



LABORATOIRE D'INFORMATIQUE EN IMAGES ET SYSTEMES D'INFORMATION
LIRIS FRE2672 CNRS, INSA LYON, UCB LYON 1, EC LYON

Université Claude Bernard, Lyon1

DEA Informatique et Systèmes de Coopération des Entreprises

Ecole Doctorale Informatique et Information pour la Société

Patrick HOFFMANN

APPARIEMENT CONTEXTUEL D'ONTOLOGIES

Stage de DEA encadré par M. Lionel MEDINI et Mme Parisa GHODOUS

Soutenance le 5 juillet 2004

Jury :

M. René SOENEN

M. Yacine OUZROUT

M. Michel MARTINEZ

M. Jacques PEYRON



Remerciements

Je voudrais commencer par remercier mes encadrants Lionel MEDINI et Parisa GHODOUS, pour le temps qu'ils ont pris pour échanger et pour me corriger ; merci Catarina Eufémia FERREIRA DA SILVA pour les nombreuses références que tu nous as fournies ; merci Samer GHAFOR (qui a effectué son stage de DEA parallèlement et sur un sujet voisin du mien) de m'avoir supporté durant ces quelques mois.

Je tiens par ailleurs à remercier Celson LIMA et Bruno FIES du CSTB (Centre Scientifique et Technique du Bâtiment) pour leur engagement à nos côtés et leur disponibilité.

Enfin, je voudrais remercier mes parents et amis pour leur soutien et leurs encouragements.



Sommaire

1	Introduction	7
1.1	Contexte du stage	8
1.2	Problématique.....	8
1.3	Organisation du rapport.....	9
2	État de l'Art – Apport du Contexte pour l'Appariement d'Ontologies.....	10
2.1	Mise en correspondance d'ontologies	10
2.1.1	Terminologie	10
2.1.2	Les configurations proposées pour réaliser l'interopérabilité sémantique.....	12
2.1.3	Les différences qui posent des difficultés lors de l'appariement.....	13
2.1.4	Les techniques proposées	15
2.1.5	Quelques méthodes, outils et frameworks existants.....	18
2.2	Le Contexte ou la mise en évidence des particularités lors d'une interaction	20
2.2.1	La notion de contexte	20
2.2.2	Diverses représentations des contextes	23
2.3	Synthèse	25
3	Proposition – Plus de Contexte pour mieux Adapter l'Appariement.....	26
3.1	Détermination des données contextuelles	26
3.1.1	Trois contextes au sein d'une seule ontologie	26
3.1.2	Les questions qui permettent de cerner le contexte.....	27
3.1.3	Application aux contextes de l'ontologie.....	28
3.1.4	Interaction avec l'utilisateur.....	29
3.1.5	Application aux contextes liés à l'utilisateur	30
3.2	Interprétation des données contextuelles propres à l'ontologie	32
3.2.1	Rôle du contexte propre aux éléments de l'ontologie	32
3.2.2	Rôle du contexte propre à l'ontologie développée.....	37
3.2.3	Rôle du contexte propre à la conceptualisation qu'on cherche à représenter ...	39
3.3	Interprétation des données contextuelles propres à l'utilisateur	41
3.3.1	Rôle du contexte propre à la recherche en cours.....	41
3.3.2	Rôle du contexte propre à l'interaction avec l'utilisateur	43
3.3.3	Rôle du contexte propre à la conceptualisation de l'utilisateur	43

3.4	Complexité du système à réaliser.....	44
3.5	Méthodologie de mise en oeuvre	45
3.5.1	Sélection d'informations contextuelles	45
3.5.2	Spécifications et structuration des informations	46
3.5.3	Développement de fonctions d'interprétations	46
3.5.4	Appariement	46
3.5.5	Appariement offline	47
3.5.6	Appariement online	48
3.5.7	Définition des distances contextuelles offline et online.....	50
4	Application	51
4.1	Intégration dans une architecture plus vaste	51
4.2	Outils utilisés.....	52
4.3	Architecture logicielle.....	52
4.4	Sélection de données	52
4.5	Spécifications	53
4.6	Fonctions d'interprétation	54
4.7	Appariement offline	54
4.8	Appariement online	54
5	Conclusion.....	55
5.1	Perspectives.....	55

1 Introduction

La mondialisation impose de nouveaux modes de fonctionnement : les entreprises coopèrent davantage, avec des fournisseurs et des clients du monde entier, qui ont d'autres outils et d'autres manières de travailler. On ne s'associe plus seulement aux entreprises de la chaîne logistique, mais parfois même à des entreprises concurrentes, afin de créer une « entreprise virtuelle » pour répondre à un projet.

Il y a donc un flux croissant d'information, et un besoin de flexibilité dans la mise en place des échanges : il faut pouvoir adapter la structure de l'information afin de permettre une communication informatisée avec un nouveau partenaire aussi rapidement que possible, sans lui demander de changer de système d'information ni d'adopter un nouveau fonctionnement.

L'adoption généralisée du World Wide Web avec les standards qui lui sont associés (langages de formatage, langages de script et protocoles de communication) permet d'accroître la circulation de documents, de messages, et donne la possibilité de faire connaître son entreprise sur le plan international à coût moindre : tout système informatique peut être configuré pour accéder à Internet et communiquer. Mais les informations échangées sur le Web ne sont comprises que par les êtres humains, et ne sont pas interprétables par les machines.

Parallèlement, Le flux croissant d'information amène un changement au sein des entreprises : [Bück 2003] souligne que ce n'est plus le travail, ni les finances qui font la richesse de l'entreprise, mais la connaissance. Le besoin se fait sentir de modéliser, stocker et rendre aisément disponible la connaissance à toute personne en ayant besoin dans l'entreprise. Parmi les formalismes de représentation qui se développent, on trouve les « terminologies », les « ontologies » et les « cartes thématiques » (« *Topic Maps* »).

La vision de [Berners-Lee 1998] pour un « Web sémantique » est celle d'un Web où les informations et les connaissances sont compréhensibles par les machines. Pour représenter ces connaissances, il porte son choix sur les ontologies, qui représentent une initiative plus formelle que les cartes thématiques (voir [Vatant 2003] pour une comparaison avec OWL) et que les terminologies (voir [Gamper et al. 1999] qui suggèrent qu'elles continuent d'être utilisées conjointement aux ontologies, parce qu'elles sont plus accessibles aux utilisateurs). En février 2004, le W3C émet une recommandation, après RDF, RDFS et DAML+OIL, pour le langage d'ontologies OWL [Horrocks et al. 2003], [OWL 2004].

De son côté, [Ray 2002] propose de réaliser l'interopérabilité au niveau sémantique en s'abstrayant des données par l'utilisation de représentations de haut niveau. Ainsi, il décrit des systèmes « auto-intégrants » comme étant composés d'un protocole de négociation, d'une distance sémantique, et de capacités de raisonnement ou d'inférence.

[Chen, Doumeingts 2003] rend compte des changements d'organisation des sociétés hiérarchiques en entités autonomes et responsables ainsi que de la situation distribuée que le Web amène, qui résultent en un environnement fédéré. Il décrit une approche reposant sur la négociation à l'aide d'ontologies.

Cette négociation nécessite « *une accommodation dynamique et l'adaptation de la sémantique des deux côtés afin d'obtenir une compréhension mutuelle* ».

1.1 Contexte du stage

Ce travail est effectué en partenariat avec le Centre Scientifique et Technique du Bâtiment [CSTB], qui, dans le cadre d'un projet Européen, a réalisé une ontologie de domaine nommée e-Cognos, afin de développer l'interopérabilité dans le secteur de la construction. Parallèlement, il a développé un serveur d'ontologies appelé e-COSer pour pouvoir la faire évoluer par des experts et pour permettre l'accès à des services Web en tirant partie de la mise en relation des multiples ressources sémantiques disponibles pour le domaine.

Actuellement, e-COSer contient un outil d'aide à la reformulation de requêtes basé sur l'ontologie et sur un ensemble de règles de correspondance écrites par les experts. Un des prochains objectifs est de pouvoir interroger différents catalogues de produits du bâtiment par ce moyen.

Ce stage s'est déroulé en parallèle de celui d'un autre étudiant de DEA, Samer Abdul Ghafour, qui avait à déterminer une méthodologie de conversion des méta-modèles de représentation des connaissances en OWL, et à établir des règles de correspondance permettant de passer d'un langage à un autre. Mon travail était de trouver comment prendre en compte le contexte lors de l'appariement entre les taxonomies résultantes.

Ces deux stages ont pu bénéficier de données fournies par le CSTB, sous la forme de ressources sémantiques issues de trois projets opérationnels, qu'il s'agit de contribuer à rendre interopérables dans e-COSer. Ce partenariat est pour nous une occasion de valider notre stage dans un environnement industriel, ce qui nous assure d'une possibilité d'application pratique de nos travaux.

1.2 Problématique

L'interopérabilité sémantique entre les ontologies reste actuellement une gageure, nécessitant l'intervention humaine. [Chen, Doumeings 2003] déclarent que « *le rôle de la sémantique est fondamental* » pour y parvenir, et considèrent l'apport de la sémantique dans le cadre de la relation entre des ressources en « *interaction* ».

Parmi les différentes étapes pour atteindre cette interopérabilité se trouvent la mise en correspondance des ontologies considérées et la conception de systèmes « *auto-intégrants* », capables de tirer parti de ces correspondances [Ray 2002]. Cet auteur souligne par ailleurs la nécessité de disposer d'une mesure de « *distance sémantique* » entre ces ressources pour réaliser de tels systèmes.

Mais la relation entre ces ressources et la sémantique qui leur est associée n'est pas directe. Sans remonter aux origines de la sémiotique de Pierce, de nombreux auteurs insistent sur la nécessité de prendre en compte le contexte dans lequel est représenté un concept dans une ontologie [Ghidini, Giunchiglia 1998], [Paiva 2003], [Bouquet et al. 2003].

L'étude du contexte est donc fondamentale pour la mise en place d'une distance sémantique qui puisse *s'adapter* à la situation dans laquelle chaque concept se trouve. Dans ce cadre, les objectifs de ce travail sont d'étudier les approches de mise en correspondance sémantique des ontologies et de

caractérisation du contexte dans ces ressources, et de proposer une méthode de prise en compte de ce contexte lors de la mise en relation de ces ressources.

1.3 Organisation du rapport

Ce rapport s'organise selon quatre grandes parties. Dans l'état de l'art, nous commençons par poser des définitions pour les termes que nous employons tout au long de ce document. Puis, nous nous intéressons aux différentes approches de mise en correspondance des ontologies. Ces deux éléments nous permettent d'aborder la notion de contexte, tout d'abord dans un cadre général, puis dans celui de l'interopérabilité des ontologies.

Dans une deuxième partie, nous proposons une contribution dans le domaine de l'appariement contextuel d'ontologies. Nous précisons les paramètres à prendre en compte pour catégoriser le contexte et définissons la façon dont nous envisageons d'effectuer la mesure de distance sémantique entre deux concepts d'une même ontologie ou d'ontologies différentes, par le parcours de relations sémantiques entre ces concepts et en prenant en compte le contexte opérationnel de l'utilisateur.

Puis, nous présentons l'application dans laquelle vient s'intégrer notre démarche. Cette application regroupe l'ensemble des opérations à réaliser pour rendre sémantiquement interopérables différentes ontologies hétérogènes. Elle tire partie des travaux présentés précédemment, ainsi que de ceux de Samer Abdel Ghafour. Cette application est destinée à être intégrée au serveur d'ontologies e-COSer du CSTB. L'architecture générale de l'application est proposée, et un exemple d'implémentation qui doit permettre de vérifier la faisabilité de notre proposition est présenté.

Enfin, nous concluons en ouvrant sur les différentes voies à explorer pour utiliser pleinement les informations dévoilées par le contexte.

2 État de l'Art – Apport du Contexte pour l'Appariement d'Ontologies

2.1 Mise en correspondance d'ontologies

2.1.1 Terminologie

2.1.1.1 Ontologie : définition

Les ontologies sont une réponse proposée à la question de l'interopérabilité au niveau sémantique, mais plusieurs définitions coexistent, dont la plus acceptée est [Gruber 1993] :

« *une ontologie est une conceptualisation rendue explicite par une spécification* » (« *An ontology is an explicit specification of a conceptualization* »).

Plus simplement, on pourra comprendre l'ontologie comme étant « *une description des concepts et relations qui peuvent exister pour un agent ou une communauté d'agents* » [Gruber]. Cette phrase souligne mieux la contextualité de l'ontologie (« *une* », « *pour un agent ou une communauté d'agents* »).

2.1.1.2 Opérations de modification d'une ontologie

La réalisation d'une ontologie est une lourde tâche, et l'on cherchera donc souvent à réutiliser une ontologie existante, plutôt que de repartir de zéro ; il y aura alors un travail d'adaptation à faire :

- *Transformation* : modification de l'ontologie afin de l'utiliser dans d'autres buts que l'original, pouvant porter sur la structuration de l'information, sur le formalisme de représentation choisi mais aussi (de façon moindre : « *small, yet pervasive* » [McGuinness et al. 2000]) sur la sémantique .
- *Traduction* (« *translation* ») : changement de formalisme de représentation (langage d'ontologie utilisé) qui cherche à préserver la sémantique de l'ontologie. [Chalupsky 2000] utilise ce terme dans un autre sens, celui d'une « *transformation de la connaissance* », pour signifier une « *réécriture syntaxique* » puis « *sémantique* » (au niveau du langage de l'ontologie puis du contenu). Enfin, le terme est également utilisé pour ([Kalfoglou, Schorlemmer 2003]) décrire une transcription d'une ressource utilisant une ontologie en une autre, transcription basée sur les *correspondances* avec une seconde ontologie.

2.1.1.3 Aspects techniques du rapprochement d'éléments d'ontologies

Une fois qu'une adaptation est faite, on peut chercher à faire correspondre les éléments de plusieurs ontologies.

- *Correspondances* (« *mappings* ») : relations entre les éléments de deux représentations (ontologies, schémas de bases de données, etc.), indiquant une similarité relative selon une mesure donnée [Klein 2001].
- *Appariement* (« *matching* ») : processus de définition d'un ensemble de fonctions permettant de spécifier des *correspondances* (qui peuvent être n-aires, voir p. ex. [Xu, Embley 2003], [He et al. 2003]) entre termes.
- *Morphisme* : établissement de *correspondances* entre ontologies sans toucher à leur structure. On trouvera plus de précisions sur le morphisme d'ontologie et de signature d'ontologie dans [Kalfoglou, Schorlemmer 2003].
- *Alignement* : établissement de *correspondances* binaires (selon [Kalfoglou, Schorlemmer 2003]) entre les concepts des deux ontologies afin de parvenir à un agrément. Il semble que [Klein, 2001] et [Noy, Musen 2000] acceptent que les ontologies soient légèrement modifiées, en ce qu'ils demandent que l'alignement aboutisse à un ensemble « *pertinent et cohérent* » (ce qui est rarement possible sans modification, pour deux ontologies développées indépendamment).
- *Fusion d'ontologies* : création d'une nouvelle ontologie rassemblant la connaissance d'ontologies existantes.
- *Articulation* : ensemble des points d'ancrage des relations lors d'une mise en correspondance entre des ontologies [Klein 2001]. Dans le cas où les ontologies sources ne doivent pas être modifiées, il s'agit d'une ontologie (« *articulation ontology* ») rassemblant l'ensemble des *correspondances* (cf. [Kalfoglou, Schorlemmer 2003]). *L'articulation* fournit une mesure de la validité de l'opération effectuée.

2.1.1.4 Aspects conceptuels du rapprochement d'éléments d'ontologies

Ces termes décrivent ce qu'on veut réaliser par le rapprochement des éléments de diverses ontologies, et non comment on va s'y prendre.

- *Combinaison* : manipulation d'au moins deux ontologies pour réaliser l'une des tâches ci-dessous, dans laquelle ce qui importe avant tout, c'est leur relation l'une à l'autre.
- *Partage* : manipulation d'ontologies en vue du partage d'information (« *knowledge sharing* »).
- *Intégration d'ontologies* : construction d'une nouvelle ontologie qui n'est pas forcément destinée à remplacer les autres (celles-ci peuvent continuer à être utilisées par ailleurs). On trouvera une discussion sur le terme dans [Pinto et al. 1999].
- *Gestion de version* (« *versioning* ») : établissement de relations entre les diverses versions d'une ontologie, l'un des objectifs étant de préserver l'accès aux données référencées pour des versions précédentes de l'ontologie (voir [Klein 2001] pour un état de l'art).

2.1.2 Les configurations proposées pour réaliser l'interopérabilité sémantique

Chaque entreprise peut développer sa(ses) propre(s) ontologie(s), correspondant aux concepts qui sous-tendent les données qu'elle manipule, soit en la(les) créant de toutes pièces, soit en réutilisant des ontologies existantes. Pour parvenir à l'interopérabilité sémantique, il faut donc parvenir à un accord entre toutes ces ontologies ; on trouve dans la littérature trois grands types d'approches ([Wache et al. 2001], [Noy, Musen 2000], [Bouquet et al. 2003a], etc.) :

2.1.2.1 Approche Intégrée

Parfois nommée « *médiation centralisée* », « *ontologie globale* » ou « *ontologie unique* », elle consiste à s'accorder sur *une seule* ontologie. Cela implique un *consensus* sur le vocabulaire utilisé, sa sémantique, la granularité de l'ontologie, le point de vue, etc. Les sources d'informations sont alors toutes reliées à cette ontologie globale. Cela revient souvent à fusionner les ontologies existantes en une seule.

Cette approche est naturelle quand on n'a qu'une ontologie, ou lorsqu'on en a plusieurs pour le même domaine, de granularité proche et réalisées dans une même optique. Elle est à éviter quand l'une des ontologies qu'on veut intégrer (ou l'une des sources de données) contient des données hétérogènes et/ou évolue de manière indépendante : on doit alors s'attendre à devoir régulièrement modifier l'ontologie globale, et également les autres sources de données.

2.1.2.2 Approche Fédérée

Parfois nommée « *médiation décentralisée* », « *médiation distribuée* » ou « *ontologies multiples* », elle consiste à considérer les ontologies comme étant des représentations approximatives représentant le point de vue d'une communauté ou d'un individu [Bouquet et al. 2003a], et à les faire correspondre l'une à l'autre. On a donc une ontologie par source de données, et chaque ontologie est indépendante.

Cette approche permet une plus grande flexibilité : elle permet d'utiliser des ontologies évoluant de manière autonome et mises à jour fréquemment. Elle n'est pas bloquante si l'une d'elles vient à disparaître. On peut facilement supprimer une source (il suffit de supprimer les correspondances avec l'ontologie locale). Mais l'appariement entre représentations autonomes est difficile, requiert parfois l'utilisation de ressources linguistiques spécialisées, et nécessite jusqu'à présent l'intervention d'experts du domaine : actuellement, aucun outil ne permet la résolution automatique de l'appariement. Parmi les principales difficultés, les cas de synonymie et d'homonymie mais surtout l'ambiguïté due à un manque d'information (cf. [Doan et al. 2002]).

Une variante considère à accorder plus ou moins de crédit aux différentes ontologies (à donner raison à l'une plutôt qu'à l'autre en cas de conflit).

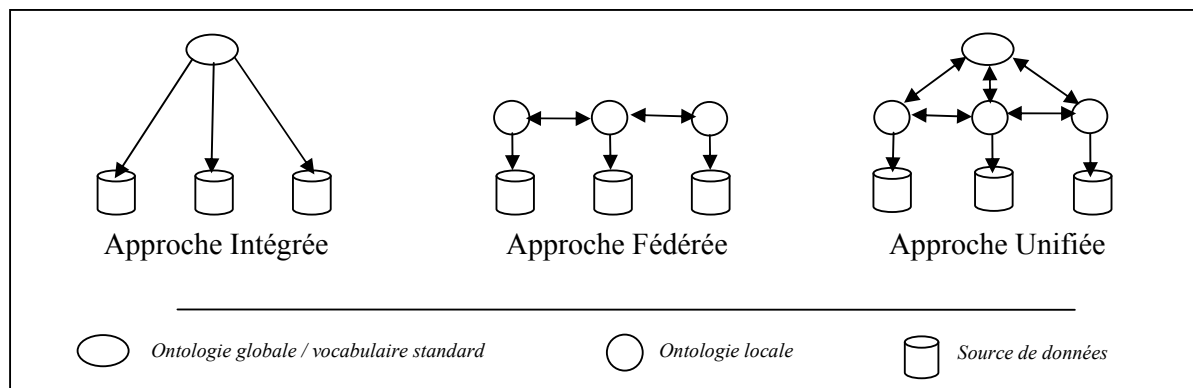
2.1.2.3 Approche Unifiée

Parfois nommée « *hybride* » ou « *ontologie de plus haut niveau* » (« *upper level ontology* »), c'est un compromis entre les deux approches précédentes : elle consiste à établir des correspondances entre les ontologies locales (une ontologie par source), et à établir pour chacune d'elles des correspondances

avec une unique ontologie de plus haut niveau (voir pour l'exemple « *Standard Upper Ontology* » [SUO])

Comme pour l'approche précédente, sources et ontologies peuvent être développées de manière entièrement autonomes, seules les correspondances sont à mettre à jour en cas d'évolution d'une ou de plusieurs ontologies. C'est cette approche que le CSTB a choisie pour faciliter la coopération entre entreprises du secteur du bâtiment (voir la partie « Application »).

