Intégration de sources extérieures dans un Web sémantique d'entreprise géré par un système multi-agents

Tuan-Dung Cao, Fabien Gandon, Rose Dieng-Kuntz

Intégration de sources extérieures dans un Web sémantique d'entreprise géré par un système multi-agents

Tuan-Dung Cao[1], Fabien Gandon[1,2], Rose Dieng-Kuntz[1]

[1] INRIA, Equipe ACACIA, 2004, route des Lucioles, B.P. 93, 06902 Sophia Antipolis France
{Tuan-Dung.Cao | Fabien.Gandon | Rose.Dieng}@sophia.inria.fr
[2] School of Computer Science – ISRI - Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213-389, U.S.A.
Fabien.Gandon@cs.cmu.edu

Résumé : Cet article décrit brièvement un système multi-agents gérant un Web sémantique d'entreprise avant de se focaliser sur l'extension de ce système par une nouvelle société d'agents responsables de l'intégration au web sémantique interne de l'organisation, de ressources d'information disponibles sur le Web ouvert et utiles à son activité. Cette nouvelle société d'agents repose entièrement sur la technologie XML pour extraire les informations et les mettre à disposition.

Mots-clés : web sémantique, système multi-agents, mémoire d'entreprise, wrapper.

1 Agents et Web sémantiques d'entreprise

Une organisation est une entité vivante, située dans un environnement, et ayant un passé, une culture et des contacts avec d'autres acteurs. L'ensemble des connaissances qu'elle mobilise pour ses activités n'est *pas* délimité par ses infrastructures ou ses structures ; les mémoires organisationnelles peuvent inclure ou pointer vers des ressources externes à l'organisation (des catalogues de références, des bibliothèques électroniques , des cours de la bourse, *etc.*).

Notre équipe de recherche étudie actuellement la matérialisation d'une mémoire organisationnelle sous la forme d'un Web sémantique interne (Dieng *et al*. 2001) ; ceci prolonge la tendance générale à déployer, dans les organisations, des systèmes d'information utilisant les technologies de l'Internet et du Web pour établir des intranets et des intrawebs. Un intraweb sémantique comporte:

- Une ontologie formalisée en RDFS (O'CoMMA (Gandon 2001));
- Des descriptions de l'organisation décrites sous forme d'annotations RDF portant sur les structures organisationnelles (modèle de l'organisation) et les personnes (profils d'utilisateurs) ;

- Des annotations RDF, qui décrivent le contenu des documents, en associant une sémantique à ces descriptions, portant sur les ressources documentaires de la mémoire. On peut les considérer comme des méta-données des documents.

```
<Comma: Article rdf:about="http://www.inria.fr/rrrt/rr-3485.html">
<Comma: Title>  Methods and Tools for Corporate Knowledge Management </Comma: Title>
< Comma:Author>    <Comma:Name> Dieng, Rose   </ Comma:Name> </ Comma:Author>
< Comma:Author>    <Comma:Name> Corby,Olivier </ Comma:Name> </ Comma:Author>
< Comma:Author>    <Comma:Name> Giboin, Alain </ Comma:Name> </ Comma:Author>
< Comma:Author>    <Comma:Name>Ribière,Myriam </ Comma:Name> </ Comma:Author>
< Comma:Keyword> CORPORATE MEMORY </Comma:Keyword>
< Comma:Keyword> KNOWLEDGE MANAGEMENT </Comma:Keyword>
</Comma:Article>
```

Fig. 1 – *Annotation RDF d'un article.*

Le résultat est un paysage d'informations hétérogènes et distribuées, annoté sémantiquement en utilisant les primitives fournies par l'ontologie. Pour gérer cet intraweb sémantique, il est intéressant d'envisager une architecture logicielle qui soit elle-même hétérogène et distribuée. L'adéquation des systèmes multi-agents a été reconnue dans un certain nombre de projets proposant des systèmes multi-agents pour traiter différents aspects de la gestion des connaissances à l'intérieur des organismes. Ainsi, CASMIR (Berney & Ferneley, 1999b) et RICOCHET (Bothorel & Thomas, 1999) se concentrent sur la collecte d'informations, l'adaptation des interactions aux préférences de l'utilisateur et l'apprentissage des centres d'intérêts afin d'établir des communautés et un filtrage collectif de l'information dans l'organisation. KnowWeb (Dzbor & Paralic, 2000) utilise les agents mobiles pour permettre l'accès à un réseau changeant dynamiquement, et exploite un modèle du domaine pour extraire les concepts représentatifs des documents afin de les utiliser pour répondre aux recherches de l'utilisateur. RICA (Aguirre *et al.* 2000) maintient une taxonomie thématique des documents et l'emploie pour émettre des suggestions aux agents d'interface en se basant sur les profils des utilisateurs. FRODO (Van Elst & Abecker, 2002) est consacré aux mémoires distribuées avec une emphase particulière sur la gestion des ontologies du domaine.

Notre équipe a participé au projet européen CoMMA (CoMMA 2000). CoMMA visait à mettre en application et tester une plate-forme de gestion d'une mémoire d'entreprise, basée sur la technologie agents, afin d'assister deux scénarios d'application: (1) l'intégration d'un nouvel employé à une organisation et (2) l'assistance aux activités de veille technologique. Le système ne gère pas directement des documents, mais des annotations à propos de documents référencés leur URI. CoMMA s'est focalisé sur trois fonctionnalités : (a) améliorer la précision et le rappel lors de la recherche de documents, en utilisant des annotations sémantiques ; (b) suggérer proactivement la consultation d'une ressource documentaire en utilisant des modèles de l'organisation et des utilisateurs ; (c) archiver intelligemment les annotations soumises. L'architecture de CoMMA telle qu'elle était à la fin du projet sera détaillée dans la section suivante.

