

**Outil d'informatique cognitive
pour transactions basées sur le Web Sémantique (CIT-SWT)**

**Rapport de projet
(Séminaire sur le Web Sémantique)**

de

Benoit Lavoie

benoit@benoit-lavoie.ca

Programme de Doctorat en Informatique Cognitive

Université du Québec à Montréal

18 août 2005

Table des matières

1. Introduction.....	1
2. Aperçu du système et des scénarios d'utilisation	1
3. Description du système implémenté	2
3.1 Aperçu de l'architecture du système CIT-SWT.....	2
3.2 Google Web APIs service.....	5
3.3 Logiciels requis pour l'installation de CIT-SWT	6
3.4 Évaluation du système	7
3.5 Présentation de l'interface via un exemple d'utilisation.....	8
4. Comparaison de CIT-SWT avec Swoogle.....	15
5. Conclusion	16
6. Références.....	16

1. Introduction

Ce document décrit un outil informatique que nous avons développé: CIT-SWT (*Cognitive Informatics Tool for Semantic Web Transactions*). Cet outil est accessible via un navigateur et permet d'offrir des services reliés au Web Sémantique (Wikipedia, 2005).

La section 2 de ce document présente un aperçu de CIT-SWT et des scénarios d'utilisation. La section 3 décrit le système plus en détails. La section 4 présente une comparaison de CIT-SWT avec un autre système comparable: Swoogle (University of Maryland, Baltimore County, 2005). La section 5 conclut en présentant l'état actuel du système et en indiquant comment obtenir une version du système pour évaluation.

Ce document a été préparé dans le cadre du Séminaire sur le Web Sémantique du programme de Doctorat en Informatique Cognitive de l'Université du Québec à Montréal.

2. Aperçu du système et des scénarios d'utilisation

Nous avons développé un logiciel, CIT-SWT (*Cognitive Informatics Tool for Semantic Web Transactions*), accessible via un navigateur, et permettant d'offrir deux types de services reliés au Web Sémantique: (1) service d'administration de librairie de documents RDF¹/OWL², et (2) service de consultation de cette librairie. Ces scénarios s'inspirent du scénario d'utilisation sur les portails Web décrit par le W3C (2004) concernant les répertoires de documents RDF tel que le Open Directory Project (2005).

Dans la version courante du système, le service d'administration de librairie de documents RDF/OWL permet à un usager pouvant être, par exemple, un administrateur dans une entreprise de (i) rechercher des documents (RDF/OWL ou autre) sur l'Internet via des paramètres de recherche utilisés pour appeler l'API³ du service de recherche Google; (ii) naviguer parmi les résultats de la recherche; (iii) ajouter à la librairie un ou plusieurs documents RDF/OWL validé selon la syntaxe RDF, et la sauvegardé avec diverses informations (nom du document, description textuelle - retournée par Google ou éditée par l'utilisateur – et URL d'origine); (iv) modifier et/ou détruire des entrées existantes de la librairie. Ce service implémente un scénario Business-to-Business (B2B) dans le sens où des documents se trouvant sur l'Internet sont importés et intégrés à la librairie d'une entreprise.

Le service de consultation de librairie de documents RDF/OWL permet à un usager pouvant être, par exemple, un client de l'entreprise, d'avoir accès au contenu de la librairie: i.e. de (i) naviguer parmi les documents RDF/OWL disponibles; et (ii) effectuer une recherche parmi les documents RDF/OWL en spécifiant une ou plusieurs des

¹ *Resource Description Framework.*

² *Web Ontology Language.*

³ *Application Programming Interface*

informations suivantes: nom d'une étiquette, nom d'un attribut, valeur d'un attribut, contenu textuel. L'interface de CIT-SWT permet d'afficher les listes des étiquettes et des noms d'attributs afin de permettre de mieux cibler les requêtes. Ce service implémente un scénario Business-à-Client (B2C) dans le sens où l'entreprise maintenant la librairie met le contenu de celle-ci à la disposition du client.

Ces services ne sont pas restreints à un domaine d'ontologie en particulier et ils peuvent faciliter l'accès aux ontologies se trouvant sur l'Internet de format RDF ou OWL. Ces services s'inspirent en partie de Swoogle (University of Maryland, Baltimore County, 2005), qui offre des fonctionnalités similaires mais beaucoup plus puissantes et plus sophistiquées. Une brève comparaison est présentée à la section 4. Cependant, il est à noter que Swoogle a été développé au cours d'un projet financé impliquant plusieurs personnes/années de travail alors que notre projet s'est fait sans financement et en quelques jours.

Le système CIT-SWT est fonctionnel. L'intégration de ses composantes dans une architecture serveur est décrite plus bas. Une version du système accompagne ce document.

Le système CIT-SWT peut être vu de façon superficielle comme un “agent” logiciel; le logiciel possède quelques caractéristiques reliées au domaine des agents incluant les suivantes:

- *Caractéristique reliée à la coopération*: le logiciel utilise des protocoles standards de communication tels que HTTP et SOAP afin de permettre et/ou faciliter l'échange de données.
- *Caractéristique reliée à l'autonomie*: le logiciel est implémenté comme une application Web pouvant être accessible en permanence de virtuellement n'importe où et n'est dépendante que de l'initialisation du serveur Web pour sa propre initialisation.

Cependant, le logiciel ne possède pas plusieurs des (autres) caractéristiques de coopération, d'autonomie et/ou d'apprentissage généralement associées aux *agents* (Obaid et Bouchard, 2005).

3. Description du système implémenté

Cette section décrit la version du système CIT-SWT qui est présentement implémenté.

3.1 Aperçu de l'architecture du système CIT-SWT

La Figure 1 illustre l'architecture du système selon la perspective de deux scénarios d'utilisation décrits plus bas.