Cette approche est surtout intéressante si les ontologies se conforment à un certain standard ; ainsi, [Wache et al. 2001] proposent que toutes les ontologies locales soient décrites à l'aide d'un *vocabulaire partagé* (qui peut être une ontologie) comprenant les termes basiques du domaine : de nouveaux termes, plus complexes, pourront être créés à partir de combinaisons des premiers, à l'aide d'opérateurs spécifiques. Cette approche requiert cependant de commencer par créer le vocabulaire commun, ainsi que les règles de combinaison des termes. De plus, si les sources sont indépendantes, ce n'est pas le cas des ontologies, qui doivent utiliser le langage commun (il faut donc un *consensus* au préalable, ou construire soi-même les diverses ontologies correspondant aux différentes sources).



Les approches que nous venons de décrire correspondent aux trois approches « Intégrée », « Fédérée » et « Unifiée » pour mettre en oeuvre l'interopérabilité décrites par le standard ISO 14258 (Concepts et règles pour la modélisation d'entreprises) dont on trouvera une présentation dans [Chen, Doumeingts 2003]. à propos de l'approche unifiée, les auteurs déclarent qu'elle impose l'utilisation d'un mécanisme de correspondance au niveau méta, et que la standardisation n'est pas nécessaire mais bienvenue, parce qu'elle réduit le nombre de correspondances à effectuer.

2.1.3 Les différences qui posent des difficultés lors de l'appariement

Pour faire correspondre les différentes ontologies, deux étapes sont nécessaire : s'abstraire de la différence entre langages d'ontologies utilisés (par exemple en traduisant les ontologies dans un même formalisme de représentation), puis chercher les concepts équivalents à appairer en tenant compte des différences de conceptualisation, de description de cette conceptualisation et de terminologie.

Nous passons assez vite la première étape (on trouvera plus de précision dans [Klein 2001]); les difficultés qui se posent sont les différences de *syntaxe* (cf. [Chalupsky 2000]), de *représentation des notions logiques*, de *l'homonymie possible entre primitives* de langage de signification différente. Voir [Abdul Ghafour 2004] pour une étude sur ce point.

Enfin, *l'expressivité* varie beaucoup entre langages : tous les langages ne peuvent exprimer la négation, le support des ensembles, des valeurs par défaut, etc. Ainsi, le nombre de sièges habituel dans une voiture pourra faire l'objet d'une valeur par défaut, ou nécessiter une nouvelle variable :

Loom	ANSI KIF
<pre>(defconcept Voiture :is-primitive Véhicule :defaults (:nombrePassagers 5))</pre>	<pre>(defrelation Voiture (?x) :=> (and (Véhicule ?x) (nombrePassagersTypique ?x 5)))</pre>

2.1.3.1 Différences de conceptualisation

- *Portée* : deux classes peuvent sembler représenter le même concept, mais elles n'ont pas exactement les mêmes instances. Par exemple, toutes les administrations ont la même compréhension du terme « employé », pourtant dans la pratique un employé dans une entreprise aura d'autres droits et devoirs que dans une autre.
- *Recouvrement* des ontologies peuvent *recouvrir* seulement une partie d'un sujet ou bien l'ensemble.
- *La granularité* avec laquelle les concepts sont décrits peut être fine comme très grossière.

Comme exemples de différences de recouvrement et de granularité, on pourra trouver une ontologie sur les vins Français, une sur les meilleurs vins rouges du monde entier, une autre encore catégorisant les vins de Bordeaux, indiquant les cépages, les meilleures années, le taux de tanins, etc.

2.1.3.2 Différences de modélisation

- Les *paradigmes* utilisés peuvent être différents ; on pourra trouver des ontologies qui décrivent un cercle par un point et un rayon, d'autres par un ensemble de trois points.
- Les *conventions de modélisation* peuvent varier : on peut distinguer deux types de classes, on peut ajouter un attribut à la classe ou créer une nouvelle classe.
- L'ontologie peut avoir été écrite de sorte à permettre l'inférence à l'aide d'un *outil particulier*. Par exemple, dans Loom, certaines inférences sont bien supportées, tandis que d'autres ne le sont qu'au niveau des instances.

2.1.3.3 Différences terminologiques

- Le même concept peut être représenté par différents termes *synonymes*, un cas particulier étant le cas des ontologies basées sur un autre langage d'ontologies. Dans tous les cas, si le concept semble le même, il faut veiller à la différence de portée sous-jacente à l'utilisation de chacun des termes.
- Au contraire, le même terme peut représenter différents concepts ; *l'homonymie* est un problème qui nécessite bien souvent l'intervention humaine.
- Finalement, *l'encodage des données* au sein de l'ontologie diffère bien souvent, que ce soit pour les dates, les unités (monnaie, distances, ...).

On trouvera une liste plus détaillée et couvrant l'ensemble des problèmes posés par la combinaison d'ontologies dans [Chalupsky 2000] et [Klein 2001].

2.1.4 Les techniques proposées

Si un grand nombre d'appariements sont encore réalisés manuellement, par des experts, ce ne sera bientôt plus possible tant le nombre et la taille des ontologies augmentent. L'utilisation d'un outil graphique semi automatique (suggérant différents appariements) permet de minimiser l'intervention humaine et d'accélérer le traitement.

Des travaux ont été menés sur ce front dans différents contextes : pour la traduction et l'intégration de schémas, dans le domaine de la représentation de la connaissance, pour l'apprentissage automatique et la recherche d'information (« Information retrieval »), plus récemment pour l'alignement et la fusion d'ontologies.

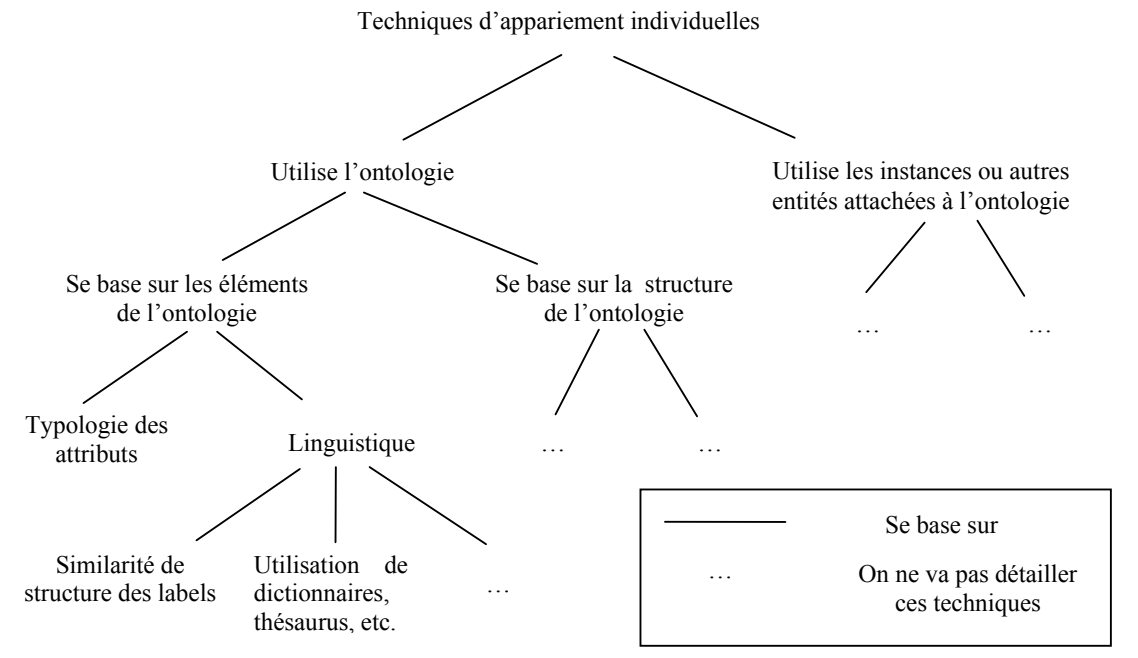
Parmi les premières recherches à ce sujet, [Rahm, Bernstein 2001] jugent que beaucoup ont été motivées par l'intégration de schémas de bases de données : l'appariement (« *matching* ») de schémas est la première étape vers l'intégration, permettant ainsi de récupérer les données dans la vue globale. L'appariement entre schémas sert également pour traduire des données sources dans le format interne d'un entrepôt de données, pour traduire les messages lors des échanges du e-commerce, etc. C'est seulement depuis 1999-2001 que des travaux similaires commencent à émerger dans le domaine des ontologies, notamment avec une publication de N. F. NOY et M. A. MUSEN en 1999, [McGuinness et al. 2000] et [Noy, Musen 2000].

La plupart des algorithmes ont fait l'objet d'une implantation spécifique et seuls Cupid [Madhavan et al. 2001], Clio, COMA [Do, Rahm 2002]) et Similarity Flooding [Melnik et al. 2002], ont cherché à rendre leurs travaux davantage accessibles pour l'ensemble des domaines. On évoquera tout de même les apports d'autres propositions, car les techniques d'appariement dépendent beaucoup plus du type d'information qu'elles exploitent que du langage de représentation sur lequel elles se basent [Rahm, Bernstein 2001].

2.1.4.1 Classification des principales techniques d'appariement

Le graphe ci-dessous montre les principales techniques individuelles utilisées pour l'appariement des ontologies ; ces techniques peuvent être ensuite utilisées ensembles en un algorithme « *hybride* » (deux techniques dans un même algorithme) ou en un paramétrage d'algorithmes exécutés indépendamment (« *combinés* »).

La classification produite est tirée de [Rahm, Bernstein 2001] et [Euzenat, Valtchev 2003] (pour son application aux ontologies).



2.1.4.1.1 Techniques linguistiques

Les *labels*¹ peuvent être comparés, après normalisation et parfois concaténation avec le chemin depuis la racine. Les labels sont plus souvent renseignés que les autres types d'annotation, c'est donc une source d'information assez fiable. C'est souvent la première donnée observée lors de l'appariement ; sans cette première indication sur le concept représenté, aucun algorithme ne peut réaliser d'appariement convenable : comment savoir s'il s'agit d'une ontologie du bâtiment ou d'un constructeur automobile ?

C'est généralement à partir des labels qu'on fera appel aux *ressources auxiliaires* : dictionnaires de synonymes, d'hyponymes, etc., mais aussi les thésaurus et des ressources sémantiques telle que [WordNet] et des dictionnaires spécifiques aux domaines étudiés.

Les *commentaires*² peuvent servir également, après traitement par des algorithmes de Traitement Automatique des Langues (TAL).

¹ La traduction littérale de la définition du W3C est : « *propriété pouvant être utilisée pour fournir une version d'un nom de ressource destinée à être lue par les humains* » (http://www.w3.org/TR/rdf-schema/#ch_label). En pratique un label désigne le nom associé à une ressource, dans un langage donné. Cf. méta-modèle OWL réalisé par Samer ABDUL GHAFOR en pages annexes.

² La traduction littérale de la définition du W3C est : « *propriété pouvant être utilisée pour fournir une description de ressource destinée à être lue par les humains* » (http://www.w3.org/TR/rdf-schema/#ch_comment). En pratique un label désigne le nom associé à une ressource, dans un langage donné. Cf. méta-modèle OWL réalisé par Samer ABDUL GHAFOR en pages annexes.

2.1.4.1.2 Typologie des attributs

La cardinalité des attributs d'une classe, l'ensemble de valeurs possibles associé peuvent aider à faire correspondre deux classes ; on trouvera beaucoup de travaux similaires relatifs au schémas, pour lesquels l'appariement entre attributs des tables se fera beaucoup d'après leur type de données.

2.1.4.1.3 Structure

L'appariement de graphes utilise fréquemment des ressemblances structurelles pour détecter des possibilités d'appariements. On pourra voir notamment les travaux sur la transformation de graphes.

2.1.4.1.4 Instances et autres entités rattachées à l'ontologie

Deux approches existent pour comparer les ontologies à partir des instances qui leur sont associées :

- soit considérer que les deux ontologies à comparer référencient les mêmes instances,
- soit faire des recherches de mots-clés dans ces mêmes instances (souvent des documents ou autres fichiers). On testera ensuite la similitude entre les instances à l'aide de ces mots-clés.

On apparie ensuite les classes qui sont liées aux instances semblables.

Les instances peuvent aider à résoudre (ou au moins à dénoncer) les problèmes de « portée ». Réutilisons l'exemple donné précédemment : le concept « employé » peut être relié à des documents sur les droits des employés d'une part et d'autre, mais pour l'une des ontologies, il peut être relié aussi à des documents indiquant l'usage des chèques-restaurant, les limites d'utilisation de la voiture de service, les informations sur la mutuelle proposée par l'entreprise, etc. On saura donc que l'employé n'a pas les mêmes droits dans les deux entreprises.

2.1.4.2 Quelques idées proposées

L'appariement entre ontologies ou schémas a été l'objet de multiples travaux, voyons ici quelques approches intéressantes et/ou originales :

- [Madhavan et al. 2001] proposent un *parcours des feuilles vers la racine* (« bottom-up »), qui est plus lourde que l'approche inverse mais peut permettre de comparer des schémas de structure très différentes et d'y trouver des candidats pour établir des correspondances.
- [He et al. 2003] utilisent la *corrélation* en appariant non pas deux, mais n schémas à la fois.
- [Rahm, Bernstein 2001] proposent de disposer de *librairies de fragments* de schémas (c'est tout aussi envisageable pour les ontologies).

2.1.4.3 Synthèse et perspectives

Beaucoup de travaux portent sur l'appariement, et un grand nombre de techniques ont vu le jour, mais le simple problème de l'équivalence concentre la majorité des efforts effectués : en effet, les algorithmes récents parviennent souvent à proposer des équivalences qui seront acceptées à 70%-

100% des cas par l'utilisateur, selon les ontologies à faire correspondre, il y a donc encore un effort à faire pour atteindre 90% de justesse pour toutes les ontologies.

Il apparaît que certains problèmes évoqués plus haut n'ont pas trouvé encore de solution par l'étude directe de l'ontologie. Nous proposons ici quelques éléments de réponse.

- Aucune recherche n'a encore résolu le problème d'*expressivité* des langages d'ontologies. Mais si l'on peut traduire toutes les ontologies dans un langage possédant l'expressivité de tous les autres, on aura trouvé un moyen de réaliser un « pivot sémantique » favorisant l'interopérabilité.
- La *couverture* des ontologies est très rarement la même ; il ne faut donc pas chercher à mettre en correspondance tous les éléments.
- Les différences de *paradigmes*, tout comme les différentes *options de modélisation*, peuvent être classifiées dans une table de correspondance.
- Les différences de terminologie due à *l'adaptation à un outil* (d'inférence, ...) ont été très peu pris en compte (sinon pas du tout).
- *L'encodage des données* peut faire l'objet d'une traduction préliminaire automatisée.

2.1.5 Quelques méthodes, outils et frameworks existants

On pourra être surpris de trouver ici des outils utilisés pour réaliser des appariements entre schémas de bases de données ; cependant ces outils se focalisent sur les techniques utilisées pour réaliser l'appariement, et non sur l'objet lui-même, comme c'est souvent le cas avec les outils proposés dans le domaine des ontologies. Certes, le problème n'est pas tout à fait identique, mais certains aspects des démarches proposées peuvent être réutilisés.

Nom	Référence	Description	Appariement	Autres commentaires
Chimerae	[McGuinness et al. 2000]	outil interactif de fusion d'ontologie; permet le diagnostic/test (et l'édition)	Chimerae aide l'utilisateur à trouver le bon terme en lui proposant une liste des termes utilisés (et aide à résoudre les difficultés d'ordre terminologique), fait apparaître les problèmes de portée .	environnement basé sur le Web. Effectue la traduction au niveau langage depuis plusieurs formalismes. Chimerae utilise des heuristiques pour trouver les parties de l'ontologie à réorganiser.
OntoMorph	[Chalupsky 2000]	système de traduction et de transformation d'ontologies	OntoMorph utilise la plupart des techniques d'appariement disponibles. Les deux seuls problèmes qu'il n'aide pas à résoudre sont les problèmes d'expressivité au niveau langage, et les problèmes de couverture différente du domaine de l'ontologie.	Ontomorph emploie deux mécanismes de réécriture : syntaxique (pattern matching influencé par PLisp) et sémantique (basé sur PowerLoom, système de représentation de la connaissance permettant l'inférence). OntoMorph est très proche de Chimerae.
Prompt (anciennement SMART)	[Noy, Musen 2000]	Outil interactif (et méthode) de fusion et d' alignement d'ontologies	Prompt commence par chercher les appariements possibles par similarité linguistique , mais se base surtout sur la structure et vérifie les actions de l'utilisateur pour détecter des <i>conflits</i> éventuels.	proche de Chimerae et d'OntoMorph, Prompt est un module dans l'éditeur d'ontologies [Protégé]. Il permet des mises à jour automatiques, se rend compte de conflits et propose une aide pour les résoudre
FCA-Merge	[Stumme, Maedche 2001]	Méthode pour la fusion d'ontologies	fait une comparaison basée sur les instances : FCA-Merge utilise des techniques linguistiques ; <i>relie les termes clés des instances avec les concepts de l'ontologie</i> (utilise le dictionnaire) tient compte de la position des éléments dans la taxonomie	Il transforme les ontologies (parcours des feuilles vers la racine) en des contextes formels (techniques d'analyse de concepts formels) puis ajoute les termes extraits des documents instances pour former un treillis que l'expert explore.
Similarity Flooding	[Melnik et al. 2002]	Algorithme permettant l' appariement de sources génériques	distance entre termes, tient compte de la structure (l'appariement de deux concepts influe positivement leur voisinage). L'appariement est robuste aux cycles.	La source est convertie en un graphe labellé. Pour chaque appariement réalisé, la similarité augmente pour les paires voisines. Une mesure juge de la qualité des appariements
Anchor-prompt	[Noy, Musen 2001]	Outil pour trouver des appariements entre ontologies	compare les labels, la typologie des attributs, la structure. (considère qu'un appariement est plus fréquent sur un 'chemin' où il en existe déjà : « contexte non local »)	Nécessite l'introduction préalable d'un ensemble d'appariements pour divers concepts des deux ontologies.
COMA	[Do, Rahm 2002]	Système générique interactif permettant de trouver des appariements entre schémas	combine 13 algorithmes (implémentant les techniques pour l'appariement) : compare les labels, leur similarité phonétique, compare la structure, la typologie des attributs, vérifie les synonymes. Utilise un historique des appariements effectués.	La librairie des algorithmes utilisée est extensible. L'utilisateur peut interagir pour sélectionner les algorithmes à utiliser et comment les combiner.
GLUE	[Doan et al. 2002]	Système permettant de trouver des appariements entre ontologies	Utilise les informations sur les documents instances (nom, taille, ...) et sur la fréquence des mots contenus : permet de prendre en compte le sens commun, les contraintes du domaine et la structure (prise en compte du voisinage)	met en oeuvre trois agents basés sur différentes stratégies d'apprentissage (adaptées au type d'information à acquérir) : un pour les noms, un pour les concepts, un pour combiner les deux approches (de manière probabiliste)
Cupid	[Madhavan et al. 2001]	Système générique permettant de trouver des appariements entre schémas	Approche hybride ; compare les labels (thésaurus, etc.) , la typologie des attributs, la structure (transforme le graphe en un arbre pour préserver le 'contexte', défini par le chemin depuis la racine),	Parcourt les ontologies des feuilles vers la racine, pour donner plus d'importance aux feuilles, afin de pouvoir appairer des schémas dont la structure intermédiaire varie mais peu.