En annotant des ressources disponibles sur le Web ouvert, la mémoire organisationnelle sort des murs de l'organisation. Dans CoMMA, des agents étaient

en charge d'aider le processus d'annotation et d'archivage mais l'annotation d'une ressource était essentiellement un processus manuel. Même si cela était acceptable dans CoMMA car des personnes impliquées dans les scénarios d'application avaient des rôles incluant la tâche d'annotation (veilleur, documentaliste, *etc.*), il est évident que des outils sont nécessaires pour assister ce travail en aidant, par exemple, l'exploitation d'indices structurels dans les ressources, en soulageant l'utilisateur de la répétition de tâches semblables et fastidieuses, en automatisant les mises à jour lorsque les ressources changent, *etc.* C'est pour cela que nous avons introduit une nouvelle société d'agents qui extraient des informations semi-structurées présentes dans des pages Web afin de générer des annotations sémantiques ; cette nouvelle société est décrite en détail dans la troisième section.

2 L'architecture initiale de CoMMA

L'architecture logicielle de CoMMA est un système multi-agents (SMA) conçu et testé pour gérer une mémoire organisationnelle basée sur les technologies du Web sémantique ; nous présentons brièvement les sociétés et les rôles de ce SMA.

2.1 Sociétés et fonctions sociales

Nous avons suivi une analyse organisationnelle descendante (Gandon, 2002a) où l'architecture multi-agents a été abordée, comme dans une société humaine, en termes de groupes, de rôles et de relations, à partir du niveau d'abstraction le plus élevé du système (*i.e.*, la société) et en détaillant par raffinements successifs (*i.e.*, sous-sociétés imbriquées) jusqu'au point où les rôles d'agent et les interactions nécessaires purent être identifiés. Ainsi le système a d'abord été divisé en quatre sociétés d'agents : trois sociétés dédiées aux ressources (ontologie et modèle organisationnel ; annotations ; pages jaunes nécessaires à la gestion des interconnexions entre agents) et une société dédiée aux utilisateurs.

En analysant les sociétés dédiées à des ressources, nous avons constaté qu'il y avait un ensemble récurrent d'organisations possibles : hiérarchique, égalitaire ou duplication. Selon le type de tâches à exécuter, la taille et la complexité des ressources manipulées, une organisation fut préférée à une autre. La société dédiée à l'ontologie et au modèle a été organisée comme une société de duplication (*i.e.*, un agent ontologiste a une copie complète de l'ontologie). La société dédiée aux annotations a été conçue comme une organisation hiérarchique (des gestionnaires et des archivistes). Les agents de la société dédiée aux pages jaunes sont fournis par la plate-forme JADE (Bellifemine *et al.* 2001) utilisée dans CoMMA et sont organisés en une société égalitaire (les requêtes sont propagées de pairs en pairs). Enfin, les agents de la société dédiée aux utilisateurs ne sont pas liés à un type de ressource comme les précédents, ils ont donc été étudiés séparément. En analysant et en organisant ces quatre sociétés, dix rôles d'agent ont été identifiés et documentés.

2.2 Les rôles des agents

La société dédiée aux utilisateurs comporte trois rôles :

- Le contrôleur d'interface (IC) contrôle et surveille l'interface utilisateur ;
- Le gestionnaire de profils utilisateur (UPM) analyse les demandes et le retour des utilisateurs pour apprendre leurs préférences dans les résultats ;
- L'archiviste des profils utilisateur (UPA) sauvegarde, fournit et questionne les profils des utilisateurs afin de répondre aux besoins des autres agents. Il compare également les nouvelles annotations et les profils d'utilisateur pour détecter les nouveaux documents intéressants pour un utilisateur et suggérer leur consultation.

CoMMA utilise la plate-forme JADE qui est compatible avec le standard FIPA. Ainsi les agents de la société d'interconnexion jouent deux rôles définis par le standard FIPA[1] :

- Le gestionnaire d'agents (AMS) met à jour les pages blanches où les agents s'enregistrent et recherchent l'adresse d'autres agents à partir de leur nom ;
- L'apparieur (DF) met à jour les pages jaunes où les agents s'enregistrent et recherchent l'adresse d'autres agents à partir d'une description de services.

La société consacrée à l'ontologie et au modèle de l'organisation a deux rôles :

- L'archiviste de l'ontologie (OA) sauvegarde et fournit l'ontologie O'CoMMA au format RDFS ;
- L'archiviste du modèle d'entreprise (EMA) sauvegarde et fournit le modèle de l'organisation au format RDF.

La société dédiée aux annotations comporte deux rôles :

- L'archiviste d'annotations (AA) sauvegarde et cherche les annotations RDF d'une archive locale à laquelle il est associé ;
- Le médiateur d'annotations (AM) décompose et distribue les tâches impliquées dans la résolution d'une requête et l'allocation d'une annotation, et notifie la soumission d'une nouvelle annotation aux agents abonnés pour cet événement.

2.3 La gestion des annotations et ses limites

La société dédiée aux annotations est responsable de la gestion des annotations et de la résolution des requêtes sur la mémoire distribuée. L'enjeu était de trouver des mécanismes pour décider où sauvegarder les annotations nouvellement soumises et comment distribuer une requête sans manquer des réponses lorsque les informations nécessaires pour obtenir la réponse sont distribuées entre plusieurs AA (Gandon *et al*. 2002b), (Gandon, 2002c).

Pour affecter à l'agent archiviste une annotation nouvellement soumise, l'AM émet un appel à proposition aux AA en utilisant un protocole d'interaction correspondant à un 'contract-net' imbriqué. Chaque AA mesure la distance sémantique entre les notions utilisées dans l'annotation et celles actuellement utilisées dans son archive.

