La représentation de connaissance est-elle soluble dans le web ?

Jérôme Euzenat

INRIA Rhône-Alpes
655 avenue de l'Europe, 38330 Montbonnot Saint-Martin (France)
Jerome.Euzenat@inrialpes.fr

RÉSUMÉ. Une double interrogation se pose concernant les rapports entre la représentation de connaissance, telle qu'elle est entendue en intelligence artificielle (c'est-à-dire une représentation formelle dotée d'une sémantique), et la notion de document telle qu'elle est actuellement comprise dans le World wide web :

— *La représentation de connaissance est-elle soluble dans le web ? C'est-à-dire peut-elle s'intégrer harmonieusement dans le paysage du web et comment, mais aussi que peut-elle apporter au web ?*

— *La représentation de connaissance va-t-elle se dissoudre dans le web ? En ces temps où toute source documentaire est nommée « base de connaissance », où les formats des documents du web sont de plus en plus structurés, la représentation de connaissance a-t-elle un avenir hors du web ou sera-t-elle dépassée par ces approches plus pragmatiques ?*

Pour cela, les activités de représentation de connaissance intégrées dans l'aspect documentaire du web (excluant les robots par exemple) sont décrites : pages web à connaissance ajoutée (par exemple, SHOE), serveurs de connaissance (par exemple, Troeps), moulins à connaissance (par exemple, AltaVista refine), éditeurs de connaissance (par exemple, Ontolingua server). Les rapports entre les systèmes de représentation de connaissance et le langage XML seront évoqués. S'il ne s'agit pas d'un langage de représentation de connaissance, les efforts à réaliser (et réalisés) pour l'en rapprocher sont précisés.

ABSTRACT. A double question is asked about the relationships between knowledge representation, as it is understood in artificial intelligence (i.e. a formal representation with a semantics), and documents, as they are displayed in the world-wide web:

— *is knowledge representation miscible with the web ? i.e. can it be smoothly embedded in the web and what can it bring to the web ?*

— *will knowledge representation be superseded by the web ? when any documentary source is called a knowledge base, when document formats are more and more structured, can knowledge representation remain independent from the web ?*

In order to answer these questions, knowledge representation activities integrated with the documentary aspect of the web (viz. excluding robots) are described :knowledge-added web pages (e.g. SHOE), knowledge servers (e.g. Troeps), knowledge mills (e.g. Altavista refine), knowledge editors (e.g. Ontolingua server). The relationships between knowledge representation and the XML language are considered. If XML does not deserve the name of knowledge representation language, the path for improving it in that direction is presented.

MOTS-CLÉS : Représentation de connaissance, WWW, XML, ACL, serveurs de connaissance, moulins à connaissance, éditeurs de connaissance.

KEYWORDS: Knowledge representation, WWW, XML, ACL, knowledge server, knowledge mill, knowledge editor.

Document numérique. Volume 3 – n° 3-4/1999, pages 151 à 167

Un mot peu utilisé jusqu'à présent a fait un retour remarqué dans le vocabulaire des "managers" : le mot connaissance. Il est question de diffusion de connaissance sur le web et d'évaluation du « capital connaissance » de l'entreprise. Ceux qui s'occupent de représentation de connaissance sont parfois surpris par cette connaissance qui n'entre pas dans leurs plans. Elle se présente principalement sous la forme de documents et, au vu de l'évolution actuelle des pratiques documentaires, diffusées à l'aide des technologies du "World wide web" (web par la suite). Pourtant, le web est une opportunité pour tous de diffuser le contenu des bases de connaissance (et valoriser l'effort qu'a suscité leur élaboration) quelle que soit la forme de la connaissance. Ainsi, la connaissance, fût-elle accumulée dans un système à base de connaissance, n'est-elle plus uniquement disponible pour être exécutée mais peut être utilisée à titre documentaire. Deux questions se posent alors :

— La représentation de connaissance est-elle soluble dans le web ? C'est-à-dire peut-elle s'intégrer harmonieusement dans le paysage du web et comment. Une question supplémentaire est que peut-elle apporter au web ?

— La représentation de connaissance va-t-elle se dissoudre dans le web ? En ces temps où toute source documentaire est nommée « base de connaissance », où les formats des documents du web sont de plus en plus structurés, la représentation de connaissance a-t-elle un avenir hors du web ou sera-t-elle dépassée par ces approches plus pragmatiques ? Est-ce la fin de la représentation de connaissance en tant que discipline ?

C'est à ces deux questions que cet article tente de répondre du point de vue de la représentation de connaissance. Il faut noter qu'ici, le mot connaissance est utilisé dans le sens qui lui est attribué dans la discipline en question : la connaissance est stockée, sans doute sous forme d'information, dans un langage formel doté d'une sémantique dénotationnelle. De même, il ne sera pas distingué entre base de connaissance et « ontologie » telle qu'elle est entendue en représentation de connaissance. Les deux termes seront utilisés indifféremment.
La seconde question sera tout d'abord abordée afin d'évaluer si la représentation n'est pas déjà dissoute dans les langages du web (et en particulier XML, §1). Après avoir répondu provisoirement par la négative, les apports variés et nouveaux que la représentation de connaissance peut avoir dans le contexte du web seront présentés. Elle peut être utilisée pour représenter formellement le contenu de documents (§2), pour exécuter/exploiter la connaissance formelle (§3), pour manipuler et organiser la connaissance disponible (§4) ou pour construire des bases de connaissance impliquant plusieurs intervenants à travers le web (§5). À chaque fois, les principes de chacune des applications sont brièvement présentés avant de décrire les originalités de divers projets de recherche impliqués dans ces activités et de tracer quelques problématiques de recherche liées à celles-ci. Ce panorama témoigne de la fertilité croisée du domaine de la représentation de connaissance et de l'aspect documentaire du web. Il permet de répondre résolument à la première question par l'affirmative.

