



HAL
open science

EvOnto une approche pour maintenir la cohérence entre une ressource termino-ontologique et des annotations sémantiques

Anis Tissaoui, Nathalie Aussenac-Gilles, Nathalie Jane Hernandez, Philippe
Laublet

► To cite this version:

Anis Tissaoui, Nathalie Aussenac-Gilles, Nathalie Jane Hernandez, Philippe Laublet. EvOnto une approche pour maintenir la cohérence entre une ressource termino-ontologique et des annotations sémantiques. 9th International Conference on Terminology and Artificial Intelligence (TIA 2011), Nov 2011, Paris, France. pp.23-30. hal-03011692

HAL Id: hal-03011692

<https://hal.science/hal-03011692>

Submitted on 18 Nov 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Copyright

EvOnto une approche pour maintenir la cohérence entre une ressource termino-ontologique et des annotations sémantiques

Anis Tissaoui, Nathalie Aussenac-Gilles et
Nathalie Hernandez

Institut de recherche en Informatique de Toulouse
Université Paul Sabatier, 118 Route de Narbonne
F-31062, Toulouse cedex 09
{Tissaoui, Aussenac,
Hernandez}@irit.fr

Philippe Laublet

Laboratoire Sens, Texte, Informatique et
Histoire, Université de Paris-Sorbonne,
1 Rue Victor Cousin, 75005 Paris
Philippe.Laublet@paris-
sorbonne.fr

Résumé

En ajoutant une composante lexicale aux ontologies, les ressources termino-ontologiques (RTO) s'avèrent bien adaptées à l'annotation sémantique de documents textuels : les termes permettent de localiser précisément l'évocation d'un concept dans un texte. Au sein d'un système de recherche d'information sémantique, le changement régulier des collections de textes, des connaissances d'un domaine ou des besoins des utilisateurs nécessitent de faire évoluer le corpus, la RTO et les annotations en cohérence. Cet article présente EvOnto, une approche pour gérer l'évolution conjointe de RTO et d'annotations sémantiques de documents textuels. Son originalité est d'exploiter l'état de chacune de ces représentations pour déclencher son évolution, et d'évaluer l'impact sur l'une du changement de l'autre, avant de décider de sa mise en œuvre.

1 Introduction

La perspective du Web sémantique est d'enrichir le contenu du Web actuel avec des descriptions formelles de sorte que les agents logiciels puissent être en mesure de traiter les informations fournies par l'homme sur les pages Web (Tim Berners Lee *et al.*, 2001). Les annotations sémantiques décrivent le contenu de ces pages à

partir de concepts et/ou de relations représentés dans une ontologie. Dans ce cadre, les ontologies jouent un rôle clé : elles fournissent le vocabulaire et la connaissance du domaine qui seront utilisées pour analyser le contenu des pages Web. L'annotation sémantique basée sur l'ontologie exige que l'ontologie soit riche : elle doit représenter non seulement les concepts et relations potentiellement mentionnés dans les documents mais aussi les termes (et leurs variantes) employés pour dénoter ces entités dans les textes. En ajoutant une composante lexicale aux ontologies, les ressources termino-ontologiques (RTO) s'avèrent donc bien adaptées.

Pour annoter de nouvelles ressources, prendre en compte les évolutions du domaine (Maedche, 2002) ou de nouveaux usages (Flouris *et al.*, 2006), une RTO doit être fréquemment modifiée. Ce processus, complexe et coûteux, soulève des questions pratiques : comment identifier la nécessité d'un changement ? Comment traiter les conséquences de ce changement sur les entités de la RTO ou sur les artefacts dépendants de cette RTO, comme les annotations posées à l'aide de la version non modifiée ? Depuis 2002, l'évolution des ontologies a fait l'objet de recherches qui ont permis d'identifier les différents types de changements possibles, les étapes d'un processus de changement, les aides possibles pour assister la gestion des versions d'une ontologie ou encore la gestion des répercussions logiques d'une modification locale sur l'ontologie toute entière (Stojanovic, 2004) (Klein, 2004). Notre recherche s'inscrit dans cette lignée, mais se concentre sur deux caractéristiques spéci-

riques : l'évolution de RTO, ontologies à composantes lexicales dans lesquelles les termes ont un statut à part entière à côté des composants habituels ; l'utilisation de la RTO pour un objectif précis : l'annotation sémantique de documents.