[1] http://www.fipa.org

L'AA le plus proche gagne l'appel. La distance sémantique est calculée en fonction de la structure taxonomique de l'ontologie (Gandon *et al*. 2002b), (Gandon, 2002c).

Pour résoudre une requête, plusieurs bases d'annotation réparties sur plusieurs AA peuvent être impliquées ; le résultat est une fusion des résultats partiels. Pour déterminer si et quand un AA doit participer à la résolution d'une requête, les AA calculent le recouvrement entre les notions utilisées dans leur base et celles utilisées dans la requête. Avec ces descriptions l'AM ayant demandé leurs services identifie à chaque étape de la décomposition de la requête les AA à consulter.

Une fois les rôles d'AA et d'AM spécifiés avec leurs interactions, des modules du moteur de recherche sémantique CORESE (Corby & Faron-Zucker, 2002) ont été intégrés dans les agents pour fournir les capacités techniques nécessaires à leur rôle.

Les soumissions des requêtes et des annotations sont produites par le contrôleur d'interface. Les utilisateurs disposent d'interfaces graphiques pour les guider lorsqu'ils annotent des documents en utilisant l'ontologie O'CoMMA. A partir de cette interface, le contrôleur d'interface produit une annotation RDF. La tâche d'annotation peut rapidement devenir fastidieuse si un large ensemble de documents pertinents est découvert et l'hypothèse de l'annotation manuelle devient alors peu réaliste. Cependant, certains sites Web ont une structure plutôt statique qui, même si elle est implicite, fournit des indices structurels (type de police, tableaux, séparateurs, *etc.*) qui peuvent être exploités pour automatiser des règles d'extraction. Ceci permet à l'utilisateur de produire automatiquement des annotations à partir de la teneur des ressources. Nous décrivons dans la suite une nouvelle société d'agents fournissant un tel service.

3 Introduction d'une société d'extracteurs

Aucune organisation n'est isolée, elle vit dans une culture, un pays, une société, un marché, *etc.* et beaucoup d'informations intéressantes (car se rapportant à l'environnement de l'organisation, aux activités dans son domaine, *etc*) sont disponibles sur le Web ouvert.. Ces ressources extérieures, mais pertinentes pour l'activité de l'entreprise, peuvent être annotées pour être intégrées le Web sémantique interne. Nous considérons ceci comme le problème dual de la vision habituelle d'un portail d'entreprise. Un portail d'entreprise offre des services de l'organisation sur le Web externe ; c'est une vitrine, un portail externe d'accès à des services internes. Réciproquement les connaissances de l'organisation peuvent être employées pour filtrer et extraire des informations du Web extérieur et fournir aux communautés d'intérêt internes un portail leur permettant d'y accéder de façon choisie et validée ; c'est un portail interne d'accès à des services externes. Dans CoMMA, une implémentation possible pour un tel portail consiste à d'introduire une société d'extracteurs comme le montré la figure 2. Les agents de cette société automatisent l'extraction de certaines informations pertinentes et leur intégration dans la mémoire de l'organisation.

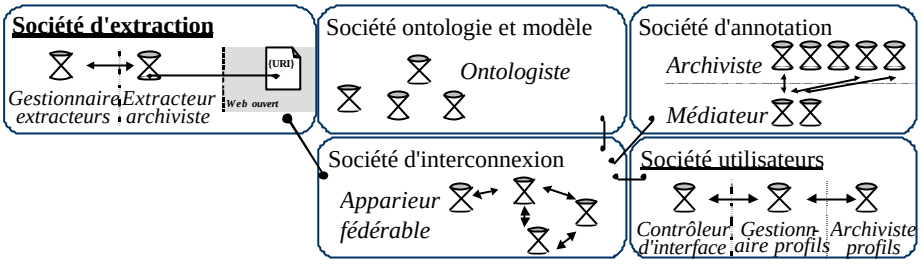

Fig. 2 – *Introduction d'une société d'extracteurs.*

Le Web ouvert est destiné aux humains et un grand nombre de ressources du Web sont non structurées ou semi-structurées. L'extraction d'annotations sémantiques fournissant à l'organisation des pointeurs internes vers des ressources externes hétérogènes, pose donc le problème du développement d'extracteurs spécifiques pour chaque source d'information jugée pertinente. Les recherches récentes sur l'extraction d'information visent à développer des systèmes de génération d'«extracteurs» : des sous-programmes qui extraient automatiquement des données à partir de sites Web et convertissent l'information en un format structuré. Alors que TSIMMIS (Chawathe *et al.* 1994) permet de questionner des sources multiples (sources de Web, systèmes de base de données et système de fichiers), STALKER (Muslea *et al*, 1999) se concentre sur les sources Web exclusivement. Dans TSIMMIS, les extracteurs sont écrits dans un langage de programmation procédural et sont compilés en code exécutable, tandis que dans STALKER (Muslea *et al*. 1999) les agents extracteurs utilisent une structuration hiérarchique de l'information pour apprendre des règles d'extraction sous la forme d'automates finis.

XWRAP (Liu *et al*. 2000) est un générateur semi-automatique d'extracteurs qui repose sur la signification sémantique des balises spécifiques de HTML (par exemple des entêtes et des tables). Les extracteurs ainsi générés dépendent de l'imbrication et de l'orientation de la table et de tout autre élément, donc ils fonctionnent seulement bien avec les sites tabulaires du Web. Le WysiWyg Web Wrapper Factory (W4F) (Sahuguet & Azavant, 1999) est un boîte à outil pour produire des extracteurs du Web. Il contient un langage d'identification et de navigation des sites Web (règles de recherche) et un langage déclaratif pour extraire des données à partir des pages Web (règles d'extraction). Il fournit également un mécanisme pour la projection des données extraites sur une structure de cible. La méthode d'extraction employée dans W4F et notre méthode sont semblables à beaucoup d'égards, mais la principale différence est que tandis que W4F utilise un langage de propre pour l'extraction de données et des règles de projection.

