Modality Conflicts in Semantics-Aware Access Control

E. Damiani
DTI - Università di Milano
26013 Crema - Italy
damiani@dti.unimi.it

S.De Capitani di Vimercati
DTI - Università di Milano
26013 Crema - Italy
decapita@dti.unimi.it

C. Fugazza
DTI - Università di Milano
26013 Crema - Italy
fugazza@dti.unimi.it

P. Samarati
DTI - Università di Milano
26013 Crema - Italy
samarati@dti.unimi.it

ABSTRACT

Security is a crucial concern for commercial and mission critical applications in Web-based environments. Semantic Web-style context descriptions aim at supporting widespread distribution of resources and cooperation of autonomous agents on the Web in a secure way. In this paper, context information associated with Access Control (AC) management policies is defined according to basic operators that can be represented using the *Web Ontology Language* (OWL). The same primitives are used, in the specification of authorizations, to compose *domain scope expressions*. Standard inference procedures of *Description Logics* (DL) can then be used to check the consistency of context information referred to by policy conditions and, more interestingly, to pre-process context information for grounding policy propagation and enabling conflict resolution. This work aims at extending the notion of *modality conflict* in the evaluation of AC policies to take into account semantic Web-style, ontology-based definitions of the entities involved.

Categories and Subject Descriptors

H.3.5 [**Information Systems**]: INFORMATION STORAGE AND RETRIEVAL; On-line Information Services

General Terms

Security

Keywords

Access Control, modality conflict

ICWE'06, July 11-14, 2006, Palo Alto, California, USA.
ACM 1-59593-352-2/06/0007.

1. INTRODUCTION

Security is a crucial concern for commercial and mission critical applications in Web-based environments. Recently, a number of advanced models and languages have been proposed for protecting Web resources and services, specifying and enforcing policies and access control constraints based on semantic Web-style context descriptions [5] aimed at supporting widespread distribution of resources and cooperation of autonomous agents on the Web in a secure way. Ontologies, rule languages and semantic Web reasoning are important ingredients of this infrastructure to enable distributed peers to negotiate access to distributed resources and services.

In this paper, context information associated with Access Control (AC) management policies is defined according to basic operators (namely, subsumption, union, intersection, and complement) that can be represented using the *Web Ontology Language* (OWL) [4]. The same primitives are used, in the specification of authorizations, to compose *domain scope expressions*. By doing this, authorizations can be directly mapped with branches of the hierarchical representation of context information and, under the assumption of a closed-world reasoning, it is possible to regulate access to resources with only positive authorizations. Moreover, by adding further DL constructors, from simple concept disjointness (i.e., the constraint requiring two concepts having no instances in common) to user-defined properties, it is possible to model more complex features, such as separation of duties. Standard inference procedures of *Description Logics* (DL) [6] can then be used to check the consistency of context information referred to by policy conditions and, more interestingly, to pre-process context information for grounding policy propagation and enabling conflict resolution.

This work aims at extending the notion of *modality conflict* in the evaluation of AC policies to take into account semantic Web-style, ontology-based definitions of the entities involved. Authorizations are propagated along partially ordered structures obtained by classifying context descriptions. Modality conflicts are then resolved on the basis of the *specificity principle*, revising the notion of *domain nesting precedence* [19] to compare concepts referenced by colliding policies. We show that *semantic similarity* measures may resolve the conflicts not mendable with the aforementioned

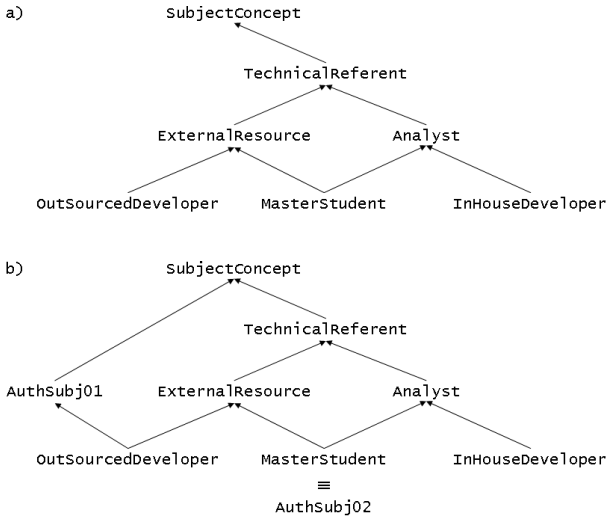

Figure 1: a) Context information for subjects; b) domain scope expressions represented as context concepts.

criterion. Finally, we describe a category of modality conflicts, arising from multiple type relationships of individuals, that cannot be detected simply by looking at the structure of context information.

