

# Ontology-based Policy Refinement Using SWRL Rules for Management Information Definitions in OWL

Antonio Guerrero<sup>1</sup>, Víctor A. Villagrà<sup>1</sup>, Jorge E. López de Vergara<sup>2</sup>,  
Alfonso Sánchez-Macián<sup>1</sup>, Julio Berrocal<sup>1</sup>

<sup>1</sup>Dpto. de Ingeniería de Sistemas Telemáticos, Universidad Politécnica de Madrid.

<sup>2</sup>Dpto. de Ingeniería Informática, Universidad Autónoma de Madrid.

antonio.guerrerocasteleiro@telefonica.es, villagra@dit.upm.es,  
jorge.lopez\_vergara@uam.es, aasmp@dit.upm.es, berrocal@dit.upm.es

**Abstract.** The goal of ontology-based management is to improve the manageability of network resources through the application of formal ontologies. Prior research work has studied their application to represent the management information definitions, the mapping and merging processes to obtain a semantic integration of those definitions, and the representation of behaviour and policy definitions. Using ontologies allows the additional advantage of integrating, in the same semantic manager, business and service level ontologies with the network management ontology, in a framework for automated management. This integration allows for policy refinement and interoperation between high level policies and low level policies.

## 1 Introduction

Network administrators need more intelligent management systems that hide the underlying complexity of the network, allowing them to manage the infrastructure at an abstract level, focusing on what the expected behaviour should be, instead of on how to specifically achieve it. In this context, Policy-Based Network Management (PBNM) [1] proposes the use of policies to administer, manage, and control network resources, in such a way that they can be centrally defined and applied to large numbers of devices uniformly. In [2] Strassner depicts the “Policy Continuum”, where policies can be defined at several layers with different levels of abstraction. This layering should allow network administrators to manage their systems at a higher level of abstraction than the mere technology configuration, therefore hiding the complexity from the administrator.

This paper presents a generic ontology-based approach to bind the behaviour specified at higher levels of abstraction to the expected behaviour at the network level, in such a way that an ontology reasoner can dynamically perform this High Level (HL) to Low Level (LL) refinement process at run-time. The next sections describe the semantic management framework within which this work is presented, and how the policy refinement process can be accomplished. Then, a simplified policy refinement example will be used to illustrate the mechanisms being presented.

## 2 Semantic Management

The ontology-based semantic management framework [3] proposes a single manager working with a unique information model, which integrates all the different definitions of the managed resources, taking into account the semantic aspects of those definitions (i.e. their meaning). In [4] it is shown how to merge and map management definitions from different domains into a Common Management Ontology. A semantic manager could then apply generic policies for all the network resources, independently of the management models in which they are originally defined.

OWL [5], the Web Ontology Language, is proposed as the language for policy and management definitions, since it contains all the necessary constructors to formally describe most of the information management definitions [6]. This semantic approach allows the integration, in the same unified management information model, of the behaviour definitions and policies for the managed resources, which can also be expressed in OWL using the SWRL language [7], as shown in [8].

A comparison of other Semantic Web policy languages is presented in [9], from a PBM point of view, stating that the possibility to represent entities and behaviours at multiple levels of abstraction makes ontology frameworks adequate to deal with several kinds of contexts at different level of specifications. The advantages of semantic policy frameworks are analysed in [10], stating that semantic approaches using RDF/OWL as standards for policy representation enable runtime extensibility and adaptability of the system, as well as the ability to work with policies relating to entities described at different levels of abstraction. The use of ontology-based PBM to provide dynamically adaptive network management solutions is also proposed in [11].

## 3 Ontology-based Policy Refinement

The proposed semantic manager can therefore work with ontologies and policies defined at different abstraction levels, which allows facing one classical problem in the PBNM area: policy refinement. Policy refinement is concerned with the process of mapping a set of HL policies to a set of LL policies [1]. Most approaches, such as in [12] and [13], attempt this decomposition of HL policies relevant to a composite system into a set of policies that are executed in its constituent parts to implement the behaviour intended by the overall higher level policies. In contrast to refinement, [14] introduces the concept of *Policy Interoperability*. While refinement is concerned with the unidirectional mapping  $HL \rightarrow LL$ , interoperability is the bi-directional mapping  $HL \leftrightarrow LL$ . The purpose of this interoperability mapping is to allow LL policies at runtime to dynamically refer to their HL parents as the need arises.

