

# Mesure de l'Inconsistance d'une Ontologie basée sur le Degré de Connectivité des Nœuds

Mouhamadou Gaye\*, Ousmane Sall\*\*, Mamadou Bousso\*\*\*, Moussa Lo\*

\*Laboratoire d'Analyse Numérique et d'Informatique – UFR Sciences Appliquées et Technologie  
Université Gaston Berger, SAINT-LOUIS, SENEGAL

[mkhadim.gaye@gmail.com](mailto:mkhadim.gaye@gmail.com), [gaye.mouhamadou@ugb.edu.sn](mailto:gaye.mouhamadou@ugb.edu.sn), [moussa.lo@ugb.edu.sn](mailto:moussa.lo@ugb.edu.sn)

\*\*Département Informatique - UFR Sciences Et Technologies

Université de THIES

THIES, SENEGAL

[osall@univ-thies.sn](mailto:osall@univ-thies.sn)

\*\*\*Département de Management Informatisé des Organisations - UFR Sciences Economiques et Sociales

Université de THIES

THIES, SENEGAL

[mbousso@univ-thies.sn](mailto:mbousso@univ-thies.sn)

**Résumé**—L'évolution de l'ontologie peut être définie comme étant la capacité de gérer les changements de l'ontologie et de leurs effets. C'est une phase inhérente au cycle de vie de l'ontologie. Cependant, la gestion des modifications n'est pas une tâche triviale car l'ontologie doit évoluer d'un état consistant à un autre. Dans cet article, nous proposons une mesure du degré d'impact d'un changement ontologique sur les autres entités de l'ontologie. La mesure se base sur le poids des nœuds du graphe ontologique et sur le sens du flux de propagation. Elle permet de déterminer l'inconsistance qu'une opération de changement peut engendrer.

**Mots clés**—Evolution, Mesure, Inconsistance, Ontologie.

## I. INTRODUCTION

L'évolution de l'ontologie se réfère au processus de modification d'une ontologie en réponse à un certain changement dans le domaine ou sa conceptualisation [5]. Selon [15], une évolution doit (1) permettre de résoudre les changements apportés à l'ontologie, garantir la consistance de l'ontologie concernée et de tous les objets dépendants, (2) être supervisée permettant aux utilisateurs de traiter plus facilement les changements et (3) offrir aux utilisateurs des conseils pour un raffinement continu de l'ontologie. Cette dernière définit un ensemble de primitives de représentation avec lesquelles on modélise un domaine de connaissances ou de discours [8]. Ces primitives évoluent pour s'adapter aux

besoins et augmenter l'historique de la connaissance dans l'ontologie. Etant une entité cohérente, l'ontologie évolue et tout changement d'une de ses composantes peut rendre inconsistants les autres composants taxonomiques et sémantiques de l'ontologie, et propager ses impacts sur les éléments du système. Il est alors important de connaître le degré ou taux d'inconsistance d'une modification d'une entité donnée sur l'ontologie, afin de pouvoir définir les actions à mener pour rendre le système consistant.

Dans cet article, nous proposons une mesure du degré d'impact d'un changement ontologique sur les autres entités de l'ontologie. La mesure se base sur le poids des nœuds du graphe ontologique et sur le sens du flux de propagation d'impact. Elle permet de déterminer l'inconsistance éventuelle de l'opération. Le poids d'un nœud se définit par le nombre de ses fils dans le graphe ontologique auxquels il est susceptible de propager des effets d'onde.

La suite de l'article est organisée comme suit. Nous présentons dans la section 2 un état de l'art des approches de mesure d'inconsistances et de propagation des changements. Dans la section 3, nous détaillons les opérations de changements que nous traitons dans notre modèle. La section 4 est consacrée à la définition de la mesure d'inconsistance et nous terminons par une synthèse de notre travail et de ses perspectives.

## II. REVUE LITTÉRAIRE

La propagation des changements est un processus qui peut affecter les composants de l'ontologie changeante mais aussi les ontologies dépendantes. Le terme « ripple effect » a été premièrement utilisé dans [7] pour décrire la manière dont un changement dans un module doit nécessiter un autre

changement dans d'autres modules. La propagation est modélisée dans [10] comme une séquence de clichés et chaque cliché représente un moment particulier du processus. Deux processus y sont discutés : « change-and-fix » et « top-down propagation ». Le processus « top-down propagation » assure l'inexistence de dépendances cycliques. Dans [12], il a été déjà fourni une approche d'analyse des effets d'onde par dérivation des changements. Cette approche propose un algorithme de propagation des changements basé sur la pré-condition, l'invariant et la post-condition de l'opération de changement. Nous avons aussi proposé dans [13], une approche de modélisation des changements basée sur la sémantique axiomatique de Hoare permettant aussi mener des tests de satisfiabilité sur les différentes opérations. Cependant, l'évaluation de l'impact sur les entités en dépendance n'est pas prise en compte par cette approche. Si la propagation des changements est encore une thématique de recherche dynamique, la mesure de l'inconsistance, elle n'en demeure pas moins.

