important improvement of retrieval relevance when concepts of documents and queries are not shared. Even if the concepts of the initial query are not shared by the document provider, our method still ensures 90% of the precision and recall obtained when the concepts are shared.

1 Introduction

In semantic web applications where query initiators and information providers do not necessarily share the same ontology, semantic interoperability generally relies on ontology matching or schema mappings. Several works in this domain focus on what (*i.e.* the concepts and relations) the peers share [9, 18]. This is quite important because, obviously if nothing is shared between the ontologies of two peers, there is a little chance for them to understand the meaning of the information exchanged. However, no matter how the shared part is obtained (through consensus or mapping), there might be concepts (and relations) that are not consensual, and thus not shared. The question is then to know whether the unshared parts can still be useful for information exchange.

In this paper, we focus on semantic interoperability and information exchange between a query initiator p_1 and a document provider p_2, which use different ontologies but share some common concepts. The problem we address is to *find documents which are relevant to a given query although the documents and the*

query may be both represented with concepts that are not shared. This problem is very important because in semantic web applications with high numbers of participants, the ontology (or ontologies) is rarely entirely shared. Most often, participants agree on some part of a reference ontology to exchange information and internally, keep working with their own ontology [18, 22].

We represent documents and queries by *semantic vectors* [25], a model based on the vector space model [1] using concepts instead of terms. Although there exist other, richer representations (conceptual graphs for example), semantic vectors are a common way to represent unstructured documents in information retrieval. Each concept of the ontology is weighted according to its representiveness of the document. The same is done for the query. The resulting vector represents the document (respectively, the query) in the n-dimensional space formed by the n concepts of the ontology. Then the relevance of a document with respect to a query corresponds to the proximity of the vectors in the space.

In order to improve information exchange beyond the "shared part" of the ontologies, we promote both *query expansion* (at the query initiator's side) and *query interpretation* (at the document provider's side). Query expansion may contribute to weight linked shared concepts, thus improving the document provider's understanding of the query. Similarly, by interpreting an expanded query with respect to its own ontology (*i.e.* by weighting additional concepts of its own ontology), the document provider may find additional related documents for the query initiator that would not be found by only using the matching concepts in the query and the documents. Although the basic idea of query expansion and interpretation is simple, query interpretation is very difficult because it requires to precisely weight additional concepts given some weighted shared ones, while the whole space (i.e. the ontology) and similarity measures change.

In this context, our contributions are the following. First, we propose a specific query expansion method. Its property is to keep separate the results of the propagation from each central concept of the query, thus limiting the noise due to inaccurate expansion. Second, given this expansion, we define the relevance of a document. Its main, original characteristic is to require the document vector to be requalified with respect to the expanded query, the result being called *image* of the document. Third, a main contribution is the definition of query interpretation which enables the expanded query to be expressed with respect to the provider's ontology. Fourth, we provide two series of experiments with still very good results although few concepts are shared.

This paper is organized as follows. Section 2 gives preliminary definitions. Section 3 presents our query expansion method and the image based relevance of a document. For simplicity, we assume a context of shared ontology. This assumption is relaxed after in Section 4, where we consider the case where the query initiator and the document provider use different ontologies and present the query interpretation. Section 5 discusses the experiments and their results. The two last sections are respectively devoted to related work and conclusion.

2 Preliminary Definitions

We define an ontology as a set of concepts together with a set of relations between these concepts. In our experiments, we consider an ontology with only one relation: the is-a relation (specialization link). This does not restrict the generality of our relevance computation. Indeed, the presence of several relations only affects the definition of the similarity of a concept wrt. another. A *semantic vector* $\overrightarrow{v_\Omega}$ is an application defined on the set of concepts $\mathcal{C}_\Omega$ of the ontology Ω : $\forall c \in \mathcal{C}_\Omega, \overrightarrow{v_\Omega} : c \rightarrow [0..1]$. A popular way to compute the relevance of a document is to use the cosine-based proximity of the document and query vectors in the space [19]. The problem with cosine is the independence of dimensions : a query on concept c_i and a document on concept c_j very close from c_i could not match. Query expansion is generally used to express these links between concepts, by propagating initial weights on other linked concepts. To define a query expansion, we need a *similarity function* [11] which expresses how much a concept is similar to another within the ontology : sim_c: $\mathcal{C}_\Omega \rightarrow [0, 1]$, is a similarity function iff $sim_c(c) = 1$ and $0 \leq sim_c(c_j) < 1$ for all $c_j \neq c$ in $\mathcal{C}_\Omega$. Then, propagation from a central concept c of weight v assigns a weight to every value of similarity with c.