The approach being presented can be summarized in the following three steps:

- 1) First, we have OWL ontologies both for the upper domain and the lower domain. Definitions of HL and LL policies could be included, as shown in [8].
- 2) Relating HL ontologies to LL ontologies can be achieved in the OWL ontology language by means of meaningful OWL relationships between HL and LL classes. These will be referred to as *Interoperability Relationships*.

3) Finally, translation SWRL rules can be used to make the semantic manager able to derive the necessary information translations in order to: 1) populate the higher level with data useful for this layer, hiding the complexity of the data at the lower level, and 2) add data to the lower level based on the information from the upper layer. The following is a generic example of 1) in SWRL logic syntax:

```
LLproperty1(?LLclassYindividual) ^
InteroperabilityRelationship1(?LLclassYindividual,
?HLclassAindividual) => HLproperty1(?HLclassAindividual)
```

More complex conditions combining classes from both layers could be expressed.

With the model and the SWRL rules programmed, the manager will be able to perform this bi-directional information mapping at run time, in such a way that changes in the HL data affect the LL data and vice versa. This way, policies defined at the HL layer can govern policies at the LL layer, achieving dynamic policy interoperability.

## 4 Proof of Concept Use Case: Backup for DSL Premium Lines

The scenario for the use case is an Internet Network Access service offered by a Service Provider for thousands of users. The service is supported by an IP backbone and an ATM access network. Each subscriber's modem-router is connected to its corresponding Broadband Remote Access Server (BRAS), through an ATM circuit, that runs over the telephone line and enters the ATM network through the DSLAM.

Since DSL circuits run over the telephone lines (POTS or ISDN), which can also be used for dial-up Internet access, the Service Provider wants to offer a backup service for some of his DSL subscribers. For this matter he has installed a Remote Access Server (RAS) that will accept incoming telephone connections from the subscribers' modem-routers. However, he only wants subscribers with a PREMIUM contract to make use of this backup network infrastructure, so he has installed a RADIUS server in order to authorize or deny access through the RAS.

The ontology model for this scenario, including the SWRL rules, has been defined in the namespace <http://www.dit.upm.es/jlopez/geseman/policy.owl#>, and a simulation has been implemented in Bossam [15], a Rule/OWL reasoner.

### 4.1 HL and LL Ontologies in OWL

At the network level we would have an Integrated Network Management Ontology, such as the Common Management Ontology proposed in [3]. For simplicity purposes, we have restricted the information used in this example to what is strictly needed. The RDF graph [16] in Fig. 1 shows the chosen HL and LL classes and properties.

In the HL specifications, we can also define the HL policies, as shown in [7]: If a subscriber has a PREMIUM service contract, and his DSL service is down, then he is allowed to use the telephone backup access service. For this HL policy, we used the following SWRL rule:

```
ServiceContract(?subscriber, ?contract) ^
swrlb:equal(?contract, "PREMIUM") ^
DSLServiceStatus(?subscriber, ?status) ^
```

```

swrlb:equal(?status, "NOT OK")
=> BackupAllowedSubscriber(?subscriber, "YES")

```

Other two HL rules were used in the simulation to set the value of the *BackupAllowedSubscriber* to “NO” when appropriate.

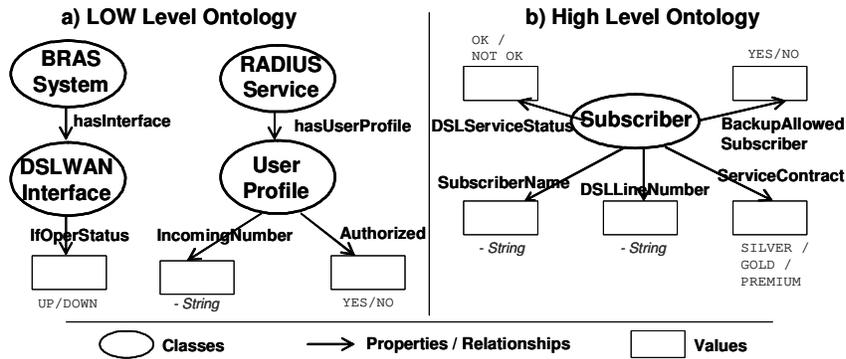


Fig. 1. RDF graph representation of High and Low Level ontologies

## 4.2 Interoperability Relationships

New relations in order to bind the HL and LL representations are required:

- A *Subscriber*  $\Leftrightarrow$  *DSLWANInterface* binding: the *hasWANInterface* relationship. For this binding, all interface data for all users will come from the provisioning system in this example, so it will be available in the semantic manager’s database of facts.
- A *RADIUS UserProfile*  $\Leftrightarrow$  *Subscriber* binding: the *relatesToSubscriber* relationship. This binding will take place at run time. It will be inferred by the semantic management system whenever an incoming call enters the RAS.

## 4.3 Translation SWRL Rules

Rules 1 and 2: *DSLServiceStatus* should be “OK” if the *ifOperStatus* of the subscriber’s WAN interface is “UP”, and “NOT OK” otherwise. This is the representation of the first rule in SWRL logic syntax:

```

hasWANInterface(?subscriber, ?wanif) ^ ifOperStatus(?wanif,
?operstatus) ^ swrlb:equal(?operstatus, "UP")
=> DSLServiceStatus(?subscriber, "OK")

```

This is an example of setting HL information from LL information.

Rule 3: A *UserProfile* relates to a specific *Subscriber* if the *IncomingNumber* of the profile matches the subscriber’s *DSLLineNumber*. This rule is an example of relating HL information to LL information. Expressed in SWRL logic syntax:

```

swrlb:equal(IncomingNumber(?userprofile),
DSLLineNumber(?subscriber))
=> relatesToSubscriber(?userprofile, ?subscriber)

```

Rules 4 and 5 (SWRL not shown): The value for the *Authorized* property of *User-Profile* should be “YES” if subscriber is allowed to use the backup service, and “NO” otherwise. These rules are examples of setting LL information from HL information.

#### 4.4 Overview of the Backup for DSL Lines Use Case

The semantic manager holds the database of facts, with all of the facts and axioms of the ontology model, including HL Policies, HL facts (subscribers’ service contracts and DSL line numbers), LL facts (operational status for all DSL WAN Interfaces), HL to LL bindings, and Translation SWRL Rules. With all this information, the ontology manager – having an inference engine – is able to act as a PDP (*Policy Decision Point*), answering the query on whether an incoming call, identified by its telephone number, should be allowed to access the network through the RAS. For a generic implementation, the RDF query in RDQL (RDF Query Language), would be

```
SELECT ?auth WHERE (dit:<subscriber> dit:Authorized ?auth)
USING dit FOR <http://www.dit.upm.es/jlopez/geseman/policy.owl#>
```

which returns a value of “YES” or “NO” for the *auth* variable depending on the instance of *<subscriber>* entered and the simulation data. The RADIUS service would finally notify the allowance or denial of the incoming call to the RAS (*Policy Enforcement Point*).

## 5 Conclusion and Further Work

A general purpose ontology reasoner can work with HL and LL ontologies, being completely independent of their abstraction levels, and allowing for interoperability, whereas some other expert systems and policy languages usually have a more specific orientation. The ontology reasoner, having an inference engine, will be able to understand the model, work with the network data, and enforce the expected behaviour, therefore becoming an implementation of a management system.

The present work presents an approach on how ontology representation could be used for dynamic policy interoperability between HL business rules and LL network policies, while maintaining the separation of concepts of HL and LL information.

In the use case presented as a proof of concept, a change at the network level, such as when a DSL connection goes down, affects the expected behaviour of the Remote Access Server, also at the network level. This LL behaviour is governed by authorization HL business policies, in a dynamic and bi-directional refinement cycle.

Unlike other methods such as those presented in [12] and [13], this approach does not attempt to directly translate policies from the upper level into a set of policies or configuration commands at the lower level. Also, it is not restricted to service oriented architectures as in [11], in which behaviour defined for a certain HL service affects the behaviour of the LL services in which the former is decomposed. It also presents the semantic web advantages for network management about working with data distributed over different systems with heterogeneous RDF-based semantics [11].