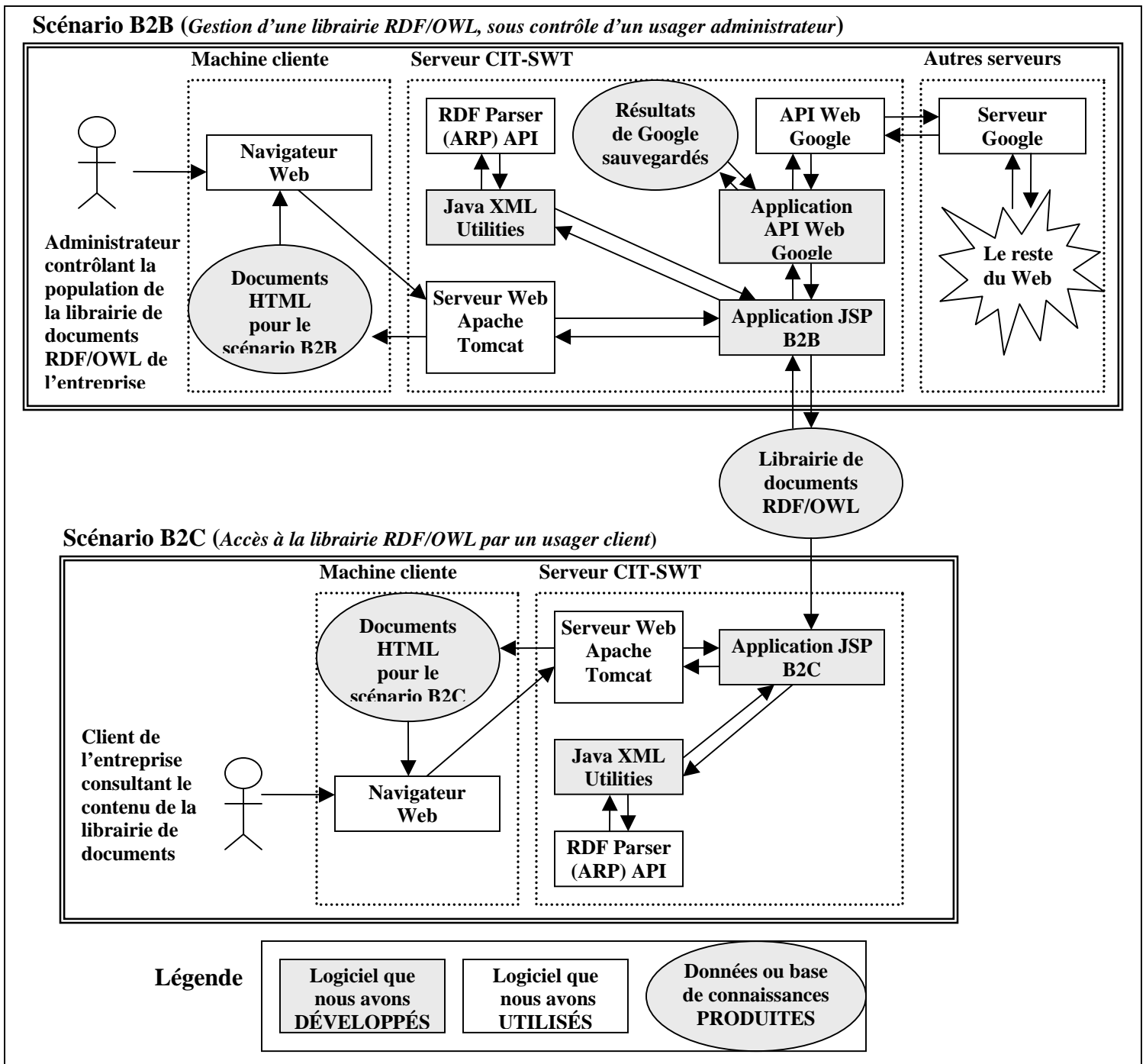


Figure 1. Illustration de l'architecture de notre système CIT-SWT selon la perspective de scénarios B2B et B2C

La partie supérieure de la Figure 1 illustre un scénario B2B où un usager administre une librairie de documents RDF/OWL en utilisant l'interface navigateur du système afin (i) de localiser des fichiers RDF/OWL sur le Web et de les importer dans la librairie, et (ii) de modifier le contenu de cette librairie. L'aspect B2B de ce scénario consiste dans l'importation de documents RDF/OWL se trouvant sur le Web vers le système.

La partie inférieure de la Figure 2 illustre un scénario B2C où un usager accède aux informations se trouvant dans la librairie de documents RDF/OWL.

Les deux scénarios sont intégrés dans le système CIT-SWT avec les pages générées qui donnent accès aux différentes fonctionnalités du système (voir détails plus bas).

Voici une description des composantes utilisées dans l'architecture du système CIT-SWT:

- *Navigateur Web*: interface usager utilisé par CIT-SWT pour les scénarios B2B et B2C. Les navigateurs Microsoft Internet Explorer 6.0 et Netscape 7.2 sont supportés.
- *Serveur Web Apache Tomcat*: serveur Web du domaine public disponible pour la plupart des systèmes d'exploitation (Apache, 2005). Ce serveur supporte les spécifications Java Server Pages (JSP: Sun Microsystems, 2005) permettant la création dynamique de contenu Web et utilisé pour générer les pages HTML de CIT-SWT.
- *Application JSP B2B*: implémentation JSP du scénario B2B pour l'administration d'une librairie RDF/OWL.
- *Application JSP B2C*: implémentation JSP du scénario B2C pour la consultation d'une librairie RDF/OWL.
- *Java XML utilities et RDF parser (ARP)*: Utilités Java pour traiter des documents XML basé sur (i) le Document Object Model de l'API Java (Sun Microsystems, 2005b) d'une part, et sur (ii) l'API du parser RDF ARP (Another RDF Parser) qui est l'analyseur syntaxique utilisé par le W3C pour son service de validation de document RDF en ligne (W3C, 2005).
- *API Web Google et application reliée à cet API*: API Java donnant accès à Google pour la recherche d'information sur l'Internet, et application de cet API que nous avons développé. Cette application sauvegarde les résultats de recherche localement afin qu'ils puissent être réutilisés plus tard s'il y a problème d'accès à l'Internet ou que le service Google n'est pas disponible (l'API permet un nombre limité d'accès au service).
- *Librairie de documents RDF/OWL*: Ensemble des documents RDF/OWL et informations associés à ceux-ci (consistant aux URLs d'où proviennent les documents RDF/OWL, et des descriptions textuelles produites par Google ou édité par l'usager lors des sessions B2B). Ces documents et informations sont sauvegardés dans des fichiers localement lors des sessions B2B, et sont accédés lors des sessions B2B ou B2C.

3.2 Google Web APIs service

Le service basé sur Google Web APIs (Google, 2005) est gratuit et permet d'accéder à des résultats de recherche Internet au moyen d'une licence donnant droit à un maximum de 1000 requêtes par jour — ce qui s'est avéré amplement suffisant pour nos besoins. Lors du développement, ce service s'est avéré très fiable excepté en quelques occasions où le service n'était pas accessible. Afin de palier à ce problème, nous avons développé le système de façon à pouvoir sauvegarder automatiquement des résultats de recherche obtenus avec ce service de façon à ce que ceux-ci puissent être automatiquement disponibles lorsque le service n'est pas accessible.