Definition 1 (Propagation function). *Let c be a concept of Ω valued by v; and let sim_c be a similarity function.*

$$\textit{A function } \mathcal{P}f_c : \begin{vmatrix} [0..1] & \mapsto [0..1] \\ sim_c(c') & \rightarrow \mathcal{P}f_c(sim_c(c')) \end{vmatrix} \textit{ is a propagation function from } c \textit{ iff}$$

- $\mathcal{P}f_c(sim_c(c)) = v$, *and*
- $\forall c_k, c_l \in \mathcal{C}_\Omega \; sim_c(c_k) \leq sim_c(c_l) \Rightarrow \mathcal{P}f_c(sim_c(c_k)) \; \leq \; \mathcal{P}f_c(sim_c(c_l))$

Among different types of propagation functions those inspired by the membership functions used in fuzzy logic work fine (see Figure 1) in our experiments. It is defined by three parameters v (weight of the central concept), l_1 (similarity value until which concepts have the same weight : v) and l_2 (similarity value until which concepts have non zero weight) such that, $\forall x = sim_c(c'), c' \in c_\Omega$:

$$\mathcal{P}f_c(x) = f_{v,l_1,l_2}(x) = \begin{vmatrix} v & if \; x \geq l_1 \\ \frac{v}{l_1 - l_2} x + \frac{l_2 \times v}{l_1 - l_2} & if \; l_1 > x > l_2 \\ 0 & if \; l_2 \geq x \end{vmatrix}$$

3 Query expansion and Image based relevance

In this section, we present our method to compute the relevance of a document wrt a query. For the sake of simplicity, we assume that the query initiator and the document provider use the same ontology. However, they can still differ on the similarity measures and the propagation functions. First, we compute a *query*

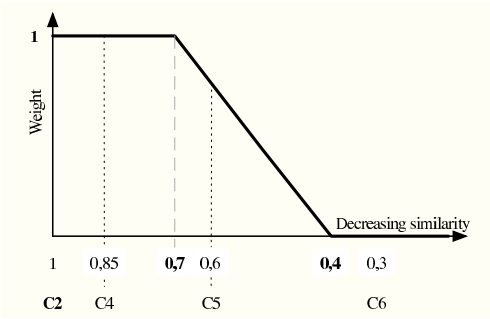


Fig. 1. Example of a propagation function $f_{1,0.7,0.4}$ with central concept c_2 .

expansion, and then an *image of a document vector* to compute the relevance of the document wrt. a query in a single space.

To our knowledge, most query expansion methods propagate the weight of each weighted concept in *the same vector*, thus directly adding the expanded terms in the original vector [13]. When a concept is involved in several propagations conducted from different central concepts, an aggregation function (e.g. the maximum) is used. We call this kind of method "rough" propagation. Although its results are not bad, such a propagation has some drawbacks among which a possible unbalance of the relative importance of the initial concepts [16]. First, let us denote by $\mathcal{C}_{\overrightarrow{q}}$ the set of the *central concepts* of query $\overrightarrow{q}$, *i.e.* those weighted concepts which represent the query. To keep separate the effects of different propagations, each central concept of $\mathcal{C}_{\overrightarrow{q}}$ is *semantically enriched* by propagation, in a separate vector.

Definition 2 (Semantically Enriched Dimension). *Let $\overrightarrow{q}$ be a query vector and let c be a concept in $\mathcal{C}_{\overrightarrow{q}}$. A semantic vector $\overrightarrow{sed}_c$ is a* semantically enriched dimension*, iff $\forall c' \in \mathcal{C}_\Omega, \overrightarrow{sed}_c[c'] \leq \overrightarrow{sed}_c[c]$.*

Definition 3 (Expansion of a query). *Let $\overrightarrow{q}$ be a query vector. An* expansion *of $\overrightarrow{q}$, noted $\mathcal{E}_{\overrightarrow{q}}$ is a set defined by:*
$\mathcal{E}_{\overrightarrow{q}} = \{\overrightarrow{sed}_c : c \in \mathcal{C}_{\overrightarrow{q}},\ \forall c' \in \mathcal{C}_\Omega, \overrightarrow{sed}_c[c'] = \mathcal{P}f_c(c')\}$

Figure 2 illustrates the expansion of a query $\overrightarrow{q}$ with two weighted concepts c_4 and c_7. It contains two semantically enriched dimensions. In dimension $\overrightarrow{sed}_{c_7}$, concept c_7 has the same value as in the query. The weight of c_7 has been propagated on c_3, c_{11} and c_6 according to their similarity with c_7. The other dimension is obtained from c_4 in the same way.

The expanded query is composed of several semantic vectors (the SEDs). Our aim is then to transform the semantic vector of a document, $\overrightarrow{d}$, in an *image* through the expanded query, i.e. to characterize the document wrt. each central concept c (dimension) of the query, as far as it has concepts related to c, in particular even if c is not initially weighted in $\overrightarrow{d}$. Given a SED $\overrightarrow{sed}_c$, we aim

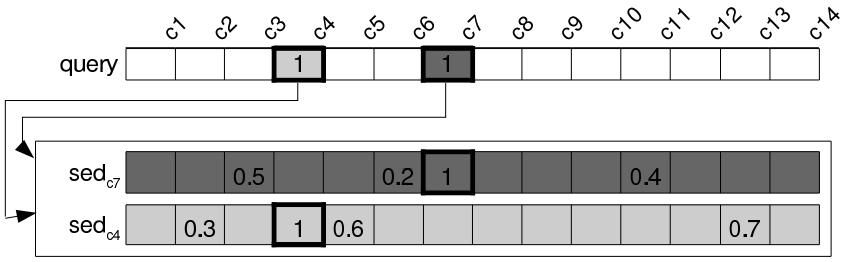


Fig. 2. A query expansion composed of 2 semantically enriched dimensions.