Une hypothèse forte de notre travail est que la prise en compte des impacts d'une modification de la RTO sur la RTO elle-même et sur les annotations, et ce dès la proposition de cette modification, permettra de réduire les effets négatifs de cet impact. Dans le cas d'annotations sémantiques, les effets négatifs peuvent être par exemple de devoir recommencer inutilement l'annotation de certains documents, ou de dégrader la qualité des annotations existantes. Le fait de prendre en compte les termes et les annotations sémantiques utilisant la RTO renouvelle les questions classiques soulevées par l'évolution d'ontologie : Comment appliquer un changement tout en assurant la cohérence au sein de RTO mais aussi la cohérence de la RTO par rapport aux applications qui l'utilisent? Comment analyser les effets d'un changement (au sein de la RTO mais aussi sur ses utilisations) et les gérer au mieux ? Et si plusieurs solutions sont possibles, quelles informations ou critères présenter à l'ontologue¹ pour l'aider à choisir ?

Dans cet article, nous présentons notre proposition pour gérer l'évolution conjointe d'une RTO et d'annotations sémantiques. La section 2 décrit le contexte de ce travail le projet DYNAMO² (*DYNAMic Ontology for information retrieval*), et son méta-modèle de RTO et d'annotation. Nous dressons dans la section 3 un état de l'art sur la gestion de l'évolution d'ontologie. La section 4 présente notre approche d'évolution, qui gère les conséquences de l'évolution de la RTO sur les annotations, et inversement, les conséquences des évolutions du corpus et des besoins en annotation sur la RTO. Pour valider cette approche, nous avons développé le logiciel EvOnto, présenté en section 5 via une étude de cas.

2 Contexte : le projet Dynamo

Le projet DYNAMO intègre la définition d'une méthode et de logiciels qui gèrent l'évolution de RTO et d'annotations sémantiques dans un objectif de recherche d'information. Pour une application donnée, l'architecture du système Dynamo repose sur trois ressources essentielles :

une collection de documents textuels, une RTO et des annotations sémantiques des documents à l'aide de la RTO (Figure 1).

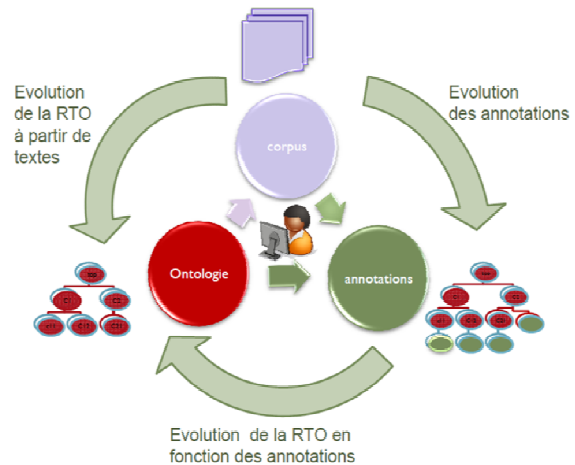


Figure 1 : Le cycle des évolutions

Les documents de la collection sont comparables dans leur structure et leur type de contenu. La RTO représente les termes, concepts et relations du domaine d'application en suivant le méta-modèle OWL présenté dans (Reymonet *et al.*, 2009). Ce méta-modèle présente l'avantage de représenter les termes (et leurs variantes) associés aux concepts de l'ontologie ainsi que les occurrences de ces termes dans les textes annotés. Un graphe d'instances des concepts, termes et de leurs relations définit les annotations.

La RTO peut être éditée et modifiée par un ontologue. Les annotations sémantiques sont créées automatiquement à l'insertion de chaque document. Ces annotations doivent respecter des critères, exprimés sous forme de contraintes : couvrir au mieux les mots du texte et contenir des instances de certains concepts et relations attendus dans chaque document.