Les sections suivantes présentent notre approche, reposant sur les technologies de XML et étant guidée par une ontologie.

3.1 Processus d'extraction de données

Le Web ouvert est destiné aux humains et un grand nombre de ressources sont non structurées ou semi-structurées. L'extraction d'annotations sémantiques fournissant à l'organisation des pointeurs internes vers des ressources externes, pose le problème du développement d'extracteurs spécifiques pour chaque source.

Pour certains sites du Web tels que les catalogues en ligne ou les bibliothèques électroniques nous avons observé qu'une majeure partie de l'information utilisée pour annoter les pages Web est présente dans le contenu du document lui-même. Ainsi nous avons conçu une solution pour produire automatiquement des annotations RDF en extrayant des données semi-structurées à partir des pages Web. La solution se base entièrement sur les technologies XML (XHTML, de XSLT et RDF) et est guidée par une ontologie formalisée en RDFS. L'outil que nous avons développé permet de produire des extracteurs qui téléchargent les documents d'un site Web et en extraient des données en utilisant un script XSLT basé sur l'ontologie O'CoMMA afin de produire des annotations sémantiques. Dans notre implémentation de l'extraction, nous avons le choix de deux options pour le fonctionnement de l'extracteur:

- La conversion au vol, où l'extracteur emploie ses modèles pour convertir l'information qui lui est demandée à chaque fois qu'il est sollicité. Cette approche a l'avantage de toujours fournir au demandeur l'information la plus à jour mais le processus de conversion peut avoir un impact sur le temps de réponse de l'agent.
- La maintenance d'une image locale, où l'extracteur contrôle la source à intervalles réguliers et, si nécessaire, applique des mécanismes de conversion pour mettre à jour une base d'annotations. Cette approche a l'avantage de fournir un accès rapide et d'être disponible même si le site Web extérieur n'est plus accessible. Cependant, selon les cas, les données peuvent ne pas être à jour et de plus la quantité d'informations recopiées en interne peut être importante.

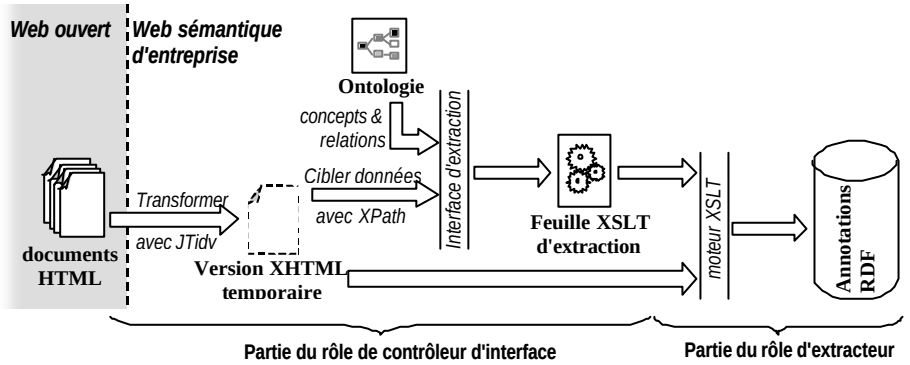

Fig. 3 – *Processus d'extraction.*

Nous avons choisi la deuxième option pour générer des annotations car elle permet que les annotations soient déjà disponibles lors du traitement d'une requête et elle découple et isole l'intranet de l'Internet, ce qui est une caractéristique appréciable pour la sécurité et la disponibilité.

Notre approche se concentre sur des sites Web bien structurés où l'information peut être fréquemment mise à jour, mais où la disposition et l'organisation de ces pages demeurent les mêmes, ou du moins ne changent pas trop. Beaucoup de sites de ce

type existent sur le Web : les bibliothèques électroniques, les sites de la météo, des catalogues de produits, des sites boursiers, etc. Comme illustré sur le schéma 2, nous procédons en trois étapes:

1. Le système télécharge le code source en HTML et le convertit en XHTML. Le modèle d'extraction est établi en utilisant une page Web choisie parmi l'ensemble des pages à annoter et représentative de leur structure répétitive. Nous supposons que les pages sont repérées par des URL contenant un préfixe, un suffixe, et un compteur au milieu. Pour extraire des données à partir d'une page Web, le système doit pouvoir désigner et accéder aux données contenues dans le document source. Une grande partie des documents HTML actuellement sur le Web sont mal formés (balises manquantes, imbrication incorrecte, *etc.*) en raison de la tolérance des «parsers » des navigateurs Web. Par conséquent, le système convertit les documents HTML en documents XHTML suivant la structure hiérarchique et corrigeant les erreurs grâce à Jtidy, une Java API du HTML-Tidy, qui est un outil recommandé par W3C. Ainsi les étapes ultérieures peuvent utiliser des outils XML pour manipuler la page Web comme un arbre DOM (modèle objet de document) que nous représentons grâce à la librairie JDOM.

2. Le système assiste la conception d'un modèle d'extraction basé sur l'ontologie O'CoMMA. Ici, une annotation contient les informations extraites à partir du document ; elle est structurée par des concepts et des propriétés choisis dans l'ontologie O'CoMMA. Pour annoter un document Web, l'utilisateur navigue dans l'ontologie pour choisir les concepts et les relations (propriétés) qui vont décrire la sémantique du document. Les valeurs de ces propriétés sont des littéraux qui apparaissent quelquefois dans le contenu de la page Web. Le rôle du processus d'extraction est de rattacher ces informations utiles aux concepts et relations correspondants. Dans notre travail, nous supposons que les concepts et les relations dans l'ontologie O'CoMMA sont suffisants pour des annotations. Par contre, des concepts où des relations manquants seront importés dans l'ontologie manuellement ou automatiquement grâce aux méthodes d'apprentissage symbolique.