1. Nouveaux langages de représentation de connaissance ?

Le développement du web donne lieu au déploiement à grande échelle de nouveaux langages. Ces langages servent de support à la connaissance et sont donc amenés à

faire figure de langages de représentation de connaissance à tel point que le développement de langages de représentation de connaissance pourrait ne plus avoir d'objet, ni d'avenir hors du web.
Deux types de langages seront examinés ci-dessous : XML et ses dérivés d'une part parce que ce sont eux qui sont destinés à la représentation des documents (renfermant la connaissance), KQML et ses descendants prévisibles parce qu'ils servent à communiquer la connaissance entre agents. Des langages tels que OKBC ("Object Knowledge Base Connectivity", [CHA 1998]) ou KIF ("Knowledge Interchange Format") ne sont pas abordés, bien qu'ils se présentent comme des standards pour l'échange de connaissance, car ils proviennent directement des travaux en représentation de connaissance mais surtout car ils sont peu liés avec l'aspect documentaire.

1.1 Langages de marquage : XML, RDF et Schema

Après l'insuccès relatif de SGML ("Standard Generalized Markup Language") et la fortune d'HTML ("HyperText Markup Language"), le langage XML ("eXtensible Markup Language", [BRA 1998]) semble promis à un brillant avenir. Il est parfois décrit comme un langage à objets c'est pourquoi il se trouve placé comme un concurrent naturel des langages de représentation de connaissance. XML est un métalangage permettant de baliser la structure d'un document (à un niveau de granularité quelconque). Pour pouvoir être interprété, il nécessite la présence d'une description du type de document (DTD) utilisé.

```
<HOUSE id="567-89" typ="flat">
  <NB-ROOMS>3</NB-ROOMS>
  <ROOM id="567-89-102"
    typ="bathroom"/>
  <ROOM id="567-89-1"/>
  <ROOM id="567-89-2"/>
  …
</HOUSE>
```

```
<!ELEMENT HOUSE (NB-ROOMS,ROOM*)>
<!ATTLIST HOUSE id CDATA #REQUIRED
  typ (flat|detached) "flat">
<!ELEMENT NB-ROOMS (CDATA)>

<!ELEMENT ROOM EMPTY>
<!ATTLIST ROOM id CDATA #REQUIRED
  typ (bathroom|…) "bedroom">
```

Figure 1. *Document XML et DTD. À gauche, un fragment de document XML décrit une maison de type "flat" identifiée par "567-89" (attributs) dont le contenu est une mention du nombre de pièces et de l'ensemble des pièces. À droite la DTD décrit chaque élément considéré par son contenu (dans !ELEMENT) et ses attributs (dans !ATTLIST).*

Le langage est, comme SGML, fondé sur une notion de marque affectant un élément de contenu particulier. Ces marques sont dotées d'attributs et peuvent renfermer un contenu qui est lui même marqué (voir Figure 1). Cette dualité marque/attribut fait grossièrement ressembler XML à un langage objet. Par ailleurs, le fait que ces marques et les attributs qu'elles acceptent soient définis strictement dans la DTD permet d'assimiler rapidement les éléments génériques définis dans la DTD à des classes.
Cependant, la comparaison s'arrête là. En effet, certaines caractéristiques communément admises des représentations de connaissance par objets ne sont pas présentes dans XML tel qu'il est actuellement : il n'est pas possible d'avoir des

objets, voire des ensembles d'objets, en valeur d'attributs ; il n'y a pas non plus de notion de spécialisation (ou d'héritage) entre éléments. Ceci peut être contourné en appelant les objets par nom ou en réutilisant les spécifications des éléments. Mais, ce n'est qu'un pis-aller : il faut, par exemple, réifier les listes d'objets ce qui n'est pas très élégant. La différence la plus importante est que le langage utilisé est un langage de marquage dont la sémantique est donnée par l'application et n'est pas intrinsèque au langage.

De nombreuses initiatives sont actuellement en cours afin d'amener XML vers un langage de représentation de connaissance. Elles sont de deux types. Les premières, menées sous l'égide du W3C ("Worldwide web consortium"), ont pour but d'étendre XML de façon à lui faire intégrer ces caractéristiques manquantes. Elles se nomment RDF ("Resource Description Framework") mais surtout XML/Schema et permettent d'étendre les types de données manipulés par XML et de généricier les attributs. Elles ne permettent cependant pas pour l'instant d'introduire la spécialisation, ni surtout de définir un standard de sémantique pour XML.

Hors du W3C, les efforts sont menés par différentes équipes pour disposer de la puissance de langages de représentation de connaissance sous XML. Pour cela, au lieu d'étendre XML lui-même, une DTD est définie pour exprimer les systèmes de représentation de connaissance classiques. Dans cette voie, SHOE définit un langage de représentation par objets pour HTML et OML ("Ontology Markup Language") a pour but de décrire des graphes conceptuels sous XML. Ces langages seront détaillés au §2.

Ce qu'il faut retenir, c'est qu'XML est un langage destiné à baliser la structure des documents et non leur contenu. En cela, il n'atteint pas l'idéal de la représentation de connaissance bénéficiant d'une sémantique [EUZ 2000]. Les efforts actuels d'extension d'XML visent à donner à XML plus de possibilités pour décrire le contenu mais ceci sans définir de sémantique.