The paper is organized as follows. In Sec. 2 we investigate the expressiveness achievable with the adoption of a semantics-aware context description and introduce the authorization model that will be used throughout the paper. The standard inference procedures of DL that are employed for context management are presented. Sec. 3 explains the advantages of our model over traditional AC context descriptions and draws a comparison with the RBAC model [25, 26]. In Sec. 3.2 we consider the two possible semantics for interpreting the knowledge base constituted by context description and domain scope expressions. In Sec. 4 we introduce policy propagation and carry out modality conflict detection, distinguishing between *terminological* conflicts, arising from the structure of context information, and *extensional* conflicts pertaining the multiple inheritance of authorizations by individuals. Sec. 5 draws the conclusions and introduces the possible extensions to the basic model.

2. MODELING CONTEXT INFORMATION

A basic motivation for using ontology-based representations to ground AC management policies is that administrative and business processes within an organization are increasingly adopting this kind of fine-grained descriptions of the entities involved to accomplish their tasks. Re-using this information is therefore advisable, as organization-wide descriptions are often normative; also, it will save time in the set-up of the AC management system and provides a categorization that has already been tested and tuned. Translating these descriptions into traditional AC structures e.g., RBAC *domains* (see Sec. 3 for a comparison) would be typically lossy and error-prone. Instead, using semantic-Web languages [3, 4] for describing context information allows to directly import these descriptions into the AC infrastructure. Moreover, when using OWL sub-languages such as OWL Lite and OWL DL, well-known results of Description Logics [6] can be applied to check the consistency of policies and derive implicit information.

In our model, context information is stored as a OWL ontology, whose root concept DomainConcept is sub-classed by concepts representing "state of affairs" that are referred to by AC authorizations (typically, *subjects*, *actions*, and *objects*). Each of these can be handled separatedly w.r.t. policy propagation and then compared for the purpose of conflict detection [20]. In the following example, we consider:

- SubjectConcepts representing the categorization of subjects that will issue access requests to the system.
- ObjectConcepts modeling the actual resources referenced by authorizations as target objects.

These distinct domain representations can be refined in two ways: either by directly modifying the structures (e.g., by means of a GUI) or else by defining domain scope expressions in authorizations (see Sec. 2.1). It is possible to create these domains by sub-classing existing concepts, although this top-down approach is prone to becoming cumbersome if multiple inheritance and more expressive constructors are introduced. Otherwise, union ($\sqcup$), intersection ($\sqcap$), and complement ($\neg$) operators of DL can be used to compose more complex concepts. In the example, primitive concept definitions InHouseDeveloper, OutSourcedDeveloper, and MasterStudent are created. In order to exemplify the *separation of duties* (SOD) our model is allowing for (see Sec. 3), OutSourcedDeveloper is made disjoint with both the other concepts, as they are distinguished by a different physical location (here, we suppose that MasterStudents may work alongside InHouseDevelopers). Finally, primitive concepts are combined by means of the union operator.

1) External human resources are grouped by the ExternalResource concept:

$$\mathsf{ExternalResource} \doteq \mathsf{OutSourcedDeveloper} \sqcup \mathsf{MasterStudent} \quad (1)$$

2) The Analyst concept represents in-house IT staff:

$$\mathsf{Analyst} \doteq \mathsf{InHouseDeveloper} \sqcup \mathsf{MasterStudent} \quad (2)$$

3) The whole IT staff, including external consultants, is represented with the TechnicalReferent concept:

$$\mathsf{TechnicalReferent} \doteq \mathsf{OutSourcedDeveloper} \sqcup \mathsf{Analyst} \quad (3)$$

The complete context is shown in Fig. 1a. Note that, the subsumption relationship between OutSourcedDeveloper and ExternalResource is not explicit in the definitions (1) to (3): indeed the hierarchies in Fig. 1 and Fig. 2 are products of the classification procedure accomplished by the reasoner. Target objects (see Fig. 2a), are structured according to:

- a taxonomical representation of the object's informative content. In our example, this is limited to Document and TechnicalReport;
- a set of mutually exclusive clearance levels (in our example, ClassifiedData and SecretData) represented as disjoint sub-classes of SensitiveData.

