

Autonomic Ontologies for Governmental Knowledge Base

Stainam Nogueira Brandao
COPPE/Federal University of Rio de Janeiro
Rio de Janeiro, Brazil
stainam@cos.ufrj.br

Tiago Silva
COPPE/Federal University of Rio de Janeiro
Rio de Janeiro, Brazil
tiagoss@cos.ufrj.br

Sergio Assis Rodrigues
COPPE/Federal University of Rio de Janeiro
Rio de Janeiro, Brazil
sergio@cos.ufrj.br

Luis Araujo
Brazilian Department of the Treasury
Brasilia, Brazil
luis.araujo@planejamento.gov.br

Jano Souza
COPPE/Federal University of Rio de Janeiro
Rio de Janeiro, Brazil
jano@cos.ufrj.br

Abstract— Ontologies are used as models to represent the semantics of the underlying data. The increasing amount of semantic data brings along important technical challenges for development and maintenance of domain ontologies. Our approach aims to provide the ontologies with capacity to evolve through the follow characteristics: (1) have a direct connection with the real world; (2) be able to execute actions in response to external stimuli; (3) execute actions faster than the human response. In other words, a system with proactive behavior must detect symptoms and must be able to handle such situations without human supervision. The paper describes the governmental knowledge base constructed from Brazilian laws and how it is linked and managed by domain ontologies through the autonomic computing paradigm to implement the proactive behavior. The autonomic characteristics were obtained through architecture that treats ontologies as knowledge that requires a management system to monitor known symptoms and execute specified actions on undesirable scenarios. The existing ontologies in the SIOP-LEGIS [3] repository are currently monitored for symptoms presented in this paper and it reached the ability to recommend actions for domain ontologies' evolution. We envision the autonomic architecture will be able to take actions regarding Service Level Agreement (SLA) and improve the human/system interaction.

Keywords— Knowledge base; Linked Data; Autonomic Computing; Ontologies.

I. INTRODUCTION

According to F.C. Albuquerque et al. [1], Open Government Data integration is possible at a global level promoting the use of standard RDF vocabularies. During the triplification process, adequate tools are thus necessary to help users map local concepts to existing RDF vocabularies, in use by other datasets in the Linked Open Data (LOD) Cloud.

A.G. Silva et al. [8] tackles that classification schemes, such as thesauri or taxonomies, are generally created and

maintained by controlled user groups. Furthermore, several methods have recently been proposed for managing ontologies and knowledge bases. However, as described below, they act on specific activities of ontology engineering.

In our work, the main source for the government knowledge base is the Federal Official Gazette, which is a PDF document and contains legislation, jurisprudence and administrative actions [3]. Published by Authority since 1808, today's Brazilian Gazette is the Brazilian Government's Official Journal. It was set up to provide King John with news while he and his court were in Brazil publishing Decree, Laws, Program and Internal Rules. With a new edition every day, today's Brazilian Gazette contains a huge amount of information and statutory notices about decisions and changes at a local and national level. The Brazilian Gazette is a natural candidate for the Government to semantically enable the reuse potential of the information it contains.

F. Bugiotti et al. [5] assert that the amount of available RDF data sources on the Web increases rapidly and, there is a constant need for scalable RDF data management tools.

Our proposal includes the assessment phase and applies its contribution within knowledge bases that use domain ontologies as semantic resources.

In this paper, autonomic ontologies are the domain ontologies that adhere to a set of active rules that deal with the actions on the configuration, healing, protection and optimization of the ontology.

The rest of this paper is structured as follows. In Section 2, we first define the concepts used in our research. Section 3 describes the related works. Section 4 analyzes the main requirements and Section 5 describes the project for an autonomic system in the context of domain ontologies. In Section 6, we describe a case study of ontologies evolution and management. Section 7 concludes the paper.

II. CONCEPTS AND RELATED TECHNOLOGIES

E.R. Sacramento et al. [13] define and relate ontology, knowledge base and data sources such as used within this research:

- (a) An ontology is a pair $O=(V,S)$ such that
 - (i) V is a finite alphabet, the vocabulary of O , whose atomic concepts and atomic roles are called the classes and properties of O , respectively, and
 - (ii) S is a finite set of inclusions in V , the constraints of O . The constraints (or Axioms) capture the semantics of the terms.
- (b) A knowledge base is a triple $KB=(V,S,A)$ such that
 - (i) (V,S) is an ontology, and
 - (ii) A is a finite set of assertions in V .
- (c) A data source is a pair $DS=(V,A)$ such that
 - (i) V is a finite alphabet, and
 - (ii) A is a finite set of assertions in V .