1.2 Langages de communication entre agents : KQML et FIPA

Jusqu'ici, il semble qu'il y ait un antagonisme entre structure et contenu (entre forme et fond). Cependant — et le glissement des utilisations d'XML est là pour le montrer —, il y a en fait une continuité entre les deux aspects et au moins un type de langage se glisse entre les deux : les langages de communication entre agents (ACL). Ces langages ont en commun avec XML de ne pas exprimer le contenu et en commun avec les langages de représentation de connaissance d'exprimer les intentions et croyances des agents (et de chercher à donner une sémantique dénotationnelle au langage).

KQML ("Knowledge Query and Manipulation Language", [FIN 1993]) et FIPA ACL ("Foundations for Intelligent Physical Agents", [FIP 1999]) ne sont pas des langages de représentation de connaissance au sens évoqué ci-dessus mais des langages de communication entre agents. La différence principale est que ceux-ci ne sont pas destinés à représenter le contenu des messages mais l'attitude d'agents envers ce contenu (ils y croient, ils veulent que l'interlocuteur y croie, etc.). Leur but est de permettre la communication sans avoir besoin de connaître le contenu (de manière similaire à XML en fait).

Les ACL, au moins dans leur problématique actuelle, envoient des messages inspirés de la théorie des actes de langage. Ces messages expriment la « fonction

illocutoire » d'une assertion tout en ne se souciant pas de ce sur quoi elle porte (le « contenu propositionnel »).

```
(inform
       :sender agent-1
       :receiver agent-2
       :in-reply-to Msg#238
       :language xml-oml-1.0
       :content "<HOUSE id=\"567-89\">
                   <NB-ROOMS>2</NB-ROOMS>
                 </HOUSE>")
```

Figure 2. *Message dans le format de [FIP 1999].*

Ce message de la Figure 2 signifie que l'agent-1 a l'intention de faire savoir à l'agent-2, en réponse à un message antérieur, le contenu, lui-même décrit en XML (à l'aide d'une DTD identifiée). Le sens de ce message est donc compréhensible par les agents sans que le contenu ait besoin de l'être. Ainsi, un agent intermédiaire ou un protocole de communication pourra manipuler le message, le transformer, en accord aves sa « fonction illocutoire », c'est-à-dire plus que sa structure mais sans connaître le « contenu propositionnel ».

Les langages utilisés sur Internet et en particulier sur le web ont donc tendance à s'approcher progressivement des langages de représentation de connaissance. Mais ils n'ont cependant pas atteint le niveau de sophistication et de précision de ces derniers. De nombreux travaux, en liaison avec les chercheurs en représentation de connaissance, ont pour but de progresser vers la représentation du contenu (ou l'avènement d'un web sémantique [BER 1998]). Ils contribueront sans doute à la distillation bénéfique de la représentation de connaissance dans le web.
Cet objectif n'étant pas pour l'instant atteint, il reste un certain nombre de thèmes sur lesquels la représentation de connaissance en tant que telle peut apporter sa pierre à l'édifice de l'approche documentaire du web. Ces thèmes sont décrits ci-dessous. Il s'agit principalement — hormis les moulins à connaissance — de systèmes serveurs (consultables comme des documents) à l'exclusion d'applications plutôt clientes que sont les agents, guichets, portails et autres médiateurs, qui nécessiteraient un article à eux seuls.

2. Pages web à connaissance ajoutée

Les pages web à connaissance ajoutée lient étroitement les documents (page web) et la connaissance formelle incluse dans le document comme une représentation de son contenu. Ce type d'application est en très fort développement actuellement, sans doute propulsé par le déploiement d'XML qui permet d'intégrer étroitement représentation formelle et représentation textuelle. Le type d'application visé par ce dispositif est, bien entendu, l'accès aux documents (et la recherche sur le web) par le contenu.
Cependant, il existe plusieurs conceptions de ce qu'est ce contenu : disposer du type d'un fichier est déjà avoir une information sur son contenu qui va permettre de trouver l'application capable de le lire. L'introduction de « méta-données » est, sous

sa forme minimale peu différente de cela, elle permet d'associer aux documents des informations sur l'auteur, la langue ou la date de dernière modification. Par contre, sous leur forme plus développée où les méta-données sont des termes bien identifiés dans une classification élaborée (telle que le Dublin-core en est l'archétype), celles-ci sont plus proches du contenu ou plutôt d'une représentation formalisée de celui-ci. L'aboutissement de cette approche serait, bien entendu, la représentation formelle complète du contenu du document. Parallèlement à ces approches actives et organisatrices du contenu, d'autres approches utilisent le contenu d'un document tel qu'il est pour en percevoir la pertinence par rapport à une interrogation du lecteur. C'est le cas des approches d'indexation en texte intégral auxquelles peuvent s'ajouter des techniques statistiques (tests de co-occurrence) et/ou linguistiques (lemmatisation, analyse grammaticale et sémantique). Cette tendance concourt au même aboutissement que la précédente : atteindre le sens d'un document et le représenter en machine.
Bien entendu, cet objectif idéal est loin d'être atteignable à l'heure actuelle. Mais, la proximité entre XML et objets d'une part et la maturité des langages de représentation de connaissance de l'autre, permettent d'envisager des projets d'indexation des documents du web par une représentation formelle de leur contenu. Ce type d'application dépasse en puissance le simple ajout de méta-donnée par la capacité à utiliser des méta-données flexibles c'est-à-dire définissables pour les besoins d'une application dans un langage permettant de décrire les types de méta-données. Ainsi, il est possible non seulement d'indexer les documents décrits mais d'appliquer des méthodes de manipulation de la connaissance beaucoup plus sophistiquées telles que la construction automatique de taxonomies à partir de ces documents (voir §3). Ici resurgit un clivage déjà présent entre bases de données et bases de connaissance où, dans les premières, les données sont amenées à changer sans affecter le schéma alors que les outils de représentation de connaissance permettent de modifier radicalement l'organisation des données en laissant celles-ci (ici les documents) inchangées.