at valuating c in the image of the document $\overrightarrow{d}$ according to the relevance of $\overrightarrow{d}$ to $\overrightarrow{sed_c}$. To evaluate the impact of $\overrightarrow{sed_c}$ on $\overrightarrow{d}$ we consider the product of the respective values of each concept in $\overrightarrow{sed_c}$ and $\overrightarrow{d}$. Intuitively, all the concepts of the document which are linked to c through $\overrightarrow{sed_c}$ have a nonnull value. The image of $\overrightarrow{d}$ keeps track of the best value assigned to one of the linked concepts if it is better than $\overrightarrow{d}[c]$, which is the initial value of c . This process is repeated for each SED of the query. Algorithm 1.1 gives the computation of the image of document $\overrightarrow{d}$, noted $\overrightarrow{i}_d$. This algorithm ensures that all the central concepts of the initial query vector are also weighted in the image of the document as far as the document is related to them. Wrt. the query, the image of the document is more accurate because it enforces the documents characterization over each dimension of the query. However, in the image, we keep unchanged the weights of the concepts which are not linked to any concept of the query (*i.e.* which are not weighted in any SED). The example of Figure 3 illustrates how the image of a document is computed.

Algorithm 1.1. Image of a document wrt a query.

```
(* Input : a semantic vector d on an ontology Ω;
         an expanded query E_q *)
(* Output: a semantic vector i_d, image of d. *)
begin
    for c ∈ C_q do
        for c' : sed_c[c'] ≠ 0 do
            i_d[c] ← max(d[c'] × sed_c[c'], i_d[c]);
    for c ∉ C_q do if ∃c' ∈ C_q : sed_c'[c] ≠ 0 then i_d[c] ← 0
                                                else i_d[c] ← d[c];
    return i_d
end;
```

We define the relevance of $\overrightarrow{d}$ wrt. $\overrightarrow{q}$ by $\cos(\overrightarrow{i}_d, \overrightarrow{q})$. Considering the image enables to take into account the documents that have concepts linked to those

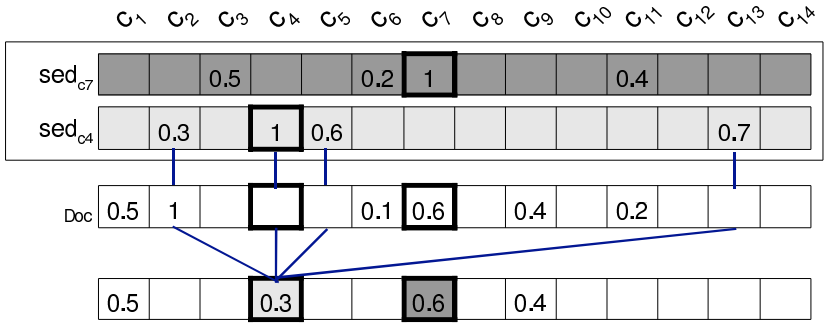


Fig. 3. Obtaining the image of a document.

of the query. Using a cosine, and thus the norm of the vectors, assigns a lower importance to the documents with an important norm, which are often very general.

4 Relevance in the context of unshared concepts

In this section, we assume that the query initiator and the document provider do not use the same ontology. We follow the approach adopted in Section 3, using a query expansion at the query initiator's side and the computation of the image of the document at the provider's side. But things get complicated by the fact that the query initiator and the document provider do not use the same vector space. An additional step is needed in order to evaluate relevance in a same and single space. Thus, we introduce a *query interpretation* step at the provider's side.

4.1 Computing Relevance: Overview

As shown in Figure 4, the query initiator, denoted by p_1, works within the context of ontology Ω_1, while the document provider, noted p_2, works with ontology Ω_2. Through its semantic indexing module, the query initiator (respectively the document provider) produces the query vector (respectively the document vector), which is expressed on Ω_1 (respectively Ω_2). Both p_1 and p_2 also have their own way of computing both the similarity and the propagation.

We assume that the query initiator and the document provider *share* some common concepts, meaning that each of them regularly, although may be not often, runs an ontology matching algorithm. Ontology matching results in an *alignment* between two ontologies, which is composed of a (non empty) set of correspondences with some cardinality and, possibly some meta-data [4]. A *correspondence* establishes a relation (equivalence, subsumption, disjointness...) between some entities (in our case, concepts), with some confidence measure.

Each correspondence has an identifier. In this paper, we only consider the equivalence relation between concepts and those couples of equivalent concepts of which confidence measure is above some threshold. We call them the *shared* concepts. For simplicity, when there is an equivalence, we make no distinction between the name of the given concept at p_1's, its name at p_2's, and the identifier of the correspondence, which all refer to the same concept. Hence, the set of shared concepts is denoted by $\mathcal{C}_{\Omega_1} \cap \mathcal{C}_{\Omega_2}$.