Similarly, RDF (Resource Description Framework) [17] is a triple subject-property-object, usually described as $P(S, O)$, where a given subject S has a property P that assumes the value O . E.R. Sacramento et al. [13] define Linked Data as a set of best practices for publishing and connecting structured data on the Web [18]. From the user's perspective, the main goal of Linked Data is the provision of integrated access to data from a wide range of distributed and heterogeneous data sources [19].

According to F.C. Albuquerque et al. [1], a reactive application advocates a paradigm shifting from human-centered to human-supervised computation. In their perspective, a proactive system must: (1) have a direct connection with the real world; (2) be able to execute actions in response to external stimuli; (3) execute actions faster than the human response. In other words, a system with proactive behavior must detect symptoms and must be able to handle such situations without human supervision. For this, R. Calhau et al. [6] apply technical and administrative procedures for developing, producing and supporting the life cycle of a product to control product evolution.

III. RELATED WORKS

According to M.C.S. Figueroa et al. [16], methodology for building ontologies mainly includes guidelines for single ontology construction ranging from ontology specification to ontology implementation, mainly targeted to ontology researchers. While NeOn Methodology [20], suggests pathways and activities for a variety of scenarios, METHONTOLOGY [21], On-To-Knowledge [22], and DILIGENT [23] were up to 2009 as the most referred methodologies for building ontologies and prescribe a rigid workflow.

A. Knowledge Management

S. Slimani et al. [15] describe distributed ontology evolution approaches, showing that ontology change management increases, especially if services ontologies are heterogeneous (like Semantic Service Architecture - SSOA).

Their approach takes into account some constraints: (1) Ontologies must be autonomous and communicate with each other in reactive way. (2) Not all changes should be managed: there are some changes, which are not interesting to manage because they do not affect the interconnection between ontologies. (3) Ontology should receive just changes that affect the mapping with its interconnected ontologies. (4) One should have a good understanding of changes, that will be translated according to the mapping between ontologies. (5) Mapping is a charred resource between two ontologies and should be managed in parallel since to access to this resource can generate conflicts. The approach is based on a distributed algorithm presenting agent behaviors': (1) initiator ontology agent (IOA) and (2) Dependant Ontology Agent DOA.

B. Proactive System

F.C. Albuquerque et al. [1] discuss basic requirements for proactive real-time monitoring applications. They propose an architecture to deploy applications that monitor moving objects, explore trajectory semantics and are sensitive to environment dynamics. This architecture uses workflows and it features a module to extract data, which helps detect changes on road conditions.

IV. CHOP: DOMAINS ONTOLOGIES WITH AUTONOMIC CHARACTERISTICS

According to M.R. Nami et al. [11], autonomic elements are the heart of an autonomic system. The autonomic elements have a control loop that regulates the workflow of different sub-components of an autonomic system.

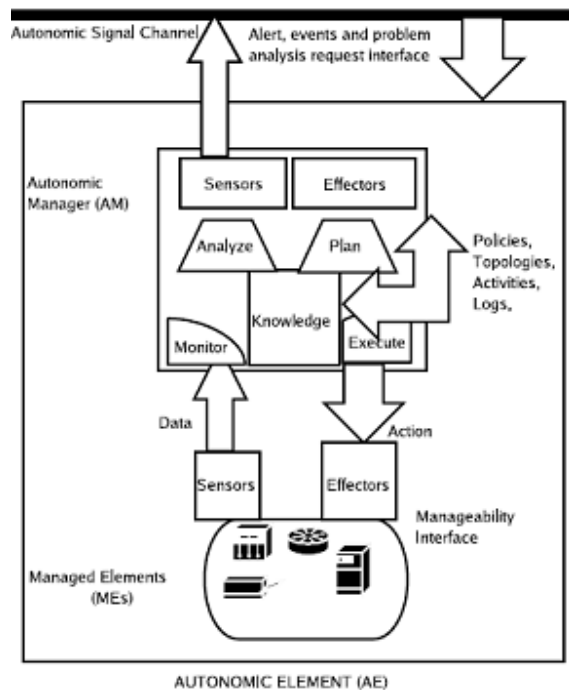


Figure 1. Autonomic Architecture [11]

Figure 1 represents the autonomic project developed by M.R. Nami et al. [11] that provides autonomic features for the domain ontologies, which defines the following components:

- Autonomic Element (AE): basic block of autonomic system, where its interaction with other AE produces the self-managing behavior;
- Managed Element (ME): any resource (in our case, the ontologies) that has its behavior controlled by the environment;
- Autonomic Manager (AM): component that monitors and controls the ME.