Une mesure de l'inconsistance consiste à quantifier la contribution de chaque axiome ou assertion d'une base de connaissances dans l'ensemble des inconsistances. Elle donne une vue sur la sévérité de l'inconsistance dans le système. La mesure permet ainsi de savoir si l'inconsistance doit être traitée ou ignorée. L'approche dans [9] consiste en une mesure qui prend en compte le nombre variables affectées par une inconsistance et la distribution de l'inconsistance sur les formules de la base de connaissances. Une méthode de réduction d'inconsistances par suppression, affaiblissement ou scission de formules est proposée dans [6]. J. Grant et A. Hunter, auteurs de cette publication, y font une présentation de plusieurs mesures d'inconsistance et d'information, toutes incompatibles les unes des autres. L'approche proposée dans [18] définit un degré d'inconsistance d'une ontologie DL-Lite utilisant une méthode nommée le « three-valued semantics ». L'algorithme ainsi proposé est d'une complexité PTime. Les valeurs de Shapley [4] sont le support de la mesure d'inconsistance proposée dans [2] dont le modèle est indépendant de tout langage de raisonnement. L'approche dans [14] présente une méthode d'estimation du coût de développement d'une ontologie en faisant d'abord état des cinq méthodologies d'estimation de coût : la méthode analogique, la méthode « bottom-up », la méthode « top-down », la méthode Delphi et la méthode algorithmique. Cette étude permet de connaître l'impact, en termes de coût, d'une modification survenue dans le cycle de vie de l'ontologie. G. Xiao et Y. Ma dans [9] proposent une mesure de l'inconsistance qui combine des approches syntaxiques et sémantiques. Cette mesure est basée sur le nombre de variables des sous-ensemble minimum insatisfiable et sous-ensemble minimum consistant. Dans [17], M. Thimm investit la probabilité logique conditionnelle pour définir une mesure de l'inconsistance qui est une simplification de la mesure  $\eta$ -inconsistance. Cette mesure s'intéresse à la plus forte probabilité qui peut être assignée de manière consistante aux conditions de la base de connaissances et définit la valeur de l'inconsistance inversement proportionnelle à cette probabilité. Dans [3] est présente une méthode de mesure de l'inconsistance s'appuyant sur la théorie de l'évidence notamment celle de Dempster-Shafer. L'algorithme calcule

l'inconsistance de chaque atome, de chaque formule, de chaque ensemble de formules et détermine l'inconsistance de l'ontologie à partir de ces mesures. Dans [1] l'approche proposée combine l'entropie de Shanon et les opérations de satisfiabilité pour mesurer le taux d'information fourni après une opération de changement d'une ontologie. Cette méthode peine cependant à donner des résultats probants dans la mesure où la comparaison de l'état de l'ontologie avant et après une opération de changement n'est pas bien établie. Cette approche, tout comme celles citées ci-dessus, utilisent souvent une base de connaissances quelconque comme champ d'application plutôt qu'un modèle formel d'ontologie rendant parfois difficile l'implémentation.

### III. CHANGEMENTS ONTOLOGIQUES

L. Stojanovic dans [16], propose une classification des changements ontologiques en changements élémentaires, changements composés et changements complexes. Un changement composé est un changement ontologique qui modifie (crée, renomme, change) le voisinage d'une entité ontologique. Un changement complexe est une collection ordonnée de changements élémentaires [11], ou un changement ontologique qui peut être décomposé en une combinaison d'au moins de deux changements élémentaires ou composés [17]. Dans notre approche, nous nous intéressons aux changements élémentaires étant donné que le vocabulaire des changements complexes n'est pas uniformisé, leur nombre d'opérations est illimité.