Given these assumptions, computing relevance requires the following steps :

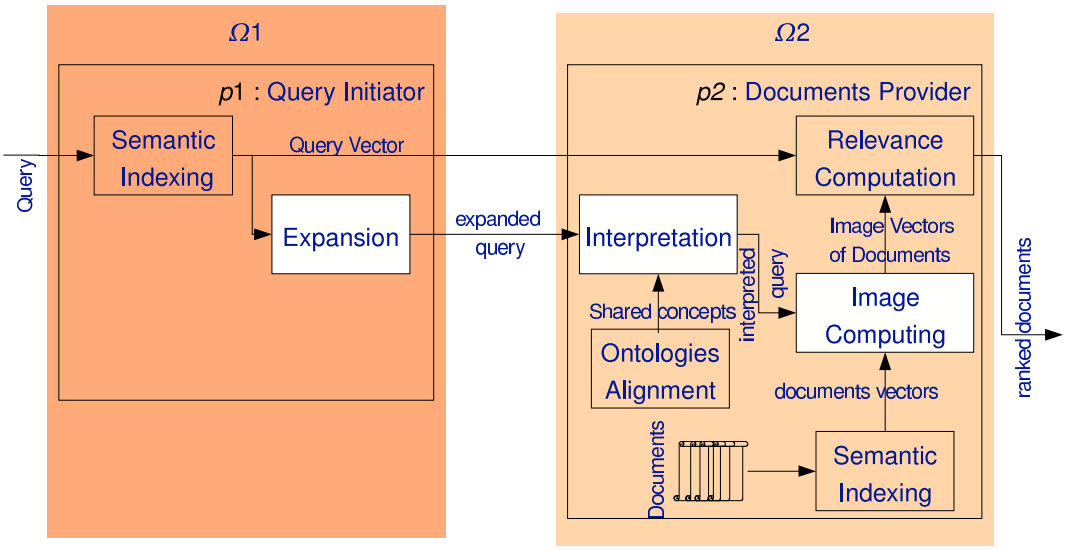


Fig. 4. Overview of relevance computation

Query Expansion. It remains unchanged. The query initiator p_1 computes an *expansion* of its query, which results in a set of SEDs. Each SED is expressed on the set $\mathcal{C}_{\Omega_1}$, no matter the ontology used by p_2. Then, the expanded query is sent to p_2, together with the initial query.

Query Interpretation. Query interpretation by p_2 provides a set of interpreted SEDs on the set $\mathcal{C}_{\Omega_2}$ and an interpreted query. Each SED of the expanded query is interpreted separately. Interpretation of a SED $\overrightarrow{sed}_c$ is decomposed in two problems, which we address in the next subsections:

- The first problem is to find a concept in $\mathcal{C}_{\Omega_2}$ that corresponds to c, noted $\tilde{c}$. This is difficult when the central concept is not shared. In this case, we use the weights of the shared concepts to guide the search. Of course, this is only a "contextual" correspondence as opposed to one that would be obtained through matching.
- The second problem is to attribute weights to shared and unshared concepts of $\mathcal{C}_{\Omega_2}$ which are linked to $\overrightarrow{sed}_c$. This amounts to interpret the SED.

Image of the Document and Cosine Computation. They remain unchanged. Provider p_2 computes the image of its documents wrt. the interpreted

SEDs and then, their cosine based relevance wrt. the interpreted query, no matter the ontology used by p_1.

In the following, we describe the steps involved in the interpretation of a given SED.

4.2 Finding a Corresponding Concept

The interpretation of a given SED $\overrightarrow{sed}_c$ leads to a major problem: finding a concept in $\mathcal{C}_{\Omega_2}$ which corresponds to the central concept c. This corresponding concept is noted $\tilde{c}$ and will play the role of the central concept in the interpretation of $\overrightarrow{sed}_c$, noted $\overrightarrow{sed}_{\tilde{c}}$. If c is shared, we just keep it as the central concept of the interpreted SED. When c is not shared we have to find a concept which seems to best respect the "flavor" of the initial SED.

Theoretically, all the concepts of $\mathcal{C}_{\Omega_2}$ should be considered. Several criterias can apply to choose one which seems to best correspond. We propose to define the notion of *interpretation function* which is relative to a SED $\overrightarrow{sed}_c$ and a candidate concept $\tilde{c}$ and which assigns a weight to each value of silmilarity wrt. $\tilde{c}$. Definition 4 consists of four points. The first one requires the interpretation function to assign the value of $\overrightarrow{sed}_c[c]$ to the similarity value 1, which corresponds to $\tilde{c}$. In the second point, we use the weights assigned by $\overrightarrow{sed}_c$ to the shared concepts (c_1, c_2, c_3 and c_6 in figure 5) and the ranking of concepts in function of $sim_{\tilde{c}}$. However, there might be several shared concepts that have the same similarity value wrt. $\tilde{c}$, but have a different weight according to $\overrightarrow{sed}_c$. Thus, we require function $f_i^{\overrightarrow{sed}_c,\tilde{c}}$ to assign the minimum of these values to the corresponding similarity value. This is a pessimistic choice and we could either take the maximum or a combination of these weights. As for the third point, let us call c_{min}, the shared concept with the lowest similarity value (c_6 in Figure 5 (a) and c_3 in Figure 5 (b)). We consider that we have not enough information to weight the similarity values lower than $sim_{\tilde{c}}(c_{min})$. Thus we assign them the zero value. The fourth point is just a mathematical expression which ensures that the segments of the affine function are only those defined by the previous points.