Within our approach, each Managed Element (ME) is autonomic domain ontology and the Autonomic Manager (AM) is the meta-knowledge describing the workflow with its specified active rules that are the policies defined by the ontologist.

The autonomic computing paradigm uses actions and predefined rules to lead a new ontology configuration, where the autonomic characteristics of configuration (C) act on ontology for normalization, mapping and alignment to other existing ontologies. Besides, healing actions treat undesirable scenarios during the autonomic evolution. Likewise, ontology instances require actions and rules to address issues related to protection. Also, ontology querying indicates the need for treatment optimization in scenarios that compromise the service quality offered by the ontology.

Then, we define autonomic ontologies as domain ontologies that obey active rules that deal with the special actions on the ontology behavior and their knowledge bases. Accordingly, the actions are related to configuration, healing, protection and optimization of the ontology (Self-CHOP).

V. AUTONOMIC ACTIONS

According to E. Hovy [9], ontologies are better accepted by traditional critics only if at least two conditions are addressed: they have well-founded methodologies for construction and evaluation and prove their usefulness in real applications. Our proposal contemplates the assessment phase by monitoring ontology metrics and applies its contribution within governmental knowledge bases that use the domain ontology as a semantic resource. Our approach makes use of autonomic computing paradigm to achieve accuracy in the evaluation and ontology management such that the ontologist is spared of the procedure details.

Firstly, the monitoring aims to guarantee the ontology quality with evaluation as an activity of their whole life cycle. This goal is addressed in two scenarios: knowledge base and ontology querying.

A. Knowledge Base Scenario (Instances)

The knowledge base uses ontologies and vocabularies that already exist and might have been developed by third parties. It is important to monitor and treat events related to these resources interaction. As the knowledge base has

concept's types represented in a domain ontology, we deal with the following events:

1) *New Meanings*

This event occurs when an ontology sub-graph has a concept referenced by a knowledge base and this concept is modified. In this case, the concept instances need to be revised to ensure the real semantic representation between the instance and the modified concept.

2) *Reuse*

This event occurs when the same instance (certified by the same unique identifier or *owl:sameAs* property) exists on different bases and it is from different concepts. In this case, we can infer there is a semantic relationship between these different concepts of ontologies.

3) *Inconsistency*

This event occurs when a concept is deleted. At this point, it is important to identify the sub-graph in which the concept was, as well as, the mappings / integration that this concept had with other ontologies / knowledge base.

B. Queries Scenarios

Even with the most advanced interfaces for user's interaction, expressing a need for information is a difficult task. There is a semantic distance between the real users needs and what they expressed on the search. The queries performed on ontology provide statistics about its use as a resource semantic related to data quality and needs for ontological management. This scenario includes three events:

1) *Concepts Accessed*

The architecture monitors central ontology concepts to collect data for statistical redistribution of instances. First, SPARQL queries [14] received from client applications are processed to analyse and identify the instances type retrieved from queries through the *rdf type property*. After, the more the concept is quoted, the more it fits in the central ontology concepts group. This means that the concept is quoted when its instances are implicitly mentioned in the query.

2) *Critical Path*

The event occurs identifying the ontology's sub-graph with the largest execution times of queries. From this point, extracting the concepts involved in the SPARQL query [14], class attributes and modifiers used, in our case, order by, projection, distinct, offset and limit (known area of database).

3) *Denial of Service*

Event identified when overload or ineffectiveness access to ontologies. Ontology as a knowledge representation and semantic resource for querying by other systems, must be concerned with the service quality offered and, most importantly, if the service is actually being offered. The two events above address quality while this event verifies availability, keeping the service history offered.

The metric used is the response time of queries to client applications, when they reach the maximum waiting time defined by the ontologist. As shown in the previous event, every query has its runtime recorded and when it achieves an unacceptable level, this event is triggered.