Le service Google peut être utilisé afin de retrouver programmatiquement des documents de types RDF, OWL ou tout autre type de documents se trouvant sur l'Internet et satisfaisant possiblement d'autres critères de recherche. Lors de nos tests, nous avons pu utiliser le service Google afin de retrouver sur l'Internet des documents RDF dont certains reliés au projet *The Open Directory Project* (2005) et les présenter en résultats à l'utilisateur pour l'ajout à la librairie.

L'une des syntaxes à base de commande-ligne pour accéder au service Google est la suivante (l'API donne un accès à ce service directement à partir de Java), et où *%LicenseKey* est la licence attribuée par Google afin d'accéder au service:

```
java GoogleAPIApplication %LicenseKey %Directive %Arguments
```

Un exemple d'appel pour la recherche de document de type RDF est la suivante:

```
java GoogleAPIApplication %LicenseKey search Google filetype:rdf
```

Un exemple d'extrait de résultat pour cet appel est illustré dans la Figure 2. Notez que cette trace ne contient que les informations sur 2 des 10 documents trouvés par Google et que ces 2 documents ne contiennent pas de bouts de texte (*text snippets*) bien que cela soit généralement le cas. Ces bouts de texte sont présentés par CIT-SWT et peuvent être sauvegardés dans la librairie avec les documents comme descripteur ou ils peuvent être édités par l'utilisateur avant la sauvegarde.

Google Web APIs supportent deux protocoles standards de communication pour les services Internet: SOAP (Simple Object Access Protocol)⁴ et WSDL⁵ (Web Service Description Language) — voir Daconta et collègues (2003) pour une description générale des services Internet. L'API que nous avons utilisé était préconfiguré avec un script SOAP par défaut.

⁴ <http://www.w3.org/TR/SOAP/>

⁵ <http://www.w3.org/TR/wsdl>

```

Parameters:
Client key = *****
Directive = search
Args = Google filetype:rdf
Google Search Results:
=====
{
  TM = 0.058234
  Q = "Google filetype:rdf"
  CT = ""
  TT = ""
  CATs =
  {
    <EMPTY>
  }
  Start Index = 1
  End Index = 10
  Estimated Total Results Number = 716
  Document Filtering = true
  Estimate Correct = false
  Rs =
  {
    [
      URL = "http://www.common-info.org.uk/thoughts/index.rdf"
      Title = ""
      Snippet = ""
      Directory Category = {SE="", FVN="" }
      Directory Title = ""
      Summary = ""
      Cached Size = ""
      Related information present = true
      Host Name = ""
    ],
    [
      URL = "http://loonyboi.com/blog/index.rdf"
      Title = ""
      Snippet = ""
      Directory Category = {SE="", FVN="" }
      Directory Title = ""
      Summary = ""
      Cached Size = ""
      Related information present = true
      Host Name = ""
    ],
    ...
  ]
}

```

Figure 2. Exemple de résultat obtenu avec Google Web APIs pour la requête *Google filetype:rdf*

3.3 Logiciels requis pour l'installation de CIT-SWT

Les logiciels requis pour l'utilisation de CIT-SWT comprennent les suivants:

Coté serveur:

- Apache Tomcat Server 4.1.31 (version testée) ou compatible
- Sun Microsystems' Java 2 SDK 1.4.2 (version testée) ou compatible
- Système d'exploitation: MS Windows XP (version testée) ou compatible

(NT/2000). L'installation sur Unix/Linux est incertaine puisqu'aucun test n'a été réalisé pour ces environnements.

Côté client:

- Microsoft Internet Explorer 6.0 (testé) ou
- Netscape 7.2 (testé)

Du côté serveur, le système doit être configuré avec une licence afin d'accéder au service Google. Cependant, le système inclue aussi quelques résultats obtenus précédemment afin de pouvoir répondre à des requêtes de base si l'Internet ou le service Google n'est pas disponible.

Les autres logiciels libres utilisés par le système sont distribués avec celui-ci.

Pour plus d'information sur l'installation de CIT-SWT, voir la documentation accompagnant la distribution du logiciel.

3.4 Évaluation du système

Nous n'avons pas effectué d'évaluation formelle du système mais nous avons testé les principaux scénarios d'utilisation auxquels nous pouvons penser (instances des scénarios décrits plus haut avec des états particuliers de données). Ces tests indiquent que toutes les fonctionnalités de CIT-SWT semblent fonctionnelles. À notre connaissance, tous les problèmes techniques rencontrés lors du développement ont été corrigés.

Le seul item important encore manquant au système au moment d'écrire ces lignes consiste dans un guide usager. La prochaine section contient une description détaillée de saisies d'écran qui sert présentement d'alternative au guide usager.

3.5 Présentation de l'interface via un exemple d'utilisation

Cette Section présente l'interface de CIT-SWT via un exemple d'utilisation du système implémenté.

La Figure 3 illustre la fenêtre obtenue lors du démarrage d'une session avec le système. L'utilisateur doit spécifier un nom et un mot de passe et choisir la langue d'interaction utilisée par l'interface (français ou anglais). Notez que les textes apparaissant sur toutes les pages générées par le système sont présentés dans la langue sélectionnée. Pour se faire, le système utilise une simple base lexicale bilingue de format XML afin de sélectionner l'expression dans la langue voulue à partir d'un identificateur d'arguments optionnels traités comme des variables.