L'algorithme appliqué pour créer les annotations consiste à identifier dans les documents des segments textuels (un mot ou une séquence de mots) correspondant à des occurrences de termes qui dénotent des concepts de la RTO. Ces fragments de textes sont alors annotés par des instances (anonymes) de ces concepts. De plus, lorsque ces concepts sont reliés dans l'ontologie et que des indices linguistiques de surface permettent d'identifier les instances en relation, des relations entre instances sont également définies. L'ensemble de ces instances et des relations sémantiques identifiées entre elles dans le document forme un ou plusieurs graphes qui constituent l'annotation sémantique. Les annota-

¹ L'ontologue est la personne qui construit l'ontologie

² <http://www.irit.fr/DYNAMO>, projet financé par la dotation ANR-07-TLOG-004 01

tions proposées par le système peuvent être vérifiées et modifiées par un annotateur humain afin d'assurer une qualité opérationnelle correspondant aux usages attendus. Le premier prototype TextViz (Reymonet *et al.*, 2009) qui réalise ces fonctions est un plug-in de Protégé.

3 Travaux existants

Les travaux qui proposent des processus d'évolution d'ontologies l'organisent en différentes phases (Rogozan, 2009), (Luong, 2007), (Handschuh *et al.*, 2001), (Plessner *et al.*, 2005), (Djedidi, 2009), (Stojanovic, 2004). Nous nous intéressons avant tout à la détection du besoin de changement, au traitement des impacts du changement sur l'ontologie ainsi qu'à la propagation des changements vers les artefacts dépendants de l'ontologie. (Stojanovic, 2004) puis (Flouris *et al.*, 2006) ont défini une typologie de changements organisée selon les structures d'une ontologie, sans traiter vraiment la composante lexicale. Notre approche se démarque en accordant plus d'importance aux termes désignant les concepts, qui sont gérés dans les évolutions.

Ainsi, EVOLVA se focalise à l'acquisition de nouveaux concepts et de relations, en exploitant des textes et en réutilisant des ressources (ontologies ou bases de données) (Zablith *et al.*, 2009). Les concepts intégrés par EVOLVA découlent de l'extraction de termes à partir d'un corpus textuel. Ils sont placés dans l'ontologie à l'aide de nouvelles relations en utilisant des ressources en ligne (comme WordNet) et des ontologies retrouvées à l'aide du logiciel Scarlet.

Une autre approche, «ECCO³», se focalise sur la dimension collaborative et contextuelle de construction d'ontologies. Elle est au cœur du système CoSWEM⁴ (Corporate Semantic Web Evolution Management) (Luong, 2007) qui gère l'enrichissement de vocabulaires structurés à partir de textes, et produit des annotations sémantiques. Il garde l'historique du processus d'élaboration et de modification d'une ontologie sous forme de métadonnées. Ces traces sont récupérées pour propager les changements de l'ontologie aux annotations qui la référencent selon des règles qui détectent et corrigent les annotations devenues obsolètes après l'évolution, en accord avec des stratégies d'évolution.

³ www-sop.inria.fr/edelweiss/projects/ewok/publications/ecco.html

⁴ www-sop.inria.fr/edelweiss/projects/ewok/publications/coswem.html

La phase de propagation doit non seulement assurer la cohérence globale de l'ontologie (Klein, 2004) mais aussi améliorer, sans les dégrader, les services rendus par les artefacts qui l'utilisent. Notre approche est de prendre en compte cette propagation des évolutions au moment de décider de réaliser ou non un changement..

4 Approche pour l'évolution conjointe de la RTO et des annotations

4.1 Le processus d'évolution d'EvOnto

Le projet Dynamo prévoit plusieurs scénarios d'évolution, qui correspondent à des parcours différents sur la figure 1. L'originalité d'EvOnto est d'assurer un support à ces différents scénarios, tout en assurant une forte imbrication entre la ressource termino-ontologique et les annotations sémantiques qu'elle permet de produire sur la collection de documents. D'une part, toute évolution de la RTO implique de détecter les annotations devenues incohérentes à cause des changements de la RTO et aussi de les corriger en vue de garantir la consistance de toute la base d'annotations. D'autre part, les évolutions de la base de documents et donc des annotations produites ou à produire peuvent indiquer la nécessité d'évolutions de la RTO, qui en retour impliquent ce maintien de cohérence sur l'ensemble des annotations.