C. Autonomic actions

Autonomic Action is any algorithm developed under autonomic computing paradigm that acts upon domain ontologies in order to generate a new configuration, healing, protection or optimization. The autonomic action is performed after an event is identified.

1) *Balancing Semantic Action*

This action includes or maps a concept to an unbalanced sub-graph through common instances between the sub-graph concepts and concepts from other ontologies. The advantage is to guarantee ontologies mapping, since the common instance ratifies the semantic relation between the concepts involved. Thus, the approach aims to restore the ontology in a coordinated and orderly way to avoid unexpected / unwanted results, maintaining consistency based on metrics already established in the literature.

As the structure taxonomic metrics evaluate the ontology quality structure, the guard expressions use the Width and Depth metrics to identify a sub-graph that reaches a value not desirable by ontologist.

This action treats the problem of sub-graph by mapping concepts with common instances. When the guard expression is triggered, the instances associated with the sub-graph concepts are used as input for the re-design of the unbalanced sub-graph through the following algorithm:

- a) Identification of the sub-graph;
- b) Sub-graph analysis;

c) If the sub-graph has reached a non-acceptable value for the width, then the treatment action will be vertical with the identification of 'NEW concepts' with semantic relation with child classes of the sub-graph (Figure 2);

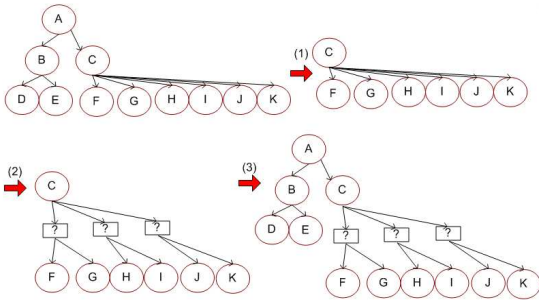


Figure 2. Semantic Vertical Balancing Strategy

d) If the sub-graph has reached a non-acceptable value for depth, then the treatment will use the concept of inclusion on leaf of the sub-graph between a father-class and child-class, expanding the ontology vertically (Figure 3).

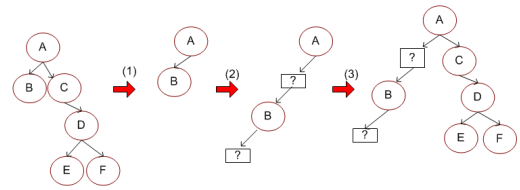


Figure 3. Semantic Horizontal Balancing Strategy

- e) Retrieve the sub-graph instances;
- f) For each instance:
 - g) Check if the resource (identified by rdf:resource) is referenced by other instances, even if in other ontologies;
 - h) Check if the concept type is different between instances identified in the previous step;
 - i) Check the existence of a common concept between the concept type of instances;
 - j) Create a semantic relationship between the concepts identified in the previous step;

2) *Fragmentation Action*

Fragmentation action occurs on the concept being highly referenced by instances and other concepts. This action finds equivalent classes to heal the critical path of ontology. This is possible through (1) equivalent relationship between instances of different ontologies (by *sameIndividuals axiom*) and (2) different instances reference the same resource (by *rdf:resource property*).

Thus, the approach heals the critical path with inclusion of existing concepts to avoid overload in query performance. The increase and enrichment of knowledge bases are the source for healing of ontologies referenced by them.

Fragmentation action has guard expressions associated with the following metrics: Importance of Class (instances distribution), Wealth of classes (instances distribution between classes) and Cost Based Evaluation (CBE - to measure performance).

When any of the ontology metrics reaches a value that triggers at least one guard expressions, Fragmentation action is performed:

- a) Identification of the concept;
- b) Instances selection of the concept identified on step (a);
- c) For each instance:
 - d) Identify instances (1) that reference the same resource (by rdf:resource tag) or (2) has the sameIndividuals axiom with an instance that has a different type (rdf type property) (according to Figure 4);
 - e) Check if the instance type identified (by rdf: type property) is different from the selected instance;
 - f) Inclusion of the equivalentClass axiom between the concept identified in step (a) and the concept of the instance identified in the step (d);