On pourra se reporter à [Klein 2001], [Kalfoglou, Schorlemmer 2003] pour davantage de précisions. [Bao 2003] propose sous « Data/Ontology integration Systems » une liste de liens vers ces outils et encore une vingtaine d'autres.

On pourra se reporter à [Do et al. 2002] pour une évaluation des systèmes recherchant à appairer des schémas (dont COMA, Cupid, GLUE et Similarity Flooding).

2.2 Le Contexte ou la mise en évidence des particularités lors d'une interaction

Les ontologies sont LA solution pour parvenir à l'interopérabilité sémantique, pour parvenir à une entente entre différents services automatisés sur le Web. Et pourtant, de plus en plus, on se rend compte qu'on n'arrivera pas toujours à un consensus. Au contraire, un grand nombre d'ontologies verra le jour sur un même sujet, mais elles diffèrent, non seulement au niveau des choix terminologiques ou de la langue choisie, mais également au niveau conceptuel : il sera nécessaire de traduire les concepts sous-jacents ; parce que les points de vue seront différents ; parce que le contexte sera différent.

2.2.1 La notion de contexte

C'est une notion bien connue en psychologie cognitive, philosophie du langage, linguistique, mais elle n'a fait son apparition dans le domaine de l'Intelligence Artificielle que depuis les années 80, avec les travaux de R. W. Weyhrauch, John McCarthy et R.V. Guha. Il semble cependant que c'est à Chambéry, en 1993, lors de la 13^{ème} conférence conjointe internationale sur l'Intelligence Artificielle, qu'il est apparu qu'il était important de considérer le contexte, avec un article de F. Giunchiglia et la proposition de John McCarthy pour introduire les contextes en tant qu'objets formels.

Le mot lui-même fait réellement partie de notre langage courant, et la prise en compte du concept qu'il représente est l'objet d'étude dans différents domaines et selon divers points de vue. Nous allons cependant n'aborder qu'une petite partie du sujet, en nous focalisant sur un certain nombre d'études faites à ce sujet dans le domaine de l' Intelligence Artificielle (IA), particulièrement celles qui ont porté à appliquer les résultats obtenus dans le domaine de la représentation des connaissances (dont les ontologies sont un sujet de recherche).

2.2.1.1 Qu'est-ce que le contexte et à quoi sert-il ?

Le dictionnaire Collins définit le contexte comme « les conditions et circonstances significatifs pour un événement, un fait, etc. ». C'est donc l'ensemble des informations qui donnent à une situation sa *particularité*. Il s'agira notamment de replacer l'action dans un cadre spatio-temporel, et de déterminer les règles et les faits qui sont constamment vérifiés (la précision de la mesure permettant la comparaison étant à déterminer selon l'influence entendue que pourrait avoir un paramètre sur l'action) pour cette situation. Par abus de langage, on parlera de contexte d'un objet pour décrire le contexte qui entoure son existence (parfois pour la seule période de l'étude).

Le contexte serait-il alors simplement la réunion des caractéristiques particulières de la situation ? Oui, et le mot fondamental, c'est « particulières » : cela implique une comparaison. Une situation n'est

donc contextuelle qu'à partir du moment où elle peut être mise en relation avec une situation semblable mais où les règles sont différentes. En résumé, le contexte prend naissance à la mise en relation de deux ensembles d'informations ; placer un objet dans son contexte, c'est le placer en relation par rapport à l'autre, aux objets auxquels il est relié, et non plus le considérer pour ce qu'il est lui-même.

Il y a plusieurs niveaux, ou couches, de contextes, car plusieurs étapes, plusieurs niveaux de détail dans la description de chaque information. Par exemple, si l'on me demande où j'habite, je répondrai différemment selon que mon interlocuteur habite mon village, mon département, mon pays, etc. ; en effet, dans le premier cas, je répondrais « j'habite dans la rue des hirondelles, près de l'église », dans le deuxième cas, « j'habite à Saint-Symphorien d'Ozon, au sud de Lyon », dans le troisième cas « j'habite près de Lyon ».

On voit donc que jamais je ne précise que j'habite dans le système solaire, mais qu'à chaque fois je m'adapte à mon interlocuteur et à ce que je pense qu'il connaît déjà sur moi, à ce qu'il peut comprendre, aux références que je pense qu'il doit posséder (« près de l'église », « au sud de Lyon »). On notera que lors de l'interaction entre contextes, une donnée peut être omise pour une autre raison : par manque de connaissance à son sujet. Dans le cadre de mon exemple, je dois avouer que je n'ai encore rencontré personne habitant ailleurs que sur le globe terrestre.

De même que les êtres humains se servent du contexte pour communiquer, parce que cela leur permet de se concentrer sur l'essentiel, sur l'information qui est nouvelle ([Brézillon 1998]), en optimisant ainsi temps et réduisant les besoins en mémoire. De même, en IA, on associe à l'étude du contexte des moyens d'interagir entre contextes : [McCarthy 1993], [Brézillon, Pomerol 1999] parlent respectivement de relations, de partage entre contextes, et essaieront de définir des « bridge rules » (« règles de liaison »), « lifting rules » (« règles de changement de niveau ») afin de passer d'un contexte à un autre, soit étranger, soit englobant ou inclus. Dans tous les cas, le contexte est étudié dans le but d'une confrontation entre contextes.

L'IA ne cherche pas à définir ce qu'est réellement le contexte (elle n'y parviendrait pas) mais elle décrit souvent le contexte comme étant un modèle local représentant un point de vue sur un domaine. [Guha, McCarthy 2003] cherche à catégoriser le contexte suivant le type de règle qui permet de passer à un autre contexte : il les nomme « *contexte de projection* », « *contexte d'approximation* » et « *contexte d'état mental* ».

Plusieurs branches de l'IA s'intéressent au contexte, que ce soit la communication Homme Machine, la reconnaissance d'images (pour obtenir des algorithmes plus robustes), les systèmes à base de connaissances (échec relatif des systèmes experts, notamment parce qu'ils ne prennent pas en compte le contexte).

2.2.1.2 Différentes perceptions du contexte

La notion de contexte est une notion abstraite et qui ne peut être définie une fois pour toute et étudiée seule, car elle n'a de sens que reliée à son objet. L'être humain enregistre inconsciemment le contexte de ses actions, qui lui revient ensuite en mémoire comme un ensemble structuré. Il pourra également associer à un contexte donné, souvent de manière inconsciente, une réaction, une attitude et un ensemble d'actions qu'il accomplira systématiquement, sans en comprendre la raison.

L'ingénieur aura une vision limitative du contexte : son but est de recueillir les informations qui concernent son problème, afin de fixer un espace de résolution ; il définira le contexte comme l'« ensemble des conditions et influences environnantes qui font de la situation une situation unique et permettent de la comprendre. » [Brézillon, Pomerol 2001] ; Il s'agit donc d'un souvent d'un contexte statique, puisque la situation est définie au moment où il détermine son problème, et n'évolue généralement pas.

Le cognitif, quant à lui, s'intéresse aux interactions et aux évolutions du contexte, il obtiendra donc un modèle dynamique. Cette différence de perception dépend du niveau du contexte observé : ainsi, si la mesure qu'un ingénieur utilise pour ses observations devient plus fine – en raison d'une augmentation des contraintes, par exemple – il n'obtiendra plus une valeur unique, moyenne, mais une fluctuations de valeurs. Si plusieurs paramètres sont en jeu (on peut penser aux contraintes portées sur un avion en vol), il peut devenir nécessaire d'établir des algorithmes permettant de prendre en compte les différentes interactions entre paramètres qui peuvent se produire.

2.2.1.3 Historique de la prise en compte du contexte en Intelligence Artificielle

Divers propositions ont été faites pour représenter le contexte comme un objet de première importance dans un système logique. On peut diviser celle-ci en deux catégories : les systèmes basés sur la logique du premier ordre et les systèmes à base de propositions, les plus nombreux, sur lesquels nous allons nous focaliser.

L'apport que l'on attend de la formalisation du contexte est multiple : Pour F. Giunchiglia, de l'équipe de Trento en Italie, le contexte est un outil pour permettre de formaliser la localité du raisonnement ; pour John McCarthy, il s'agit davantage d'introduire par là la notion de sens commun, et d'avoir plusieurs espaces contextuels dans lesquels une même assertion peut être validée ou non. L'essentiel de [McCarthy 1993] ne vise pas à définir le contexte (il le dit lui-même) mais est tourné vers les relations entre concepts (section 2) et les règles pour passer d'un contexte à un autre (section 3 et suivantes).

Pour Guha (qui a soutenu sa thèse à Standford en 1991 sous la direction de J. McCarthy), il existe un langage global dans lequel on peut exprimer tout ce qui peut l'être dans divers langages et différents contextes. C'est certainement ce qui a été la base pour l'intégration de la notion de contexte dans la plus grande base de sens commun, CYC, sous la forme de « micro langages » (« Microtheories »). La formalisation du contexte permet ici de rassembler des informations de provenances diverses, mais également de limiter la quantité de mémoire requise pour stocker les millions d'assertions (1,5 millions en fin 1991).

[Buvac, Mason 1993] propose alors une formalisation partielle des travaux de Guha avec le système PLC (« Propositional Logic of Context »). Parmi les systèmes à base de propositions proposé à cette époque, on aura celui de P. Nayak en 1994, suivi par F. Massacci en 1995, avec une version revisitée de PLC. Ces divers travaux sur le contexte donnent lieu à de nouveaux résultats dans plusieurs domaines de recherche en IA, dont celui de la gestion des points de vue, du raisonnement à base de croyances, et des systèmes multi agents.

En 1999, l'équipe de Trento réalise un travail remarqué et réellement différent (système qui n'utilise pas la modalité) avec [Ghidini, Giunchiglia 1998] décrivant un système LMS (« Local Models Semantics », « sémantique de modèles basés sur la localité »), parfois nommé LMS/MCS (MCS pour

«Multi Context System», soit «Système pour prendre en compte plusieurs contextes»). Les auteurs insistent sur la localité du raisonnement, mais également, cette fois, sur la question de compatibilité entre les raisonnements : à partir d'un exemple sur une «boîte magique», ils montrent que si deux personnes voient la boîte d'un côté différent sans se rendre compte de la profondeur, ils verront chacun autre chose, et ne pourront pas distinguer (même ensemble) l'ensemble des possibilités.

[Bouquet, Serafini 2001] font une comparaison entre PCL et LMS et montrent des points communs pour les deux principaux systèmes : des «faits» sont entrés dans l'un ou l'autre contexte (ou déduit de l'un à l'autre) ; l'évaluation d'un fait dépendra alors en général du contexte dans lequel on se trouve (tous les faits sont considérés comme dépendant du contexte). Les contextes permettent de formaliser le sens commun plus facilement ; enfin, chaque système inclut une possibilité de raisonner entre contextes, mais souvent l'inférence d'une connaissance d'un contexte dans un autre n'est possible qu'à travers un troisième contexte. [Bouquet, Serafini 2003] poursuit le travail effectué en démontrant la plus grande généralité au sens strict de LMS/MCS par rapport à PLC.

On trouvera une comparaison des systèmes logiques à base de propositions pour représenter le contexte dans [de Paiva 2003], ainsi qu'une bibliographie plus complète sur le sujet.

2.2.2 Diverses représentations des contextes

Dès 1995, les premiers résultats sur le contexte servent de base à des travaux sur l'acquisition de données, l'intégration de sources de données hétérogènes, etc. Depuis, divers travaux sont venus s'ajouter dans le domaine. Parallèlement, les langages utilisés pour représenter les connaissances évoluent, et des tentatives sont faites pour adapter la notion de contexte à ces langages.

Les hiérarchies (arbres étiquetés, etc.) sont beaucoup utilisés. Les graphes conceptuels sont également souvent à l'honneur ; [Rivière 1998] utilisera par exemple ce formalisme pour représenter une gestion des points de vue adaptée au domaine de l'ingénierie concurrente. On rappelle que la notion du point de vue, élaborée dans les années 70, est essentielle pour le raisonnement contextuel, plus peut-être encore que le raisonnement à base de croyances.

L'utilisation de plus en plus importante des langages d'ontologies à base de logiques de description, et leurs caractéristiques (possibilité d'utiliser plusieurs types de relation pour relier des concepts, moteur d'inférence, extensibilité aisée...), en font également des langages privilégiés.

Ainsi, A. Delteil et C. Faron-Zucker ont tenté d'intégrer des balises contextuelles au niveau (uniquement) des annotations dans RDF(S). Les «cartes thématiques» («Topic Maps»), qui ne sont pas réellement un langage d'ontologies, intègrent un paramètre «scope» («portée»), à propos duquel [Vatant 2003] dénonce le manque de formalisation, résultant en diverses interprétations.

2.2.2.1 Mise en correspondance de contextes

Nous avons vu que le contexte est la réunion des caractéristiques particulières d'une situation. Comme une situation peut être comparée à une multitude d'autres, une représentation du contexte aura besoin d'avoir accès à une représentation de la connaissance contenant une réunion de caractéristiques (pouvant être particulières) concernant la situation. Il est donc tout à fait envisageable de considérer que le contexte puisse inclure une ontologie.

Nous avons également vu à plusieurs reprises que la relation entre contextes est un point primordial pour les différents auteurs, un point clé et difficile : « le principal problème » selon [Guha, McCarthy 2003] (« *the central issue in modeling contexts is that of lifting* »). On ne pourra donc parler d'une représentation du contexte que s'il est fourni un moyen d'échanger des informations entre contextes.

On notera que bien souvent, les ontologies sont utilisées comme un modèle partagé, sont établies sur la base d'un consensus, pour permettre aux différents participants d'avoir un même « vocabulaire » pour communiquer. Par opposition, les contextes sont des modèles locaux, réalisés de manière autonome et exprimant le point de vue d'une communauté ou d'un individu. On considérera une ressource ontologique comme un contexte lorsqu'il y aura nécessité ou obligation de préserver son autonomie, et qu'on pourra se limiter à une mise en correspondance forcément moins achevée (par exemple dans la gestion de la connaissance distribuée, la classification de documents, l'intégration de catalogues, ...).

2.2.2.2 Contexte et ontologies

[Benslimane et al. 2003] proposent de représenter les points de vue divers sur un même sujet au sein d'une même ontologie (basée sur un langage tel que DAML, OIL, ...) , au lieu d'avoir plusieurs ontologies séparées. Pour cela ils définissent une « ontologie contenant plusieurs représentations » (« MUltiRepresentation Ontology MuRO»), en s'aidant des travaux déjà effectués dans le domaine des bases de données spatio-temporelles et de la modélisation orientée objet. Cette approche permet de résoudre des problèmes propres à la gestion des points de vue (filtrage des données, etc.).

Le passage d'une représentation particulière à une autre est réalisé par la connaissance des concepts qu'elles utilisent toutes deux : chaque concept n'est défini qu'une fois pour toute l'ontologie. On notera cependant que pour mettre en œuvre cette approche, il faut avoir à disposition l'ensemble des ontologies de base ; si les ontologies sont réutilisées, récupérées ailleurs, il faut arriver à suivre leur évolution en traduisant les concepts nouvellement ajoutés en des concepts propres à l'ontologie MuRO. Et cette approche ne permet guère (elle perd sinon son intérêt) d'utiliser des concepts proches mais légèrement différents.

[Bouquet et al. 2003b] proposent quant à eux d'utiliser le langage d'ontologies recommandé par le W3C, et de l'adapter pour l'utiliser comme un contexte. Les auteurs dénoncent les non-sens que l'on peut obtenir par inférence lors d'une fusion de deux ontologies en OWL, lorsque le domaine est le même, et qu'un même axiome est utilisé avec des significations bien différentes. Ils proposent alors de rajouter les correspondances directionnelles selon la sémantique décrite dans [Bouquet et al. 2002].

Une critique qu'on pourrait faire, c'est celle de la complexité inutile d'OWL que décrivent [Horrocks et al. 2003], due à la compatibilité descendante recherchée avec RDF, jusqu'au niveau de la sémantique (pour OWL FULL).

2.2.2.3 Diverses propositions d'utiliser le contexte pour l'appariement entre ontologies

Quelques travaux sont parus sur l'utilisation du contexte pour l'appariement d'ontologies. On peut catégoriser l'utilisation du contexte :

- Appariement entre contextes (en tant que systèmes logiques à base de propositions)
- Prise en compte, lors de l'appariement, des concepts voisins

- Utilisation du chemin (du concept à la racine) pour comparer les concepts

L'équipe de l'université de Trento en Italie a réalisé divers travaux pour ajouter une perspective (« *background theory* ») aux ontologies : [Bouquet et al. 2002],[Bouquet et al. 2003a] [Bouquet et al. 2003b], considérant celles-ci comme des points de vue locaux sur un domaine. [Stumme, Maedche 2001] utilise également ces formalisations du contexte pour réaliser la fusion d'ontologies.

Beaucoup d'algorithmes prennent en compte la proximité d'éléments voisins, mais sans la nommer « contexte », parmi lesquels [Doan et al. 2002], [Melnik et al. 2002], [Do, Rahm 2002]. [Madhavan et al. 2001] parlent de correspondances dépendant du contexte (« context-dependant mapping ») pour l'utilisation des éléments voisins. Ils transforment préliminairement le schéma en entrée pour le mettre sous la forme d'un arbre, afin que la réutilisation d'un élément (telle que « adresse ») dans un schéma ne soit pas un obstacle à cette prise en compte du voisinage.

[Do, Rahm 2002] concatènent les labels sur le chemin pour faire la comparaison au format chaîne de caractères. Pour [Madhavan et al. 2001], pour qui « *chaque chemin définit un contexte* », le contexte est essentiellement composé des liens de parenté (même non immédiats). [Noy, Musen 2001] utilisent un algorithme permettant de découvrir de nouvelles paires d'ancres (faisant partie de l'articulation entre les deux ontologies à comparer) à partir d'un ensemble de paires d'ancres données ; leur algorithme se base sur la recherche des concepts se trouvant sur le chemins chacune de ces ancres, en essayant de chercher les concepts correspondants dans l'ontologie voisine. Ils appellent cette utilisation du chemin complet « *contexte non local* ».

2.3 Synthèse

Les changements au niveau de la société, l'expansion du Web mettent en place un environnement fédéré ; le besoin d'interopérabilité et de flexibilité dans ça mise en place semble nécessiter une négociation au niveau sémantique, et donne une importance cruciale au contexte.

Les ontologies représentent le point de vue d'une communauté ou d'un individu [Bouquet et al. 2003a], et sont le formalisme de représentation choisi comme support pour cette négociation. Mais l'appariement entre ontologies, bien que pouvant être appréhendé à l'aide de diverses techniques combinées, ne peut se faire sans intervention humaine : les ontologies sont rarement suffisamment explicite.

Si les ontologies contiennent une part d'implicite, cette donnée peut être explicitée dès lors qu'on connaît suffisamment le contexte ; c'est un avantage certain que possède l'expert métier. L'état de l'art sur l'apport du contexte pour la mise en correspondance des ontologies montre que l'utilisation du contexte lors de l'appariement a consisté jusqu'à présent à considérer l'information explicite que sont les relations entre labels (représentant des concepts), et à l'interpréter.

3 Proposition – Plus de Contexte pour mieux Adapter l'Appariement

Chacun des différents travaux réalisés jusqu'ici sur l'appariement mettant en avant le contexte comme un aspect important de la méthodologie déployée, n'en font état que pour un aspect particulier. Qu'est-ce que le contexte, lorsqu'on veut établir des correspondances entre ontologies ? Quel est l'intérêt de l'étudier, que peut-on gagner en utilisant les informations apportées par le contexte ?

3.1 Détermination des données contextuelles

Notre situation de départ est la suivante : nous cherchons à apparier des ontologies en utilisant le contexte. Comme il s'agit d'une situation elle a un contexte, que nous allons étudier maintenant : examinons ses caractéristiques particulières.

Quelles situations, quels objets ayant une « existence », pouvant déborder le cadre temporel, spatial (et autre) de notre situation présentée plus haut, pouvons-nous rencontrer lors de cet appariement ? Principalement les ontologies, mais aussi les correspondances mises en place.

L'étude du contexte de ces correspondances durant leur « existence » aurait un sens s'il était prévu qu'elles évoluent, et que nous nous intéressions à l'évolution de l'appariement. Cela ne rentre pas dans le cadre de notre étude, nous allons donc considérer le contexte des ontologies, mais non celui des correspondances entre ontologies.