Or les termes sont au cœur de cette imbrication, dans la mesure où le processus d'annotation de TextViz s'appuie sur les termes de la RTO pour annoter les documents par un graphe d'instances de concepts en relation. Le processus d'évolution d'Evonto concerne l'évolution de la RTO et des annotations. L'évolution de la RTO se déroule selon les étapes classiques de l'état de l'art (Figure 2). En revanche, chaque étape est innovante en couplant ontologie et annotations. De plus, l'étape de propagation est particulièrement fouillée, et correspond au processus d'évolution des annotations.

4.2 Déclencher un changement de la RTO

Alors que la plupart des travaux font l'hypothèse que les évolutions sont déclenchées à partir d'une analyse de l'ontologie elle-même (de son incohérence ou de ses lacunes), dans EvOnto, nous avons prévu trois motifs de son évolution : (i) une évolution des connaissances du domaine (apparition d'un nouveau terme ou d'un concept,

modification du sens d'un terme, etc.), (ii) le repérage d'erreurs ou d'améliorations possibles par l'ontologie (un terme n'est pas associé au bon concept, la signature d'une propriété n'est

pas correcte, deux concepts devraient être fusionnés, etc.), (iii) une analyse des nouvelles annotations suite à une évolution du corpus.

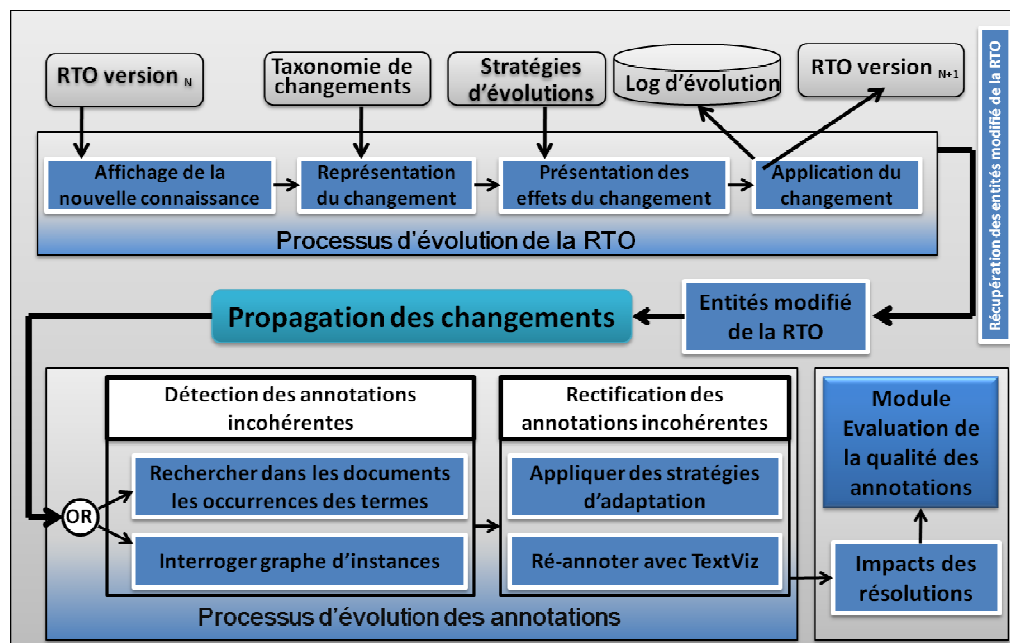


Figure 2: Méthodologie d'évolution d'une RTO et d'annotations sémantiques

Cette dernière situation est une des originalités d'EvOnto : faire évoluer la RTO à partir des annotations. Ce processus est basé sur 3 étapes.

Tout d'abord, des documents sont ajoutés ou retirés de la collection de textes. Ces documents sont ensuite annotés automatiquement à partir du module dédié. Cependant l'ontologie peut ne plus être en adéquation avec le corpus, ce qui mène à la création automatique d'annotations incomplètes ou fausses. En effet, le retrait de documents peut rendre des termes et/ou des concepts inutiles. De plus, l'ajout de nouveaux documents peut mettre en évidence des termes contenus dans ces documents, pertinents pour leur annotation mais absents de la RTO avec le sens considéré. Dans ce cas, le système n'est pas capable d'annoter correctement le document.