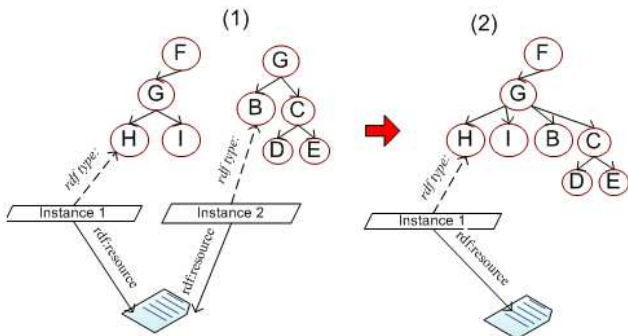


Figure 4. In (2) a relationship was created between the concepts H and B due to their particular instances to represent the same feature (1)

Figure 4 shows the case where two individuals of different types represent the same resource. In this case, they are considered identical individuals according to our approach.

VI. CASE STUDY

Domain ontologies are Managed Elements (ME), in which the metrics are monitored in the form of Jess production rules (Figure 5) [10], implementing the workflow transition conditions of each ontology management. Moreover, the ontology is registered as a web service, whose desirable values are filled by the ontologist. At this moment begins the self-management.

```
(defrule INSTANCES_OVERLOAD
  ?o <- (OntologyVO (e_av_cnt_c ?e_av_cnt_c) (e_max_cnt_c ?e_max_cnt_c > (* e_av_cnt_c 3)))
  =>
  (save-problem "PROTECTION" "INSTANCES_OVERLOAD")
  (printout t "Self-protection -> e_av_cnt_c: " ?e_av_cnt_c
    " e_max_cnt_c: " ?e_max_cnt_c crlf)
  )
```

Figure 5. Jess production rules

This section presents a brief case study to demonstrate how the architecture works. The case came from Secretary of Federal Treasury responsible for control and oversight of federal spending in accordance with the legislation, case law and administrative acts. The Knowledge Organizational System - SIOP-LEGIS [3] is a project that aims to provide knowledge management for legislative domain through changeable representation, which deals with trends in the law.

Nowadays, according to S.N. Brandao et al.[25], the system represents the knowledge from Official Gazette, allowing to answer questions that were required during the monitoring, auditing and oversight. This knowledge base is linked to other Brazilian Open Data and represents one more effort in Open Government Partnership [12] to reflect the country's commitment to strengthen the transparency of government actions to prevent and combat corruption [4].

The government knowledge base is constructed from Brazilian laws, the events that surround them and authorities responsible for these. The knowledge base is built on RDF language through the Brazilian Official Gazette, which is the access for official information.

The initial study was done with two different ontologies: Social Security and Legislative domain [7]. The first ontology has an overload concept called 'Law' (Figure 6), while, Legislative ontology used by Chamber of Deputies treat specifically the federal law documents with other 14 concepts (Figure 7). Given the need to deal with the overloaded concept 'Law' on Social Security ontology, which reference the same official documents of Legislative ontology, which more specific types allowing to treat the overload 'Law' concept in the Social Security ontology. The Fragmentation action maps the Legislative ontology that provides new concepts as view that is a faithful and attentive to changes in the ontology provider. Note that there is a copy or a mapping of new concepts.

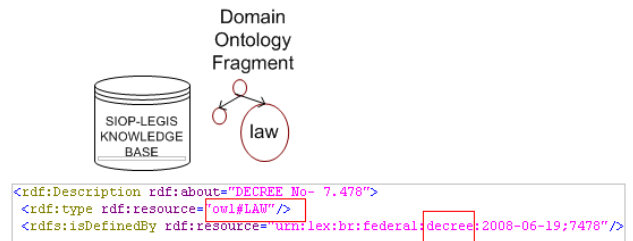


Figure 6. An knowledge base instance associated to concept 'Law' trough 'rdf: type' despite being a decree

A second study was conducted with the inclusion of the concept 'Law' in the Legislative ontology. In this case, the Balancing Semantic action could be performed with the inclusion of hierarchical relationship (containing 14 concepts) in the Social Security ontology. In this case, even the fragmentation action being thrown to the inclusion of the axiom equivalentclass, there was the possibility of performing semantic balancing action.