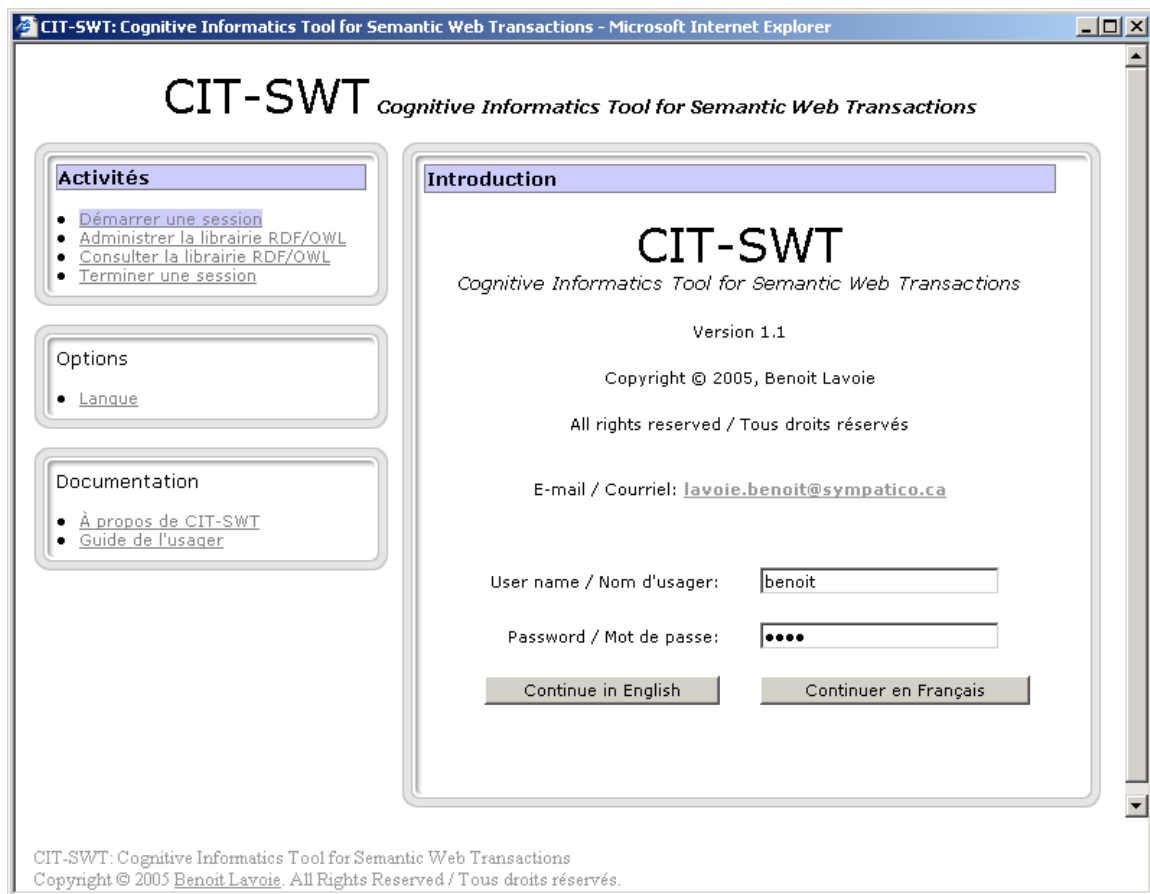


Figure 3. Saisie d'écran de CIT-SWT pour le démarrage d'une session

La Figure 4 illustre la fenêtre obtenue lors de la sélection de l'activité "Administrer la librairie RDF/OWL" se trouvant en haut à gauche de la fenêtre. La fenêtre obtenue permet (i) la navigation dans la librairie de documents existants; (ii) la modification d'un document existant; et (iii) une recherche Web d'un document (RDF/OWL ou autre); et (iv) l'ajout d'un document RDF/OWL à syntaxe valide à la librairie.

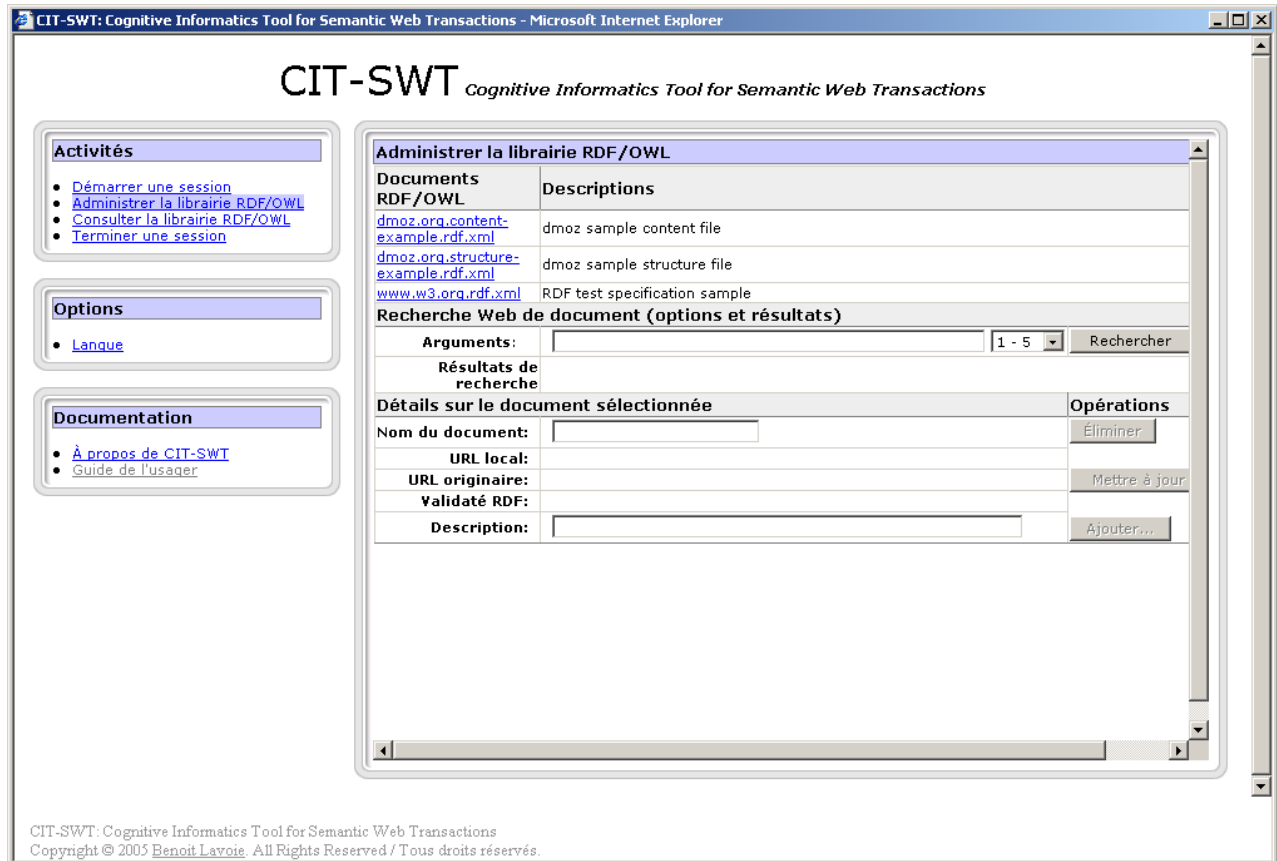


Figure 4. Saisie d'écran de CIT-SWT pour l'administration de la librairie RDF/OWL: illustration d'une librairie préexistante

La Figure 5 illustre la fenêtre obtenue lors d'une recherche Web pour la requête *Google filetype:rdf* limitée aux 5 premiers résultats. La partie inférieure droite de la fenêtre affiche le contenu du document sélectionné (l'affichage du document est celui par défaut du Navigateur utilisé – Microsoft Internet Explorer dans notre exemple).