De nombreux projets s'attaquent à cette problématique avec des approches et des langages de représentation de connaissance très différents.
Historiquement, le premier projet de ce type, LogicWeb, associait une description en Prolog des pages web [LOK 1996]. L'intérêt de ce projet est que les annotations sont de véritables théories logiques (équivalentes à des programmes en réalité) et que leur évaluation devait être faite par le client HTTP ("HyperText Transfert Protocol") dans lequel un interpréteur Prolog est intégré (à l'instar des machines virtuelles Java).
Le second projet, plus axé sur les notions récentes d'ontologie (ou de référentiel organisé) est SHOE ("Simple HTML Ontology Extension" [HEF 1998]) à l'université du Maryland, qui malgré son nom a été dès le début doté d'une DTD ce qui le rend naturellement utilisable hors de HTML. Le langage SHOE est un langage de type objet permettant d'exprimer des classes d'objets organisées en hiérarchies et des relations entre ces objets. Ce projet est destiné aux agents logiciels qui « parcourent » le web à la recherche d'information précise. Un éditeur a été développé pour pouvoir annoter plus facilement les documents.

Le système Ontobroker [FEN 1998] conçu à l'université de Karlsruhe a pour but d'indexer les documents à l'aide d'une représentation par objets (FLogic) et en référence à des « ontologies » décrites dans le même langage. Il est assez massivement utilisé dans l'initiative $(KA)^2$ ("Knowledge Annotation for the Knowledge Acquisition community" [BEN 1998]) où de nombreux chercheurs en acquisition de connaissance aident à la création d'ontologies et l'annotation de pages web sur ce sujet. Ontobroker est doté d'un langage de requête (SiLRI) très expressif et capable d'inférence et surtout capable de traiter les expressions de RDF comme des expressions dans le langage utilisé par Ontobroker.

Le langage OML [KEN 1999] ("Ontology Markup Language") issu de l'université de l'état de Washington est une DTD XML permettant de décrire des graphes conceptuels. Le langage est utilisé pour indexer des documents du web, en particulier dans une application, Wave, à l'indexation des communiqués de presse de la société Intel. Cette application, suivant le profil de l'utilisateur, est capable de lui présenter les communiqués les plus susceptibles de l'intéresser vis-à-vis de leur contenu et de son profil. Contrairement à SHOE dont l'expressivité est délibérément restreinte, les représentations d'Ontobroker et OML sont très expressives (par conséquent, la réponse à une requête peut-être longue indépendamment de la quantité de données à explorer).

Ce type de travaux fait actuellement l'objet de nombreux développements. Bien entendu, le premier problème est la difficulté d'annoter manuellement de nombreux sites web. Cependant, à titre de test de concept, les projets présentés sont très intéressants. Ils préparent sans doute des applications plus automatiques dans lesquelles le traitement de la langue prend tout son sens.
Enfin, si de nombreuses équipes se lancent dans ce type d'application, le choix du langage de représentation de connaissance est réalisé sans étude préalable alors que d'importants compromis entre expressivité, complexité et explicabilité sont à faire. C'est pourquoi des travaux de recherche permettant d'identifier les formalismes les plus adaptés à certaines utilisations seront nécessaires.

3. Serveurs de connaissance

Les serveurs de connaissance sont les applications les plus immédiates de la représentation de connaissance au web. Il faut distinguer deux types de services : la mise à disposition d'une ressource de type système à base de connaissance (pour la résolution de problème) sur le web et la possibilité d'accéder par le web à la connaissance formalisée. Force est de constater que peu de publicité est faite sur des systèmes implémentant le premier type bien qu'ils aient été théorisés assez tôt ("problem-solving documents") [GAI 1992 ; CHA 1997]. Sans doute parce qu'ils apparaissent comme une simple interface vers des systèmes dont la contribution est ailleurs. Le système DME ("Device modeling environment") modélise et simule un système dynamique. Il est capable de donner des explications sur la simulation en répondant à des questions (quelle quantité à eu un impact sur le résultat ?) ou de relancer une simulation en changeant les conditions initiales. Ces opérations sont accessibles à partir du web [GRU 1997], les résultats peuvent être affichés sous la forme de courbes documéntés par des liens vers les documents de spécifications.

Les bases de connaissance formalisées, telles qu'elles sont considérées en intelligence artificielle, sont plus que des documents et, à ce titre, engendrer un ensemble de documents statiques à partir d'une base de connaissance c'est figer ce qui est encore vivant. Il faut donc tirer parti de la dimension dynamique du web permettant d'exploiter pleinement la perspective cognitive (ou la possibilité d'utiliser ou d'exécuter la connaissance) dans un contexte documentaire. Ainsi, il devient possible d'accéder à la connaissance stockée par le biais de requêtes s'appuyant sur la structure (filtrage ou classification).
Cet aspect est pris en compte dans les notions de livres de connaissance électronique [CHA 1997] ou de documents actifs [GAI 1999] qui permettent non seulement d'accéder par le biais d'un document à la représentation formelle de la connaissance (un peu comme les pages web à connaissance ajoutée) mais de mobiliser l'aspect opératoire de cette connaissance.
D'autres raisons poussent à utiliser le web en tant qu'interface à une base de connaissance :

— diffusion de la connaissance sans se soucier de problèmes de portage (les clients HTTP, le protocole du web, étant disponibles dans le monde entier) ;

— mise à jour instantanée de la connaissance à partir d'un unique serveur ;

— possibilité d'atteindre des utilisateurs non spécialistes grâce à l'universalité d'HTML ;

— connexion de la base de connaissance à son contexte (bibliographie, projets, textes, lexiques, images) à l'aide de liens vers d'autres sites.