Definition 4 (Interpretation function). *Given a SED $\overrightarrow{sed}_c$ and a concept $\tilde{c}$, $f_i^{\overrightarrow{sed}_c,\tilde{c}} : [0..1] \rightarrow [0..1]$, noted f_i if no ambiguity, is an* interpretation function *iff it is a piecewise affine function and:*

- $f_i(1) = \overrightarrow{sed}_c[c]$;
- $\forall c' \in \mathcal{C}_{\Omega_1} \cap \mathcal{C}_{\Omega_2},\ f_i(sim_{\tilde{c}}(c')) = \min_{\substack{c'' \in \mathcal{C}_{\Omega_1} \cap \mathcal{C}_{\Omega_2} \\ sim_{\tilde{c}}(c') = sim_{\tilde{c}}(c'')}} (\overrightarrow{sed}_c[c''])$;
- $\forall x \in [0..1],\ x < sim_{\tilde{c}}(c_{min}) \ \Rightarrow\ f_i(x) = 0$;
- $Seg = ||\{x : \exists c' \in \mathcal{C}_{\Omega_1} \cap \mathcal{C}_{\Omega_2}, c' \neq \tilde{c} \text{ and } sim_{\tilde{c}}(c') = x\}|| + 1$ *where Seg is the number of segments of f_i.*

Intuitively, the criterias for choosing a corresponding concept among all the possible concepts can be expressed in terms of the properties of the piecewise affine function f_i. Of course, there are as many different function f_i as candidate concepts. But the general idea is to choose the function f_i wich resembles the more to a propagation function. Let us consider the example of Figure 5 (a) and (b) where c_1, c_2, c_3 and c_6 are shared. The function in Figure 5 (a) is obtained considering c'_1 as the corresponding concept (and thus ranking the other concepts in function of their similarity with $c'1$). The function in Figure 5 (b) is obtained similarly, considering c'_2. Having to choose between c'_1 and c'_2 we would prefer c'_1 because function $f_i^{\overrightarrow{sed}_c,c'_1}$ is monotonically decreasing whereas $f_i^{\overrightarrow{sed}_c,c'_2}$ shows a higher "disorder" wrt. the general curve of a propagation function.

Several characteristics of the interpretation function can be considered to evaluate "disorder". For example, one could choose the function which minimizes the number of local minima (thus minimizing the number of times the sign of the derivated function changes). Another example is to choose the function which minimizes the variations of weight between local minima and their next local maximum (thus penalizing the functions which do not decrease monotonically). A third could combine these criteria.

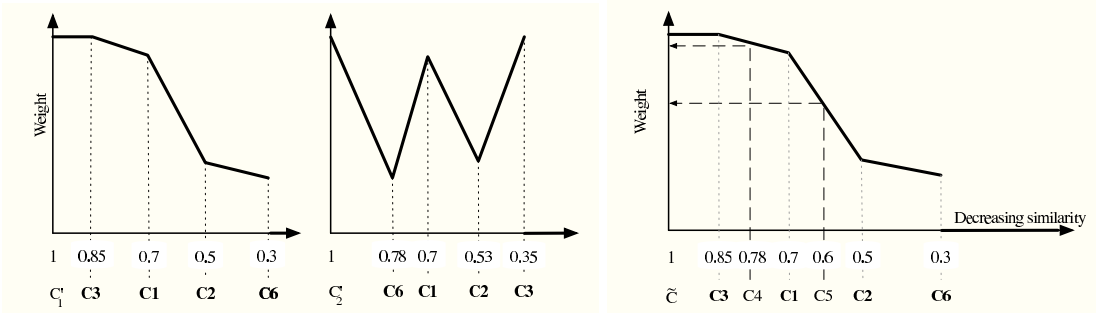


Fig. 5. Two steps of the interpretation of a SED : (a) f_i for candidate concept c'_1, (b) f_i for candidate concept c'_2 and (c) weighting the unshared concepts.

4.3 Interpreting a SED

We define the interpretation of a given SED $\overrightarrow{sed}_c$ as another SED, with central concept $\tilde{c}$ which has been computed at the previous step. We keep their original weight to all the shared concepts. The unshared concepts are weighted using an interpretation function as defined above.

Definition 5 (Interpretation of a SED). *Let $\overrightarrow{sed}_c$ be a SED on $\mathcal{C}_{\Omega_1}$ and let $\tilde{c}$ be the concept corresponding to c in $\mathcal{C}_{\Omega_2}$. Let $sim_{\tilde{c}}$ be a similarity function and let $f_i^{\overrightarrow{sed}_c,\tilde{c}}$, noted f_i, be an interpretation function. Then SED $\overrightarrow{sed}_{\tilde{c}}$ is an interpretation of $\overrightarrow{sed}_c$ iff:*

- $\overrightarrow{sed}_{\tilde{c}}[\tilde{c}] = f_i(1)$;
- $\forall c' \in \mathcal{C}_{\Omega_1} \cap \mathcal{C}_{\Omega_2},\ \overrightarrow{sed}_{\tilde{c}}[c'] = \overrightarrow{sed}_c[c']$;
- $\forall c' \in \mathcal{C}_{\Omega_2} \setminus \mathcal{C}_{\Omega_1},\ \overrightarrow{sed}_{\tilde{c}}[c'] = f_i(sim_{\tilde{c}}(c'))$;

Figure 5 (c) illustrates this definition. Document provider $p2$ ranks its own concepts in function of $sim_{\tilde{c}}$. Among these concepts, some are shared ones for which the initial SED $\overrightarrow{sed}_c$ provides a given weight. This is the case for c_1, c_2, c_3 and c_6 which are in bold face in the figure. The unshared concepts are assigned the weight they obtain by function f_i (through their similarity to $\tilde{c}$). This is illustrated for concepts c_4 and c_5 by a dotted arrow.