#### A. Modèle formel de l'ontologie

Un modèle de l'ontologie est formalisé selon [12] comme suit :

$$O = \langle S, L \rangle$$

où  $S$  est la structure et  $L$  le niveau lexical. La structure d'une ontologie  $O$  est alors définie avec le nupplet :

$$S = \{C, R, A, T, CAR_R, H^C, \sigma_R, \sigma CAR_R, \sigma A\}$$

- $C, A, T$  représentent respectivement l'ensemble des concepts, l'ensemble des attributs et l'ensemble des types des attributs ;
- $R \subseteq (C \times C)$  définit l'ensemble des relations associatives entre concepts ;
- $CAR_R$  représente la caractéristique de la relation  $R$  ;
- $H^C \subseteq (C \times C)$  définit la relation de subsumption entre deux concepts  $C_1$  et  $C_2$  ;
- $\sigma_R: R \rightarrow C \times C$  est la signature de la relation associative  $R$  ;
- $\sigma_{CAR}: R \rightarrow CAR_R$  est la signature de la relation spécifiant la caractéristique de la relation  $R$  ;
- $\sigma_A: A \rightarrow C \times T$  définit la signature de la relation d'attribut entre un concept et un attribut.

Exemple 1: nous considérons cette ontologie relative à la mécanique automobile décrite ci-dessous selon notre modèle.  
 $C = \{\text{camion, voiture, moteur, boite, portières, roues, cylindre, essence, radiateur, eau, gasoil}\}$

$R = \{\text{est\_compose, est\_forme, entraine, fait\_tourner, consomme, refroidit}\}$

$A = \{\text{coûte}\}$

$T = \{\text{chaîne de caractères}\}$

$CAR_R = \{\text{synonymie, transitivité, symétrie, inversion}\}$

$SH^C = \{H^c(\text{camion, voiture}), H^c(\text{gasoil, essence})\}$

$S\sigma_R = \{\sigma_R(\text{voiture, est\_compose, boite}), \sigma_R(\text{voiture, est\_compose, portieres}), \sigma_R(\text{voiture, est\_compose, moteur}), \sigma_R(\text{voiture, est\_compose, roue}), \sigma_R(\text{moteur, est\_forme, cylindre}), \sigma_R(\text{moteur, est\_forme, radiateur}), \sigma_R(\text{boite, entraine, moteur}), \sigma_R(\text{moteur, consomme, essence}), \sigma_R(\text{moteur, consomme, eau}), \sigma_R(\text{eau, refroidit, radiateur}), \sigma_R(\text{moteur, fait\_tourner, roues})\}$

$S\sigma_A = \{\sigma_A(\text{gasoil, coule, « 690 »})\}$

### B. Opérations de changement simples

Nous traitons trois catégories d'opérations de changement: la création (créerX), la suppression (supprimerX) et l'ajout (ajouterX).

TABLE I. OPERATIONS DE CHANGEMENT SIMPLES

Opérations	Significations
CréerConcept( $C_i$ )	Créer un concept $C_i$
CréerRelationAssociative( $R_i$ )	Créer une relation associative $R_i$
CréerAttribut( $A_i, T_i$ )	Créer un attribut $A_i$ de type $T_i$
CréerType( $T_i$ )	Créer un type $T_i$
CréerRelation( $R_i, T_i$ )	Créer une relation $R_i$ de type $T_i$
SupprimerConcept( $C_i$ )	Supprimer un concept $C_i$
SupprimerRelationAssociative( $R_i$ )	Supprimer une relation associative $R_i$
SupprimerAttribut( $A_i, T_i$ )	Supprimer un attribut $A_i$ de type $T_i$
SupprimerType( $T_i$ )	Supprimer un type $T_i$
SupprimerRelation( $R_i, T_i$ )	Supprimer une relation $R_i$ de type $T_i$
AjouterRelationAttribut( $A_i, C_j, T_i$ )	Ajouter un attribut $A_i$ de type $T_i$ au concept $C_j$
AjouterSubsompction( $C_i, C_j$ )	Ajouter une relation de subsomption entre les concepts $C_i$ et $C_j$
AjouterCaractRelation( $R_i, CAR_{R_i}$ )	Ajouter une caractéristique $CAR_{R_i}$ à la relation $R_i$
AjouterRelationAssociative( $R_k, C_i, C_j$ )	Ajouter une relation associative $R_k$ entre les concepts $C_i$ et $C_j$
AjouterRelation( $R_k, T_k, C_i, C_j$ )	Ajouter une relation associative $R_k$ de type $T_k$ entre les concepts $C_i$ et $C_j$

## IV. MESURE DE L'INCONSISTANCE

La mesure d'inconsistance de notre modèle se base sur deux paramètres essentiels : le degré de connectivité des nœuds modifiés et le sens du flux d'impact. Le graphe ontologique est un graphe orienté acyclique. Chaque entité de l'ontologie (concept, relation, attribut) représente un nœud et les relations entre les entités sont les arcs du graphe.