To exemplify the second category of conflicts arising from a semantics-aware context description, see Sec. 4.2, the document instance Note123 is asserted to be a member of both TechnicalReport and ClassifiedData.

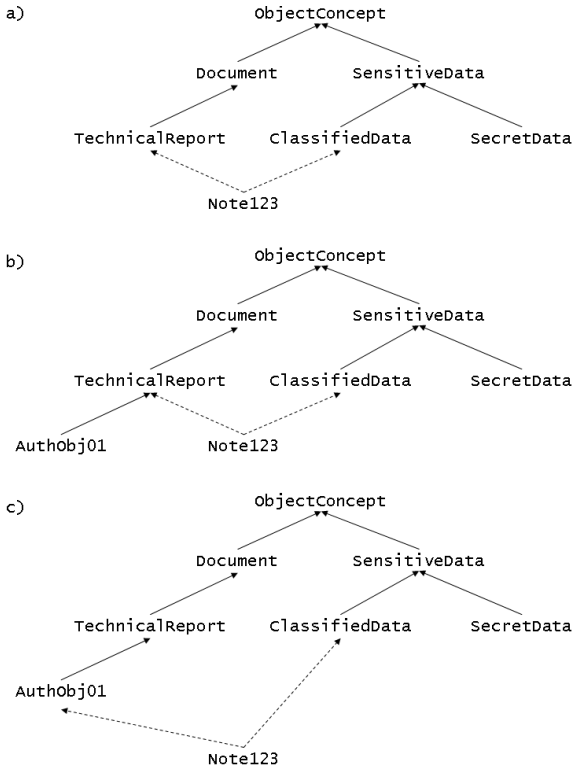

Figure 2: a) Context information for target objects; b), c) domain scope expressions represented as context concepts.

2.1 Authorizations and Domain Scope Expressions

The model introduced so far allows for referencing a fine-grained description of context information in AC authorizations. More interestingly, we allow for using the same modelling primitives in the definition of domain scope expressions combining elements in the context description. Authorization A5 in Fig. 3 is addressing as subjects all human resources except master students by using the complement ($\neg$) operator:

$$\mathsf{AuthSubjA5} \doteq \neg\mathsf{MasterStudent} \quad (4)$$

Similarly, authorization A2 applies to individuals being both an analyst and an external resource by using the intersection ($\sqcap$) operator:

$$\mathsf{AuthSubjA2} \doteq \mathsf{ExternalResource} \sqcap \mathsf{Analyst} \quad (5)$$

In our example, authorizations apply to subject and object descriptions; of course, they could also directly apply to individuals such as MasterStudent(JohnSmith), but it is of little interest to our purpose. Other than the subject/object pair an authorization is applying to, we only need to specify the authorization mode (+ or −) that will be used for carrying out modality conflict detection in Sec. 4. Fig. 3 displays all six authorizations considered by the example of conflict detection presented in this paper. The *Subject* and *Object* columns display the domain scope expressions referenced to by the authorization. For the sake of clarity, singleton expressions (i.e., those not making use of operators to combine concepts from the context description) are

Auth.	Mode	Subject	Object
A1	+	MasterStudent	TechnicalReport
A2	−	AuthSubjA2	TechnicalReport
A3	+	ExternalResource	Document
A4	−	TechnicalReferent	TechnicalReport
A5	+	AuthSubjA5	TechnicalReport
A6	−	ExternalResource	SensitiveData

Figure 3: The set of authorizations considered by the example.

indicated with the concept itself. Instead, authorizations A2 and A5 address the more complex domain scope expressions AuthSubjA2 and AuthSubjA5 introduced in (5) and (4), respectively.