Nous utilisons XSLT pour décrire les règles d'extraction dans le document XML, en employant des chemins XPath pour localiser les données à extraire. L'utilisation des expressions XPath est particulièrement efficace pour l'extraction de données significatives bien localisées, comme le prix des produits, le nom d'un auteur, les mots-clés d'un article, des tableaux sur les cours de la Bourse, *etc.* Pour assurer la précision et l'automatisation du processus d'extraction, nous avons créé des modèles XSLT génériques fournissant des fonctions de haut niveau comme l'extraction récursive d'une liste de données délimitées par un séparateur fixe (ex. une liste d'auteurs - c.f. figue 3), l'extraction itérative d'une liste de données consécutives dans une structure balisée (ex. les données d'une même colonne d'un tableau), le remplacement de données extraites par des concepts correspondant dans l'ontologie (ex. extraction de mots-clés). Ces modèles sont transparents aux utilisateurs du système, et sont inclus dans les modèles d'extraction qu'ils produisent. L'ensemble est facilement extensible. Enfin, le mécanisme d'extension de XSLT permet de combiner des expressions

régulières dans XPATH, et donc de donner la possibilité d'extraire plus efficacement des données.

```
<xsl:template name="getListItem">
<xsl:param name="list" />
<xsl:param name="delimiter"/>
<xsl:param name="openning"/>
<xsl:param name="closing"/>
<xsl:choose>
 <xsl:when test ="$delimiter = 'br'">
  <xsl:for-each select="$list">
 <xsl:value-of select="$openning" disable-
output-escaping="yes"/>
  <xsl:value-of select="normalize-space()" />
 <xsl:value-of select="$closing" disable-
output-escaping="yes"/>
 </xsl:for-each>
 </xsl:when>
 <xsl:otherwise>
  <xsl:choose>
  <xsl:when test="string-length($list)=0" />
  <xsl:otherwise>
    <xsl:choose>
      <xsl:when test="contains($list,
$delimiter)">
      <xsl:value-of select="$openning"
             disable-output-escaping="yes"/>
       <xsl:value-of select="concat(normalize
-space(substring-before($list,$delimiter)),
'&#10;')" />
      <xsl:value-of select="$closing"
             disable-output-
escaping="yes"/>
       </xsl:when>
       <xsl:otherwise>
     <xsl:value-of select="$openning"
             disable-output-
escaping="yes"/>
    <xsl:value-of select="concat(normalize-
space

($list),'&#10;')" />
     <xsl:value-of select="$closing"
           disable-output-escaping="yes"/>
       </xsl:otherwise>
    </xsl:choose>
   <xsl:call-template name="getListItem">
    <xsl:with-param name="list"
select="substring-after($list, $delimiter)"
/>
  <xsl:with-param name="delimiter"
                        select="$delimiter"
/>
    <xsl:with-param name="openning"
                         select="$openning"
/>
    <xsl:with-param name="closing"
                         select="$closing"
/>
   </xsl:call-template>
  </xsl:otherwise>
  </xsl:choose>
</xsl:otherwise>
</xsl:choose>
</xsl:template>
```

Fig. 4 –*Modèle XSLT générique d'extraction d'une liste de données*.

3. Application du modèle d'annotation XSLT aux sources XHTML pour établir une base d'annotation. Une fois que le modèle est créé, il est employé par un moteur de feuilles de styles XSLT pour transformer toutes les pages en une archive d'annotations pointant vers ces documents.

3.2 Scénario d'interaction entre agents

Dans le système CoMMA, les agents communiquent en échangeant des messages basés sur le langage de communication FIPA ACL. La figure 5 montre le diagramme d'interaction entre agents pour ce scénario qui comprend 6 étapes :

1. Dans le contrôleur d'interface (IC), les utilisateurs choisissent une source de pages Web et construisent un exemple d'annotation RDF pour l'une des pages ;
2. L'IC en dérive une feuille de style XSLT pour extraire l'information à partir du site Web et créer automatiquement les annotations correspondantes en RDF ;
3. Une fois la feuille validée, l'IC contacte un gestionnaire d'extracteurs (WM) et demande la création d'un extracteur archiviste (AWA) en charge de gérer cette nouvelle source des annotations ;
4. L'IC envoie la feuille XSLT et la description des URL à l'AWA créé ;
5. L'AWA crée sa base d'annotations en appliquant la feuille de style, puis s'enregistre auprès d'un médiateur d'annotation (AM) comme n'importe quel archiviste d'annotation (AA), afin de participer à la résolution de requêtes ;
6. L'AWA met à jour sa base en surveillant les changements dans la source.

Ce scénario permet aux utilisateurs de développer et de lancer une population d'AWA, chacun d'eux surveillant la source particulière qui lui est affectée.