### A. Définitions

#### Définition 1 Graphe ontologique

Un graphe ontologique est un couple  $G = (E, \Gamma)$  où  $E$  est un ensemble d'entités qui peuvent être des concepts, des relations, des attributs ou des types, et  $\Gamma$  une application de  $E$  vers l'ensemble  $P(E)$  des parties de  $E$ .

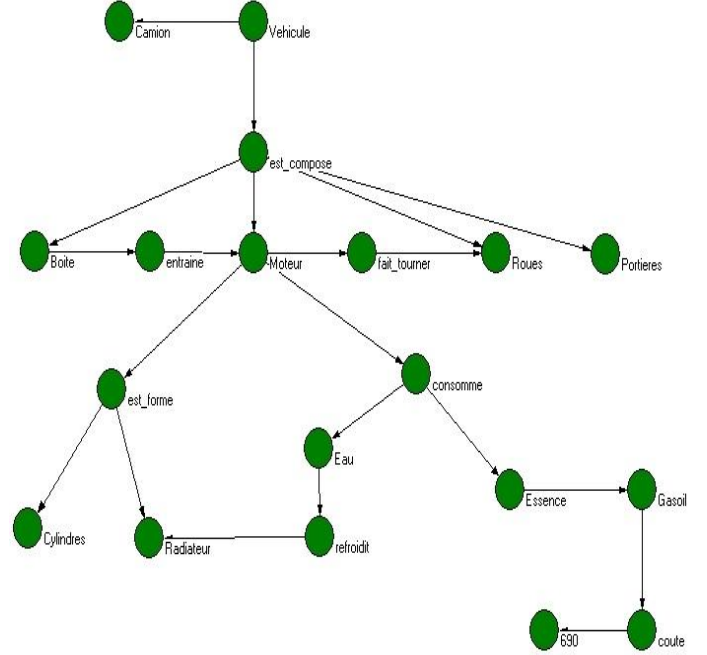


Figure 1 : Graphe ontologique de l'exemple 1

**Flux de propagation** Nous considérons trois relations pouvant propager des impacts aux entités cibles. La relation de subsomption  $H^c(C_i, C_j)$  indique, comme définie dans [12], que toute modification de  $C_j$  peut impacter sur  $C_i$ . Si  $\Delta$  désigne une opération de modification, alors le flux d'impact est  $\Delta(C_k) \mapsto \Delta(C_i)$ . Dans ce même article, il a été établi que pour les relations associatives  $\sigma_R(C_i, R_j, C_k)$ , le sens de flux est  $\Delta(C_k) \mapsto \Delta(C_i)$  sauf pour les relations d'équivalence et d'opposition que notre étude ne prend pas en charge. Pour la relation d'attribut entre le concept  $C_i$  et l'attribut  $A_k$   $\sigma_A(C_i, A_k, T_i)$ , une modification de l'attribut  $A_k$  peut entrainer des conséquences sur le concept  $C_i$  qui l'utilise, donc  $\Delta(A_k) \mapsto \Delta(C_i)$  est le sens du flux d'impact.

#### Définition 2 Impact Entrant

Soit  $E$  un ensemble d'entités d'un graphe ontologique  $G$  et  $C_i$  élément de  $E$ . Nous définissons un impact entrant de  $C_i$  comme suit :

$$I_e(C_i) = \{C_k \in E : \sigma_R(C_i, R_j, C_k) \text{ ou } H^c(C_i, C_k)\}$$

#### Définition 3 Impact Sortant

Soit  $E$  un ensemble d'entités d'un graphe ontologique  $G$  et  $C_i$  élément de  $E$ . Nous définissons un impact sortant de  $C_i$  comme suit :

$$I_s(C_i) = \{C_k, A_i \in E : \sigma_R(C_k, R_k, C_i) \text{ ou } H^c(C_k, C_i) \text{ ou } \sigma_A(C_i, A_i, T_k)\}$$

#### Définition 4 Nœud fils

Soit  $E$  un ensemble d'entités d'un graphe ontologique  $G$  et  $C_i$  et  $C_j$  deux éléments de  $E$ .  $C_j$  est un nœud fils de  $C_i$  s'il existe un arc dans le graphe ontologique de  $C_j$  à  $C_i$ . Nous notons par  $N(C_i)$  l'ensemble des nœuds fils de  $C_i$ .