5 Experimental Validation

In this section, we use our approach based on *image based relevance* to find documents which are the most relevant to given queries. We compare our results with those obtained by the *cosine based method* and the *rough propagation method*. In the former method, relevance is defined by the cosine between the query and document vectors. In the latter, the effects of propagating weights from different concepts are mixed in a single vector; then relevance is obtained using the cosine.

5.1 General Setup for the Experiments

We use the Cranfield corpus, a testing corpus consisting of 1400 documents and 225 queries in natural language, all related to aeronautical engineering. For each query, each document is scored by humans as relevant or not relevant (boolean relevance). Our ontology is lightweight, in the meaning of [7], *i.e.* an ontology composed of a taxonomy of concepts : WordNet [5]. In Information Retrieval, there was a debate whether WordNet is suitable for experimentation (see the discussion in [24]). However, more recent works show that it is possible to use WordNet, and sometimes other resources, and still get good results [8]. Semantic indexing [20] is the process which can compute the semantic vectors from documents or queries in natural language. The aim is to find the most representative concepts for documents or queries. We use a program made in our lab : RIIO [3], which is based on the selection of synsets from WordNet. Although it is not the best indexing module, one of its advantages is that there is no human intervention in the process. The semantic similarity function we use is that of [2], because it has good properties and results which are discussed in Section 6. We slightly modified that function due to normalization considerations. Following the framework of membership functions presented in Section 2 we can define many propagation functions. We tested three different types of functions : "square" (of type f_{v,l_1,l_1}), "sloppy" (of type $f_{v,1,l_2}$), or hybrid (of type f_{v,l_1,l_2} with $l_1 = 2 \times l_2$). Our experiments show no important difference, but sloppy propagation has slightly better results. So we use only this propagation function, adding ten concepts in average for a given central concept.

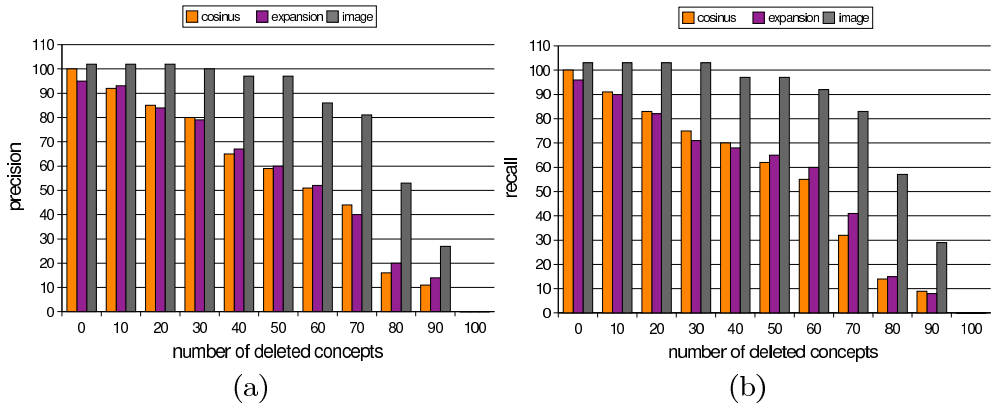


Fig. 6. Evolution of (a) precision and (b) recall in function of the random removal percentage of mappings.

In order to evaluate whether our solution is robust, we would need ontologies which agree on different percentages of concepts : 90%, 80%, 70%, . . . , 10%. This is very difficult to obtain. We could build artificial ontologies, but this would force us to give up the experiments on a real corpus. Thus, we decided to stick to WordNet and simulate semantic heterogeneity. Both the query initiator and the provider use WordNet, but we make so that they are not able to understand each other on some concepts (a given percentage of them). To do so, we remove some mappings between the two ontologies. Thus it simulates the case where the query intiator and the document provider use the same ontology but are not aware of it. It is then no more possible to compare queries and documents on those concepts. The aim is to evaluate how the answers to queries expressed with removed matchings, change. Note that the case with no removed matching reduces to a single ontology.

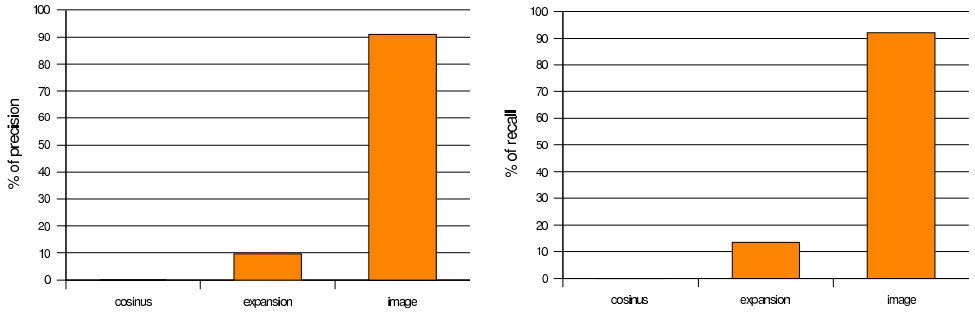
In a first experiment, we progressively reduce the number of mappings, thus increasing the percentage of removed mappings (10%, 20%, . . . until 90%). The progressive reduction in their common knowledge is done randomly. In a second experiment, we remove the mappings concerning the central concepts of the queries in the ontology of the document manager. This is now an intentional removing, which is the worst case for most of the techniques in IR : removing only the elements that match. For both experiments, we take into account the results obtained with the 225 queries of the corpus.