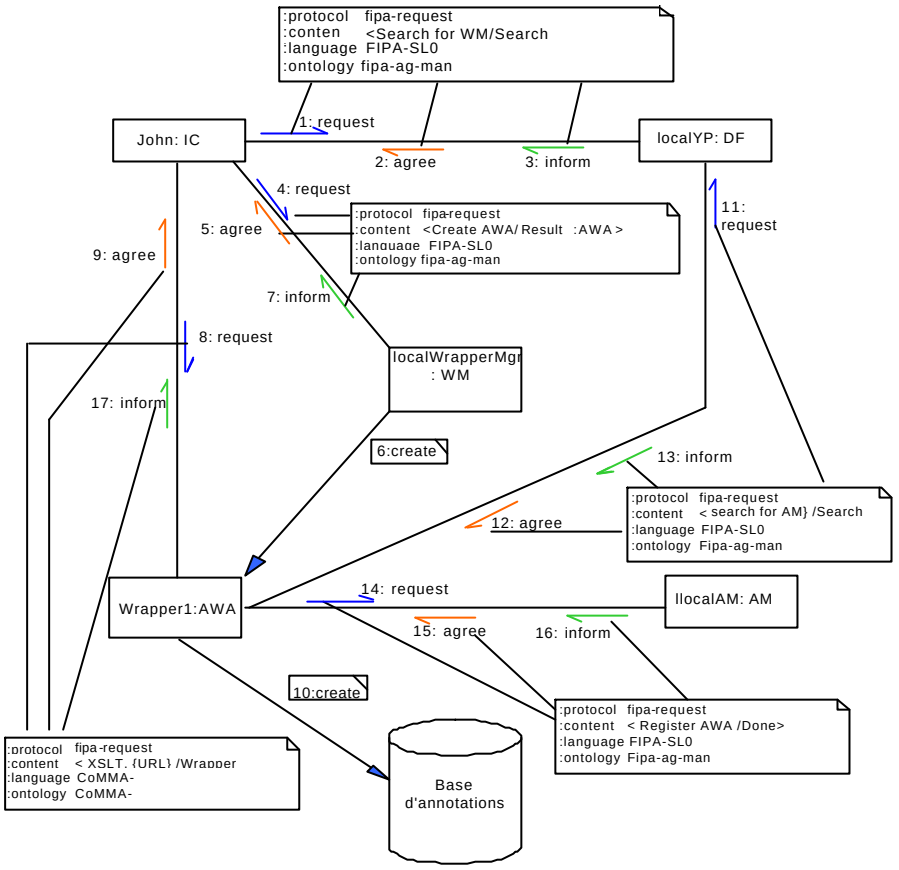

Fig. 5 – *Interactions entre agents pour l'extraction d'annotations.*

3.3 Modifications apportées au contrôleur d'interface

Pour aider les deux premières étapes du scénario, nous avons développé un outil graphique appelé WebAG (Générateur d'Annotations du Web) qui a été inséré dans l'interface existante de l'IC de CoMMA. La figure 6 est une copie d'écran.

L'utilisateur peut indiquer la page Web échantillon dans la zone de texte en haut du cadre (1). La page Web est alors déchargée et convertie en XHTML. La structure hiérarchique résultante (DOM) est visualisée et permet à l'utilisateur de choisir les données à extraire et d'indiquer leur XPath simplement par «drag and drop ». Le cadre (2) permet de naviguer dans l'ontologie et de choisir les concepts et les relations à utiliser dans les annotations. Le cadre (3) permet de définir le modèle d'annotation. Les données sélectionnées dans le cadre (1) et les concepts et les relations choisies dans le cadre (2) sont déposés dans ce cadre. Le cadre (4) montre le code de la feuille XSLT, comme il a été dérivé par le système. S'il le veut, un expert peut intervenir ici pour améliorer le résultat du processus d'extraction au cas où une manipulation particulière

serait nécessaire. L'utilisateur visualise et contrôle l'annotation résultante dans le cadre (5) et dans le cadre (6) il peut demander la création d'un AWA en indiquant l'ensemble des pages auxquelles ce modèle s'applique et en soumettant une requête au WM (URL et feuille XSLT).