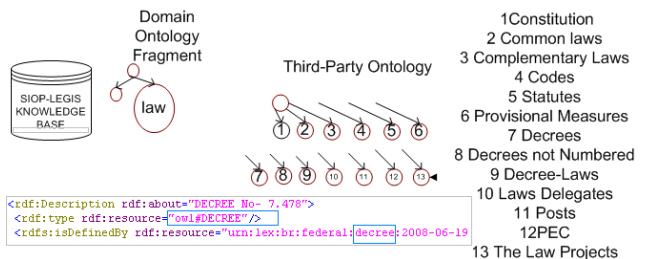


Figure 7. After Fragmentation action, the instance of the Figure 6 now associated to concept 'Decree' trough 'rdf: type', what really represents the 'rdfs: isDefinedBy' property

This suggests as future work to create a workflow with actions containing priority, treatment of infinite loop and treatment of undesirable behavior and scenarios.

VII. CONCLUSION

The SIOP-LEGIS [3] project is part of Federal Budget Secretary initiative to provide information to society. The project allows the development of tools to read the data provided by own government, since information is linked and interoperable in our open knowledge base.

In general terms, the methodology presents as main advantages: (i) the semi-automation process of domain ontologies management, minimizing human intervention, (ii) ontology monitor through ontology metrics, since the knowledge base is constantly updated and consequently under failures. Therefore, if an known symptoms occurs, the proposal allows a new configuration, healing, optimization or protection. The existing ontologies in the SIOP-LEGIS [3] repository are currently monitored for symptoms and it reached the predictive level 3 (according to Figure 8) with the ability to monitor symptoms and recommend actions as a form of domain ontologies' evolution. We envision the next autonomic level, where the architecture will be able to take actions regarding Service Level Agreement (SLA) and improve the human/system interaction.

	Basic Level 1	Managed Level 2	Predictive Level 3	Adaptive Level 4
Characteristics	Multiple sources of system generated data	Consolidation of data and actions through management tools	System monitors, correlates and recommends actions	System monitors, correlates and takes action
Skills	Requires extensive, highly skilled IT staff	IT staff analyzes and takes action	IT staff approves and initiates actions	IT staff manages performance against SLAs
Benefits		Greater system awareness Improved productivity	Reduced dependency on deep skills Faster/better decision making	Balanced human/system interaction IT agility and resiliency
Manual				

Figure 8. Autonomic level define by [24]

We also aim to find inconsistencies in the knowledge base and indicate them for the domain specialist. With this, self-management characteristics will be added to the knowledge base.

ACKNOWLEDGMENT

We thank CAPES, CNPq and Brazilian Department of Treasury for supporting this work.

REFERENCES

[1] F.C. Albuquerque, I. Barbosa, M.A. Casanova, M.T.M. Carvalho, and J.A.F. Macedo, "Proactive Monitoring of Moving Objects", ICEIS, Wroclaw, Poland, 2012, pp. 191-194.

[2] K. Breitman, P. Salas, M.A. Casanova, D. Saraiva, and M. Chaves, "Open government data in Brazil", IEEE Intelligent Systems, 27(3), Rio de Janeiro, Brazil, 2012, doi: {10.1109/MIS.2012.25}, pp.45 -49.

[3] S.N. Brandao, T.S. Silva, S.A. Rodrigues, L.S. Araujo, and J.M. Souza, "SIOP-LEGIS: Thesaurus for Selection and Management of Brazilian Treasury Domain", International Conference on Knowledge Management and Information Sharing, Paris, France, 2011, pp. 195-200.

[4] S.N. Brandao, T.S. Silva, S.A. Rodrigues, L.S. Araujo, and J.M. Souza, "Knowledge Representation of Brazilian Official Gazettes for Chronological Recovery of Laws", 10° IADIS International Conference WWW/INTERNET, Rio de Janeiro, Brazil, 2011, pp. 149-157.

[5] F. Bugiotti, F. Goasdoué, Z. Kaoudi, and I. Manolescu, "RDF data management in the Amazon cloud", Proceedings of the 2012 Joint EDBT/ICDT Workshops. EDBT-ICDT '12. New York, USA: ACM, pp. 61-72.

[6] R. Calhau and R.A. Falbo, "A Configuration Management Task Ontology for Semantic Integration". Proceedings of the 27th Annual ACM Symposium on Applied Computing. SAC '12. New York, USA, 2012, pp. 348-353.

[7] BCD, Legislação. Available at: <http://www4.planalto.gov.br/legislacao> [retrieved: February, 2013].