Dans notre problématique, l'appariement entre ontologies est effectué en vue de présenter le résultat à un utilisateur, ce qui est une utilisation fréquente des résultats de la mise en correspondance. Il y a donc interaction entre le contexte des ontologies et le contexte de l'utilisateur, que nous devons donc considérer également.

3.1.1 Trois contextes au sein d'une seule ontologie

Le contexte, par définition, c'est ce qui entoure l'objet, mais qui n'est pas l'objet. Le contexte a trait à toutes les évolutions de l'objet, en commençant par sa création. Nous avons défini l'ontologie comme étant « *une conceptualisation rendue explicite par une spécification* », énoncé qui nous démontre trois aspects importants pour une ontologie :

- elle est décrite de manière explicite, avec des atomes qui sont les labels de concepts, les relations, les attributs, les liens vers les instances.
Objet : élément de l'ontologie.
- le passage de la conceptualisation à l'ontologie nécessite une spécification, et donc un ensemble de développements.

Objet : l'ontologie stricto sensu, soit la construction réalisée par l'assemblage des différents éléments cités plus haut.

- elle représente une conceptualisation.

Objet : la conceptualisation.

De même qu'on reconnaît une abstraction croissante entre les objets « *élément* », « *ontologie* » et « *conceptualisation* », on reconnaîtra que le contexte propre à chaque objet suit le même phénomène, et que les contextes sont de plus en plus généraux, depuis le contexte de l'objet « *élément* » jusqu'au contexte de l'objet « *conceptualisation* ».

Le contexte le plus restreint concerne l'ajout, la modification, la suppression des différentes éléments de l'ontologie. Ensuite vient le contexte plus général de l'environnement de développement de l'ontologie, avec les intérêts des différents organismes qui y ont pris part, leur interaction, etc. . Finalement, on considère le contexte englobant, soit le contexte commun à l'ensemble des personnes ayant travaillé sur l'ontologie : la culture qui a influencé leur conceptualisation, leur façon de la structurer, leur expérience, leur métier d'origine, etc.

3.1.2 Les questions qui permettent de cerner le contexte

Dans toute situation, il y a un besoin de replacer dans leur contexte les événements (« Lundi matin, un avion s'est écrasé en Allemagne, au sud de Stuttgart... »), les leçons apprises (« lorsqu'il pleuvra, même un peu, je ferai attention de suffisamment ralentir avant chaque virage»), les nouveaux contacts (« Bonjour, je m'appelle Andie, je suis Française, je travaille chez Hewlett-Packard et j'habite à Thonon-les-Bains »), les divers problèmes à résoudre, ... dans leur contexte.

La méthode la plus connue pour récupérer et classifier les informations, employée pour l'étude de textes, pour la rédaction des articles de journaux, par les scientifiques pour la résolution de problèmes, etc., est dénommée parfois QQQQCP pour « *Quoi ? Qui ? Où ? Quand ? Comment ? Pourquoi ?* » :

- *Quoi ?* De quoi s'agit-il ? Quelles sont les principales données connues ? On peut ici déterminer les principales contraintes.
- *Qui ?* ...est concerné par l'atteinte de l'objectif ? Déterminer quelle est l'équipe, les rôles respectifs, les intervenants extérieurs,...
- *Où ?* Quel est le périmètre géographique concerné ? Quels flux ont été nécessaires ?
- *Quand ?* Depuis quand ? Pendant combien de temps ?
- *Comment ?* Quelles ont été les étapes ? Quels outils ont été utilisés ?
- *Pourquoi ?* Qu'est-ce qui a motivé toutes ces personnes à s'impliquer ? Pourquoi est-on actuellement dans cette situation ?

On notera des variantes, selon qu'on a affaire à des journalistes, des gestionnaires, etc. ; selon aussi qu'on cherche à comprendre une situation ou à déterminer toutes les caractéristiques d'une solution (c'est le premier cas qui nous intéresse!), ajoutant les questions :

- *Pour quoi ?* Dans quel but ? Quel est l'objectif visé ? (cette question est parfois incluse dans le « *Quoi ?* »). Quel est le besoin ?

- *Combien ?* Quelle unité ? Combien ça coûte ? Quel gain ?

Mais la méthode ne résout pas tout : il reste encore à déterminer précisément, et de manière la plus exhaustive possible, les questions particulières pour chaque cas étudié.

3.1.3 Application aux contextes de l'ontologie

3.1.3.1 Contexte propre aux éléments de l'ontologie

Ce contexte ne prend sens que si l'ontologie a une histoire, car ce qui nous intéresse ici, c'est l'ontologie dans son évolution.

- *Quoi ?* De quelle opération s'agit-il : ajout, modification, suppression ? De quel type d'élément s'agit-il : classe, relation, attribut, lien vers une instance ?
- *Qui ?* Quelle personne a rajouté, supprimé ou modifié cet élément ? Quel est son métier de base, son niveau d'expertise ?
- *Où ?* A quel endroit l'élément a-t-il été placé dans l'ontologie, à quel niveau de « hiérarchie », et avec quels autres éléments a-t-il été relié ?
- *Quand ?* Quelle est la date précise de l'opération ?
- *Comment ?* S'agit-il d'une modification automatique, semi-automatique (comme dans le cas d'une intégration de plusieurs ontologies), ou manuelle ?
- *Combien ?* Combien d'éléments ont été modifiés en même temps ? en tout ?
- *Pourquoi ?* La personne a-t-elle modifié ces éléments dans le cadre de son travail, par plaisir, parce qu'elle a besoin de l'ontologie et qu'elle doit l'adapter ?
- *Pour quoi ?* Dans quel but l'élément a-t-il été rajouté : pour diminuer la granularité, pour étendre l'ontologie, pour combler un manque ? Dans quel but l'élément a-t-il été supprimé : parce qu'il est déprécié, n'existe plus, qu'il a été mis en place par erreur, qu'il y a désaccord, qu'il y a abandon d'une partie de l'ontologie, ... ? Dans quel but l'élément a-t-il été modifié : pour utiliser un terme plus adéquat, pour préciser davantage la relation, pour modifier la granularité, ... ?

3.1.3.2 Contexte propre à l'ontologie développée

Il s'agit ici de ne plus considérer les éléments de l'ontologie comme étant les « atomes », mais on voit celle-ci comme un tout, et on cherche à la caractériser, dans ce qui peut la différencier des autres ontologies qu'on pourra rencontrer. C'est un peu le « *contexte commun* » des éléments de l'ontologie (ce qui est vrai pour la plupart des éléments).

- *Quoi ?* Quel langage d'ontologie est utilisé ? Quelle est la structure choisie (hiérarchique, graphe, etc.) ? Quelle langue est utilisée ? Quelles sont les règles de nommage ?
- *Qui ?* Quel organisme a créé l'ontologie ? Pour qui ? Avec l'aide de qui ? Prévue pour être utilisée par qui ? Qui la met à jour ? Qui classe les informations en les reliant à l'ontologie ?

- *Où ?* Où a-t-elle été développée : dans quel pays, dans quelle entreprise ? Où est-elle sensée servir ? Où est-elle réellement utilisée ?
- *Quand ?* Quand a-t-elle été créée ? Quand a-t-elle été mise à jour ? Quelle est la fréquence de sa mise à jour ?
- *Comment ?* Quelles ont été les informations de base à la création de l'ontologie : dictionnaires, thésaurus, ... ? Comment a-t-elle été créée : par fusion, transformation, ... ?
- *Combien ?* Combien de concepts met-elle en relation ? Combien compte-t-on de relations, d'attributs, de liens identiques vers des instances ?
- *Pourquoi ?* Pourquoi cet organisme a-t-il décidé de développer l'ontologie ? Quel est le rapport entre l'organisme et le besoin (évoqué question suivante) ?
- *Pour quoi ?* Dans quel but l'ontologie a-t-elle été développée ? Pour répondre à quel besoin ?

3.1.3.3 Contexte propre à la conceptualisation que l'on cherche à représenter

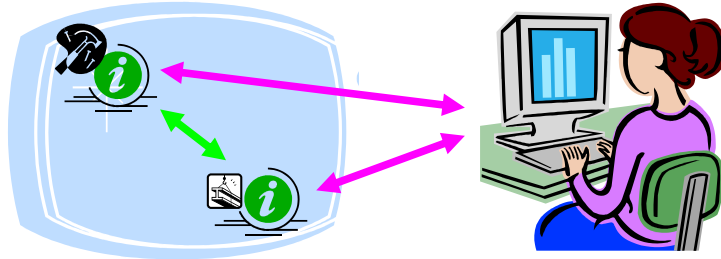
L'ontologie est une tentative de représenter, sous un format donné et avec une terminologie rigoureuse, une conceptualisation commune à l'ensemble de ceux qui ont participé à son évolution. C'est cette conceptualisation sous-jacente dont on va chercher à déterminer les tenants et aboutissants.

- *Quoi ?* Quel est le sujet traité dans l'ontologie ? Quelle en est la portée ? Que veut-on couvrir ? Quelle est la finesse de granularité recherchée ?
- *Qui ?* Quelles ont été les organismes qui, de près ou de loin, sont impliqués dans la création et l'évolution de cette ontologie ? Quels sont les organismes acteurs du domaine ?
- *Où ?* De quel origine sont les personnes concernées ? Travaillent-elles dans la même entreprise, dans un même pays ?
- *Quand ?* Quelle année l'ontologie a-t-elle été créée ? Quelles sont les dates des différentes étapes de son existence : création, intégration, réutilisation par un autre organisme, ... ?
- *Comment ?* Quelles ont été les moyens utilisés pour la faire évoluer ?
- *Combien ?* Dans quelle proportion couvre-t-elle le sujet ?
- *Pourquoi ?* Pourquoi les personnes impliquées ont-elles imaginé une telle conceptualisation ? Quelles sont les raisons de voir les choses de telle ou telle manière particulière ?
- *Pour quoi ?* Quels sont les besoins auquel la spécification de cette conceptualisation (par une ontologie) doit répondre (et qui peuvent donc influencer sur les paradigmes, structurations choisis) ?

3.1.4 Interaction avec l'utilisateur

L'interaction entre l'utilisateur et les ontologies est primordiale : l'utilisateur va interagir avec les ontologies selon ce qu'il connaît, selon sa conceptualisation du domaine, etc. Lorsqu'il effectue une recherche, cela revient à chercher un appariement entre le concept auquel il pense et le concept équivalent (ou à une variante près) tel qu'on peut le trouver dans l'une ou l'autre des ontologies. Sa

recherche est fortement influencée par son contexte : il écrira la requête dans sa langue, il utilisera les mots qu'il connaît, en supposant connu le but de sa requête ;



Par exemple, si dans un moteur de recherche traditionnel, un maçon cherche « brique », il ne s'attendra certainement pas à recevoir des réponses concernant des « casse briques » ; il triera lui-même les informations, ou ajoutera « maçonnerie » (par exemple) dans sa requête. Cela veut dire que c'est l'utilisateur qui s'adapte plutôt que l'outil.

Le contexte consistant à décrire les informations particulières dans le cadre d'une interaction, on pourra décrire à l'utilisateur les caractéristiques particulières des ontologies l'une par rapport à l'autre. Mais la conceptualisation de l'utilisateur est différente de celle des ontologies considérées ; pour qu'il puisse réellement comprendre l'information qu'on lui soumet, il est donc essentiel d'adapter l'information. Pour cela, il faut connaître son contexte.

Beaucoup d'informations du contexte de l'ontologie apportent une connaissance lorsqu'on connaît suffisamment l'utilisateur. Quelle est en effet la valeur ajoutée de savoir que tel fournisseur ne fournit que du matériel pour maison individuelles, si on ne sait pas que l'utilisateur qui fait la requête est spécialisé dans la construction d'immeubles ?

De même, si on sait que l'utilisateur comprend uniquement le français et l'anglais, on peut lui éviter de parcourir les résultats en chinois (on peut les rendre disponibles par ailleurs). Un apport de l'utilisation du contexte est en effet de fournir l'information particulière de manière explicite, et non de la proposer parmi un ensemble d'autres informations où elle pourrait passer inaperçue, résultant en une perte de temps : pas d'information superflue.

La difficulté de l'interaction avec l'utilisateur, c'est qu'on connaît bien peu l'utilisateur et qu'on a accès à très peu d'informations sur ses besoins : ce qu'on lui aura demandé de renseigner et ce qu'on pourra déduire de ses différents choix au cours de ses recherches via l'outil.

3.1.5 Application aux contextes liés à l'utilisateur

L'utilisateur, lorsqu'il interagit avec les ontologies, le fait de manière uniforme à travers une interface unique d'interrogation ; il ne devrait pas avoir besoin de s'adapter à une ontologie particulière. Lorsqu'il pose une requête, il interagit avec les ontologies (et avec leur appariement, s'il existe) via l'outil.

Quelles situations, quels objets, pouvant déborder le cadre temporel, spatial (et autre) de cette situation, pouvons-nous rencontrer ?

- L'utilisateur effectue une recherche en utilisant l'outil.
Situation : recherche de l'utilisateur
- Son utilisation évolue avec les ontologies via l'outil.
Situation : interaction générale de l'utilisateur avec l'outil
- L'utilisateur a développé une conceptualisation, qui influe sur ce qu'il va rechercher, les termes qu'il va utiliser, et comment il va comprendre le résultat.
Objet : conceptualisation de l'utilisateur

Comme précédemment, nous allons rechercher pour chaque situation un ensemble de questions permettant de déterminer chaque contexte.

3.1.5.1 Contexte propre à la recherche en cours

L'utilisateur interagit avec les ontologies à travers un ensemble de requêtes, dans un certain ordre, et peut-être également en explorant l'arborescence d'une ou de plusieurs ontologies.

- *Quoi ?* Quel est l'objet de la recherche ? Quel concept a-t-il recherché ? A-t-il trouvé ce qu'il cherché ?
- *Quand ?* Quelle est la date de l'interrogation ? Combien de temps a-t-il passé ? En combien de temps ses requêtes ont-elles été traitées ?
- *Comment ?* Comment s'y est-il pris : quels sont les requêtes qu'il a successivement lancées ? Par quels concepts est-il passé pour arriver au résultat ?
- *Combien ?* Combien de requêtes lui a-t-il fallu pour avoir une réponse (négative ou positive)
- *Pour quoi ?* Quel était son besoin ?
- *Qui, Où, Pourquoi ?* -

3.1.5.2 Contexte propre à l'interaction avec l'utilisateur

L'utilisateur a interrogé l'outil de manière (plus ou moins) régulière. On peut connaître mieux maintenant ce qu'il recherche. Il a peut-être changé sa manière d'effectuer ses recherches

- *Quoi ?* Quels sont les divers concepts qu'il a recherchés ? Quels sont les mots qu'il a utilisés pour les trouver ? Dans quel(s) domaine(s) effectue-t-il ses recherches ? Dans quelle(s) langue(s) ?
- *Qui ?* Quelles informations a-t-il renseigné sur lui-même ?
- *Où ?* Dans quel pays se trouve l'utilisateur ? Dans quelle entreprise ?
- *Quand ?* Quand a-t-il effectué ses requêtes ?
- *Comment ?* Utilise-t-il l'outil directement, ou via un autre outil informatique ?
- *Combien ?* Combien de fois a-t-il effectué des recherches ? Pendant combien de temps ?
- *Pourquoi ?* Pourquoi utilise-t-il ce service (et pas un autre) ?
- *Pour quoi ?* Quel type d'information recherche-t-il par ce service ?

3.1.5.3 Contexte propre à la conceptualisation de l'utilisateur

L'utilisateur a un arrière-plan particulier, et des connaissances dans divers domaines, de manière plus ou moins spécialisée. Son interaction avec les ontologies via l'outil sera influencée également par sa culture, et par les langues qu'il pratique.

- *Quoi ?* Quels sont les domaines qu'il connaît ? Quelles sont les langues qu'il pratique ?
- *Où ?* Dans quels pays a-t-il séjourné ? Pour quelles (type d')entreprises a-t-il travaillé ?
- *Quand ?* Combien de temps a-t-il exercé les divers métiers ? Pendant combien de temps a-t-il travaillé dans chaque domaine ? Depuis combien de temps n'a-t-il plus travaillé dans ce domaine ?
- *Comment ?* Par quel type de métier a-t-il acquis ces connaissances ?
- *Combien ?* Dans quelle mesure connaît-il ces domaines, pratique-t-il ces langues ?
- *Qui, Pourquoi, Pour quoi ?*

3.2 Interprétation des données contextuelles propres à l'ontologie

Nous avons cherché à déterminer de manière relativement rigoureuse quel type d'information fait partie du contexte de l'ontologie. Pour cela, nous avons vu que l'on pouvait identifier trois contextes imbriqués, ce qui nous a permis de définir les questions pour lesquelles l'information contextuelle doit nous donner une réponse.

Mais quelles que soient les réponses, elles nous sont de peu d'utilité si nous ne passons pas par une étape d'interprétation, ou tout au moins de classification. Si toutes les données ne peuvent faire l'objet d'un traitement informatisé, elles peuvent être fournies directement à l'utilisateur pour lui donner des éléments de comparaison entre deux informations divergentes.

Nous allons tâcher de décrire ci-après diverses règles d'interprétation qu'on peut déduire des informations que nous pourrions récupérer. Il est en effet essentiel d'automatiser la tâche d'interprétation (ou de réduire l'intervention humaine au minimum) : ce travail demande beaucoup de temps. On notera que le résultat de l'interprétation est subjectif, et dépend de la complétude des données récupérées.

3.2.1 Rôle du contexte propre aux éléments de l'ontologie

Le contexte propre aux éléments de l'ontologie fournit des informations sur l'évolution de l'ontologie. Une difficulté est qu'un tel développement est rarement linéaire, et fait état de changements de stratégie ; ceux-ci doivent être détectés d'une autre manière, sans cela ce qu'on pourra déduire de toutes les données récoltées à ce niveau sera faussé.

Le maniement des concepts et leur mise en relation nous donne les renseignements les plus déterminants pour comprendre la conceptualisation sous-jacente : les développeurs de l'ontologie cherchent à rendre concrète la conceptualisation qui les préoccupe, et ils vont tâcher de traduire les principes, les images qu'ils se font du sujet en concepts distincts et hiérarchisés (en général), pour lesquels ils vont essayer de trouver les termes adéquats. Une difficulté sera alors de bien choisir le terme (nécessite une bonne connaissance du langage, et/ou de bons outils) ; sans doute y aura-t-il un

effort d'ôter les ambiguïtés possibles, et peut-être faudra-t-il se conformer à un vocabulaire commun ou tout au moins à un ensemble de normes.

Nous voyons donc que pour bien interpréter ces informations, il nous faut des informations du contexte englobant, qui est le contexte propre à l'ontologie *stricto sensu*. Mais le contexte propre à la conceptualisation sous-jacente donnera également des éléments permettant d'orienter notre compréhension.

3.2.1.1 Labels

Nous avons vu que le développeur porte son effort sur la traduction de principes et concepts abstraits, et en général peu structurés, vers une représentation concrète, sous forme de labels et de relations. Il nous faut donc effectuer le chemin inverse pour parvenir à la conceptualisation.

3.2.1.1.1 Ajout d'un label

Soit donc un label, ajouté à l'ontologie. Il nous faut chercher à comprendre quels sont les concepts/principes qui peuvent faire penser à ce label.

Pour y arriver, il nous faut une terminologie qui définisse rigoureusement chaque concept : pour comparer des concepts entre eux, il est nécessaire de pouvoir les identifier de manière unique. Supposons qu'on dispose d'une telle terminologie rigoureuse et extensible, et d'un ensemble de concepts ainsi définis.

Il nous faut chercher tout d'abord pour quel(s) concept(s) le label est utilisé, mais aussi dans quelle proportion il est utilisé pour chaque concept qu'il peut représenter. Pour cela, on a bien entendu besoin de connaître la langue utilisée, les contraintes imposées à ce niveau par le langage de l'ontologie choisi, et l'arrière-plan du développeur (sa provenance géographique et ses divers séjours, les professions qu'il a exercées, ...). Ce sont autant de facteurs à prendre en compte pour déterminer, pour chaque concept, la fréquence moyenne d'utilisation du terme utilisé dans le label par rapport aux autres termes possibles. Mieux, on pourrait effectuer préliminairement une analyse lexicale et sémantique de documents rédigés par ses soins, et rechercher dynamiquement si le terme y est utilisé, dans quel proportion et quel(s) concept(s) lui est(ont) rattaché(s).