Ainsi, WebCokace développé à l'INRIA Sophia-Antipolis permet de naviguer au sein de bases CML ("Conceptual Modelling Language", issu de KADS "Knowledge Acquisition and Documentation Structure") [COR 1997]. Il est utilisé pour visualiser diverses bibliothèques classiques de modèles KADS.

Le système de représentation de connaissance Troeps [SHE 1998] est utilisable dans sa version serveur de connaissance pour naviguer dans les structures formelles. Celles-ci sont liées à des élément informels (pages web) ou plus structurés (bases de données, lexique). Il est bien entendu possible de demander interactivement le résultat de filtrage ou de classification sur les structures contenues dans la base ; les personnes habilitées à éditer la base peuvent par ailleurs la modifier (voir §5) mais aussi utiliser des outils permettant de construire automatiquement des taxonomies sur des ensembles d'objets. Le système étant extensible, il est aussi possible d'adjoindre à la base formalisée des opérations adaptées à une application particulière (rejoignant ainsi le premier type de serveur de connaissance). La base de connaissance Knife [EUZ 1997], accessible librement depuis le web, permet par exemple de rechercher les cycles ou les chemins dans les graphes d'interaction génique ou de diagnostiquer les incomplétudes de la base elle-même.

La notion de serveur de connaissance, au sens strict, c'est-à-dire mettant la connaissance opératoire à disposition sur le réseau est assez peu détaillée dans la littérature, sans doute parce que d'une part, les systèmes sont habillés sous un autre éclairage, mais aussi parce que les efforts ont porté sur la construction d'« ontologies » pour lesquelles seul l'inventaire des êtres (et non les êtres eux-

mêmes) compte. Ainsi, il n'y a pas d'exemples sur lesquels appliquer la connaissance. Le second type de serveur de connaissance est en fait beaucoup plus répandu que ne le laisse penser cette section grâce aux éditeurs abordés au §5.
L'un des aspects de la connaissance en-ligne est de pouvoir être manipulée par des systèmes capables de l'organiser, même si elle n'a pas vocation à perdurer. Cet aspect est étudié dans la section suivante.

4. Moulins à connaissance

Ce troisième aspect est moins directement lié à la représentation de connaissance et plus lié en principe à la recherche d'information. Il consiste à manipuler la connaissance disponible pour en donner une représentation plus synthétique. Il recouvre donc la construction automatique de regroupements de pages web comme c'est le cas dans les outils décrits au tableau 1. Seule l'organisation des documents est considérée ici et non pas les différentes fonctions des agents, robots et autres médiateurs dont les moulins à connaissance peuvent tirer parti.
Ces outils, à partir d'un ensemble de documents, construisent un graphe dont les nœuds sont des regroupements de documents. Ils diffèrent par la manière de regrouper les documents (regroupement) et par la manière de relier les différents groupes (liaison). Mais tous sont fondés sur l'analyse du texte. Un courant important de "text-mining" commence à se développer autour de cette thématique [KOD 1999]. Certains systèmes, tels SemioMap, sont capables de construction de taxonomie (c'est-à-dire une hiérarchie dont chaque nœud représente un ensemble de documents et dont l'ensemble des fils d'un nœud représente un ensemble de documents plus restreint que celui de son père). Mais la plupart d'entre eux se bornent à lier les « concepts » obtenus sans orienter le graphe, ni donner de sens précis à ces liens.

Société	Produit	Recherche	Regroupement	Liaison
AltaVista	Refine/Cow9	Texte	Texte	Statistiques
Verity	Topics	Texte	Requêtes symboliques	Texte
NorthernLight	Search folders	Texte	Texte	Statistique
Trivium	Umap	Texte	Combinatoire	Algorithmique
Semio	SemioMap	Texte	Linguistique	Statistique
IRIT	Tétralogie	Structuré	Linguistique	Bibliométrique

Tableau 1. *Comparaison de divers moulins à connaissance basés sur le texte.*

Cependant, lorsqu'au lieu de documents sont manipulées des bases de connaissance ou des pages web à connaissance ajoutée, il est possible de tirer parti de nombreuses techniques développées en représentation et en acquisition de connaissance pour aider à l'organisation du corpus obtenu.
Ainsi, WebGrid, développé à l'université de Calgary [GAI 1995], permet à un utilisateur de formaliser sa connaissance à partir de "repertory grids", techniques issues de la psychologie cognitive. Pour cela, il présente à l'utilisateur des triplets d'éléments (ici de documents dont le contenu est modélisé par des structures attributives) et lui demande de choisir l'élément correspondant le mieux et celui correspondant le moins bien à une classe particulière (par exemple, les documents

recherchés). Le système apprend alors les attributs et les valeurs d'attributs discriminants entre classes.