[8] A.G. Silva, O.A. Corcho, H. Alani, and A.G. Pérez, "Review of the State of the Art: Discovering and Associating Semantics to Tags in Folksonomies", Knowl. Eng. Rev., 27(1), 2012, doi: 10.1017/S02698891100018X, pp.57-85.

[9] E. Hovy, "Methodologies for the Reliable Construction of Ontological Knowledge", Eds. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2005, pp. 91-106.

[10] Jess, "Jess: Rule Engine and Scripting Environment", Available at: <http://www.jessrules.com/jess/docs/index.shtml> . [retrieved: February, 2013]

[11] M.R. Nami and K. Bertels, "A Survey of Autonomic Computing Systems", In Proceedings of the Third International Conference on Autonomic and Autonomous Systems. ICAS 2007. Washington, DC, USA: IEEE Computer Society, pp. 1-26.

[12] OGP, Open Government Partnership, Available at: <http://www.opengovpartnership.org/countries/brazil>[retrieved: February, 2013].

[13] E.R. Sacramento, M.A. Casanova, K.K. Breitman, A.L Furtado, and V.M.P. Vidal, "Dealing with Inconsistencies in Linked Data Mashups", Proceedings of the 16th International Database Engineering & Applications Symposium. IDEAS '2012. New York, USA: ACM, pp. 175-180.

[14] E. Sirin and B. Parsia, "SPARQL-DL: SPARQL Query for OWL-DL", 3rd OWL Experiences and Directions Workshop (OWLED-2007), Innsbruck, Austria, pp. 16-21.

[15] S. Slimani, S. Baïna, and K. Baïna, "A Framework for Ontology Evolution Management in SSOA-Based Systems", Proceedings of the 2011 IEEE International Conference on Web Services (ICWS). Washington, DC, USA, pp. 724-725.

[16] M.C.S. Figueroa, R.G. Castro, B.V. Terrazas, and A.G. Pérez, "Essentials In Ontology Engineering: Methodologies, Languages, and Tools", Proceedings of the 2nd Workshop Organized by the EEB Data Models Community, Sophia Antipolis, France, 2011, pp.9-21.

[17] F. Manola and E. Miller, 2004. "RDF Primer". Available at: <http://www.w3.org/TR/2004/REC-rdf-primer-20040210/>. [retrieved: February, 2013].

[18] C. Bizer, R. Cyganiak, and T. Heath, "How to Publish Linked Data on the Web", Available at: <http://www4.wiwiw.de/bizer/pub/LinkedDataTutorial/>. [retrieved: February, 2013].

- [19] C. Bizer, T Heath, and T.B. Lee, "Linked Data – The Story So Far", *International Journal on Semantic Web and Information Systems*, 5(3), 2009, pp.1–22, doi:10.4018/jswis.2009081901.
- [20] A.G. Perez and M.C.S. Figueroa, "Scenarios for building ontology networks within the NeOn methodology", *Proceedings of the fifth international conference on Knowledge capture. K-CAP '2009*. New York, USA, pp. 183–184.
- [21] M.F. Lopez, A.G. Perez and, N. Juristo, "METHONTOLOGY: from Ontological Art towards Ontological Engineering", *Proceedings of the AAAI 1997 Spring Symposium*. Stanford, USA, pp. 33–40.
- [22] C. Fluit, H.T. Horst, J.V.D. Meer, C.D. Fensel, and E. Ab, "On-To-Knowledge: Content-driven Knowledge management Tools through Evolving Ontologies On-To-Knowledge Consortium", 2000, Available at: <http://www.ontotext.com/research/otk>. [retrieved: February, 2013]
- [23] H.S. Pinto, S. Staab, C. Tempich, Y. Sure, "Distributed Engineering of Ontologies (DILIGENT)", in: Staab, S., Stuckenschmidt, H. (Eds.), *Semantic Web and Peer-to-Peer*. Springer Berlin Heidelberg, 2006, ISBN 978-3-540-28347-8, pp. 303–322.
- [24] IBM, IBM Research | Almaden Research Center | Computer Science [WWW Document]. Available at: <http://www.almaden.ibm.com/cs/projects/autonomic/>. [retrieved: February, 2013]
- [25] S.N. Brandao, S.A. Rodrigues, T.S. Silva, L.S. Araujo, and J.M. Souza, "Open Government Knowledge Base". *The Seventh International Conference on Digital Society*, Nice, France, ISBN: 978-1-61208-249-3, pp. 13-19.