On choisit alors un concept parmi les concepts que le label peut représenter, qui soit compatible avec le sujet de l'ontologie. On prendra donc de préférence le concept pour lequel la fréquence du label par rapport aux autres termes pouvant le décrire est la plus forte.

Les données récoltées doivent être conservées car ce choix peut être remis en cause ; s'il s'agit d'un concept pour lequel la fréquence du label parmi les autres termes possibles est plus faible (et que la différence est conséquente), cela nous permet de connaître que :

- soit le calcul des fréquences des mots est erroné (un degré de fiabilité doit être associé à ce calcul, tenant en compte la différence entre les dates des documents consultés et de la date de la création du label, la divergence géographique, etc.),
- soit le développeur est insuffisamment connu,

- soit il s'agit d'un choix délibéré de la part du développeur (un degré de fiabilité doit être associé à cette décision, car il peut s'agir d'un concours de circonstances), ce qui nous donne un indice sur sa volonté d'orienter l'ontologie dans un sens particulier.

3.2.1.1.2 Modification d'un label

Sous réserve (voir plus haut) d'un changement de stratégie, la modification d'un label obéit aux mêmes règles, dès lors que le concept représenté est le même : cela indique une orientation. Si le nouveau terme ne correspond pas au même concept, la question se pose : la modification est-elle due à une erreur de conceptualisation ou bien à un manque de précision au niveau de la granularité ou au niveau de la séparation des concepts (dans ce cas, les concepts doivent être voisins), ou encore d'une évolution ?

Il faudra savoir déterminer s'il s'agit d'une réelle modification ou d'une suppression et création : on pourra reconnaître cette dernière possibilité par la suppression, à un intervalle de temps très court, de relations superflues, et/ou l'ajout, approximativement au même endroit, d'autres labels (un ajout d'information est rarement unique).

Il faudra bien entendu tenir compte du développeur qui aura effectué la modification : s'agit-il du même ? Auquel cas, on devrait pouvoir en déduire qu'il s'agit d'un gain en précision ou en justesse. Dans le cas contraire, il serait utile de pouvoir comparer l'expérience de chacun, leur connaissance du sujet et/ou de la représentation de la connaissance, et enfin de connaître leur relation : si en effet le développeur modifiant le label est bien moins qualifié que son prédécesseur, on associera à l'interprétation associée à la modification un faible indice de fiabilité.

Cet indice de fiabilité pourra être diminué ou augmenté selon les autres informations disponibles ; par exemple, si on constate que le premier développeur avait créé le label parmi un grand nombre d'autres et dans un temps très court, on pourra considérer que c'était un travail pour poser les fondements de l'ontologie, pouvant nécessiter des ajustements ; dans ce cas, on pourra augmenter l'indice de fiabilité associé à l'interprétation de la modification, s'il est faible.

Si on constate par la suite que le label est modifié à nouveau pour reprendre sa valeur initiale, cela peut signifier :

- soit une erreur d'inattention rectifiée (si la correction survient assez tôt)
- soit une erreur plus subtile, concernant la conceptualisation, et qui a pu faire l'objet de discussions (la correction vient plus tard)
- soit un conflit (si les modifications sont faites par deux développeurs alternativement).
- soit un problème d'incompétence (idem, mais de façon répétée).
- soit un changement de stratégie (on notera qu'une reprise des objectifs initiaux après une longue dérive peut être considérée comme un changement de stratégie, puisque la première n'avait pas été suffisamment claire ou mise en pratique)

S'il s'agit d'une erreur d'inattention, d'un problème évident d'incompétence ou d'un changement de stratégie, l'interprétation associée à la « correction » sera la même que l'interprétation associée à l'opération précédant l'« erreur ». Autrement, il faut tenir compte que la modification intermédiaire représentait certainement une alternative proche d'être retenue. La nouvelle interprétation sera donc :

$$\mathbf{I}(modification(t_{n+2}, Y, X) = \mathbf{I}(opération(t_{n+1}, X) * f_{I_{n+2}} + (1 - F_{I_{N+2}}) * \mathbf{I}(modification(t_{n+1}, X, Y)$$

Où \mathbf{I} est la fonction d'interprétation et $F_{\mathbf{I}}$ est l'indice de fiabilité correspondant.

3.2.1.1.3 Suppression d'un label

Enfin, la suppression d'un label peut indiquer :

- soit une erreur de conceptualisation,
- soit une redondance (qui sera en général déterminée par la présence de modifications d'autres termes reliés au même parent),
- soit une restriction de la couverture de l'ontologie (dont on pourra se rendre compte par la suppression de tout un pan de l'ontologie),
- soit encore un changement de stratégie (p. ex. un élagage de ce qui n'est pas utilisé dans l'application en vue de laquelle l'ontologie est développée).

À noter que si l'opération est effectuée par un autre développeur, il faudra en tenir compte comme précédemment.

3.2.1.2 Relations

Nous avons parlé jusque là de labels correspondant à des concepts. On aura une considération similaire (mais moindre) pour les relations créées/modifiées/supprimées de manière indépendante : elles servent à définir la structuration intégrant le nouveau concept, représenté par son label, avec les autres concepts déjà représentés.

3.2.1.2.1 Ajout d'une relation

Souvent plusieurs relations sont créées dans la même occasion : lors de l'ajout d'un label représentant un concept, le développeur le mettra immédiatement en relation avec plusieurs autres ; il faut donc regrouper ces relations lorsqu'elles sont posées dans un intervalle de temps rapproché (et les associer à la création du label) ; par ailleurs, un oubli étant vite arrivé, l'ajout ultérieur d'une relation n'impliquera pas forcément un changement d'orientation.

3.2.1.2.2 Modification d'une relation

Pour ce qui est des modifications, là aussi, les relations étant généralement l'objet de moins d'attention lors de la construction de l'ontologie, on ne pourra guère en déduire une évolution ; sinon peut-être lorsque l'ontologie aura atteint un niveau de stabilité élevé et que les relations seront réellement diversifiées. L'indice de fiabilité sera donc en général plus faible que dans le cas d'ajout/modification/suppression de label.

Nous avons cependant un avantage : une bonne partie des types de relations se retrouve dans la plupart des ontologies, et il y a donc moyen de créer des tables permettant de savoir immédiatement si une modification correspond à un gain de précision, à un changement total, etc.

On notera que lorsqu'on relie un label avec d'autres, c'est que l'on vise à replacer le concept ainsi représenté dans son contexte : cela permet d'en donner une « définition » pour quelqu'un qui connaîtrait les concepts associés, et de préciser des particularités : un concept existe en général sous plusieurs variantes : le concept de « fleuve » correspond plus ou moins à celui de « stream », mais ce dernier peut être défini à partir du concept « rivière » (« river ») à l'aide d'un critère de taille (« big river »), alors que le fleuve, est défini à l'aide du critère « qui se jette dans la mer ou dans l'océan » (cf. [Sowa 1999]). L'ajout de relations peut donc être une *aide pour savoir si on a affaire exactement au même concept ou à une variante*.

3.2.1.3 Instances

Une ontologie n'étant pas toujours reliée à des instances, et celles-ci variant beaucoup d'une application à l'autre, nous ne nous attarderons pas sur cet aspect. Néanmoins, si une telle information existe, on pourra souligner l'importance qu'elle a pour montrer ce que *les utilisateurs* comprennent par tel label. La difficulté, c'est qu'il peut y avoir mésentente entre le groupe des développeurs et celui des utilisateurs. La connaissance d'une telle information (de divergence entre les concepts représentés et les ceux qui sont associés au label fournit) *permet d'anticiper certains « changements de stratégie »*, qui auront lieu pour mettre à jour l'ontologie de manière à l'adapter davantage aux utilisateurs (ou en changeant la paternité des instances « fautives »).

3.2.1.4 Modifications automatiques

De plus en plus, les manipulations ontologiques se font de manière semi-automatisée, voire automatisée. Que peut-on en déduire ? Sans doute que l'automate a une expertise moindre du sujet, des subtilités de la représentation de la connaissance (mais des progrès sont accomplis dans ce domaine), et des finesses de la langue employée. L'indice de fiabilité pour chacun de ses aspects sera adapté, si possible à partir d'une évaluation de l'outil utilisé.

On pourra en déduire certainement davantage si on sait, du contexte englobant, que l'outil a été utilisé pour réaliser une fusion, un alignement, etc. : il peut être utile alors de garder une trace de l'articulation ainsi mise en place, afin de pouvoir associer à chaque division de l'ontologie un ensemble de propriétés pouvant évoluer de manière sensiblement différente : non que ce soit souhaitable, mais simplement parce qu'il est impossible dans les faits que deux ontologies, développées dans des endroits différents par deux équipes différentes, puissent avoir exactement la même granularité, la même précision, etc. (le langage peut être le même, ainsi que la normalisation des termes, toutefois).

3.2.1.5 Quelques remarques avant de passer à la suite

Un lecteur attentif aura noté que nous avons passé beaucoup de temps à définir le périmètre et à analyser des éléments du contexte afin de déterminer (dans la mesure du possible) ... d'autres éléments du même contexte (p. ex. le but des diverses opérations) !

Certaines informations ne peuvent être connues que si le développeur lui-même les renseigne. Cela lui demanderait beaucoup de temps, à moins que l'opération soit semi-automatisée. On peut également imaginer avoir un outil adapté qui, après avoir effectué toutes sortes de calculs pour réaliser une fusion, un alignement ou une autre opération, sauvegarde sous un format adéquat une explication de la

modification de la conceptualisation qu'il veut obtenir en effectuant une telle modification. Connaître de manière fiable. Dans un cas comme dans l'autre, cela ajouterait à la fiabilité de l'information récupérée, mais n'est qu'une partie de l'interprétation.

Le rôle du contexte que l'on vient de décrire est donc de nous donner les moyens de comprendre une ontologie imparfaite, dans son évolution, de ne pas la considérer comme un tout mais comme un ensemble de parties. C'est en quelque sorte la « mémoire métier » de son développement. Cette connaissance peut être une aide pour confirmer, infirmer, orienter le choix de concepts, particulièrement lors de l'appariement.

3.2.2 Rôle du contexte propre à l'ontologie développée

Si le contexte de l'ontologie *stricto sensu* est une aide pour comprendre le contexte des éléments de l'ontologie, son intérêt premier est de donner des éléments permettant de juger si l'ontologie est adaptée à l'utilisation qu'on veut en faire.

Ce contexte est beaucoup plus statique que celui étudié précédemment, seules les informations de mise à jour et celles indiquant la taille de l'ontologie (en nombre de concepts, relations, etc.) sont modifiées régulièrement. La plupart des informations ne sont pas destinées à être modifiées, et seront certainement renseignées manuellement, mais ce n'est pas gênant, vu qu'il y a relativement peu d'informations à fournir : la difficulté est plutôt de pouvoir récupérer l'ensemble des informations, et les manipuler de manière automatique.

3.2.2.1 Une aide pour comprendre le contexte des éléments de l'ontologie

Nous avons vu dans la première section que le *langage d'ontologies* pouvait avoir un impact sur l'expressivité permise. Cela peut être pris en compte lors de la traduction de l'ontologie, par exemple en cherchant à voir si le concepteur n'a pas cherché à exprimer davantage par des moyens détournés. Une telle prise en considération doit être réalisée effectivement par un algorithme spécifique au langage d'ontologies concerné car ces détours, lorsqu'ils existent, sont propres à chaque langage d'ontologies.

Cependant, c'est un point difficile à appréhender : on pourra rarement être sûr qu'un choix conceptuel est peu expressif parce que le langage utilisé manque d'expressivité, vu que ça peut être le résultat d'un choix délibéré. Il y a donc une part d'incertitude en tout point où l'expressivité fera défaut, et où ce manque peut être dû au langage original.

Ce manque d'expressivité peut ne pas poser de problème lorsqu'on considère l'ontologie seule. Quand on la considère en relation avec d'autres, on comparera les informations de par et d'autre : si l'une des ontologies est moins expressive sur un point, on vérifiera si cela peut être dû à un manque d'expressivité du langage ; dans ce cas, on n'en pas tiendra pas compte.

On pourra en plus (c'est une information redondante, il est vrai) indiquer le type de *structure*. En effet, il existe beaucoup de langages à partir desquels on peut imaginer transférer les informations vers un vrai langage d'ontologies, et il est donc impossible de connaître leurs particuliers à chacun. Si donc un langage n'est pas connu, la structure qu'il utilise (le nombre de structures possibles est plus limité) pourra nous donner tout de même quelques indications !

Connaître *la langue* est essentiel pour réaliser l'appariement : même s'il est possible de modifier les termes lors de la traduction de l'ontologie, cela peut être dangereux, puisque les informations sont souvent prises en-dehors du contexte, et un faux ami peut être choisi (ce qui pourra résulter en un faux sens par la suite). Il faut donc savoir associer différents dictionnaires selon la langue trouvée.

Enfin, la connaissance de *règles de nommage* va être une aide précieuse pour l'interprétation des termes : cela nous donne une information sur la structuration des informations textuelles. Par exemple, un des catalogues que nous avons à apparier a choisi de nommer ses labels en suivant la « norme » des identifiants dans Java : lié, avec une majuscule pour chaque début de mot. Si on recherche le mot complet dans le dictionnaire, on risque de ne pas trouver grand chose !

Pour résoudre ce problème, certains algorithmes essaient d'intégrer toutes les possibilités de nommage. Nous proposons que l'information soit recueillie sous forme de spécifications, suivant une forme d'expressions régulières par exemple (afin que plusieurs algorithmes d'appariement basés sur différents langages puissent s'en servir).

Les ontologies étant créées et évoluant de manière autonome, on ne peut pas s'attendre à ce que ces règles soient respectées tout le temps. La création des règles pourrait cependant être intégrée à l'éditeur d'ontologie pour écrire tout nouveau label automatiquement dans la bonne forme, ce qui permettrait également d'automatiser tout changement de règle ultérieur.

3.2.2.2 Une aide pour juger de l'intérêt de l'ontologie pour ce qu'on veut en faire

Si les ontologies sont créées par des entités autonomes, la question de la *validité des données* se pose :

- Les données sont-elles suffisamment à jour ?
- Quelle est la part de subjectivité des informations ?

Il est vital de pouvoir utiliser les réponses à ces questions de manière automatique : actuellement, lorsqu'un utilisateur fait un achat sur le Web, il vérifie qu'il est sur un site crédible, et que son achat pourra être fait en sécurité ; lorsqu'il recherche des informations, il vérifie que la source est valable. Avec la venue du Web sémantique, ce seront des robots qui iront chercher à sa place ; s'ils ne tiennent pas compte de la validité des sources, l'utilisateur peut avoir de surprenants résultats !

De même, une ontologie dont la dernière mise à jour remonte à un an peut être oubliée s'il s'agit d'une ontologie qui détaille les nouvelles technologies, mais cette même information ne remet pas en cause une ontologie sur les locomotives à vapeur ! De même, on préférera souvent une ontologie réalisée par un organisme indépendant que par un organisme commercial...

3.2.2.2.1 Mise à jour des données : détermination d'un indice de l'activité

Pour évaluer si les données sont suffisamment à jour, il faut avoir accès à un *indice de l'activité* des différents domaines/sujets (la difficulté étant d'être exhaustifs). A défaut de pouvoir bénéficier d'un tel service, on pourra dresser une liste de sites Internet pour un grand nombre de domaines, et déterminer leur fréquence de mise à jour moyenne mensuelle (soit par la date de mise à jour, soit par le renouvellement du contenu).

On aura alors un indice de l'activité du Web pour ce domaine. Il faudra bien sûr corriger le résultat pour ôter les effets des fluctuations de l'activité globale sur le Web. On pourra alors comparer la

moyenne de l'activité pour le domaine à la fréquence de mise à jour de l'ontologie, et obtenir un pourcentage d'activité pour l'ontologie.

3.2.2.2 Subjectivité des informations : indice de biais

Dans le deuxième cas, il faut déterminer un indice de « danger de *biais* » qu'on pourra calculer par exemple en faisant le produit du pourcentage de parti pris (approximation manuelle) de l'organisme dans l'affaire avec un pourcentage donnant l'estimation de l'importance des modifications que cela peut amener sur l'ontologie.

3.2.2.3 Couverture, granularité de l'ontologie : indicateur de taille relative

La *taille* de l'ontologie est également un facteur discriminant pour son choix. Elle peut indiquer une complétude de l'information, une granularité plus fine, ou une couverture plus grande. Mais cela peut être trompeur : il est possible que la granularité soit très fine juste sur un aspect très spécifique qui ne va pas intéresser l'utilisateur.

Dans tous les cas, la taille est une information qui peut difficilement s'évaluer de prime abord : comment savoir quelle serait la « taille idéale » pour une ontologie d'un domaine ? Une solution est la comparaison entre ontologies, et de prendre comme indicateur le rapport entre la taille de l'ontologie (en nombre de concepts par exemple) et la moyenne de taille des ontologies (la mesure devient de plus en plus pertinente à mesure que le nombre d'ontologies considérées croît).

3.2.2.4 Utilisation de l'ontologie : indication par les métiers, les domaines, etc.

Savoir dans quel(s) but(s) l'ontologie est (ou peut être) réellement utilisée est extrêmement pratique ... mais c'est une information très difficile à récupérer, tout comme à prendre en compte de manière automatique. En fait, on peut prendre ici l'ensemble des réponses aux questions « où est-elle censée servir ? », « où est-elle réellement utilisée ? », « qui était le destinataire originel ? », « dans quel but a-t-elle été développée, pour répondre à quel besoin ? », et déterminer plusieurs *utilisations possibles de l'ontologie* (parmi un nombre de possibilités répertoriées), les *domaines* qu'elle couvre ainsi que les *métiers* qui l'utilisent (parmi un nombre de possibilités répertoriées).

3.2.3 Rôle du contexte propre à la conceptualisation qu'on cherche à représenter

Ce contexte, qui est le contexte commun à tout le développement de l'ontologie, nous donne le moyen de confronter la conceptualisation que nous avons cherché à déterminer (grâce au contexte des éléments de l'ontologie) avec une conceptualisation du même sujet mais dans un autre contexte (provenant donc d'une autre ontologie) : on s'y intéressera pour savoir s'il n'est pas trop difficile de la réutiliser, et si oui, dans une vue d'adapter la sémiotique de l'ontologie.

L'ontologie est une tentative de représenter, sous un format donné et avec une terminologie rigoureuse, une conceptualisation globalement commune à l'ensemble de ceux qui ont participé à son évolution. C'est cette conceptualisation sous-jacente dont on va chercher à déterminer les tenants et aboutissants.

3.2.3.1 **Sujet (souvent le domaine) de l'ontologie : ensemble de mots-clés**

Pour commencer, le plus évident : le *sujet* de l'ontologie est sans doute l'information la plus utile lorsqu'on veut consulter une ontologie. L'information peut être fournie de manière explicite pour chaque ontologie, ou sinon récupérée en considérant les concepts rencontrés : par exemple en mettant ensemble tous les mots-clés provenant des nœuds ainsi que leurs synonymes; la comparaison entre ontologies se fera donc sur la proportion de mots-clés communs.

Pour pouvoir effectuer des comparaisons de domaines, il faut réaliser une classification hiérarchique préalable, qui soit assez précise pour qu'on puisse associer plusieurs sous-domaines à une même ontologie. S'il s'agit d'un ensemble d'ontologies portant sur des sujets variés, cela n'est plus possible, il faudra donc considérer de récupérer l'information à partir du contenu de l'ontologie.

3.2.3.2 **Culture**

La *culture* (pays, entreprise) est à prendre en compte, requérant une introspection particulière, car les différences de portées peuvent être grandes. C'est un critère difficile que nous ne pouvons pas détailler ici.

3.2.3.3 **Organismes ayant trait à l'ontologie : indice de garantie**

Les *organismes sources* de l'ontologie, où celles qui sont impliquées dans la création de l'ontologie, ont un rôle de garant de la qualité de l'ontologie. Si ce sont les acteurs principaux du domaine, c'est un critère de qualité important. Une liste des principaux organismes pour chaque domaine peut être récupérée, afin de la comparer à la liste des organismes à l'origine de l'ontologie. On peut imaginer un indice de garantie de 90% si un des organisme est reconnu internationalement, resp. 70% , 50% et 40% si un des organismes est reconnu à l'échelle de plusieurs pays industrialisés, de plusieurs entreprises phares dans le domaine, ou d'un pays industrialisé respectivement.