2.2 Context Pre-processing

One of the advantages of modeling context information with DL is the opportunity of re-using the underlying theoretical framework for dealing with implicit knowledge. A broad range of inference procedures are available for deriving information not explicitly stated in a *knowledge base* (KB); refer to [6] for a comprehensive description of these. Whereas expressive DL feature critical worst-case complexities, these cases are very difficult to obtain in practical examples. Note that, in our model, the union operator is the only source of non-deterministic complexity in the structures processed by tableaux algorithms and constraint systems.

In our model, we first need to keep context information consistent when new concept definitions and domain scope expressions are introduced. Given concepts A and B, a *subsumption check* allows to determine whether $A \sqsubseteq B$ or $B \sqsubseteq A$; this standard inference procedure has efficient decision algorithms for very expressive DL sub-languages [11]. The *classification* inference procedure, generalizing the subsumption check to the whole terminology (often called *TBox*), is applied to create the concept hierarchies in Fig. 1 and 2.

Moreover, we need to associate authorizations with parameters in actual AC requests i.e., identify the individuals targeted by domain scope expressions as subjects and objects. When reasoning in the closed-world (see Sec. 3.2), this can be exhaustively carried out, determining the *Access Control Lists* associated with resources. The inference procedure used for this computation is often called *instance checking*. In our working example, the Pellet reasoner [1] was employed for carrying out both classification and instance checking.

3. COMPARISON WITH TRADITIONAL ACCESS CONTROL

Traditional AC relies on generic *directed acyclic graphs* (DAG) to describe context information such as the structure generated by containment and overlapping relationships between RBAC domains [26]. Set-theoretic operators have also been introduced to model context information and enable policy conflict detection and resolution [19]. Still, it is difficult to support set-theoretic operators within domain scope expressions of authorizations. Considering the branch of context information describing target objects (Fig. 2a)

a positive authorization for allowing access to technical reports that are not considered sensitive data would use the following domain scope expression as the target object:

$$\mathsf{AuthObj01} \doteq \mathsf{TechnicalReport} \sqcap \neg\mathsf{SensitiveData} \quad (6)$$

In traditional AC frameworks, this authorization is generally split into a positive authorization on concept TechnicalReport and a negative authorization on concept SensitiveData. This leads to a conflicting configuration for those items being asserted as members of both classes (e.g., the individual Note123 in Fig. 2a) and the information conceived by (6) as a whole (i.e., the negative authorization on SensitiveData being the exception to the general case) is lost. Moreover, resolving the conflict on the basis of the specificity principle would see the positive authorization taking over, since TechnicalReport would prove more specific than SensitiveData w.r.t. Note123 in most measure notions [7, 9]. This is clearly not the intended behavior.

In our model, domain scope expressions become part of the ontology modelling context information. Therefore, it is possible to state a single positive authorization having the concept description in (6) as the target object (Fig. 2b). By doing this, the apparent conflict generated by the authorization pair presented above can be avoided, specifically, Note123 in not anymore in the scope of the authorization. Should we restrict the scope of (the negative part of) the authorization to SecretData

$$\mathsf{AuthObj01} \doteq \mathsf{TechnicalReport} \sqcap \neg\mathsf{SecretData} \quad (7)$$

Note123 would enter the scope of the authorization, see Fig. 2c. Finally, under the assumption of a closed world when evaluating policies (see Sec. 3.2) we can dispense with negative authorizations when specifying authorizations by complementing the subject being denied access directly within a positive authorization, as in A5.

3.1 Close-up on RBAC

To further clarify the expressiveness gain achievable with the adoption of a semantics-aware context description, we compare it with the capabilities of classic Role-Based Access Control (RBAC) [25, 26]. This section, primarily intended to readers knowledgeable in the area of AC, aims at categorizing our ontology-based model according to the parameters traditionally used for RBAC. In our model, context representation is primarily made of concept definitions that roughly correspond to *domains* in RBAC. In our model, however, domain scope expressions referenced to by authorizations originate new domains, as shown in Fig. 1b. Context information is then classified as a partially ordered structure where D dominates C iff $C \sqsubseteq D$, where $\sqsubseteq$ is the subsumption operator of DL. This structure can ground the *permission-inheritance interpretation* of policy propagation i.e., rights assigned to concepts can be inherited by subsumed concepts. The structure considered so far can be assimilated to a *Hierarchical Access Control* where the structure of roles, determined at context-setup, is complemented at policy-writing time by the definition of domain scope expressions.