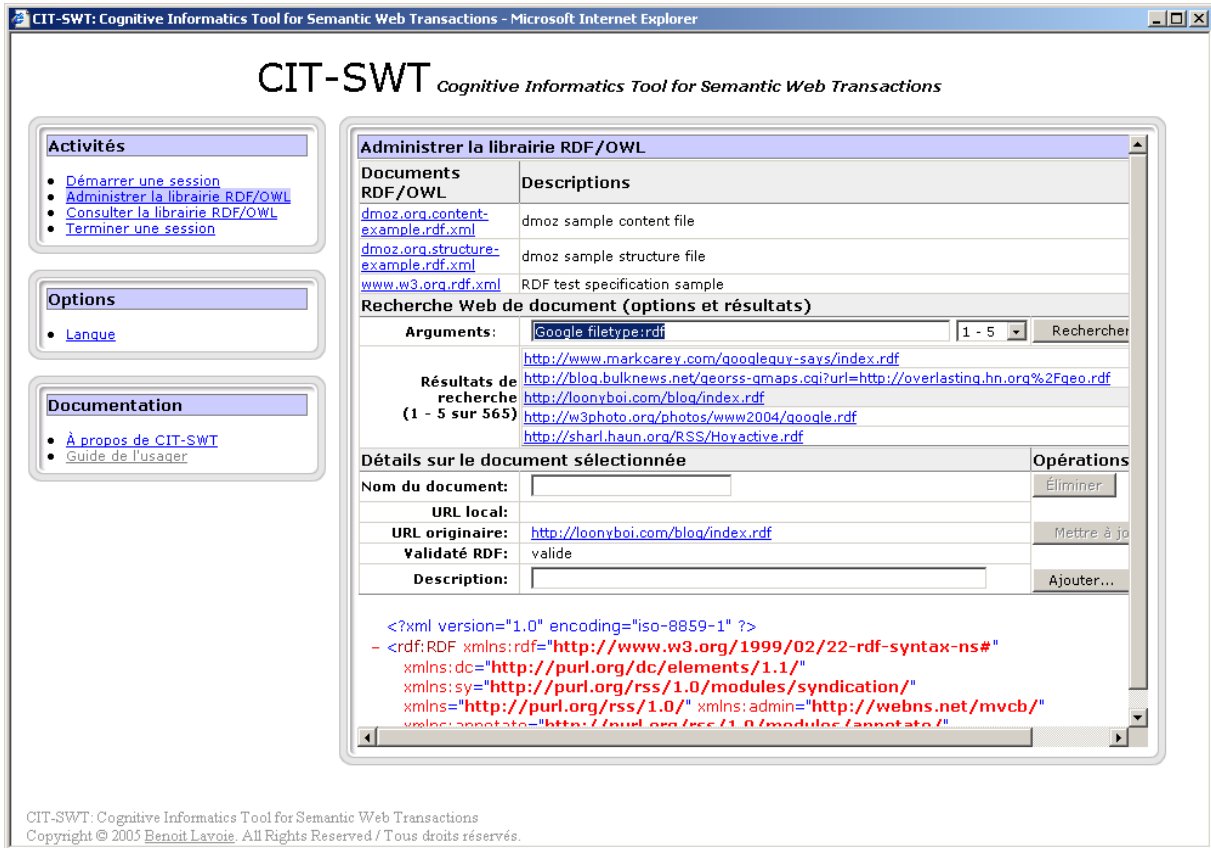


Figure 5. Saisie d'écran de CIT-SWT pour l'administration de la librairie RDF/OWL: illustration des résultats d'une recherche Web sur des documents de type RDF

La Figure 6 illustre la fenêtre obtenue en ajoutant à la librairie un document RDF résultant de la recherche.

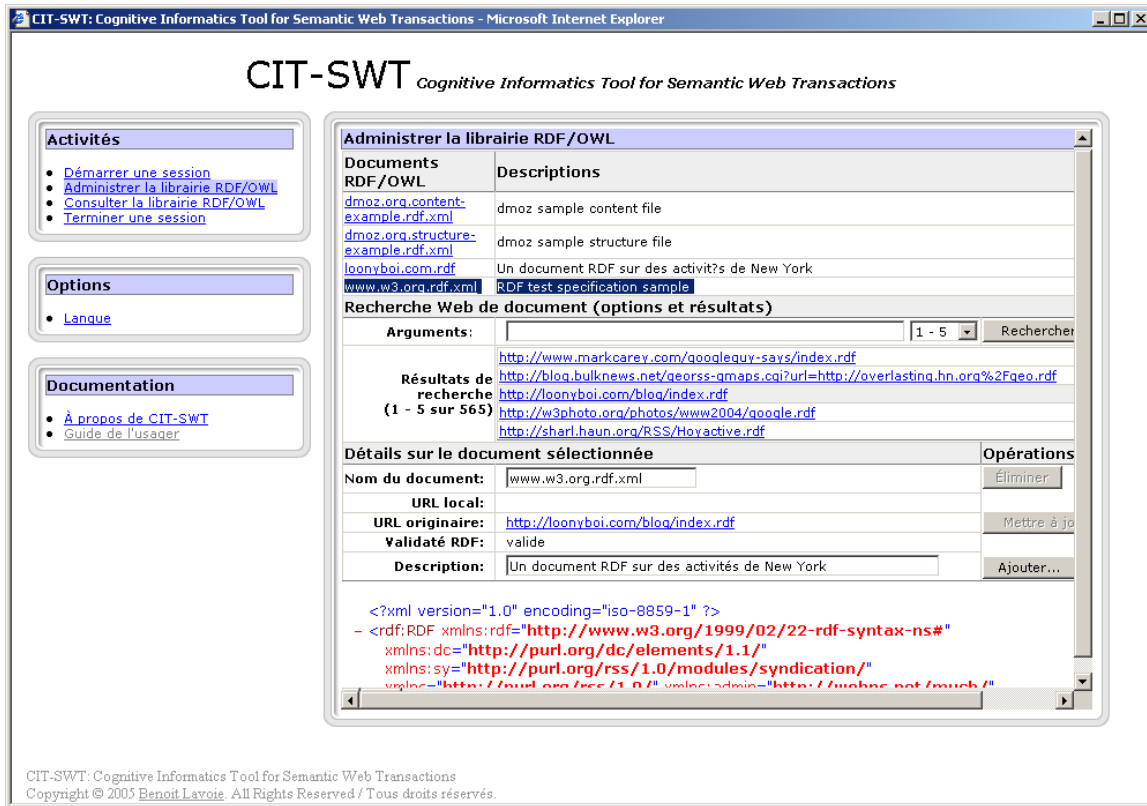


Figure 6. Saisie d'écran de CIT-SWT pour l'administration de la librairie RDF/OWL: illustration de l'ajout d'un document à la librairie

La Figure 7 illustre la fenêtre obtenue lors de l’affichage d’un document résultant de la recherche et qui n’est pas de syntaxe RDF valide, et d’un message d’erreur correspondant. L’interface permet la navigation dans les documents qui ne sont pas de format RDF/OWL, mais elle restreint l’ajout à la librairie qu’aux documents qui sont de syntaxe RDF valide.