Ensuite, les annotations produites automatiquement pour ces documents sont évaluées. Cette étape est une originalité de notre approche. Une mauvaise annotation peut être détectée par l'annotateur à la lecture des annotations, avec des risques d'oubli. Pour repérer rapidement et systématiquement tous les documents mal annotés, EvOnto propose un *module d'évaluation de la qualité des annotations* (Figure 2).

Ce module demande à l'utilisateur de définir des critères de qualité des annotations propres au

corpus et au domaine d'application. Un critère regroupe un ensemble de concepts et / ou des relations qui doivent être trouvés dans chaque annotation. Une annotation ne sera valable que si elle contient au moins une instance de chacun de ces éléments. En général, ces concepts sont des classes de très haut niveau dans l'ontologie. Un autre critère est le nombre de mots du texte auxquels on a pu associer un concept. Dans certains domaines, une bonne annotation est celle qui permet de couvrir le plus possible de termes dans une partie considérée du document (résumé, partie « symptôme » etc.). Pour chaque document, un score évalue le pourcentage d'éléments reconnus, indiquant dans quelle mesure les critères sont vérifiés.

Finalement, l'ontologie peut vérifier une à une les annotations des documents dont le score est faible. Bien que ce processus soit manuel, il est efficace parce qu'il guide l'identification de concepts ou de termes manquants ou à modifier. Ceci revient à suggérer des évolutions.

4.3 Représentation du changement

Par le biais de l'éditeur d'ontologies, l'ontologue apporte les modifications souhaitées. Chaque modification peut avoir de nombreuses conséquences sur les éléments dépendants de celui

modifié, que ce soit localement dans la RTO, ou au niveau des applications utilisant la RTO telles que les annotations. La représentation du changement a pour but d'en présenter les conséquences avant d'en assurer la mise en œuvre.

Pour exprimer l'évolution, nous avons recensé l'ensemble des modifications que l'on peut apporter aux RTOs et nous leur avons donné une signification bien précise (Tissaoui, 2009). Les changements peuvent être *élémentaires* (ajouter ou effacer un terme, un concept ou une relation), ou *complexes* (déplacement ou fusion de concepts). Cette typologie élargit celle de (Stojanovic, 2004) par la prise en compte du lexique de la RTO, qui peut évoluer lorsque les changements portent sur les termes et les liens de dénotations.

Pour chaque type de changement, nous avons identifié différentes stratégies d'évolution, c'est-à-dire les manières de gérer les conséquences de ce changement sur les entités de la RTO liées à l'entité modifiée initialement et sur les usages de la RTO. Ces stratégies permettent d'anticiper les différentes conséquences possibles d'un changement avant de le rendre effectif : par exemple, les fils d'un concept détruit peuvent être associés à son père, à Top ou à un autre concept choisi par l'utilisateur. Les stratégies sont présentées à l'ontologue avec leurs conséquences. Une fois une stratégie sélectionnée et adaptée, le changement est répercuté sur la RTO et les annotations.