3.2.3.4 **Évolution de l'ontologie et changements de stratégie**

L'année de création de l'ontologie peut donner une indication sur sa maturité. Des *statistiques* décrivant l'évolution du nombre d'utilisateurs et du nombre de concepts pourraient nous permettre de savoir si l'ontologie est en « progression », si le nombre de concepts se stabilise, si l'ontologie a subi des gros changements (p. ex. : une réduction brutale d'un grand nombre de concepts peut indiquer un élagage réalisé pour se refocaliser sur le sujet spécifique de l'ontologie).

3.2.3.5 **Objectif initial de l'ontologie**

Enfin, savoir quel était l'*objectif initial* de la création de l'ontologie, et connaître les raisons qui font qu'elle soit maintenue sont des points-clés pour se rendre compte de sa pérennité ; cela peut apporter des indices concernant un biais éventuel ; enfin, cela peut impliquer des choix conceptuels (sont-ils compatibles avec l'utilisation que nous voulons en faire ?)

3.2.3.6 Couverture de l'ontologie : indicateur de couverture

Pour juger de la couverture du sujet par l'ontologie il est nécessaire d'effectuer des appariements avec d'autres ontologies portant sur le même sujet. Si on réalise une ontologie de correspondances (« *articulation ontology* ») pour chaque couple d'ontologies mises en correspondance, on pourra déterminer un indicateur de couverture par la formule $I_{couverture} = \frac{1}{n \cdot M} \cdot \sum_{i=1}^n N(i)$ où $N(i)$ représente le nombre de concepts pour l'articulation i (n étant le nombre total d'ontologies de correspondance pour l'ontologie), et M est le nombre de concepts moyen pour toutes les ontologies d'articulation du domaine.

3.3 Interprétation des données contextuelles propres à l'utilisateur

Maintenant que nous avons donné les bases de l'interprétation des données contextuelles relatives aux ontologies, étudions les contextes liés à l'utilisateur. Tout comme les contextes de l'ontologie, il s'agit de contextes imbriqués : une recherche d'un utilisateur est en général un cas particulier de l'ensemble des interactions de l'utilisateur via l'outil, et l'un et l'autre dépendent largement de la conceptualisation que s'est forgée l'utilisateur.

Il faut noter que les contextes que nous avons déterminés sont propres à l'utilisateur, mais dans la limite de l'interaction avec les ontologies, et dans la limite de ce qu'on peut connaître de l'utilisateur : ainsi, on ne peut pas savoir comment l'utilisateur a construit sa conception du monde, ni comment elle évolue ; on peut juste tenter d'en cerner quelques aspects rencontrés lors de l'utilisation de l'outil.

3.3.1 Rôle du contexte propre à la recherche en cours

Lorsque l'utilisateur se connecte à l'outil à la recherche d'une information, il veut obtenir une réponse positive (de préférence) ou négative, rapidement et de manière fiable. Pour cela, il va, selon sa connaissance de l'outil informatique en général, de l'outil particulier qu'il utilise, et des données qu'il manipule, chercher à adapter ses requêtes et comprendre également les réponses de cette manière.

On peut donc noter cinq grand types d'utilisateurs possibles auxquels il faudra s'adapter. :

	Connaissance	Informatique	Outil particulier	Domaine	Ontologie particulière
Néophyte	Aucune	-	-	-	-
Habitué de l'outil l'informatique	Syntaxique	X	?	-	-
Spécialiste	Sémantique	?	?	X	
Utilisateur régulier	Syntaxique, contextuelle	X	X	?	?
Spécialiste utilisateur régulier	Syntaxique, sémantique, contextuelle	X	X	X	?

La question de l'adaptation sur le point de vue syntaxique ne nous intéresse pas ici : on se reportera à des revues traitant de l' IHM (Interface Homme Machine) si l'on veut que l'outil propose une interface de requête conviviale, respectant les conventions informatiques et permettant de poser les requêtes de manière intuitive.

Au niveau sémantique, l'adaptation consiste à chercher à « comprendre » les terme que l'utilisateur utilise s'il s'agit effectivement de termes exprimant des concepts d'un domaine couvert (sinon entièrement, au moins en partie) par l'une au moins des ontologies. « Comprendre », c'est-à-dire lui répondre, si possible selon le langage qu'il utilise, sinon en définissant au moins nos termes par rapport aux siens.

Une adaptation intermédiaire consiste à proposer à l'utilisateur un ensemble de terme proches de celui qu'il a saisi, et de lui demander de s'exprimer à l'aide de ces termes ; l'utilisateur doit donc s'adapter au niveau sémantique, mais l'outil a fait le premier pas permettant de parvenir à l'entente. C'est l'approche choisie par le CSTB.

Lorsqu'on considère plusieurs ontologies, on ne s'attendra pas à avoir une terminologie homogène, il est donc particulièrement important de s'adapter autant que possible à la terminologie utilisée par l'utilisateur ou au moins de lui proposer une terminologie uniforme pour atteindre l'ensemble des informations.

3.3.1.1 Adaptation à l'utilisateur grâce au contexte

3.3.1.1.1 Connaissance du « type » d'utilisateur

L'étude du contexte va nous permettre d'identifier le type d'utilisateur : selon la *terminologie* qu'il utilise, on saura si c'est un simple visiteur, ou s'il connaît tel ou tel domaine (on s'aidera pour cela de ressources linguistiques spécialisées). C'est important car il est facile de mal cerner le besoin de l'utilisateur en terme d'outil, et de l'obliger ainsi à s'adapter davantage pour obtenir les résultats qu'il souhaite, en devant de plus contourner les nouvelles « adaptations » mises en place.

L'utilisateur interagit avec les ontologies à travers un ensemble de requêtes, dans un certain ordre, et peut-être également en explorant l'arborescence d'une ou de plusieurs ontologies. *La suite des requêtes* de l'utilisateur, et ses actions, permettent de comprendre comment il utilise l'outil : si l'utilisation est différente de l'utilisation prévue, cela peut venir d'un fonctionnement insuffisamment clair ou intuitif, ou encore que l'utilisateur préfère avoir un moyen plus direct (par exemple) d'accéder aux données.

3.3.1.1.2 Compréhension de la démarche de l'utilisateur

L'observation chronologique des actions de l'utilisateur peut permettre de départager le temps de sa connexion en un certain nombre de requêtes bien distinctes (si c'est le cas), et aider ainsi à connaître les différents objets de ses requêtes. On pourra ainsi essayer de comprendre sa démarche, l'évaluer (combien d'étapes lui a-t-il fallu pour parvenir au résultat) et peut-être la faciliter ultérieurement.

A nouveau, la *terminologie* rencontrée donne des indications précieuses sur ce que l'utilisateur s'attendait à trouver ; si la terminologie est inadéquate, cela peut venir d'une mauvaise information sur ce qui est proposé ; s'il s'agit de termes appartenant (ou pouvant appartenir) au domaine, on reliera

l'information à l'objet de la requête qu'on aura pu déterminer : à une prochaine requête sur ce terme (par le même utilisateur ou un autre), on pourra demander s'il est intéressé par la même information au final. Par ailleurs, cette information peut toujours servir à titre statistique, par exemple pour connaître les termes réellement utilisés, et leur sens réel pour les utilisateurs de l'outil.

3.3.1.1.3 Couverture des ontologies pour le domaine : indicateur de couverture utilisateurs

Enfin, *savoir si le résultat a été positif* (pour l'utilisateur) ou négatif peut nous donner des indications sur la couverture des ontologies pour le domaine concerné. On peut ainsi construire un deuxième indicateur de couverture pour chaque ontologie en notant le nombre de réponses validées (un utilisateur sélectionne le concept et affiche les informations correspondantes sur le nombre de réponses validées pour le domaine entier.

Il faut noter que cet indicateur ne dépend pas uniquement de la couverture, mais aussi de l'objectif de l'utilisateur lors de sa recherche (comme les différentes ontologies sont reliées à différentes sources de données, il se peut que l'utilisateur soit plus intéressé par une source de données que par une autre).

3.3.2 Rôle du contexte propre à l'interaction avec l'utilisateur

Ce contexte permet de recenser les informations qu'on connaît pour un utilisateur particulier à travers l'ensemble de ses accès, de ses requêtes. Lorsqu'il a interrogé l'outil de manière (plus ou moins) régulière, on se rend mieux compte de ce qu'il recherche, et on saura déterminer une distance donnant la proximité d'un concept avec l'utilisateur.

Pour le contexte précédent, on cherchait à comparer l'utilisation d'un utilisateur à celle qui était attendue, ou peut-être à celle d'autres utilisateurs ; cette fois, on comparera la démarche de l'utilisateur dans le temps.

3.3.2.1 Adaptation au langage de l'utilisateur

Comme cela a été proposé au paragraphe précédent, on peut sauvegarder des liens entre les concepts rentrés par l'utilisateur et l'objet final de la requête : en effet, la démarche habituelle est d'abord de faire une recherche sur le premier terme qui nous vient à l'esprit pour représenter ce que nous recherchons, puis nous nous adaptons en fonction des résultats. Ce premier terme peut donc être une représentation possible de l'objet final recherché.

Ainsi, pour un utilisateur donné, on peut réaliser une liste de « correspondances » spécifiques entre les termes qu'il a cherchés (qu'il connaît et utilise) et ceux qui sont codés dans les ontologies. On pourra ensuite proposer les différentes listes à l'ensemble des utilisateurs, et étendre ainsi l'ontologie de manière itérative, comme le proposent [Wache et al. 2002], mais en gérant les points de vue lorsque les concepts ajoutés n'auront pas fait l'objet d'un consensus.

3.3.3 Rôle du contexte propre à la conceptualisation de l'utilisateur

L'utilisateur a un arrière-plan particulier, et des connaissances dans divers domaines, de manière plus ou moins spécialisée. Son interaction avec les ontologies via l'outil sera influencée également par sa culture, et par les langues qu'il pratique.

L'utilisateur n'acceptera pas de renseigner des informations qui caractérisent sa personnalité, mais il acceptera sans doute de saisir des informations s'il croit que cela pourra lui être bénéfique par la suite : ainsi, les langues qu'il pratique, ses domaines de connaissances, les métiers qu'il a exercés, ... Les informations saisies pourront ensuite être confrontées avec ce qu'on aura pu déduire lors de l'utilisation de l'outil.

Pour être sûr d'avoir une bonne qualité pour les réponses, on pourra demander la plus grande partie des renseignements après un certain nombre d'utilisations de l'outil, parce que l'utilisateur aura déjà vu l'utilité de l'outil ; on pourra de plus adapter le questionnaire.

3.3.3.1 Langues, culture

L'utilité de connaître les *langues* que pratique l'utilisateur ne fait aucun doute : s'il ne comprend pas le résultat, celui-ci ne lui est d'aucune aide. Pareillement, si l'utilisateur a une bonne connaissance d'une *culture* particulière, il sera plus à même d'interroger une ontologie réalisée par une communauté ayant cette même culture : il sera mieux préparé que d'autres pour se rendre compte des portées différentes que peuvent avoir les différents concepts.

3.3.3.2 Domaines et métier

De même, connaître les *domaines* maîtrisés par l'utilisateur permettra de lui proposer en priorité des termes qui devraient le concerner davantage, en tous cas des termes qu'il connaît et qu'il pourra rejeter sans difficulté en cas d'erreur. La connaissance des *métiers* qu'il a exercé peut apporter un complément d'information dans le cas d'une ontologie à granularité très fine : certains types de métiers utilisent des mots particuliers, et leur connaissance du domaine n'est pas la même.

3.4 Complexité du système à réaliser

La complexité du système mettant en oeuvre cette prise en compte du contexte pour réaliser l'appariement dépend du nombre d'informations prises en compte et de leur choix. On peut se limiter volontairement à un sous-ensemble d'informations pour ne pas avoir de difficultés de cet ordre. L'adaptation à l'utilisateur, même dans le cas où l'on s'adapte à son langage et qu'on fait évoluer l'ontologie de cette manière, ne pose pas de problèmes algorithmiques particuliers.

La difficulté vient plutôt lorsqu'on cherche à comprendre les concepts de par leur évolution. Dans ce cas, moins on a d'informations, et moins les informations sont pertinentes, et plus l'algorithme sera complexe. Néanmoins, il s'agit ici d'algorithmes « offline », et donc le problème n'est pas aussi gênant que si l'algorithme était exécuté à la demande de l'utilisateur.

On a donc tout intérêt à développer en parallèle un module qui s'insère dans un éditeur d'ontologie, afin de renseigner le plus tôt possible ce qu'on peut connaître sur les diverses intentions de conception (l'outil peut semi automatiser les saisies de l'utilisateur à ce niveau)

3.5 Méthodologie de mise en oeuvre

Nous avons cherché à connaître les informations contextuelles associées aux ontologies et ce que chacune d'elles pouvait apporter dans la compréhension de la conceptualisation, afin de pouvoir notamment faire des choix de concepts en connaissance de cause. Nous nous sommes renseignés sur le contexte connu de l'utilisateur afin de prendre en compte ses besoins. Maintenant, il nous faut déterminer comment développer des systèmes permettant d'en prendre compte.

3.5.1 Sélection d'informations contextuelles

Pour commencer, il nous faut effectuer une *sélection d'informations contextuelles* car on ne peut pas prendre en compte tout le contexte : les informations ne sont pas forcément toutes disponibles, ou souhaitables (on prendra garde de ne pas porter atteinte à la vie privée). Il nous faut prendre du contexte les éléments les plus pertinents, qui peuvent nous aider à mieux comprendre l'ontologie et l'utilisateur. Les éléments significatifs pouvant varier selon les ontologies considérées, on ne se limitera pas à une sélection trop restreinte.

On pourra classer les éléments de cette sélection en :

- Informations sur les ontologies
 1. Informations d'ordre général sur l'ontologie
 2. Informations sur des éléments d'une ontologie
- Informations sur les utilisateurs
 3. Informations d'ordre général sur l'utilisateur

Lors de cette sélection, il nous faut faire attention d'avoir pour chaque donnée concernant l'utilisateur une donnée correspondante pour l'ontologie ou un élément de l'ontologie, à moins qu'elle n'ait une utilité statistique voulue.

Comme exemple, considérons l'information « métier(s) de *l'utilisateur* ». Il nous faudra alors une information correspondante « métier(s) pour lesquels *l'ontologie* propose des concepts », ou une information « métier(s) pour lequel le *concept* a un sens ».

Deux autres types d'informations sont ajoutés lors de l'appariement des ontologies :

4. Indicateurs calculés à partir des données fournies ou récupérées sur l'ontologies
5. Évaluations (pour différentes mesures données) de la proximité entre deux concepts

Enfin, pendant la connexion de l'utilisateur respectivement :

6. Informations sur chaque connexion de l'utilisateur

A chaque paire d'ancres correspond un ensemble de valeurs numériques exprimant chacune l'évaluation de la distance entre les deux ancres pour une information (attachée aux éléments des ontologies) donnée. Il est envisageable de rattacher aux paires d'ancres d'autres informations spécifiques qu'on aura obtenues lors de l'appariement.

3.5.2 Spécifications et structuration des informations

Ceci fait, il nous faut *réaliser les spécifications* correspondantes, soit déterminer la typologie des informations choisies, et déterminer comment structurer les données pour les utilisateurs, les ontologies, etc. Plusieurs variantes sont possibles, il n'y a donc pas de schéma prédéfini.

L'information recherchée doit tout d'abord être quantifiable, afin de pouvoir faire l'objet d'un traitement automatique, et comparable de la manière la plus fiable possible ; on évitera par exemple les chaînes de caractères, à moins de fournir un ensemble de valeurs prédéfinies.

Pour notre exemple, l'idéal est sans doute de fournir une taxonomie de métiers exhaustive pour le domaine concerné, et de contraindre la sélection d'une ou plusieurs valeurs dans cette hiérarchie : cela permet de bien délimiter les différentes possibilités, d'avoir la précision désirée (selon l'utilisation qu'on veut en faire), de réduire le temps de calcul (comparaison suivant la parenté entre les valeurs), et permet une optimisation (utile si l'information est stockée pour chaque concept). On peut trouver un exemple d'une mise en oeuvre d'une telle démarche dans [Link-Pezet, Ertscheid, 2002].

Ces spécifications écrites, il restera à déterminer sous quelle forme enregistrer les données (base de données, journal...), à initialiser la base de données le cas échéant, et à développer des algorithmes pour récupérer et mettre à jour les données.

3.5.3 Développement de fonctions d'interprétations

Nous avons besoin ensuite de *fonctions d'interprétation* pour chaque information décrites, sauf celles décrivant le contexte de l'utilisateur ; les valeurs numériques associées à chaque paire d'ancre sont déjà le résultat d'interprétation, et ne sont donc plus évaluées. Le résultat de chaque fonction sera une valeur numérique dans un intervalle déterminé, par exemple entre 0 et 1 inclus ou entre 0 et 100 inclus.

On en a besoin pour interpréter les données de l'ontologie (par exemple en les comparant aux autres) lors de l'appariement offline

3.5.4 Appariement

Comme il peut s'avérer coûteux en temps et en mémoire d'effectuer de nouveaux appariements d'ontologies à chaque connexion de l'utilisateur, où même à la première fois que l'utilisateur veut utiliser une nouvelle ontologie (ce qui est irréaliste, car il faut mettre à jour l'appariement d'après les évolutions des ontologies), une solution intéressante semble être de réaliser l'appariement en deux étapes. On peut ainsi appairer les ontologies deux à deux (ou plusieurs à la fois, pour reprendre la méthodologie de [He et al. 2003]. Cependant, sauvegarder les articulations par couple d'ontologies permet de limiter le nombre d'ancres à vérifier à chaque demande, en important uniquement celles correspondant aux ontologies sélectionnées).

Nous proposons donc de réaliser l'appariement en deux étapes :

- une étape « *offline* » **générique**, pouvant être réalisée hors connexion, prendre beaucoup de temps, nécessiter une intervention humaine,

- une étape « *online* » (dynamique) **spécifique à l'utilisateur**, réalisée pour répondre à une requête effectuée par l'utilisateur, qui doit donc être quasi-instantanée (d'autant plus que beaucoup d'ontologies peuvent être concernées), et totalement automatisée.

Pour parvenir à diminuer le temps d'exécution de L'appariement online se basera donc l'appariement générique effectué, et utilisera les valeurs propres à l'utilisateur pour le personnaliser, de manière dynamique.

3.5.5 Appariement offline

Il s'agit ici de réaliser l'appariement entre ontologies. Après une mise à jour de celles-ci, une première interprétation de données contextuelles de l'ontologie permet de déterminer un premier *vecteur d'interprétation de l'ontologie*.

Un parcourt de l'ontologie est alors effectué, afin de déterminer quels concepts sont comparables, selon une distance sémantique ou plutôt selon une combinaison de ces distances : les concepts sont alors comparés selon la *distance sémantique contextuelle offline*, afin de former un *vecteur contextuel offline*. Ce vecteur est alors sauvegardé (pour chaque paire d'ancres) dans une ontologie de correspondance.

L'appariement est donc constitué de quatre étapes principales :

1. *Mise à jour* les données sur les ontologies à appairer (nombre de concepts, etc.)
2. *Interprétation* des données, notamment par comparaison avec des autres ontologies du même domaine, afin d'obtenir un vecteur *d'indicateurs* pour l'ontologie (cf. Figure 1. Interprétation des données des ontologies).

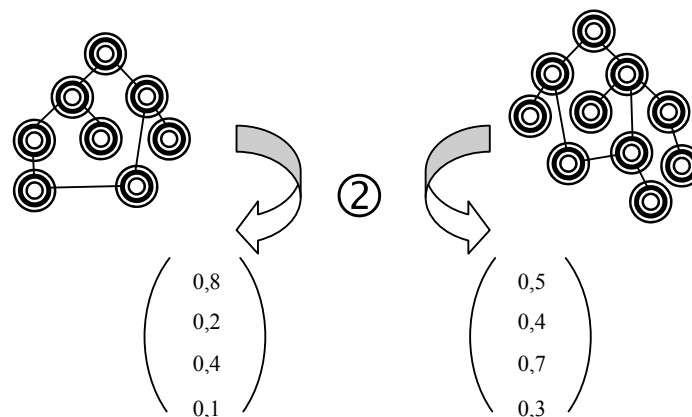


Figure 1. Interprétation des données des ontologies

3. *Sélection* des concepts à appairer à l'aide d'un *algorithme d'appariement sémantique*
4. *Comparaison* des concepts des deux ontologies à appairer à l'aide d'une *distance sémantique contextuelle offline* afin d'obtenir un *vecteur contextuel offline*(cf. Figure 2. Comparaison des concepts des ontologies).