5.2 Results

Figure 6 shows the results obtained in average for the all 225 queries of the testing corpus. The reference method is the cosine one when no matching is removed, which gives a given reference precision and recall. Then, for each method and each percentage of removed matching, we compute the ratio of the precision obtained (respectively recall) by the reference precision. When the percentage of randomly removed matchings increases, precision (Figure 6 (a)) and recall

(Figure 6 (b)) decrease *i.e.* the results are less and less relevant. However, our "image and interpretation based" solution shows much better results. When the percentage of removed matchings is under 70%, we still get 80% or more of the answers obtained in the reference case.

In the second experiment, we consider that the document manager does not understand (*i.e.* share with the query initiator) the central concepts of the query (see Figure 7). With the cosine method, there is no more matching between concepts in queries and concepts in documents. Thus no relevant document could be retrieved. With the query expansion, some of the added concepts in the query allow to match with concepts in documents that are close to the central concepts of the query. This leads to precision and recall at almost 10%. Our image-based retrieving method has more than 90% of precision and recall in the retrieval. This is also an important result. Obviously, as we have the same ontology and the same similarity function, the interpretation can retrieve most of the central concepts of the query. But the case presented here is hard for most of the classical techniques (concepts of the query unshared) and we obtain a very important improvement.



(a) (b)

Fig. 7. Precision (a) and recall (b) when the central concepts of the query are unshared.

6 Related Work

The similarity that we use in our experiments is the result of a thorough study of the properties of different similarity measures. We looked for a similarity which is not a distance (does not satisfy similarity nor triangle inequality), based on the result of [23]. Hence we use one classical benchmark of this domain : the work of Miller and Charles [15] on the human assessments of similarity between concepts. Thirty eight students were asked to mark how similar thirty couples of concepts were. We have implemented four similarity measures: [26, 21, 12, 2], respectively noted *Wu and P.*, *Seco*, *Lin* and *Bidault* in table 1. Correlation is the ratio between those measures on the human results. The results show that only

Bidault's measure does not meet symmetry nor triangle inequality. Moreover, it obtains a slightly better correlation. Hence, it was preferred to rank the concepts according to their (dis)similarity with a central concept.

	Wu & P.	Seco	Lin	Bidault
symmetry	yes	yes	yes	**no**
triangle inequality	no	no	no	no
correlation	0.74	0.77	0.80	0.82

Table 1. Comparison of similarity measures.

The idea of query expansion is shared by several fields. It was already used in the late 1980's in Cooperative Answering Systems [6]. Some of the suggested techniques expanded SQL queries considering a taxonomy. In this paper, we do not consider SQL queries, and we use more recent results about ontologies and their interoperability. Expansion of query vectors is used for instance in [17, 24]. However, this expansion produces a single semantic vector only. This amounts to mix the effects of the propagations from different concepts of the query. Although this method avoids some silence, it often generates too much noise, without any highly accurate sense disambiguation [24]. Consequently, the results can be worse than in the classical vector space model [1]. Our major difference with this approach is that (1) the propagations from the concepts of the query are kept separate and that (2) they are not directly compared with the document. Rather, they are used to modify its semantic vector. In our experiments, our method gives better results. Also, we join [16] on their criticism of the propagation in a single vector, but our solutions are different.

Our approach also relies on the correspondences resulting from the matching of the two ontologies. Several existing matching algorithms could be used in our case [4]. In the interpretation step, we provide a very general algorithm to find the concept corresponding to the central concept of a SED. In case the concept is not shared, one could wonder whether matching algorithms could be used. In the solution we propose, the problem is quite different because the *weights of the concepts are also used* to find the corresponding concept (through the interpretation function). This is not the case in traditional ontology matching, which aim is to find general correspondences. In our case, one can see the problem as finding a "contextual" matching, the results of which cannot be used in other contexts. Because it is difficult to compute all the interpretation functions, one can use an *approximation algorithm* (for example, taking the least common ancestor as we did in our experiments). In that case, existing proposals can fit like [10, 14]. But it is clear that they do not find the best solution every time.

Finally, the word *interpretation* is used very often and reflects very different problems. However, to the best of our knowledge, it never refers to the case of interpreting a query expressed on some ontology, within the space of another ontology, by considering the weights of the concepts.