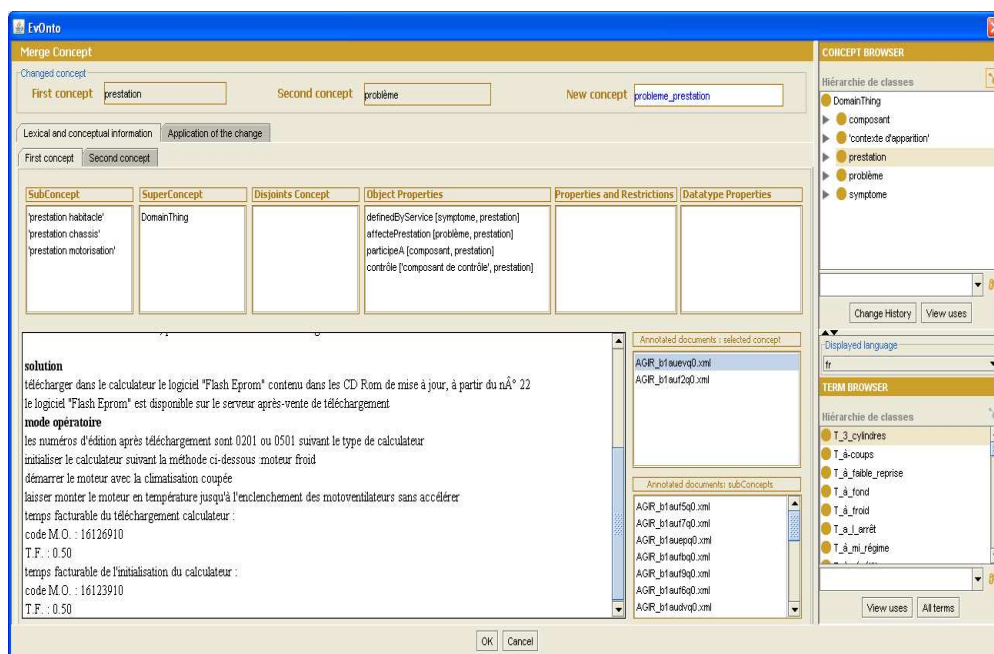


Figure 3 : Présentation des informations relatives à un concept modifié par MergeConcept

Les stratégies d'évolution sont adaptables. Elles reposent sur un ensemble de conséquences pré-définies pour chaque type de changement, mais l'ontologue peut ajuster les conséquences entité par entité s'il le juge nécessaire. Les informations présentées comprennent les éléments de RTO reliés et les documents annotés par l'élément modifié, mais aussi les documents annotés par les éléments de la RTO impactés par cette modification (Figure 3). Les stratégies sont adaptables pour mieux répondre à la diversité des raisons à l'origine d'un changement. En effet, pour une même modification, l'utilisateur adopte telle ou telle conséquence selon les documents ou les concepts. Or le système ne peut proposer que des choix systématiques. Gérer cette diversité

de manière interactive est donc une nécessité. Une interface homme-machine propose à l'utilisateur une séquence d'informations modifiables qui le guident pour ajuster les conséquences ou revoir la modification demandée.

4.4 Impact sur les annotations

Lorsque l'ontologue a choisi de faire évoluer la RTO, ce changement est répercuté sur les annotations. Pour gérer l'évolution conjointe des annotations, nous proposons un processus en deux temps : détecter les annotations qui impliquent au moins une entité de la RTO qui a évolué suite au changement puis modifier ces annotations devenues incohérentes.

Pour détecter les annotations à modifier, deux méthodes sont utilisées. La première est de rechercher dans les graphes d'annotation les noeuds qui correspondent à des entités de la RTO qui ont été modifiées lors du changement. Par exemple, un triplet RDF où apparaît un concept qui a été supprimé doit clairement être remis en cause. Pour la recherche des annotations concernées, nous utilisons des requêtes SPARQL sur l'ensemble de ces graphes correspondant aux annotations des documents.

Une fois les annotations incohérences déterminées, elles doivent être corrigées. Nous proposons de relancer le module d'annotation automatique directement sur les documents concernés. Le temps de calcul reste raisonnable si la base de documents ne comprend que quelques centaines de documents courts. Les annotations devant dans certaines applications être vérifiées par l'ontologue, cette solution n'est pas pleinement satisfaisante. Afin de réduire ce coût, nous sommes en train de définir des stratégies d'adaptation d'annotations qui pourront être directement appliquées à partir de la stratégie choisie pour faire évoluer la RTO.

5 Outil d'évolution : EvOnto

Pour mettre en œuvre les processus que nous venons de présenter, nous avons développé EvOnto (*Evolution d'Ontologie*) (Tissaoui *et al.*, 2011) qui se présente comme un complément à TextViz. Il est intégré dans Protégé sous forme d'un plugin. Nous illustrons le processus d'évolution d'une RTO avec EvOnto à l'aide d'un exemple issu du domaine des incidents logiciels. Un extrait de la RTO du domaine est présenté en figure 4. Selon le modèle de TextViz (Reymonet *et al.*, 2009), les termes sont des classes, reliées aux classes représentant les concepts par la propriété «dénote». Les instances des termes sont alors les occurrences identifiées dans les documents. Sur la figure 4, par exemple, les termes «T_error» et «T_edition_default» dénotent le concept [Edition_Default], fils du concept [Default] dont il hérite la propriété «located-in» qui le lie à [Component].