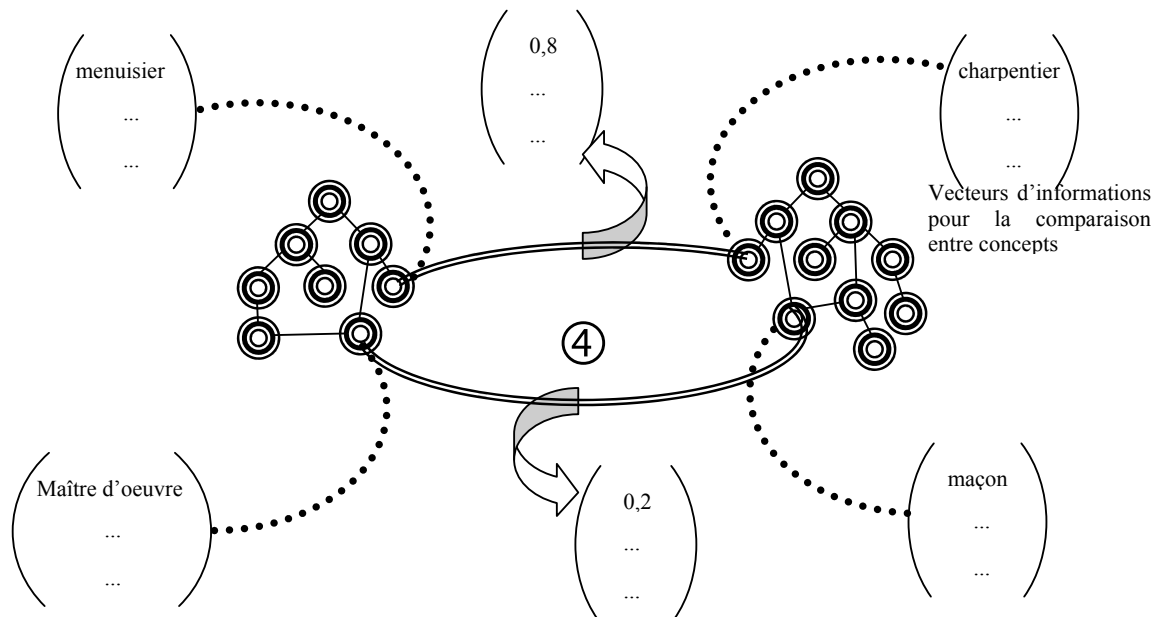


Figure 2. Comparaison des concepts des ontologies

3.5.6 Appariement online

L'appariement online fait usage des informations disponibles sur l'utilisateur et de la configuration que celui-ci aura indiquée. Une sélection d'ontologies est réalisée préalablement selon ce fichier et selon le *vecteur d'interprétation* de l'ontologie, ainsi que selon les informations comparables à l'utilisateur (la langue par exemple, si on choisit de la définir pour l'ontologie entière et non pour chaque concept).

Nous avons considéré que la requête utilisateur portait sur un concept d'une ontologie donnée, afin de rechercher les concepts équivalents dans les autres ontologies. On utilise le résultat de l'appariement offline : on cherche dans l'ontologie de correspondance les concepts équivalents, pour lesquels on récupère les *informations contextuelles comparables à celles de l'utilisateur*. On utilise la *distance sémantique contextuelle online* pour comparer les valeurs des concepts avec celles de l'utilisateur, ce qui nous donne un *vecteur contextuel online*.

Le fichier de configuration permet alors de pondérer le vecteur constitué par le vecteur d'interprétation de l'ontologie, le vecteur contextuel offline, le vecteur contextuel online. Un produit de matrices permet d'obtenir une mesure à une dimension, permettant de faire le tri. La proximité entre concepts et la proximité avec l'utilisateur peuvent être indiquées pour chaque concept.

L'appariement online est donc constitué de trois étapes :

5. Examen du fichier de *configuration de l'utilisateur*, et *sélection* des ontologies utiles d'après la liste d'indicateur correspondant à chaque ontologie
6. Récupération (à partir d'un concept donné) des concepts correspondants dans ces ontologies (en utilisant les appariements déjà réalisés) pour les *comparer* à l'aide d'une *distance contextuelle online*, et récupération d'un *vecteur contextuel online* (cf. Figure 3. Comparaison entre caractéristiques des ontologies et caractéristiques de l'utilisateur).

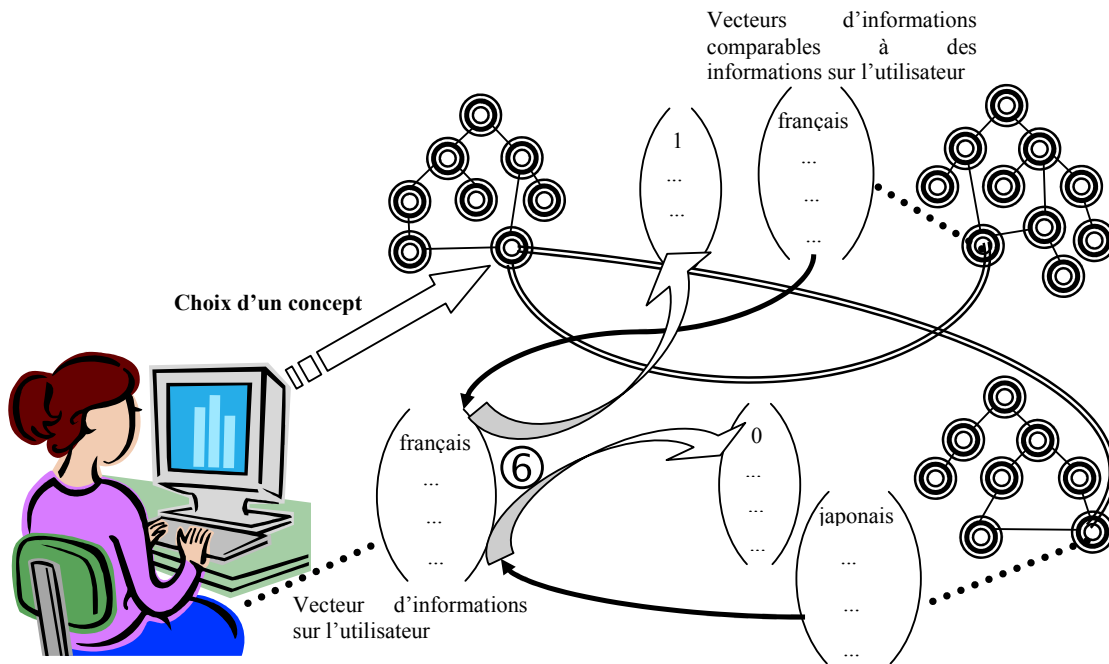
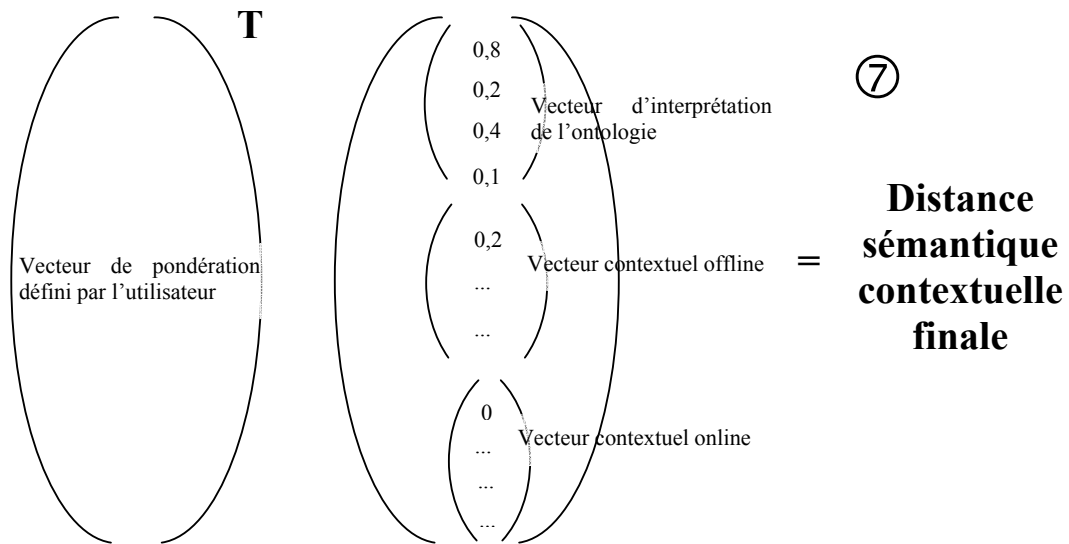


Figure 3. Comparaison entre caractéristiques des ontologies et caractéristiques de l'utilisateur

- Application de *coefficients de pondération*, selon le fichier de configuration par défaut paramétrable par l'utilisateur.



On peut remarquer qu'on n'a pas tenu compte ici de la possibilité d'avoir des informations sur l'ontologie comparables à des informations de l'utilisateur, qui pourraient servir d'élément de comparaison (en plus de son utilité pour la sélection).

3.5.7 Définition des distances contextuelles offline et online

Les distances contextuelles offline et online consistent en une combinaison d'algorithmes permettant de mesurer le rapprochement entre deux valeurs différentes pour une information contextuelle. Chacun de ces algorithmes est appliqué séparément sur l'information correspondante présente dans le vecteur d'information du concept (pour la distance offline) dans le vecteur d'information du concept comparable aux informations de l'utilisateur (pour la distance online).

Par exemple, un algorithme peut mesurer le rapprochement entre deux langues données. Si les valeurs sont « français » et « français canadien », le résultat obtenu peut être 0,9. Si les valeurs sont : italien, italien, le résultat obtenu sera 1. Deux langues de même origine (latine, germanique....) pourront donner un résultat intermédiaire : entre « italien » et « espagnol », on pourra donner 0,3 .

4 Application

Rappelons que notre travail s'inscrit dans un partenariat avec le CSTB, afin de participer à l'évolution du serveur d'ontologies e-COSer. Ainsi, le travail de [Abdul Ghafour 2004] permet de disposer d'ontologies correspondant à divers catalogues du secteur du bâtiment dans le même langage OWL DL. Ainsi, e-COSer, à travers son outil de reformulation de requête, basée sur l'ontologie de plus haut niveau eCognos, a la possibilité de renvoyer aux différents termes des divers catalogues.

Mais pour cela, il faut encore associer à chaque concept d'eCognos les concepts équivalents des catalogues. Ma contribution est de déterminer une distance qui puisse prendre en compte le contexte particulier de chaque catalogue, ainsi que l'utilisateur.

4.1 Intégration dans une architecture plus vaste

Le calcul de distances sémantiques s'insère dans une vision d'architecture logicielle pour l'interopérabilité (voir schéma en page annexe).

Cette architecture comprend deux modules permettant la liaison avec les catalogues (traduction préalable des ontologies dans un format commun puis lien entre concepts identiques de l'une et l'autre ontologies à l'aide de l'URI (pour les taxonomies en XML ou autres, il est possible de configurer un espace de nom particulier).

L'appariement est réalisé à l'aide d'un algorithme recherchant et sélectionnant les concepts similaires entre deux ontologies. Une fois ceux-ci sélectionnés, plusieurs mesures de distances sémantiques et contextuelles sont appliquées, afin d'obtenir un vecteur d'appariement offline, qui est sauvegardé avec la relation entre concepts dans une ontologie de correspondance.

Lors de la connexion de l'utilisateur à e-COSer pour effectuer une requête, le serveur d'ontologies récupère l'information propre à l'utilisateur pour opérer une sélection parmi les ontologies à considérer. L'outil d'aide à la reformulation de requête transmet les choix de concepts au module général d'appariement, qui recherche alors parmi les ontologies sélectionnées, les concepts similaires ; il compare alors les informations associées à ces concepts avec les informations propres à l'utilisateur, afin de réaliser un vecteur d'appariement online.

Un fichier de configuration proposé par défaut à l'utilisateur et paramétrable permet de choisir de tenir plus ou moins compte des différents calculs de distance pour l'un et l'autre vecteur, et de déterminer la proportion de prise en compte de l'utilisateur dans le calcul global.

Le résultat est ensuite présenté à l'utilisateur dans l'ordre décroissant selon la mesure décrite dans le chapitre trois, sous « Méthodologie de mise en oeuvre ».

4.2 Outils utilisés

Dans la mesure du possible, nous avons voulu utiliser les outils logiciels choisis par le CSTB et utilisés pour le développement du serveur d'ontologies e-COSer : le moteur de servlets [Tomcat], le langage de programmation [Java], enfin l'API de manipulation d'ontologies en RDF, DAML et OWL [JENA].

Jena étant davantage prévu pour manipuler la syntaxe RDF que pour manipuler les objets OWL de manière intuitive, nous avons étudié la possibilité d'utiliser OWL API [Bechhofer et al. 2003]. Cette API permet de manipuler des ontologies en OWL en s'abstrayant de la syntaxe RDF habituellement sous-jacente, et donc est plus adapté à une utilisation uniquement sémantique. La simplicité de notre implémentation nous a permis d'utiliser Jena malgré tout.

Nous avons choisi de sauvegarder les informations contextuelles en XML, qui est un langage standard, souple (notre première idée était d'utiliser Jena pour manipuler ces informations supplémentaires, et donc de les enregistrer au format RDF ou OWL, qui présentent une structure beaucoup plus rigide) et relativement simple à manipuler et à comprendre. Pour manipuler les informations depuis Java, nous avons choisi l'outil [JAXB] de Sun, qui permet de faire du « *XML binding* » : mise en correspondance automatique entre documents XML et objets JAVA.

4.3 Architecture logicielle

L'implémentation que nous voulons réaliser peut donc être divisée en quatre parties :

- Une partie « Interface », développée en Java (servlets) pour une utilisation sous Tomcat
- Une partie « Appariement online », développée en Java, utilisant le « *XML binding* »
- Une partie « Appariement offline », développée en Java et utilisant Jena et le « *XML binding* »
- Une partie « *XML binding* » contenant les classes générée automatiquement en Java à partir des fichiers XSD (« XML Schema Definition »), ainsi que l'ensemble des fichiers XML permettant de sauvegarder l'information.

Ce stage m'a donné l'occasion d'apprendre à me servir de Tomcat et à développer des servlets, à utiliser Jena pour manipuler des ontologies, à manipuler les XML Schemas, et à utiliser JAXB pour générer des classes Java, que j'ai appris à manipuler également.

Le manque de temps m'a fait m'arrêter à l'écriture des fichiers XML et XSD correspondant aux informations contextuelles choisies (paragraphe suivant), mais j'espère pouvoir faire une démonstration légère prochainement, implémentant les quatre parties décrites ci-dessus.

4.4 Sélection de données

Les ontologies que nous devons manipuler ont été traduites à partir de catalogues développés indépendamment. La méthodologie indiquée pour prendre en compte le contexte des éléments de l'ontologie n'a donc pas été suivie, et nous n'avons d'ailleurs aucune autre information qui soit similaire. Cependant, pour l'exemple, nous allons attacher quelques informations aux « concepts » de l'ontologie, afin de montrer comment les utiliser.

Les données concernant l'interaction avec l'utilisateur nécessitent un grand nombre de connexions pour apporter des éléments intéressants. Comme nous sommes limités en temps et que nous voulons que l'exemple puisse présenter des résultats visuels, nous avons choisi de ne pas développer cet aspect.

Enfin, nous n'avons que deux ontologies à disposition pour le domaine. Nous ne pouvons donc pas implémenter l'indicateur de couverture dont nous avons parlé plus haut.

Reprenons donc la classification des informations indiquée dans la méthodologie de mise en oeuvre, et complétons-la avec les informations disponibles. Nous avons :

1. Informations d'ordre général sur les ontologies
 - Le langage d'ontologies d'origine
 - La structure sur laquelle repose ce langage
 - La règle terminologie permettant de reconnaître les labels
 - Le nombre de concepts
 - Les domaines
2. Informations sur des concepts d'une ontologie
 - Les métiers pour lesquels ce concept fait sens. Cette information peut être déduite automatiquement pour la plupart des concepts si l'on dispose de dictionnaires spécialisés. Dans notre cas, elle sera décrite manuellement.
 - La langue utilisée pour décrire le concept. Cette information pourrait tout aussi bien être attachée à l'ontologie, mais l'ontologie utilisée par e-COSer contenant des concepts en plusieurs langues, nous avons jugé préférable de nous adapter.
3. Informations d'ordre général sur l'utilisateur
 - Langues pratiquées
 - Domaines connus
 - Métiers pour lesquels l'utilisateur a développé une expérience
4. Indicateurs calculés à partir des données fournies ou récupérées sur l'ontologies
 - Indicateur d'activité
 - Indicateur de biais possible
 - Indicateur de taille relative
5. Informations sur la relation entre deux éléments (ou paire d'« *ancres* ») dans le cadre d'un appariement entre deux ontologies
 - (*nous n'avons rien prévu d'ajouter ici*)

4.5 Spécifications

On trouvera deux schémas entité-relations montrant la structuration des informations en annexe.

J'ai développé une partie des fichiers exemples correspondants en XML, validés par des schémas XML spécifiques (également en annexe).

4.6 Fonctions d'interprétation

Non réalisées par manque de temps.

4.7 Appariement offline

Non réalisées par manque de temps.

4.8 Appariement online

Non réalisées par manque de temps.

5 Conclusion

L'originalité de notre approche est de prendre en compte à la fois le contexte des éléments des ontologies comparées lors de l'appariement et celui de l'utilisateur. On soulignera également la possibilité qui est donnée à l'utilisateur de paramétrer lui-même comment les informations sont prises en compte, et qui est donc l'« auteur » de l'appariement final.

Il reste encore un travail d'implémentation plus conséquent à faire, afin de démontrer la pertinence des solutions proposées. Nous nous sommes limité également à rechercher l'apport du contexte dans l'appariement, et n'avons donc pas conçu d'algorithme qui recherche les concepts à appairer suivant différents critères sémantiques comme indiqué dans l'état de l'art.

5.1 Perspectives

Nous avons commencé à explorer le contexte de l'ontologie, qui nous donne des informations permettant de juger la pertinence de l'information dispensée pour une recherche donnée d'un utilisateur particulier. Nous nous sommes restreints pour cette étude à des relations d'équivalence simple entre concepts, mais il serait intéressant de pouvoir prendre en compte des relations plus complexes dans le calcul de distance sémantique.

Nous avons présenté dans le chapitre précédent une architecture qui permet de mettre en relation un grand nombre d'ontologies et taxonomies pour le domaine de la construction, en établissant pour chacune des sources le méta-modèle approprié. Cette architecture que nous venons de présenter reste encore un défi, son implémentation permettra de démontrer la pertinence de l'approche et des solutions décrites, car les sources de données (catalogues) sont développés de manière tout à fait indépendante, dans des buts différents et pour des métiers très variés.

Dans le cadre de cette architecture, on notera l'intérêt possible d'utiliser une approche basée sur la corrélation [He et al. 2003], car la philosophie du projet est de réaliser un « pivot sémantique » entre un grand nombre de catalogues ; les informations communément présentes dans ces catalogues peuvent permettre de réaliser des appariements plus sûrs.

Enfin, on pourra chercher à utiliser les multi-agents apprenants pour atteindre des sources pour lesquelles aucun méta-modèle n'a été réalisé. [Gasser, Wang 2002], [SU et al. 2003] sont des exemples d'utilisation des multi-agents au service de l'appariement des ontologies. On pourra également consulter [Beslon 2002], [Guessoum 2002] et [Kaplan 2002] à ce sujet. L'application du contexte dans ce cas là peut emprunter l'approche basée sur les médiateurs proposée par [Bressan et al. 1997].