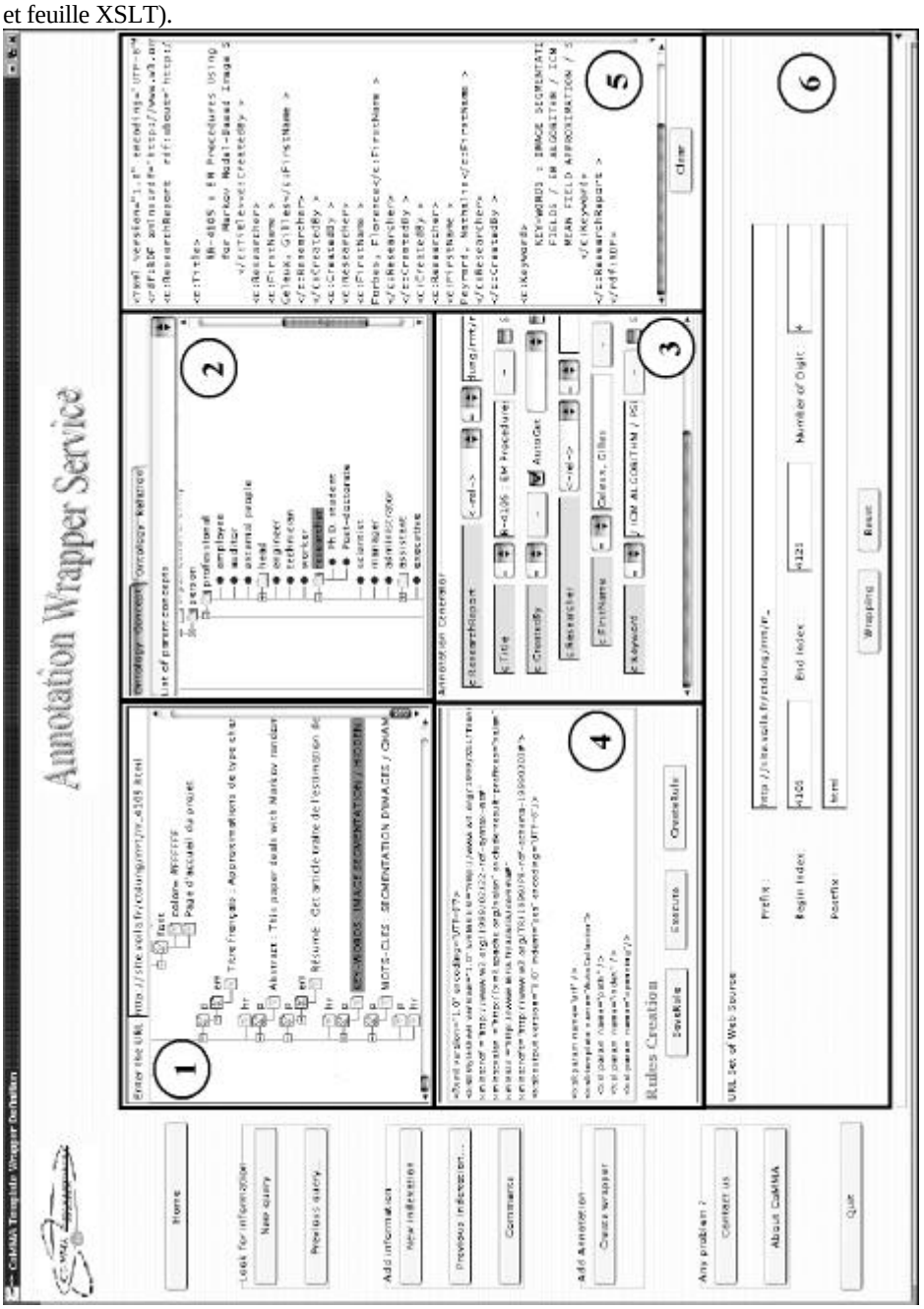

Fig. 6 – *Interface pour produire des feuilles d'extraction d'annotation.*

3.4 Société hiérarchique des extracteurs

La société des extracteurs est hiérarchique, avec un rôle de gestionnaire d'extraction (WM) responsable de la création et du contrôle des extracteurs archivistes (AWA) responsables de l'extraction pour une source d'information ciblée.

Le Gestionnaire d'extracteurs (WM) : Le WM contrôle les AWAs. Actuellement sa tâche est de créer un AWA quand il reçoit une requête d'un IC. Il le crée ainsi en envoyant une demande à l'AMS pour créer un nouvel agent de type AWA. A l'avenir, nous envisageons d'ajouter des services au WM tel que recréer un AWA ayant disparu, contrôler les adresses des AWAs actifs dans le système, tuer un AWA dont l'information n'est plus intéressante, *etc.*

L'extracteur archiviste (AWA) : L'AWA est le travailleur principal dans la société d'extraction. Il fournit deux services : l'extraction d'une source et la mise à disposition de ses annotations pour résoudre des requêtes.

Ainsi le rôle d'AWA est divisé en deux rôles:

- Le premier rôle d'extracteur de page Web est un nouveau rôle dans CoMMA : l'AWA reçoit de l'IC une demande pour extraire des annotations à partir d'une source Web ; elle contient le modèle d'extraction et l'emplacement de la source. L'AWA procède à l'extraction, comme expliqué précédemment, et produit une base d'annotations.
- Le deuxième rôle existait déjà dans CoMMA ; c'est le rôle d'archiviste d'annotation (AA). Ce rôle a ainsi été intégré dans l'AWA avec cette seule modification qu'il refuse systématiquement d'archiver des annotations autres que celles qu'il a produites par l'extraction. Le rôle d'AA enregistre l'AWA auprès d'un médiateur d'annotation (AM), et participe au processus de résolution distribué des requêtes.

4 Discussion et perspectives

Nous avons testé la société d'extraction essentiellement sur trois sites Web : la bibliothèque électronique des rapports de recherches de l'INRIA et la bibliothèque électronique des rapports techniques de l'Institut Technologique de Programmation de l'Université de Carnegie Mellon, la bibliothèque électronique des médicaments PubMed. Les résultats sont encourageants : les extracteurs ont correctement annoté ces sites en utilisant un modèle produit sur une page typique. La figure 7 donne un exemple simple d'une annotation extraite.

Plusieurs améliorations peuvent être envisagées. Tout d'abord, dans les feuilles d'extraction, l'emplacement des données du document est représenté comme un chemin XPath absolu, cette solution est moins robuste lors d'un changement de la structure des pages Web que des chemins relatifs. Nous envisageons d'utiliser un chemin XPath relatif, de sorte que l'emplacement des données puisse être indiqué en fonction de l'emplacement d'autres données et pas systématiquement à partir de la racine du document. De plus, pour améliorer la génération et la mise à jour des feuilles

d'extraction, l'application de techniques d'apprentissage symbolique peut être envisagée à plus long terme.

Fig. 7 – *Exemples d'extraction d'annotation des pages Web de l'INRIA et de PubMed.*

En utilisant des technologies XML, l'implantation du processus d'extraction a été simplifiée par l'utilisation des outils disponibles pour la manipulation de XML, en particulier pour le pré-traitement et l'analyse des documents. Cela devrait également réduire les coûts de maintenance et rendre l'outil plus pérenne. L'utilisation des feuilles XSLT pour la représentation des règles d'extraction, nous permet de reposer sur une norme et les composants développés pour ce système peuvent ainsi être réutilisés. De plus, en utilisant XSLT et XML, il est possible de fusionner des informations extraites

de plusieurs ressources dans un nouveau résultat, donc cette approche peut être utilisée avec des sources hétérogènes.