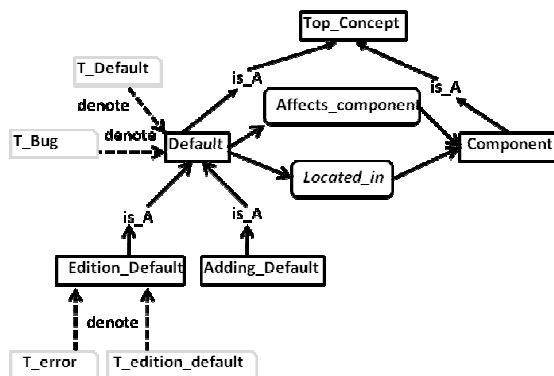


Figure 4. Extrait d'une RTO relative aux incidents logiciels

La figure 5 présente les annotations sémantiques de deux documents (fiche 1077 et fiche 1084) générées à partir la partie de la RTO de la figure 4. Les annotations des deux fiches contiennent le concept [Edition_Default] car une occurrence du terme «T_error» a été identifiée dans chacune.

```

1. <Term_fr rdf:ID="T_error">
2. <dénote>
3. <Concept rdf:ID="#Edition_Default">
4. < rdfs:subClassOf rdf:resource="#Default"/>
5. < Edition_Default rdf:ID="Onto_Individual_1051">
6. < Edition_Default rdf:ID="Onto_Individual_1084">
7. </Concept>
8. <T_Error rdf:ID="Onto_Individual_1052">
9. <document_id rdf:datatype="&xs:string">Fiche-1077.xml</document_id>
10. <designer rdf:resource="#Onto_Individual_1051"/>
11. <field_id rdf:datatype="&xs:string">Resum&#233;</field_id>
12. <spotted_words rdf:datatype="&xs:string">[Error]</spotted_words>
13. <term_offset rdf:datatype="&xs:string">[0]</term_offset>
14. </T_Error>
15. <T_Error rdf:ID="Onto_Individual_1085">
16. <document_id rdf:datatype="&xs:string">Fiche-1084.xml</document_id>
17. <designer rdf:resource="#Onto_Individual_1084"/>
18. <field_id rdf:datatype="&xs:string">Resum&#233;</field_id>
19. <spotted_words rdf:datatype="&xs:string">[Error]</spotted_words>
20. <term_offset rdf:datatype="&xs:string">[0]</term_offset>
21. </T_Error>
22. </dénote>
23. </Term_fr>
  
```

Figure 5. Annotations sémantiques réalisées à partir de la RTO de la figure 3

Considérons un besoin d'évolution de la RTO qui se traduit par la modification du concept dénoté par le terme «T_error». A la lecture d'un nouveau document, l'ontologue considère en effet que le terme «T_error» ne dénote pas spécifiquement un défaut d'édition mais plus généralement n'importe quel défaut. Ce changement correspond à l'opération «Move Term», qui modifie l'association d'un terme à un concept via la relation de dénotation : le concept associé est remplacé par un concept nouveau ou existant. Suite à ce choix, l'interface propose à l'utilisateur deux stratégies d'évolution à savoir : associer le terme à un concept existant dans la RTO ou associer le terme à un nouveau concept

que l'interface permettra de créer. Si l'utilisateur choisit d'associer le terme « T_error » au concept existant [Default] (première stratégie), les conséquences du changement de concept dénoté par le terme sont présentées à l'ontologie : toutes les fiches annotées par le terme « T_error » (fiches [Fiche-1077, Fiche-1084]) seront désormais annotées par une instance de [Default] et non de [Edition_Default]. Si l'utilisateur est d'accord pour appliquer ces différentes conséquences, il valide l'opération et le système effectue tous les changements pour obtenir la nouvelle version de la RTO. Si l'utilisateur n'est pas d'accord, l'interface lui permet d'ajuster les conséquences sur les entités liées à l'objet modifié. Ainsi, au vu des conséquences potentielles, il peut renoncer à cette modification, choisir un autre concept relié ou encore « forcer » une des annotations en sélectionnant un autre concept pour annoter.