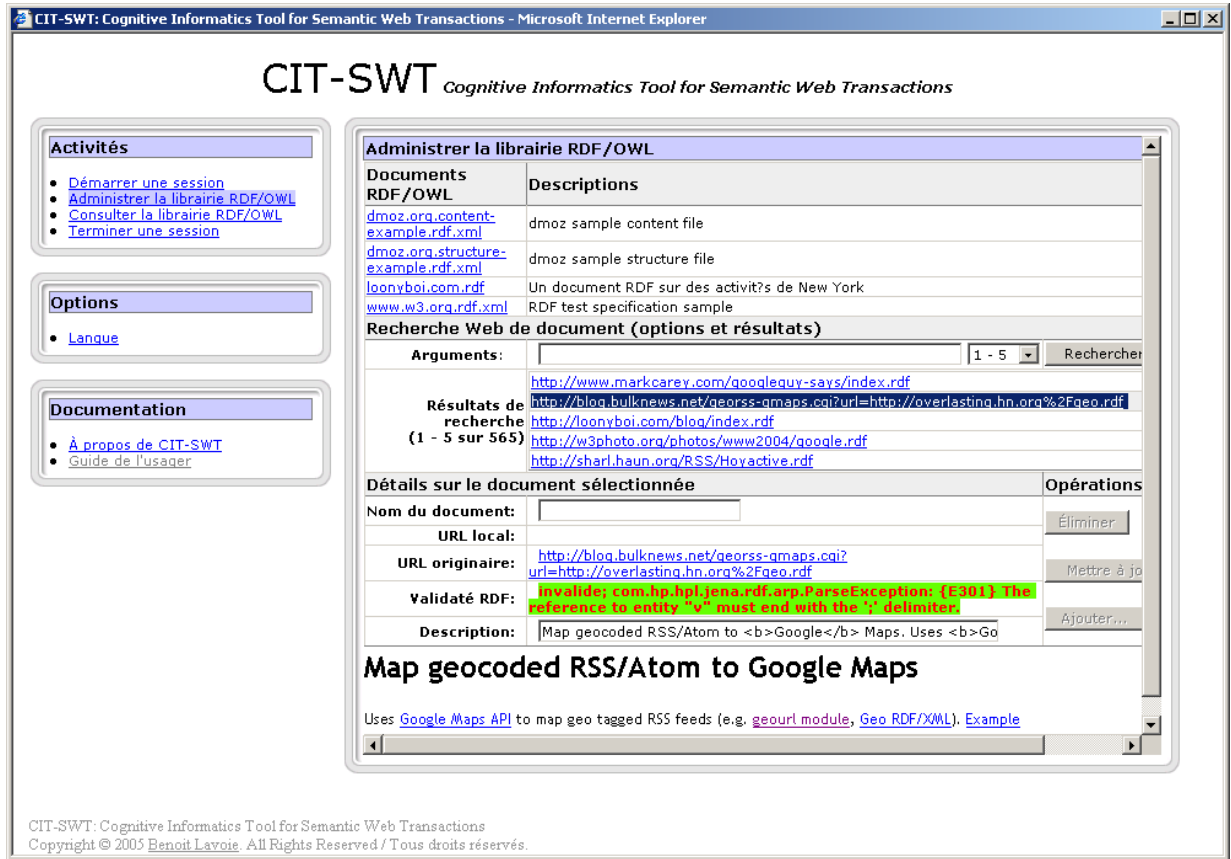


Figure 7. Saisie d’écran de CIT-SWT pour l’administration de la librairie RDF/OWL: illustration d’une erreur de validation RDF de document

La Figure 8 illustre la fenêtre obtenue lors de la sélection de l'activité "Consulter la librairie RDF/OWL" se trouvant en haut à gauche de la fenêtre. La fenêtre obtenue permet (i) la navigation dans la librairie de documents existants; et (2) la recherche de documents à partir d'arguments tels que le nom d'une étiquette, le nom d'un attribut, la valeur d'un attribut et le contenu textuel.