APECKS ("Adaptative Presentation Environment for Collaborative Knowledge Structuring"), développé à l'université de Nottingham, a pour but d'aider les utilisateurs à créer des « ontologies individuelles » en les comparant à celles des autres [TEN 1998]. Pour cela le système utilise un langage à base d'objets et traduit les ontologies pour les comparer à l'aide de WebGrid. Il signale aux utilisateurs les différences entre leurs ontologies.
Enfin, le projet Wave (voir §2) a pour but d'organiser les documents décrits par des structures OML à l'aide des techniques d'analyse de concepts formels (issues de l'analyse de données) permettant de construire une taxonomie à partir d'un ensemble d'objets.
En règle générale, il existe de nombreuses techniques issues de l'analyse de données ou de l'apprentissage automatique permettant d'organiser l'information représentée symboliquement.

Les moulins à connaissance, qu'ils partent du texte ou de représentation de connaissance intégrée aux documents, sont pour l'instant relativement frustes dans les organisations qu'ils produisent. Cependant, l'aide qu'ils apportent pour organiser des quantités importantes de documents est toujours la bienvenue. Ce type de travaux est amené à être un aiguillon pour la recherche en analyse de texte à tous les niveau. Mais tant que celle-ci n'est pas en mesure de fournir une représentation formalisée du contenu, il est nécessaire de pouvoir créer les structures formelles à même d'annoter les documents. Le faire sur le web et qui plus est de manière collaborative est le sujet de la section suivante.

5. Éditeurs de connaissance et collaboration

Le web peut non seulement permettre de diffuser la connaissance mais aussi de l'acquérir. La dimension documentaire est quelque peut absente de ce dernier aspect dû à la fonction communicante du web. En effet, la connexion de nombreux « travailleurs de la connaissance » au même réseau permet de leur proposer un outil d'élaboration de connaissance issant le web au rang de « co-laboratoire ». Cette perspective a séduit le public intéressé par les systèmes d'aide à la recherche scientifique et celui dirigé vers la construction d'« ontologies ».
De la même manière qu'il est possible de naviguer dans une base de connaissance (voir §3) il faudrait bien vite la modifier, l'éditer. Éditer la connaissance à l'aide de HTTP est donc souhaitable. Mais le protocole HTTP est un protocole sans état ce qui signifie que les requêtes ne modifient pas le contenu du serveur. Il est donc nécessaire de prendre en compte cet aspect. Certains systèmes, tels que WebGrid, ont cherché à rester dans la philosophie du web en ne stockant rien sur le serveur et en transmettant à chaque page toute la connaissance en champ caché. Cependant, afin de permettre la confrontation de la connaissance exprimée par différents utilisateurs, WebGrid-II permet maintenant le stockage momentané des structures décrites sur le serveur. D'autres systèmes, parmi lesquels l'ensemble des systèmes de bases de données permettant l'édition, rompent radicalement avec cette contrainte irréaliste pour des applications de taille importante.

D'autre part, l'édition est plus complexe à traiter que la consultation car tant que la base n'est pas modifiée peu d'erreurs peuvent se produire en son sein. Mais dès que la modification est possible il est nécessaire de traiter divers problèmes :

— lors de la modification des erreurs d'entrée peuvent se produire : il faut les rattraper et les expliquer ;

— si des objets disparaissent (sont détruits par exemple) leur URL ("Uniform Resource Locator") peut être encore présent dans le client d'un utilisateur : ce problème doit aussi être détecté.

Le premier des systèmes permettant d'éditer des structures de connaissance fut sans doute WebGrid qui permet l'acquisition au travers du web. Ontosaurus, développé à l'université de Californie du sud est un éditeur web plutôt mono-utilisateur pour le système LOOM [SWA 1996]. Mais l'édition simultanée d'une base de connaissance nécessite la résolution de problèmes techniques, juridiques et sociaux qui en découlent , parmi lesquels :

— la gestion de l'interaction et de la communication entre les individus ;

— le contrôle de l'accès aux données ;

— la reconnaissance d'un droit moral sur la connaissance (attribution) ;

— le rattrapage et la gestion des erreurs ;

— la gestion de la modification concurrente des données.

Sur un plan technique le dernier problème est fondamental. Peu de systèmes le traitent et diverses approches coexistent qui sont comparées ci-dessous.

Le site interactif de systématique développé à l'université Paris 6 permet l'annotation concurrente plus que l'édition et utilise pour cela le service de modérateurs. C'est-à-dire que les utilisateurs peuvent toujours ajouter des éléments mais jamais en enlever. Ces éléments sont gérés par des modérateurs (qui peuvent déléguer leur pouvoir sur une sous-partie de leur domaine). Le rôle des modérateurs se borne à estampiller une contribution comme acceptée ou non. Ainsi, les conflits pouvant survenir sont réglés par une autorité de régulation.

WebOnto est un éditeur de base de connaissance dans le langage OCML ("Operational Conceptual Modelling Language") développé à Open university [DOM 1998]. WebOnto permet à l'éditeur de propager les modifications qu'il vient d'effectuer et, à chaque utilisateur, de recevoir ou non les nouvelles modifications. Les utilisateurs sont identifiés afin de connaître leurs droits pour l'édition d'une ontologie. Une ontologie ne peut être éditée que par un utilisateur à la fois (il doit verrouiller le mode d'édition pour tous les autres utilisateurs). Le protocole utilisé est donc des plus simples mais résout les problèmes de conflits lors de l'édition. Ce dispositif risque de se révéler problématique si l'édition devient une activité fréquente d'un nombre important d'utilisateurs.

Ontolingua server, développé à Stanford-KSL, est un éditeur d'ontologies partagées sur le web [FAR 1995, 1997]. Il est un peu le père de tous les systèmes présentés ici. La connaissance est représentée à l'aide des langages Ontolingua et KIF. Il a été utilisé pour créer de nombreuses ontologies (une cinquantaine en accès libre sur le

site comme l'ontologie médicale InterMed). Ontolingua permet de protéger l'accès en édition en créant des espaces de travail particuliers (sessions) qui peuvent être partagés par plusieurs utilisateurs. Il n'exerce aucun contrôle sur les modifications effectuées au sein d'une session mais notifie les modifications [ALE 1998].