The possibility to represent and work with meaningful and reusable interrelations between different abstraction levels could be useful for other specific purposes of ontology-based network management. Areas for further work include its application to event enrichment, service composition, and event correlation.

## References

1. A. Westerinen, J. Schnizlein, J. Strassner, M. Scherling, B. Quinn, S. Herzog, A. Huynh, M. Carlson, J. Perry, S. Waldbusser: Terminology for Policy-Based Management. IETF Request For Comments 3198 (2001)
2. J. Strassner: Policy-Based Network Management – Solutions for the Next Generation. Morgan Kaufman (2003)
3. J. E. L pez de Vergara, V. A. Villagr , J. I. Asensio, J. Berrocal: Ontologies: Giving Semantics to Network Management Models. IEEE Network, Vol. 17, Issue 3 (2003) 15-21.
4. J. E. L pez de Vergara, V. A. Villagr , J. Berrocal: Benefits of Using Ontologies in the Management of High Speed Networks. Proc. 7th IEEE Intl. Conf. on High Speed Networks and Multimedia Communications (HSNMC'04), LNCS 3079, Toulouse, France (June 2004) 1007-1018.
5. M. K. Smith, C. Welty, D. L. McGuinness: OWL Web Ontology Language Guide. W3C Recommendation, (February 2004)
6. J. E. L pez de Vergara, V. A. Villagr , J. Berrocal: Applying the Web Ontology Language to management information definitions. IEEE Communications Magazine, Vol. 42, Issue 7 (2004) 68-74.
7. I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosz, M. Dean: SWRL: A Semantic Web Rule Language Combining OWL and RuleML. W3C Member Submission (21 May 2004)
8. A. Guerrero, V. Villagr , J. E. L pez de Vergara, J. Berrocal: Ontology-Based Integration of Management Behaviour and Information Definitions Using SWRL and OWL. Proc. 16th IFIP/IEEE Intl. Workshop on Distributed Systems: Operation and Management (DSOM'05), Barcelona, Spain, LNCS 3775 (October 2005) 12-23.
9. G. Tonti, J. M. Bradshaw, R. Jeffers, R. Montanari, N. Suri, A. Uszok: Semantic Web Languages for Policy Representation and Reasoning: A Comparison of KAoS, Rei, and Ponder. Proc. 2nd Intl. Semantic Web Conference, Sanibel Island, Florida, USA, LNCS 2870 (October 2003) 419-437.
10. F. J. Garc a, G. Mart nez, J. A. Bot a, A. F. G mez Skarmeta: Representing Security Policies in Web Information Systems. Proc. Policy Management for the Web (PM4W), 14th Intl. WWW Conference, Chiba, Japan (May 2005)
11. D. Lewis, K. Feeney, K. Carey, T. Tiropanis, S. Courtenage: Semantic-based Policy Engineering for Autonomic Systems. Proc. 1st IFIP Intl. Workshop on Autonomic Communication (WAC 2004), Berlin (October 2004)
12. R. Darimont, A. van Lamsweerde: Formal Refinement Patterns for Goal-Driven Requirements Elaboration. Proc. 4th ACM Symposium on the Foundations of Software Eng. (FSE4) (1996) 179-190.
13. A. Bandara, E. Lupu, J. Moffet, A. Russo: A Goal-based Approach to Policy Refinement. Proc. 5th IEEE Workshop on Policies for Distributed Systems and Networks (Policy 2004)
14. S. Magrath, R. Braun, F. Cuervo: Policy Interoperability and Network Autonomics. Proc. 1st IFIP Intl. Workshop on Autonomic Communication (WAC 2004), Berlin (October 2004)
15. Minsu Jang, Joo-chan Sohn: Bossam: an extended rule engine for the web. Proc. 3rd RuleML Intl. Workshop (RuleML 2004), LNCS Vol. 3323, (November 2004) 128-138.
16. F. Manola, E. Miller: RDF Primer, W3C Recommendation (10 February 2004)