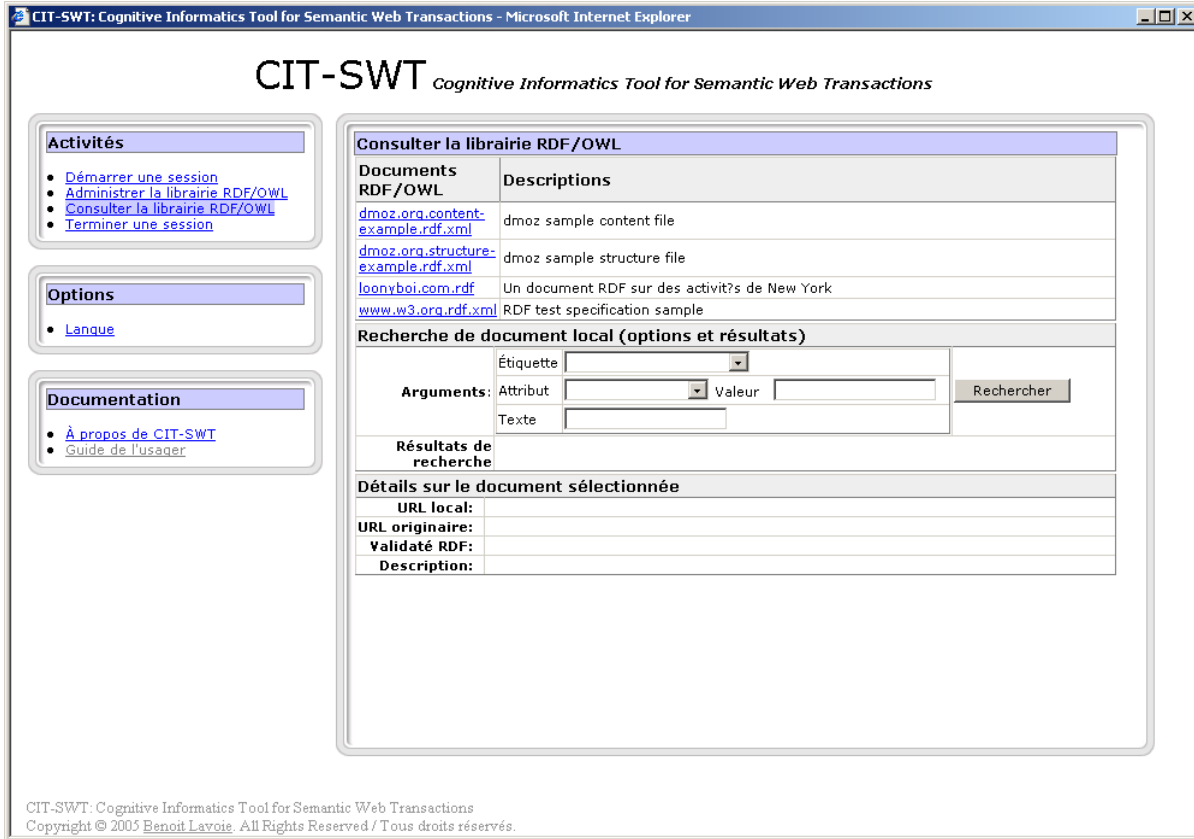


Figure 8. Saisie d'écran de CIT-SWT pour la consultation de la librairie RDF/OWL: illustration avec l'état de la librairie administrée précédemment.

La Figure 9 illustre la fenêtre obtenue lors de la sélection d'un des documents disponibles dans la librairie.

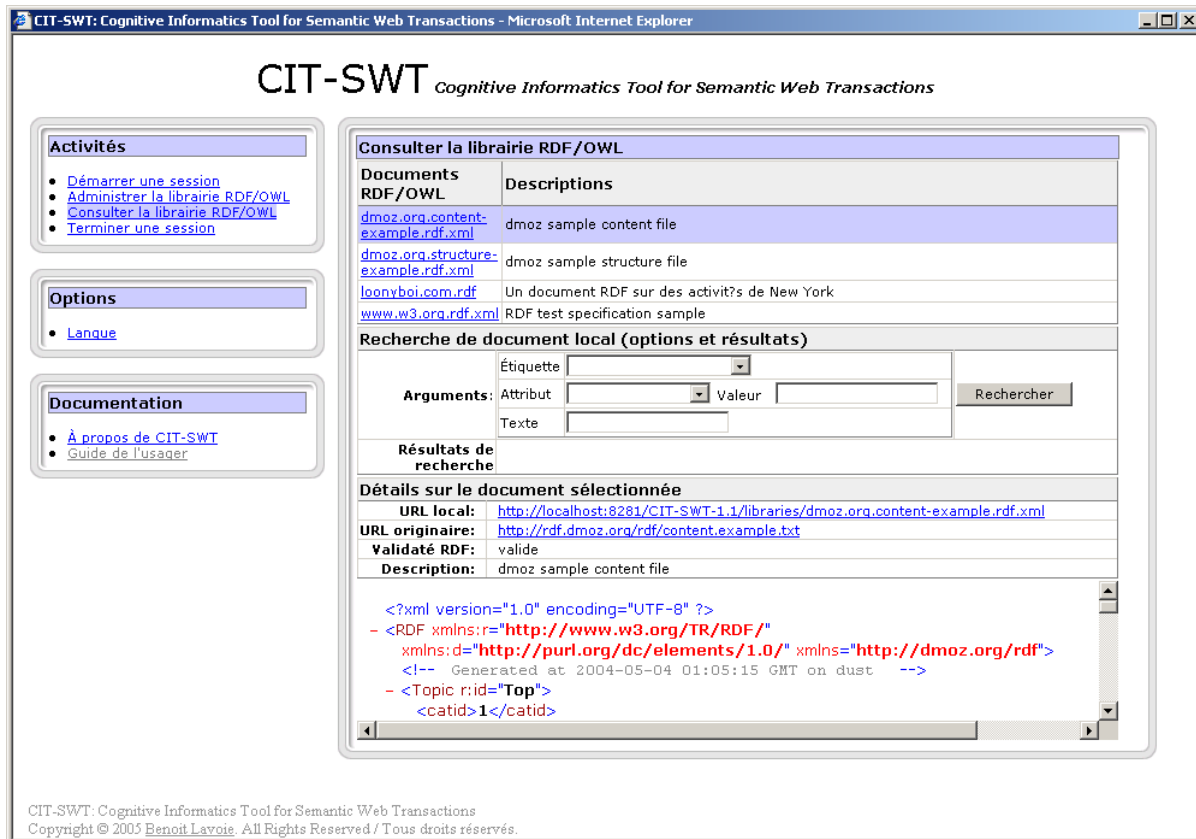


Figure 9. Saisie d'écran de CIT-SWT pour la consultation de la librairie RDF/OWL: illustration de la consultation d'un document parmi ceux disponibles

La Figure 10 illustre la fenêtre obtenue lors de la sélection d'une recherche de document contenant une étiquette "d:Title" dont le contenu texte contient la chaîne de caractères "The 13th Warrior". Notez que la fenêtre du bas a été déroulée manuellement de façon à afficher cette chaîne de caractères dans la saisie d'écran. Cette fonctionnalité peut être obtenue programmatiquement mais n'a pas encore été implémentée.