GKB-Editor, développé à SRI, est un éditeur d'objets fondé sur le "Generic Frame Protocol" (aujourd'hui OKBC [CHA 1998]). Il doit être intégré dans une boite à outils de construction d'ontologies [KAR 1997]. GKB-Editor est utilisé dans la fameuse base EcoCyc sur le métabolisme du colibacille développée par Peter Karp. Ce système met en œuvre un mécanisme de contrôle optimiste qui permet à chaque utilisateur d'éditer une copie et qui tente de régler les problèmes lors de l'intégration ("commit") de la connaissance, élaborant sur les travaux de Chaudhri [CHA 1992]. Les conflits sont traités de la manière suivante : le système gère une succession de versions des objets modifiés en considérant que la dernière version réintroduite est toujours la bonne. Deux utilisateurs peuvent solliciter le même objet en édition mais lorsque le second soumet une modification alors que l'objet a été modifié depuis le moment où il est entré en édition, le système créé deux versions et signale le conflit qui doit être réglé manuellement.

Co_4 développé à l'INRIA Rhône-Alpes [EUZ 1996b], a pour but de construire à plusieurs une base de connaissance exprimée dans la représentation à base d'objets Troeps [SHE 1998]. Le système est composé de l'éditeur de bases de connaissance Troeps [ALE 1998] et du protocole de Co_4 implémenté sous forme d'une bibliothèque [EUZ 1997]. Le protocole ne sert qu'à gérer l'interaction entre les différents contributeurs. Pour cela chaque utilisateur dispose de sa propre base qui est une base de plein droit et qui adhère à une base commune, dite consensuelle. Cette base consensuelle peut elle-même être adhérente d'une base plus vaste.
Co_4 exerce un contrôle lors de l'intégration d'un fragment de base individuelle dans la base consensuelle. Cette intégration est subordonnée à la soumission et l'acceptation d'éléments de connaissance suivant un protocole inspiré de la soumission d'articles scientifiques. Les utilisateurs doivent soumettre la connaissance qu'ils désirent intégrer à la base consensuelle, celle-ci est rediffusée aux autres adhérents qui doivent donner leur opinion (acceptation, refus ou demande de modification). Chaque utilisateur peut tester la connaissance soumise dans sa propre base. Co_4 gère aussi la soumission anonyme et l'attribution des éléments de connaissance (une fois acceptés) à leurs auteurs.

Le domaine des éditeurs de connaissance peut sembler éloigné de la problématique documentaire. Mais la convergence entre documents et connaissance ne se fera que s'il est possible de s'entendre sur les concepts mis en œuvre dans le contenu des documents. Pour cela il est nécessaire de pouvoir établir, de manière collaborative, les ontologies servant de support à l'expression formelle de ce contenu. Par ailleurs, bien des techniques développées ci-dessus concernant l'édition d'ontologies sont empruntées à l'édition collaborative de documents. Il y a donc lieu d'envisager la convergence entre les éditeurs de document informel et les éditeurs de connaissance formelle. Bien entendu, la problématique n'en est qu'à ses balbutiements mais elle est promise à un brillant avenir.

Conclusion

Il y a bien convergence, dans le cadre des outils proposés par le web, des documents et des connaissances. Cette convergence nécessaire n'en est qu'à ses début. Elle se réalise progressivement en rapprochant les langages de représentation de connaissance et les langages de description de documents. Ces derniers, s'ils permettent de baliser les documents, ne sont pas encore dotés d'une sémantique permettant de s'assurer de la validité des opérations appliquées aux documents. C'est pour cette raison que de nombreuses équipes de représentation de connaissance travaillent à l'intégration de leurs langages dans le contexte du web. Une perspective qui mériterait l'attention du monde de la représentation de connaissance consisterait à développer, à côté de la définition de la structure (DTD) une définition du sens à accorder aux éléments (par exemple, en théorie des modèles). La réussite d'une telle initiative n'est en rien assurée mais les avantages d'une telle approche seraient très important (par exemple, la possibilité de valider les transformations opérées à l'aide du langage XSLT "XML Stylesheet Language Transformations").

Cette convergence document-connaissance, pour l'instant embryonnaire, devrait permettre une seconde révolution dans la gestion de la documentation et du savoir qui tire pleinement parti de l'aspect opératoire des ordinateurs. La connaissance ainsi représentée dans des documents naguère statiques est ouverte à des procédés issus de la représentation de connaissance et capables d'en tirer le meilleur parti : les résultats obtenus par les agents et les guichets devraient être plus pertinents ; des algorithmes particuliers (comme la construction automatique de taxonomie fondée sur des structures formelles) pourront être appliqués sur le contenu des documents ; la connaissance contenue dans les documents devrait pouvoir être directement exécutable pour les lecteurs.

Disposer de documents exécutables au contenu formalisé devrait faciliter la collaboration par une implication accrue de l'ordinateur dans l'aide aux collaborateurs (par exemple par des tests de cohérence ou de généralité entre contenu). Les éditeurs de connaissance eux-mêmes devraient tirer parti de cette assistance dans la collaboration.