Bibliographie

- [Abdul Ghafour 2004] Samer ABDUL GHAFOUR. Méthodes et outils pour l'intégration des ontologies. Rapport de Stage de DEA.
- [Bao 2003] Jie BAO. Bibliography on Ontology integration [en ligne]. Disponible sur : <http://www.cs.iastate.edu/~baojie/acad/reference/2003-02-25_ontomas.htm> (consulté le 13/05/2004)
- [Bechhofer et al. 2003] S. BECHHOFFER, R.VOLZ, P. LORD,. Cooking the Semantic Web with the OWL API. **In**: *First International Semantic Web Conference 2003 (ISWC 2003)*, October 21-23, 2003, Sanibel Island, Florida. p 659-674.
- [Benslimane et al. 2003] Djamal BENSLIMANE, Christelle VANGUENOT, Catherine ROUSSEY et all. Multirepresentation in ontologies. *Advances in Databases and Information Systems, Lecture notes in Computer Science* [en ligne]. 2003, vol 2798, pp4-15. ISSN: 0302-9743
- [Berners-Lee 1998] Tim BERNERS-LEE. Semantic Web Road Map [en ligne]. Disponible sur : <<http://www.w3.org/DesignIssues/Semantic.html>> (consulté le 22/06/04)
- [Beslon 2002] BESLON Guillaume. Coopération implicite et émergence dans des groupes d'agents. **In** : GUILLOT Agnès, DAUCE Emmanuel. *Approche dynamique de la cognition artificielle*. Lavoisier, Paris, 2002, pp269-283.
- [Bouquet et al. 2002] Paolo BOUQUET, Antonia DONA, Luciano SERAFINI et al. ConTeXtualized local ontology specifications via CTXML. **In** : Paolo BOUQUET, Frank van HARMELEN, Fausto GIUNCHIGLIA et all. *AAAI-02 workshop on Meaning Negotiation*. AAAI press. 28 July 2002, Edmonton, Alberta.
- [Bouquet et al. 2003a] Paolo BOUQUET, Bernardo MAGNI, Luciano SERAFINI et all. A SAT-Based Algorithm for Context Matching. **In** : *4th International and Interdisciplinary Conference, CONTEXT 2003*. June 23-25, 2003, Stanford, CA, USA. Springer, 2003, Lecture Notes in Computer Science, vol. 2680, pp 66-79.
- [Bouquet et al. 2003b] Paolo BOUQUET, Fausto Giunchiglia, Frank van Harmelen et al. . C-OWL : Contextualizing ontologies. **In** : *2nd International Semantic Web Conference (ISWC2003)*. 20-23 October 2003, Sundial Resort, Sanibel Island, Florida, USA. Disponible sur : <www.cs.vu.nl/~frankh/postscript/ISWC03.pdf> (consulté le 22/03/04)
- [Bouquet, Serafini 2001] Paolo Bouquet, Luciano Serafini. Two formalizations of context: a comparison. **In** : *Proceedings of the Third International and Interdisciplinary Conference on Modeling and Using Context (CONTEXT'01)*, July 27-30, 2001, Dundee, Scotland. Springer Verlag, Lecture Notes in AI Volume 2116, July 2001 [en ligne]. Disponible sur : <<http://dit.unitn.it/~bouquet/pers-publ-engl.html>> (consulté le 26/06/04)
- [Bouquet, Serafini 2003] Paolo BOUQUET, Luciano SERAFINI. On the difference between bridge rules and lifting axioms. **In** : *Proceedings of CONTEXT 2003: Fourth International and Interdisciplinary Conference on Modeling and Using Context*, June 2003, Stanford, USA. Lecture Notes in Artificial Intelligence, volume 2680, pp 80-93 [en ligne]. Disponible sur : <<http://dit.unitn.it/~bouquet/papers/CONTEXT2003-Comparison.pdf>> (consulté le 26/06/04)
- [Bressan et al. 1997] S. BRESSAN, C. GOH, K. FYNN, et al. Demonstration of the context interchange mediator prototype. **In** : *Proceedings of ACM SIGMOD/PODS Conference on*

- Management of Data*, May 1997, Tucson, AZ. Disponible sur : <<http://context2.mit.edu/coin/publications/sigmod97/sigmod97.pdf>> (consulté le 21/04/04)
- [Brézillon 1998] Patrick BRÉZILLON. Context in Problem Solving : A Survey. Paris, 1998.
- [Brézillon, Pomerol 1999] Patrick BRÉZILLON, J. –Ch. POMEROL. Contextual knowledge sharing and cooperation in Intelligent Assistant Systems. *Le Travail Humain*, 1999, Vol. 62, n°3, pp223-245.
- [Brézillon, Pomerol 2001] Patrick BRÉZILLON, J. –Ch. POMEROL. Some comments about knowledge and context [en ligne]. Disponible sur : <<http://www-ftp.lip6.fr/ftp/lip6/reports/2001/lip6.2001.022.pdf>> (consulté le 01/04/04).
- [Bück 2003] Jean-Yves BÜCK. Le management des connaissances et des competences en pratique. Paris, France: Editions d'organisation, 2003, 329pp. ISBN : 2-7081-2870-1
- [Buvac, Mason 1993] Sasa BUVAC, Ian A. MASON. Propositional Logic of Context. **In** : *Eleventh National Conference on Artificial Intelligence*, 1993.
- [Chalupsky 2000] H. CHALUPSKY. OntoMorph: A Translation System for Symbolic Knowledge. **In** : Anthony G. COHN, Fausto GIUNCHIGLIA, Bart SELMAN. *KR 2000, Principles of Knowledge Representation and Reasoning, Seventh International Conference*, Breckenridge, Colorado, USA, April 11-15, 2000 [en ligne]. Breckenridge : Morgan Kaufmann Publishers, 2000, pp 471-482. Disponible sur : <<http://citeseer.ist.psu.edu/chalupsky00ontomorph.html>> (consulté le 22/05/04)
- [Chen, Doumeingts 2003] David CHEN, Guy DOUMEINGTS . Basic Concepts and Approaches to Develop Interoperability of Enterprise Applications. **In** : *Fourth Working Conference PRO-VE'03, Processes and Foundations for Virtual Organisations*, 29-31 October 2003, Lugano, Switzerland. Kluwer Academic Publisher, pp. 323-330. ISBN 1-4020-7638-X.
- [CSTB] CSTB. Centre Scientifique et Technique du Bâtiment [en ligne]. Disponible sur : <<http://www.cstb.fr/>> (consulté le 24/05/04)
- [Do et al. 2002] H. Do, S. Melnik, E. Rahm. Comparison of schema matching evaluations. **In** : *Proceedings of the 2nd Int. Workshop on Web Databases (German Informatics Society) at NETObjectDays'02 - LNCS 2593*, Octobre 2002, Erfurt, Germany [en ligne]. Disponible sur : <<http://www.netobjectdays.org/pdf/02/papers/ws-webdb/02-Do.pdf>> (consulté le 26/06/04)
- [Do, Rahm 2002] Hong-Hai DO, Erhard RAHM. COMA - A System for Flexible Combination of Schema Matching Approaches. **In** : *Proceedings of the 28th Intl. Conference on Very Large Databases (VLDB)*, Août 2002, Hongkong [en ligne]. Disponible sur : <<http://www.vldb.org/conf/2002/S17P03.pdf>> (consulté le 25/06/04).
- [Doan et al. 2002] A. Doan, J. Madhavan, P. Domingos et al. . Learning to Map between Ontologies on the Semantic Web. **In** : *the eleventh International World Wide Web conference (WWW2002)*, May 7-11, 2002, Honolulu, Hawaii, USA ACM 1-58113-449-5/02/0005 [en ligne]. Disponible sur : <<http://www.cs.washington.edu/homes/alon/site/files/glue.pdf>> (consulté le 25/06/04)
- [Euzenat, Valtchev 2003] Jérôme EUZENAT, P. VALTCHEV. An Integrative proximity measure for ontology alignment. **In** : DOAN A., HALEVY A., NOY N. . *Semantic Integration Workshop*, October 20, 2003, Sanibel Island, Florida [en ligne]. Sanibel Island : CEUR, 2003. Disponible sur : <<http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-82/>> (consulté le 22/05/04)
- [Gamper et al. 1999] J. GAMPER., W. NEJDL , M. WOLPERS . Combining ontologies and terminologies in Information Systems [en ligne]. Disponible sur : <<http://www.kbs.uni-hannover.de/Arbeiten/Publikationen/1999/tke99/>> (consulté le 23/05/04)
- [Ghidini, Giunchiglia 1998] F. GIUNCHIGLIA and C. GHIDINI. Local Models Semantics, or Contextual Reasoning = Locality + Compatibility. **In** : *Proceedings of the Sixth International Conference on Principles of Knowledge Representation and Reasoning (KR'98)*, June 2-5, 1998,

Trento, Italy. Morgan Kaufmann, pages 282-289 [en ligne]. Disponible sur : sra.ite.it/people/ghidini/publications/AIJ01.pdf (consulté le 26/06/04)

[Gruber 1993] Thomas R. GRUBER. A translation approach to portable ontology specifications. Elsevier Science, Knowledge Acquisition, vol. 5, n°2, Juin 1993 [en ligne]. Disponible sur : <http://www.dbis.informatik.hu-berlin.de/lehre/WS0203/SemWeb/lit/KSL-92-17.pdf> (consulté le 25/06/04)

[Gruber] Gruber T. . What is an ontology ? [en ligne]. Disponible sur : <http://www-ksl.stanford.edu/kst/what-is-an-ontology.html> (consulté le 26/04/04).

[Guessoum 2002] GUESSOUM Zahia, Modèles multi agents et systèmes dynamiques. **In** : GUILLOT Agnès, DAUCE Emmanuel. *Approche dynamique de la cognition artificielle*. Lavoisier, Paris, 2002, pp255-268.

[Guha, McCarthy 2003] R. Guha, John McCarthy. Varieties of Contexts. . **In** : *4th International and Interdisciplinary Conference, CONTEXT 2003*. June 23-25, 2003, Stanford, CA, USA. Springer, 2003, Lecture Notes in Computer Science, vol. 2680, pp 164-177

[He et al. 2003] Bin HE, Kevin Chen-Chuan CHANG, Jiawei HAN. Automatic Complex Schema Matching across Web Query Interfaces: A Correlation Mining Approach. Technical Report UIUCDCS-R-2003-2388, Department of Computer Science, UIUC, December 2003. Disponible sur : www.cs.uiuc.edu/Dienst/Repository/2.0/Body/ncstrl.uiuc_cs/UIUCDCS-R-2003-2388/pdf (consulté le 13/05/2004)

[Horrocks et al. 2003] Ian HORROCKS, Peter F. PATEL-SCHNEIDER, Frank van HARMELEN. From SHIQ and RDF to OWL : The Making of a Web Ontology Language. Journal of Web Semantic [en ligne]. 2003, vol 1, n°1, 31pp. Disponible sur : <http://www.websemanticsjournal.org/volume1/issue1/Horrocksetal2003/Horrocksetal2003.pdf> (consulté le 23/03/2004)

[Wache et al. 2001] H. WACHE, T. VOGELE, U. VISSER et al.. Ontology-based integration of information --- a survey of existing approaches. **In**: H. STUCKENSCHMIDT. *IJCAI-01 Workshop: Ontologies and Information Sharing*, 2001, Seattle, WA., pp.108-117. Disponible sur : <http://www-agki.tzi.de/buster/new/publications.html> (consulté le 03/05/2004).

[Java] Sun. Java Technology [en ligne]. Disponible sur : <http://java.sun.com/> (consulté le 27/06/04)

[Jaxb] Sun. Java Architecture for XML binding [en ligne]. Disponible sur : <http://java.sun.com/xml/jaxb/> (consulté le 27/06/04)

[JENA] Hewlett-Packard Development Company LP. Jena – A Semantic Web Framework for Java [en ligne]. Disponible sur : <http://jena.sourceforge.net/> (consulté le 25/06/04)

[Kalfoglou, Schorlemmer 2003] Yannis KALFOGLOU, Marco SCHORLEMMER. Ontology mapping: the state of the art. Knowledge Engineering Review, 2003, issue 2, volume 18, pp 1-31.

[Kaplan 2002] KAPLAN Frédéric. Construction sociale du sens. **In** : GUILLOT Agnès, DAUCE Emmanuel. *Approche dynamique de la cognition artificielle*. Lavoisier, Paris, 2002, pp201-217.

[Klein 2001] KLEIN M. . Combining and relating ontologies : an analysis of problems and solutions. **In** : GÓMEZ-PÉREZ A., GRUNINGER M., [STUCKENSCHMIDT H.](#) *IJCAI-01 Workshop on Ontologies and Information Sharing*, August 4-5, 2001. Seattle, USA [en ligne]. Seattle : CEUR, 2001, pp53-62. Disponible sur : <http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-47/> (consulté le 23/05/04)

- [Link-Pezet, Ertscheid 2002] Jo LINK-PEZET, Olivier ERTZSCHEID. FoRSIC : de l'ingénierie des compétences à l'ingénierie des connaissances. Communication. 21 juin 2002. 6pp. Disponible sur : <http://archivesic.ccsd.cnrs.fr/sic_00000074.html> (consulté le 12/03/2004).
- [Madhavan et al. 2001] Jayant MADHAVAN, Philip A. BERNSTEIN, Erhard RAHM. Generic Schema Matching with Cupid. **In** : *Proceedings of Very Large Databases*, September 11-14, 2001, Roma, Italy. The {VLDB} Journal, 2001, pp49-58. Disponible sur : <<http://research.microsoft.com/~philbe/CupidVLDB01.pdf>> (consulté le 21/04/04)
- [SU et al. 2003] X. SU, M. MATSKIN, J. RAO., Implementing Explanation Ontology for Agent System. **In** : *IEEE/WIC International Conference on Web Intelligence (WI'03)*, October 13-17, 2003, Halifax, Canada [en ligne]. Disponible sur : <www.idi.ntnu.no/~xiaomeng/paper/WI2003CRC.pdf> (consulté le 17/05/04)
- [McGuinness et al. 2000] Richard Fikes, Deborah L. McGuinness, James Rice, et al. . An Environment for Merging and Testing Large Ontologies. **In** : *Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning (KR2000)*, 12-15 April 2000, Breckenridge, Colorado, USA
- [McCarthy 1993] John McCARTHY. Notes on Formalizing Context. **In** : *13th International Joint Conference on Artificial Intelligence*, 1993, Chambéry, France, pp 555-560.
- [Melnik et al. 2002] Sergey MELNIK, Hector GARCIA-MOLINA, Erhard RAHM. Similarity Flooding: A Versatile Graph Matching Algorithm and Its Application to Schema Matching. **In** : *Proceedings of the 18th International Conference on Data Engineering*, 26 February - 1 March 2002, San Jose, CA. IEEE Computer Society, 2002, pp 117-128. ISBN 0-7695-1531-2 [en ligne]. Disponible sur : <http://www-db.stanford.edu/~melnik/pub/melnik_ICDE02.pdf> (consulté le 25/06/04)
- [Noy, Musen 2000] N. F. NOY, M. A. MUSEN. PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment. **In** : *Seventeenth National Conference on Artificial Intelligence (AAAI-2000)*, July 30 - August 3, 2000, Austin, Texas, USA. AAAI Press / The MIT Press, 2000, pp 450-455. Disponible sur : <www-smi.stanford.edu/pubs/SMI_Reports/SMI-2000-0831.pdf> (consulté le 13/05/2004). ISBN 0-262-51112-6.
- [Noy, Musen 2001] N. F. Noy, M. A. Musen. Anchor-PROMPT: Using Non-Local Context for Semantic Matching. **In** : *Proceedings of workshop on Ontologies and Information Sharing at the International Joint Conference on Artificial Intelligence (IJCAI)*, août 2001, Seattle, WA. Disponible sur : <www-smi.stanford.edu/pubs/SMI_Reports/SMI-2001-0889.pdf> (consulté le 25/06/04)
- [OWL 2004] WebOnt. Les cas et conditions d'utilisation du langage d'ontologie Web OWL [en ligne]. Disponible sur : <<http://www.voyodesign.org/doc/w3c/webont-req-20040210/>> (consulté le 25/05/04)
- [de Paiva 2003] Valeria de PAIVA. Natural Deduction and Context as (Constructive) Modality. **In** : *Proceedings of CONTEXT 2003: Fourth International and Interdisciplinary Conference on Modeling and Using Context*, June 2003, Stanford, USA. Lecture Notes in Artificial Intelligence, volume 2680, pp 116-129.
- [Pinto et al. 1999] H. PINTO, A. GOMEZ-PEREZ J. P. MARTINS. Some Issues on Ontology Integration. **In** : Richard BENJAMINS, B. CHANDRASEKARAN, A. GÓMEZ PÉREZ et al. . *In : Workshop on Ontologies and Problem Solving Methods : Lessons Learned and Future Trends*, 1999, Stockholm, Sweden. Amsterdam, The Netherlands : CEUR Publications and University of Amsterdam, 1999, Volume 18. ISBN 90-5470-085-8.
- [Protégé] Protégé-2000. Welcome to the Protégé Project [en ligne]. Disponible sur : <<http://protege.stanford.edu/>> (consulté le 22/05/04)

- [Rahm, Bernstein 2001] Erhard RAHM, Philip A. BERNSTEIN,. A survey of approaches to automatic schema matching. *VLDB Journal: Very Large Data Bases*, vol. 10, n° 4, pp 334—350, 2001. Disponible sur : <<http://research.microsoft.com/~philbe/VLDBJ-Dec2001.pdf>> (consulté le 22/04/04)
- [Ray 2002] Steven R. RAY .Interoperability standards in the semantic web. *Journal of Computing and Information Science in Engineering* [en ligne]. 2002, vol. 2, n° 1, pp. 65-71. Disponible sur : <www.nist.gov/msidlibrary/doc/ray020.pdf> (consulté le 03/05/2004).
- [Ribière 1998] Myriam RIBIÈRE. Using Viewpoints and CG for the Representation and Management of a Corporate Memory in Concurrent Engineering. **In** : Marie-Laure MUGNIER, Michel CHEIN. *Conceptual Structures: Theory, Tools and Applications, 6th International Conference on Conceptual Structures, ICCS '98*, August 10-12, 1998, Montpellier, France. Springer, Lecture Notes in Computer Science, 1998, 438pp. Disponible sur : <www.springerlink.com/index/9T4R0HRPA4W99LRM.pdf> (consulté le 13/04/04).
- [Sowa 1999] J. F. SOWA. Building, Sharing, and Merging Ontologies [en ligne]. Disponible sur : <http://www.jfsowa.com/ontology/ontoshar.htm> (consulté le 12/05/2004)
- [Stumme, Maedche 2001] G. STUMME, A. MAEDCHE. Ontology Merging for Federated Ontologies on the Semantic Web. **In** : GÓMEZ-PÉREZ A., GRUNINGER M., [STUCKENSCHMIDT H.](#) *IJCAI-01 Workshop on Ontologies and Information Sharing*, August 4-5, 2001. Seattle, USA [en ligne]. Seattle : CEUR, 2001, pp91-99. Disponible sur : : <<http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS//Vol-47/>> (consulté le 23/05/04)
- [SUO] IEEE. Standard Upper Ontology Working Group [en ligne]. Disponible sur : <<http://suo.ieee.org/>> (consulté le 25/05/04)
- [Tomcat] The Apache Software Foundation. The Jakarta Site – Apache Tomcat [en ligne]. Disponible sur : <<http://jakarta.apache.org/tomcat/>> (consulté le 27/06/04)
- [Vatant 2003] Bernard VATANT. OWL and Topic Map Pudding [en ligne]. August 2003 ; Paris, France. Disponible sur : <<http://www.mondeca.com/owl/owltm.htm>> (consulté le 23/03/04)
- [W3C] World Wide Web Consortium. World Wide Web Consortium, leading the Web to Its Full Potential [en ligne]. Disponible sur : <<http://www.w3c.org/>>(consulté le 22/06/4)
- [Wache et al. 2002] H. WACHE, U. VISSER, T. SCHOLZ. Ontology Construction - An Iterative and Dynamic Task. **In**: *Florida Artificial Intelligence Research Society Conference (FLAIRS)*, 2002, Pensacola, FL, USA, pp.445-449.
- [Wang, Gasser 2002] Jun WANG, Les GASSER. [Mutual Online Ontology Alignment](#). **In** : Stephen CRANFIELD, Tim FININ, Steve WILLMOTT. *1st International Joint Conference on Autonomous Agents and Multi-Agent Systems*. Bologna, Italy, 16 July 2002, CEUR Workshop Series Volume 66. Disponible sur : < www.isrl.uiuc.edu/~gasser/papers/oas-aamas2002-wang-gasser.pdf> (consulté le 13/05/2004).
- [WordNet] Princeton University. WordNet – a lexical database for the English language [en ligne]. Disponible sur : <<http://www.cogsci.princeton.edu/~wn/>> (consulté le 25/06/04)
- [Xu, Embley 2003] Li XU, David EMBLEY. Using Domain Ontologies to Discover Direct and Indirect Matches for Schema Elements. **In** : *Second International Semantic Web Conference (ISWC-03)*, October 20, 2003, Sanibel Island, Florida. CEUR, 2003. Disponible sur : <<http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS//Vol-82/>> (consulté le 10/05/2004)

Annexes