7 Conclusion

The main contribution of this paper is a proposal improving information exchange between a query initiator and a document provider that use different ontologies, in a context where semantic vectors are used to represent documents and queries. The approach only requires the initiator and the provider to share some concepts and also uses the unshared ones to find additional relevant documents. To our knowledge, the problem has never been addressed before and our approach is a first, encouraging solution. In short, when performing query expansion, the query initiator makes more precise the concepts of the query by associating an expansion to each of them (SED). The expansion depends on the initiator's characteristics: ontology, similarity, propagation function. However, as far as shared concepts appear in a SED, expansion helps the document provider interpreting what the initiator wants, especially when the central concept is not shared. Interpretation by the document provider is not easy because the peers do not share the same vector space. Given its own ontology and similarity function, it first finds out a correspondent concept for the central concept of each SED, and then interprets the whole SED. The interpreted SEDs are used to compute an image of the documents and their relevance. This is only possible because the central concepts are expanded separately. Indeed if the effects of propagations from different central concepts were mixed in a single vector, the document provider wouldn't be able to interpret the query as precisely.

Although our approach builds on several notions (ontology, ontology matching, concept similarity, semantic indexing, relevance of a document wrt a query...) it is not stuck to a specific definition or implementation of them and seems compatible with many instantiations of them. It is important to notice that there is no human intervention at all in our experiments, in particular for semantic indexing. Clearly, in absolute, precision and recall could benefit from human interventions at different steps like indexation or the definition of the SEDs. Results show that our approach significantly improves the information exchange, finding up to 90% of the documents that would be found if all the concepts were shared.

As future work, we plan to test our approach in several different contexts in order to verify its robustness. Many different parameters can be changed: similarity and propagation functions, ontologies, indexing methods, corpus...Complexity is another point that should be considered carefully. Indeed, naive implementations would lead to unacceptable execution time. Although an implementation is running for the experiments within admissible times, it could benefit from a more thorough study of theoretical complexity.

References

1. M. W. Berry, Z. Drmac, and E. R. Jessup. Matrices, vector spaces, and information retrieval. *SIAM Rev.*, 41(2), 1999.
2. A. Bidault, C. Froidevaux, and B. Safar. Repairing queries in a mediator approach. In *ECAI*, 2000.

3. E. Desmontils and C. Jacquin. *The Emerging Semantic Web*, chapter Indexing a web site with a terminology oriented ontology. 2002.
4. J. Euzenat and P. Shvaiko. *Ontology matching*. Springer-Verlag, Heidelberg (DE), 2007.
5. C. Fellbaum. *WordNet : an electronic lexical database*. 1998.
6. T. Gaasterland, P. Godfrey, and J. Minker. An overview of cooperative answering. *J. of Intelligent Information Systems*, 1(2):123–157, 1992.
7. A. Gómez-Pérez, M. Fernández, and O. Corcho. *Ontological Engineering*. Springer-Verlag, London, 2004.
8. J. Gonzalo, F. Verdejo , I. Chugur, and J. Cigarran. Indexing with wordnet synsets can improve text retrieval. In *COLING/ACL '98 Workshop on Usage of WordNet for NLP*, 1998.
9. Z. G. Ives, A. Y. Halevy, P. Mork, and I. Tatarinov. Piazza: mediation and integration infrastructure for semantic web data. *Journal of Web Semantics*, 2003.
10. G. Jiang, G. Cybenko, V. Kashyap, and J. A. Hendler. Semantic interoperability and information fluidity. *Int. J. of cooperative Information Systems*, 15(1):1–21, 2006.
11. J. Jiang and D. Conrath. Semantic similarity based on corpus statistics. In *International Conference on Research in Computational Linguistics*, 1997.
12. D. Lin. An information-theoretic definition of similarity. In *International Conf. on Machine Learning*, 1998.
13. C. D. Manning and H. Schtze. *Foundations of statistical natural language processing*. MIT Press, 1999.
14. E. Mena, A. Illaramendi, V. Kashyap, and A. Sheth. Observer: An approach for query processing in global information ssytems based on interoperation across preexisting ontologies. *Int. J. distributed and Parallel Databases*, 8(2):223–271, 2000.
15. G. A. Miller and W. G. Charles. Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 1991.
16. J.-Y. Nie and F. Jin. Integrating logical operators in query expansion invector space model. In *SIGIR workshop on Mathematical and Formal methods in Information Retrieval*, 2002.
17. Y. Qiu and H. P. Frei. Concept based query expansion. In *SIGIR*, 1993.
18. M.-C. Rousset. Small can be beautiful in the semantic web. In *International Semantic Web Conference*, pages 6–16, 2004.
19. G. Salton and M. MacGill. *Introduction to Modern Information Retrieval*. MacGraw-Hill, 1983.
20. M. Sanderson. Retrieving with good sense. *Information Retrieval*, 2000.
21. N. Seco, T. Veale, and J. Hayes. An intrinsic information content metric for semantic similarity in wordnet. In *ECAI*, 2004.
22. C. Tempich, H. S. Pinto, and S. Staab. Ontology engineering revisited: An iterative case study. In *ESWC*, pages 110–124, 2006.
23. A. Tversky. Features of similarity. *Psychological Review*, 84(4), 1977.
24. E. M. Voorhees. Query expansion using lexical-semantic relations. In *SIGIR*, Dublin, 1994.
25. W. Woods. Conceptual indexing: A better way to organize knowledge. Technical report, Sun Microsystems Laboratories, 1997.
26. Z. Wu and M. Palmer. Verb semantics and lexical selection. In *ACL*, 1994.