#### Définition 5 Poids d'une entité ontologique

Soit  $G = (E, \Gamma)$  un graphe ontologique. Nous définissons le poids d'une entité ontologique  $e_i$ , noté  $w(e_i)$ , comme suit :

$$w(e_i) = |Is(ei)| + \sum_{ek \in N(ei)} |Is(ek)| \quad (1)$$

#### B. Mesure de l'inconsistance

Soit  $Op(e_1, e_2, \dots, e_k)$  une opération de changement impliquant les composantes  $e_1, e_2, \dots, e_k$ . Nous proposons une mesure qui estime le degré d'inconsistance d'une opération de changement. Cette mesure se définit ainsi :

$$I(Op) = \frac{\sum_{i=1}^k w(e_i)}{\sum_{i=1}^N w(e_i)} \quad (2)$$

- $w(e_i)$  est le poids de la composante  $e_i$  ;
- $\sum_{i=1}^k w(e_i)$  est le poids total des entités impliquées dans l'opération ;
- $\sum_{i=1}^N w(e_i)$  est le poids total des entités de l'ontologie qui a  $N$  composantes de l'ontologie.

Les entités de l'ontologie sont des composantes connexes, c'est-à-dire qu'il existe des chemins entre les composantes. Ainsi, on peut définir un poids pour chaque entité qui est égal au nombre total de ses entités descendantes. La somme des poids des nœuds du graphe constitue le poids de ce dernier. Mesurer une inconsistance d'une opération revient à déterminer l'impact qu'elle peut avoir sur les entités de l'ontologie. Cet impact se propage à travers les chemins d'où l'intérêt des poids définis. Le rapport de la somme de l'impact lié à degré de connectivité sur le poids total de l'ontologie donne une probabilité d'impact des entités de l'opération sur

le reste de l'ontologie. Cette probabilité définit une mesure. Elle renseigne sur la sévérité de l'opération sur l'ontologie.

#### C. Illustration sur l'exemple

Le graphe ontologie de la figure 1 comporte dix-neuf entités, et nous appelons  $N$  ce nombre.

TABLE II. BASIC AND GENERAL ASSERTIONS

Entité $e_i$	$ N(ei) $	$w(e_i)$
camion	0	0
vehicule	2	33
est_compose	4	31
boite	1	15
entraîne	1	14
moteur	3	13
fait_tourner	1	1
portieres	0	0
roues	0	0
consomme	2	7
est_forme	2	2
cylindres	0	0
radiateur	0	0
essence	1	3
eau	1	2
gasoil	1	2
refroidit	1	1
coute	1	1
« 690 »	0	0
$\sum_{i=1}^N w(ei)$		125

$$I(\text{creerConcept}(\text{bus})) = \frac{w(\text{bus})}{\sum_{i=1}^N w(ei)} = \frac{0}{125} = 0$$

$$I(\text{AjouterSubsption}(\text{bus}, \text{vehicule})) = \frac{w(\text{bus}) + w(\text{vehicule})}{\sum_{i=1}^N w(ei)}$$

$$I(\text{AjouterSubsption}(\text{bus}, \text{vehicule})) = \frac{0 + 33}{125} = 0.264$$

$$I(\text{SupprimerRelationAssociative}(\text{consomme}, \text{moteur}, \text{essence})) = I$$

$$I = \frac{w(\text{consomme}) + w(\text{moteur}) + w(\text{essence})}{\sum_{i=1}^N w(ei)}$$

$$I = \frac{7 + 13 + 3}{125} = 0.184$$

## V. CONCLUSION

Nous avons présenté dans cet article une mesure d'impact d'un changement ontologique sur les composantes connexes de l'ontologie. La mesure prend en compte le poids des nœuds ayant été impliqué dans la modification et le sens du flux d'impact. Elle permet ainsi d'anticiper sur les effets de bord qu'un changement peut produire et d'envisager sa résolution. L'approche est implémentée sous Eclipse avec l'API Jena et testée sur des ontologies de petite taille.

Nous envisageons d'étendre cette approche sur les ontologies lourdes incluant les axiomes. Nous projetons également de travailler sur une méthode permettant de mesurer le degré d'impact d'un changement complexe sur l'ontologie. Notre objectif étant de définir une approche de résolution des inconsistances engendrées par les changements ontologiques aussi bien simples que complexes.