Des fonctionnalités supplémentaires ont été suggérées pour le gestionnaire d'extracteurs tel que tuer ou ressusciter un AWA afin de gérer la population des extracteurs. De même des fonctionnalités supplémentaires sont envisagées pour l'AWA, en particulier des services de notification : informer d'un changement de la base d'annotation, informer de l'échec de l'application d'un modèle, informer d'un mot-clé qui n'a pas pu être traduit en un concept de l'ontologie, *etc.*

Même si les sources initiales sont structurées de différentes façons, nous pouvons les traduire et les restructurer selon notre ontologie d'une façon qui peut être complètement différente de la structure initiale de l'information. Etant guidé par une ontologie, le processus d'extraction de données peut exploiter un modèle du domaine pour produire non seulement la structure des annotations mais aussi, par exemple, des concepts se substituant à des mots-clés, afin d'être utilisés dans des inférences. La qualité du procédé de recherche dans la mémoire dépend de la qualité des annotations puisque CoMMA emploie des inférences pour exploiter la sémantique des annotations, par exemple pour généraliser ou spécialiser des requêtes de l'utilisateur. Reposer sur des annotations produites automatiquement peut ne pas fournir la qualité exigée et un environnement semi-automatisé dans lequel un opérateur humain est encore impliqué, comme proposé ici, est la garantie d'une qualité acceptable.

Remerciements : Nous remercions tous nos collègues du projet ACACIA pour les discussions très inspiratrices.

Références

AGUIRRE & BRENA & CANTU-ORTIZ. (2000). Multiagent-based Knowledge Networks. *In the special issue on Knowledge Management of the journal Expert Systems with Applications.*

BELLIFEMINE F. & POGGI A. & RIMASSA G. (2001). Developing multi-agent systems with a FIPA-compliant agent framework. *Software Practice & Experience, (2001) 31:103-128, See also JADE : Java Agent Development Framework at http://sharon.cselt.it/projects/jade.*

BERGAMASCHI S. & BENEVENTANO D. (1999). Integration of Information from Multiple Sources of Textual Data, *In Intelligent Information Agent: Agent-Based Information Discovery and Management on the Internet* p53-77, Matthias Klusch, Springer 1999

BERNEY & FERNELEY. (1999). CASMIR: Information Retrieval Based on Collaborative User Profiling, *In Proceedings of PAAM'99*, pp. 41-56. www.casmir.net

BOTHOREL & THOMAS. (1999). A Distributed Agent Based-Platform for Internet User Communities, *In Proceedings of PAAM'99, Lancashire*, pp. 23-40.

CHAWATHE S. & GARCIA-MOLINA H. & HAMMER J. & IRELAND K. & PAPAKONSTANTINOU Y. & ULLMAN J & WIDOM J. (1994). "The TSIMMIS Project: Integration of Heterogeneous Information Sources", *in Proceedings of IPSJ Conference 1994.*

CoMMA Consortium. (2000). Corporate Memory Management through Agents, *In Proceedings E-Work & E-Business 2002, Madrid*, pp 383-406.

CORBY O. & FARON-ZUCKER C. (2002). Corese: A Corporate Semantic Web Engine, *Workshop on Real World RDF and Semantic Web Applications at 11th International World Wide Web Conference, 2002 Hawaii*

DIENG-KUNTZ R & CORBY O. & GANDON F. & GIBOIN A. & GOLEBIOWSKA J. & MATTA N. & RIBIÈRE M. (2001). Méthodes et outils pour la gestion des connaissances : une approche pluridisciplinaire pour le "knowledge management", *2ème édition, DUNOD, 2001, INFORMATIQUES, Série Stratégies et systèmes d'information*

DZBOR & PARALIC. (2000). Knowledge Management in a Distributed Organisation, *In Proceedings of the BASYS'2000 - 4th IEEE/IFIP International Conference on Information Technology for Balanced Automation Systems in Manufacturing, Kluwer Academic Publishers, London, September 2000, ISBN 0-7923-7958-6*, pp. 339-348

GANDON F. (2001). Engineering an Ontology for a Multi-Agents Corporate Memory System, *In Proceedings of ISMICK'01, Université de Technologie de Compiègne*, p209-228.

GANDON F. (2002a). A Multi-Agent Architecture for Distributed Corporate Memories, *Proceedings of the 16th European Meeting on Cybernetics and Systems Research (EMCSR) April 3 - 5, 2002, Vienna, Austria*, pp 623-628.

GANDON F. & BERTHELOT, L. & DIENG-KUNTZ R. (2002b). A Multi-Agent Platform for a Corporate Semantic Web, *in Proceedings of AAMAS'2002, Castelfranchi, C., Johnson, W.L., (eds)* p.1025-1032, July 15-19, 2002, Bologna, Italy

GANDON F. (2002c). Distributed Artificial Intelligence and Knowledge Management: ontologies and multi-agent systems for a corporate semantic web. *PhD in Informatics, Nov. 7th 2002*, INRIA and University of Nice - Sophia Antipolis. Doctoral School of Sciences and Technologies of Information and Communication (S.T.I.C.)

LIU L. & PU C. & HAN W. (2000). XWRAP : An XML Enabled Wrapper Construction System for Web Information Sources. *In Proceedings of International Conference on Data Engineering* (ICDE), San Diego, 2000.

MUSLEA I. & MINTON S. & KNOBLOCK C. (1999). A Hierarchical Approach to Wrapper Induction, *In Proceedings of the Third Annual Conference on Autonomous Agents,* p190-197, Seattle, WA USA MAY 1-5, 1999 Edited By Oreb Etzioni ; Jörg P. Müller ; Jeffrey M. Bradshaw, ACM Press / ACM SIGART

SAHUGUET A. & AZAVANT F. Building Light Weight Wrapper Construction System for Legacy Web Data Sources Using W4F. *In Proceedings of International Conference on Very Large Databases (VLDB)*, Edinburgh, Scotland, 1999.

VAN ELST L. & ABECKER A. (2002). Domain Ontology Agents in Distributed Organizational Memories in Knowledge Management and Organizational Memories, *Dieng-Kuntz, R., Matta, N., (eds), Kluwer Academic Publishers*, p.145-158,Boston, July 2002, ISBN 0-7923-7659-5