Si le changement est validé, certains triplets des annotations (dont les triplets 3, 9 et 16, figure 5) deviennent incompatibles avec la nouvelle RTO. Pour actualiser les annotations, l'ontologue peut activer le module d'annotation seulement pour les documents concernés, ou exécuter leur ré-annotation locales par requêtes SPARQL. Ces fiches seront ensuite présentées à l'ontologue pour qu'il en valide l'annotation ou non.

6 Conclusion

Nous avons proposé une approche et un outil pour gérer l'évolution d'une ressource termino-ontologique. Cette approche prend en compte les annotations sémantiques qui sont produites sur la base des termes de la RTO reconnus en corpus. Notre étude apporte plusieurs innovations par rapport aux travaux de la littérature. Tout d'abord, nous sommes intéressés à des ontologies avec une composante lexicale (RTO), définies à partir d'un méta-modèle, où les termes jouent un rôle central. Deuxièmement, notre approche assiste le processus d'évolution en offrant à l'ontologue des informations sur les conséquences d'un changement avant qu'il ne soit mis en œuvre. Ces conséquences prennent en compte les structures liées à la modification dans la RTO ainsi que les annotations sémantiques utilisant cette structure. Cette information appuie la prise de décision et évite un processus très coûteux en tentatives. Nous continuons à améliorer EvOnto en ajoutant de nouveaux types de changements et les stratégies correspondantes.

References

- Berners-Lee T., Hendler J., Lassila O. (2001), The Semantic Web, *The scientific American*. May.
- Djedidi R. (2009). Approche d'évolution d'ontologie guidée par des patrons de gestion de changement. Thèse de doctorat, université Paris-Sud XI Orsay.
- Flouris G., Plexousakis D. and Antoniou G. (2006), A classification of ontology change, Proc. of *the 3rd Italian Semantic Web Workshop* Scuola Normale Superiore, Pisa, Italy. CEUR-WS201.
- Klein, M. (2004). Change management for distributed ontologies. Ph.D.Thesis, Dutch Graduate School for Information and Knowledge Systems. (G).
- Luong P.H. (2007). Gestion de l'évolution d'un web sémantique d'entreprise. Thèse de doctorat, école de Mines de Paris.
- Mäedche A., (2002). *Ontology learning for the Semantic Web*, vol. 665, Kluwer Academic Pub. .
- Plessers, P, De Troyer O. (2005). Ontology change detection using a version log, In Y.Gil, E. Motta, V.R. Benjamins, & M. Musen (Eds.), LNCS: Vol. 3729. *The Semantic Web – ISWC 2005*. Berlin, Germany: Springer-Verlag, 578-592.
- Reymonet A., Thomas J. & Aussenac-Gilles N. (2009). Ontology Based Information retrieval: an application to automotive diagnosis. *International Workshop on Principles of Diagnosis (DX 2009)*. Stockholm M. Nyberg, E. Frisk, M. Krisander, J. Aslund (Eds.), Linköping University, pp 9-14.
- Rogozan D. (2009). Gestion de l'évolution des ontologies : méthodes et outils pour un référencement sémantique évolutif fondé sur une analyse des changements entre versions d'ontologie. Thèse de doctorat, université du Québec à Montréal.
- Stojanovic L. (2004). Methods and Tools for Ontology Evolution, Ph.D. Thesis, Karlsruhe University. Germany.
- Tissaoui A. (2009). Typologie de changements et leurs effets sur l'évolution de Ressources Termino-Ontologiques (Poster) *20e Journées Francophones d'Ingénierie des Connaissances IC2009*, Hammamet (Tunisie) ic2009.inria.fr/docs/posters/Tissaoui_Poster_IC2009.pdf
- Tissaoui A., Aussenac-Gilles N., Laublet P. et Hernandez N. (2011). EvOnto : évolution de ressource termino-ontologique pour l'annotation sémantique. *Journées Francophones sur les Ontologies (JFO 2011)*, Montréal (Canada), 22-23/06/2011, ACM SIGAPPFR, p. 5-17.
- Zablith F., Sabou M., d'Aquin M. and Motta E. (2009). Ontology Evolution with Evolve. *Proceedings of the 6th European Semantic Web Conference (ESWC) LNCS 5554*. eds. L. Aroyo et al., Springer-Verlag, Berlin, Heidelberg, 908-912.