Bibliographie

[ALE 1998] ALEMANY C., Étude et réalisation d'une interface d'édition de bases de connaissances au travers du World Wide Web, Mémoire CNAM, Grenoble (FR), 1998

[BEN 1998] BENJAMINS R., FENSEL D., Community is knowledge in $(KA)^2$, Actes 11th KAW, Banff (CA), 1998

[BER 1998] BERNERS-LEE T., Semantic web roadmap, 1998. http://www.w3.org/DesignIssues/Semantic.html

[BRA 1998] BRAY T., PAOLI J., SPERBERG-MCQUEEN C. M. (éds.), Extensible Markup Language (XML) 1.0, Recommendation, W3C, 1998 http://www.w3.org/TR/REC-xml

[CHA 1997] CHAILLOT M., ERMINE J.-L., Le livre de connaissance électronique, Document numérique 1(1):75-98, 1997

[CHA 1992] CHAUDHRI V., HADZILACOS V., MYLOPOULOS J., Concurrency control for knowledge bases, Actes 3rd KR, Cambridge (MA US), pp762-773, 1992

[CHA 1998] CHAUDHRI V., FARQUHAR A., FIKES R., KARP P., RICE J., OKBC : a programmatic foundation for knowledge base interoperability, Actes 15th AAAI, Madison (WI US), pp600-607, 1998

[COR 1997] CORBY O., DIENG R., A commonKADS expertise model web server, Actes 5th ISMICK, Compiègne (FR), pp97-117, 1997

[DOM 1998] DOMINGUE J., Tadzebao and WebOnto : discussing, browsing, and editing ontologies on the web, Actes 11th KAW, Banff (CA), 1998

[EUZ 1996] EUZENAT J., Corporate memory through cooperative creation of knowledge bases and hyper-documents, Actes 10 th KAW, Banff (CA), 1997

[EUZ 1997a] EUZENAT J., A protocol for building consensual and consistent repositories, Rapport de recherche 3260, INRIA Rhône-Alpes, Grenoble (FR), 1997

[EUZ 1997b] EUZENAT J., CHEMLA C., JACQ B., A knowledge base for *D. melanogaster* gene interactions involved in pattern formation, Actes 5th international conference on intelligent systems for molecular biology, Halkidiki (GR), pp108-119, 1997

[EUZ 2000] EUZENAT J., XML est-il le langage de représentation de connaissance de l'an 2000 ?, Actes 6es journées « langages et modèles à objet », Mont Saint-Hilaire (CA), 2000 à paraître

[FAR 1995] FARQUHAR A., FIKES R., PRATT W., RICE J., Collaborative ontology construction for information integration, Rapport de recherche 63, Knowledge system laboratory, Stanford university, Stanford (CA US), 1995

[FAR 1997] FARQUHAR A., FIKES R., RICE J., (1997). The Ontolingua server: a tool for collaborative ontology construction, *International journal of human-computer studies* 46:707-727, 1997

[FEN 1998] FENSEL D., DECKER S., ERDMANN M., STUDER R., Ontobroker or how to enable intelligent access to the WWW, Actes 11th KAW, Banff (CA), 1998

[FIN 1993] FININ T., WEBER J., WIEDERHOLD G.,GENESERETH M., FRITZSON R., MACKAY D., MACGUIRE J., PELAVIN R., SHAPIRO S., BECK C., Draft specification of the KQML agent communication language, 1993 ftp://ftp.cs.umbc.edu/

[FIP 1999] Foundation for Intelligent Physical Agents, Agent communication language, FIPA Spec 2, 1999 http://www.fipa.org

[GAI 1992] GAINES B., SHAW M., Documents as expert systems, Actes 9th British society expert systems conference, Cambridge university press, Cambridge (UK), pp331-349, 1992

[GAI 1995] GAINES B., SHAW M., WebMap: concept mapping on the Web, Actes 4th WWW conference, Boston (MA US), 1995

[GAI 1999] GAINES B., SHAW M., Embedding knowledge models in active documents, *Communication of the ACM* 42(1):55-63, 1999

[GRU 1997] GRUBER T., VEMURI S., RICE J., Model-based virtual document generation, *International journal of human-computer studies* 46(6):687-706, 1999

[HEF 1998] HEFLIN J., HENDLER J., LUKE S., ZHENDONG Q. SHOE: a knowledge representation language for internet applications, submitted 1998. http://www.cs.umd.edu/projects/plus/SHOE/aij-shoe.ps

[KAR 1997] KARP R., CHAUDHRI V., PALEY S., A collaborative environment for authoring large knowledge bases, 1997, soumis à publication http://www.ai.sri.com/pubs/papers/Karp9704:Collaborative/document.ps

[KEN 1999] KENT R., Conceptual knowledge markup language, Actes 12th KAW, Banff (CA), 1999

[KOD 1999] KODRATOFF Y., Knowledge discovery in texts : a definition and applications, *Lecture notes in computer science* 1609:16-29, 1999

[LOK 1996] LOKE S.-W., DAVISON A.,STERLING L., Lightweight deductive databases on the world-wide web, Actes 1st Workshop on Logic Programming Tools for Internet Applications, Bonn (DE), 1996

[SHE 1998] PROJET SHERPA, TROEPS 1.2 reference manual, Rapport interne, INRIA Rhône-Alpes, Grenoble (FR), 1998 ftp://ftp.inrialpes.fr/pub/sherpa/rapports/tropes-manual.ps.gz

[SWA 1996] SWARTOUT B., PATIL R., KNIGHT K., RUSS T., Toward distributed use of large scale ontologies, Actes 10th KAW, Banff (CA), 1996

[TEN 1998] TENNISON J., SHADBOLT N., APECKS: a tool to support living ontologies, Actes 11th KAW, Banff (CA), 1998