## REFERENCES

- [1] M. Bousso, O. Sall, M. Thiam, M. Lo, E. H. B. Touré, "Ontology Change Estimation Based on Axiomatic Semantic and Entropy Measure", IEEE Conference- Track Signal and Image Technology of the 8th International Conference on Signal-Image Technology and Internet-Based Systems (SITIS 2012) 25-29, November 2012 - Sorrento - Naples, Italy. pp. 458-465.
- [2] X. Deng, V. Haarslev, N. Shiri, "Measuring Inconsistencies in Ontologies", In *The Semantic Web: Research and Applications*, volume 4519 of LNCS, pages 326–340. Springer, 2007.
- [3] L. Dongmei, L. Youfang, H. Houkuan, H. Shudong, W. Jianxin, "Dempster-Shafer Inconsistency Values", In *Chinese Journal of Electronics* Vol. 23, No. 2, Apr. 2014.
- [4] T. S. Ferguson "Game Theory", Second Edition, 2014, Part IV, p. 12-21.
- [5] G. Flouris, D. Plexousakis, G. Antoniou, "A Classification of Ontology Change", In *Proceedings of the 3rd Italian Semantic Web Workshop, Semantic Web Applications and Perspectives (SWAP)*, 2006.
- [6] J. Grant, A. Hunter, "Measuring Consistency Gain and Information Loss in Stepwise Inconsistency Resolution", In *ECSQARU*, pages 362–373, 2011.
- [7] F. M. Haney, "Module connection analysis- a tool for scheduling of software debugging activities", *Proceedings Fall Joint Computer Conference*, p. 173-179, 1972.
- [8] IEEE, 2003, <http://ieeexplore.ieee.org/>
- [9] Y. Ma, P. Hitzler, "Distance-based Measures of Inconsistency and Incoherency for Description Logics", *Proc. 23rd Int. Workshop on Description Logics (DL2010)*, CEUR-WS 573, Waterloo, Canada, 2010.
- [10] V. Rajlich, "A Model for Change Propagation Based on Graph Rewriting", In the *Proceedings of ICSM'97*, September 28 - October 2, 1997 in Bari, Italy, ISBN 0-8186-8013-X.
- [11] D. C. Rogozan, "Gestion de l'évolution des ontologies : méthodes et outils pour un référencement sémantique évolutif fondé sur une analyse des changements entre versions d'ontologies", Université du Québec de Montréal, 2008.
- [12] O. Sall M. Thiam, M. Lo, H. Basson, "A model for ripple effects analysis of cascading problems in ontology evolution", *Int. J. Metadata, Semantics and Ontologies*, Vol. 7, No. 3, 2012.
- [13] O. Sall, M. Thiam, M. Bousso, M. Lo "Using Hoare's Axiomatic Semantics For Checking Satisfiability of Ontology Change Operations", In *IEEE Conference - 8th International Conference on Information Science and Digital Content Technology ICIDT'2012 (5th ICIS)* - June 26 to 28, 2012 in Jeju Island, Republic of Korea, p. 61-66, Vol. 1, ISBN: 978-1-4673-1288-2.
- [14] E. P. B. Simperl, C. Tempich, Y. Sure, "ONTOCOM: A Cost Estimation Model for Ontology Engineering", *The Semantic Web-ISWC 2006*, p. 625-639, Springer Berlin Heidelberg, 2006.
- [15] L. Stojanovic, A. Maedche, B. Motik, and N. Stojanovic. "User-driven Ontology Evolution Management", *Proceedings of the 13th European Conference on Knowledge Engineering and Knowledge Management EKAW'02*, Madrid, 2002.
- [16] L. Stojanovic, J. Schneider, A. Maedche, S. Libischer, R. Studer, T. Lumpp, A. Abecker, G. Breiter, J. Dinger, "The role of ontologies in autonomic computing systems", To appear in *IBM Systems Journal*, Vol. 43, No. 3, 2004.
- [17] M. Thimm, "Inconsistency Measures for Probabilistic Logics", In *Artificial Intelligence*, 197:1-24, April 2013.
- [18] L. Zhou, H. Huang, G. Qi, Y. Ma, Z. Huang, Y. Qu, "Measuring Inconsistency in DL-Lite Ontologies", *IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology – Workshops*, 2009.