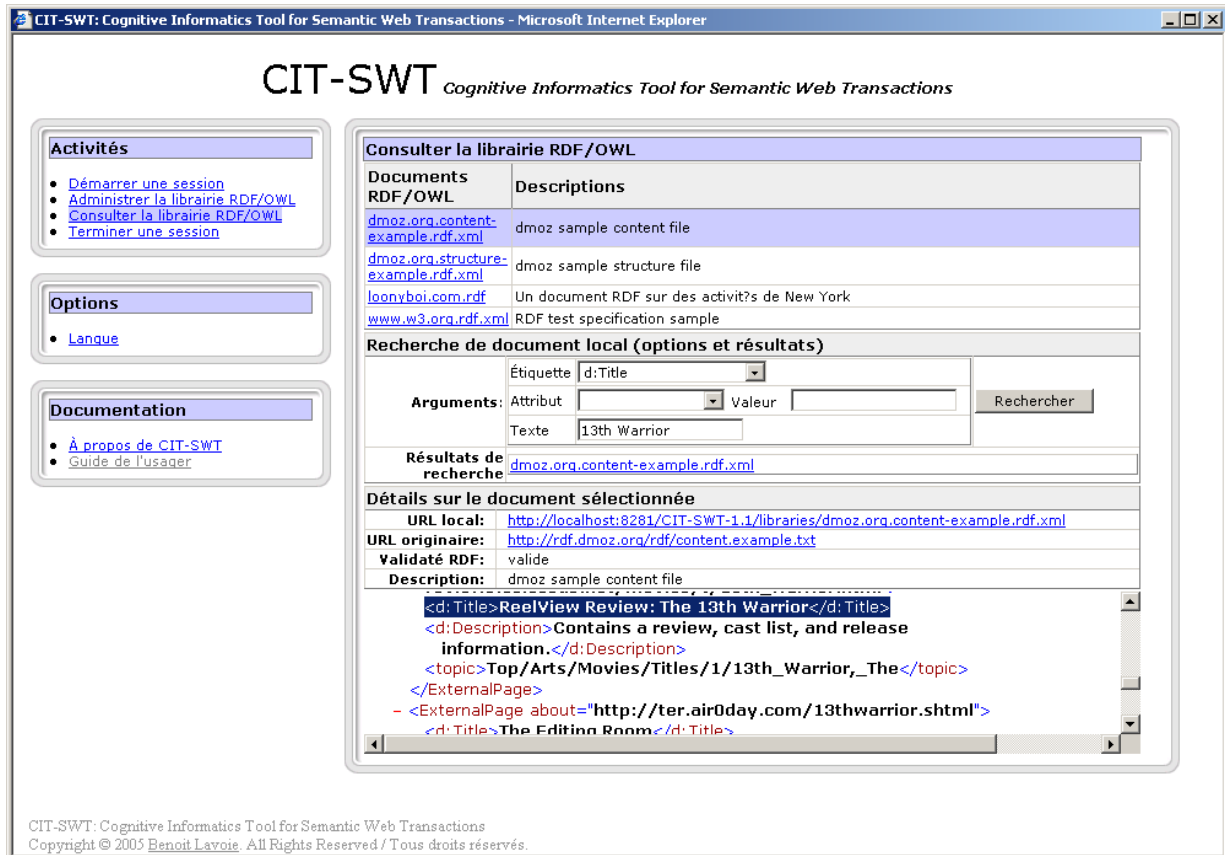


Figure 10. Saisie d'écran de CIT-SWT pour la consultation de la librairie RDF/OWL: illustration des résultats d'une recherche par étiquette et texte

4. Comparaison de CIT-SWT avec Swoogle

La conception de CIT-SWT s'inspire de Swoogle (University of Maryland, Baltimore County, 2005) mais la comparaison est difficile étant donné que CIT-SWT n'offre qu'une fraction des fonctionnalités de Swoogle.

Une différence importante entre CIT-SWT et Swoogle consiste dans le traitement des documents RDF/OWL retrouvés sur l'Internet. CIT-SWT ne traite les documents que pour en vérifier la validité syntaxique RDF et en extraire des informations ponctuelles telles que la liste des étiquettes et la liste des attributs contenus dans les documents. Swoogle extrait plus de métadonnées que ces simples listes items et calcule les relations entre ceux-ci afin de déterminer les similarités dans les documents et procéder à une

indexation raffinée de ceux-ci. Swoogle détermine aussi le rang ontologique d'un document, i.e. son importance comme document du Web Sémantique et sa pertinence par rapport à la requête de l'utilisateur. Ce sont là des fonctionnalités qui sont probablement très utiles à avoir dans le contexte du Web Sémantique mais que nous n'avons pas pu considérer à cause des complexités techniques qu'ils impliquent et qui dépassent le cadre de notre projet.

Nous rappelons que Swoogle a été développé au cours d'un projet financé impliquant plusieurs personnes/années de travail alors que notre projet s'est fait sans financement, par une seule personne et en quelques jours. Cela est un détail important car il illustre qu'avec les technologies présentement disponibles, on peut implémenter en quelques jours un service de Web Sémantique permettant la recherche, le traitement et la l'accès à des documents ontologiques se trouvant sur l'Internet.

5. Conclusion

Dans ce document nous avons présenté un système implémenté, CIT-SWT. Ce logiciel est un serveur Web accessible via un navigateur par protocole HTTP. Ce logiciel permet l'administration (incluant acquisition) et l'accès de ressources du Web Sémantique (des documents RDF/OWL en particulier). Ces fonctionnalités permettent de supporter différents cas d'utilisation dont un Business-à-Business et un Business-à-Client. CIT-SWT est indépendant du domaine ontologique bien qu'il permette la création et l'accès à des bibliothèques de documents RDF/OWL pouvant représenter des ontologies de domaines variés.

Une version du logiciel est disponible pour évaluation. Les personnes intéressées à évaluer le système peuvent adresser leurs requêtes à l'auteur de ces lignes au courriel suivant:

benoit@benoit-lavoie.ca

6. Références

Apache (2005). Apache Tomcat.

<http://jakarta.apache.org/tomcat/index.html>

Daconta, Michael C., Leo J. Obrst, Kevin T. Smith (2003). Chapter 4: Understanding Web Services. *The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management*. John Wiley & Sons.

Google (2005). Google Web APIs.

<http://www.google.ca/apis/>

HP (2005). ARP: Another RDF Parser (The Jena RDF/XML Parser).

<http://www.hpl.hp.com/personal/jjc/arp/>

Obaid, Abdellatif; & Lorne Bouchard (2005). *Qu'est-ce qu'un agent*. Notes de cours (Séminaire sur le Web Sémantique), Programme de Doctorat en Informatique Cognitive, Université du Québec à Montréal.

<http://www.info2.uqam.ca/~obaid/DIC9380/PDF/Agents.pdf>

Open Directory Project (2005) Site Open Directory Project.

<http://www.dmoz.org>

Sun Microsystems (2005a). JavaServer Pages Technology.

<http://java.sun.com/products/jsp/>

Sun Microsystems (2005b). API of Java 2 Platform Standard. Edition. V.1.4.2

<http://java.sun.com/j2se/1.4.2/docs/api/>

University of Maryland, Baltimore County (2005). Swoogle.

<http://swoogle.umbc.edu>

W3C (2004). *Web Ontology Language (OWL) Use Cases and Requirements*. W3C Recommendation, February 10. Jeff Heflin (editor).

<http://www.w3.org/TR/webont-req/>

W3C (2005). RDF Validation Service

<http://www.w3.org/RDF/Validator/>

Wikipedia (2005). *Semantic Web*. Online article from Wikipedia encyclopedia.

http://en.wikipedia.org/wiki/Semantic_Web