By allowing asserting disjointness between concepts, our model makes it possible to express a strong notion of *static separation of duties*, in the sense that a subject cannot belong to two or more classes at a time (i.e., a subject be assigned roles that should be separated). In Sec. 5 we will argue for a weaker notion of separation of duties. Moreover, in our model user-role and permission-role relationships correspond, respectively, to type and property assignments that are stored and processed as assertions. Therefore the main requirement of *Symmetric RBAC* i.e., supporting permission-role review of authorizations with performance comparable with user-role review [21], is implicitly met. Finally, since rights are propagated according to concepts classification, revoking an individual's membership to a given concept automatically revokes previously inherited rights as well. This property of our model can be assimilated to the *strong revocation of roles* in RBAC.

3.2 Reasoning in different worlds

So far we exemplified the novel use we make of some DL operators in the definition of context information and domain scope expressions. Another specificity of our model is the different approach we can take to evaluate *implicit* knowledge. In our example, this would mean deciding, for instance, whether Note123 is in the scope of the domain scope expression (7). Our model supports two different assumptions leading to different outcomes.

The first one assumes context information to be *complete*: knowledge not explicitly stated is considered negative knowledge. Under this *closed-world assumption* (CWA), which is the interpretation of a knowledge base underlying the relational data model, Note123 is considered ¬SecretData because there is no assertion in the KB stating the contrary. The second assumption states that available information is incomplete and that when an assertion is not explicitly stated one cannot assume its contrary to be true. Under this *open-world assumption* (OWA), Note123 is considered ¬SecretData because it is asserted to be ClassifiedData and clearance levels are mutually disjoint. Similarly, under the CWA the authorization subject in (4) comprises both InHouseDevelopers and OutSourcedDevelopers, while under the OWA only the latter can be inferred to be ¬MasterStudent, because it was made disjoint with MasterStudent.

Choosing between CWA and OWA is often less awkward than it may seem at first sight. While the closed world assumption applies to many aspects of context information (e.g., target objects can be generally assumed to be known in advance) this may not be the case for authorization subjects. As an example, in access control infrastructures based on negotiation of credentials, nothing is known in advance on which concepts a subject can be ascribed to [2]. As credentials are negotiated and exchanged, type assignments relate the subject to concept definitions in the context ontology. In this scenario, the closed-world assumption might even turn out to be deceptive in combination with the complement operator: in principle, in order to be ascribed to ¬MasterStudent, a user could just refrain from providing the credentials that would ascribe she to MasterStudent.

On the other hand, the open-world approach can cope with this kind of incomplete information and derive information not achievable with traditional information retrieval systems [10]. This behavior can be assimilated to a *discretionary role assignment*, where user-role relationships are determined by run-time decisions e.g., as a consequence of negotiation. Unfortunately, it is necessary to introduce negative authorizations when evaluating authorizations in the open-world because of the different semantics of the complement operator of DL w.r.t. the set-theoretic counter-

part. Referring to Fig. 2b and 2c, should Note123 not be asserted to be ClassifiedData, it would not be inferred of type ¬SecretData as well. Moreover, investigating modality conflicts is particularly interesting in the open-world because, as described in the beginning of this section, our model can dispense with negative authorizations when reasoning in the closed-world and hence no modality conflict can occur.

4. MODALITY CONFLICT DETECTION

Traditionally, modality conflicts are determined by authorizations of opposite mode (indicated by + and −) that apply to the same subject, action, and target simultaneously. The works [18, 19] distinguish between *authorizations* (A) and *obligations* (O), both of which can be positive or negative. Note that, in this model, a negative obligation should be interpreted as *obliged not to*; other approaches (e.g., when using Deontic Logics) obligations are interpreted as *not obliged to*. Three possible kinds of conflicts can then be defined:

A^+/A^- when the subject is both allowed and forbidden to do an action on the object;

O^+/O^- when the subject is required to and should refrain from doing an action at the same time;

O^+/A^- when the subject is required to do an action which is forbidden to her.

Abstracting from the interplay of authorizations and obligations, we only consider the former for the purpose of detecting modality conflicts. For the sake of conciseness, we refrain from extending the results to the complete model. Authorizations in Fig. 3 are propagated according to the subsumption relationships between concepts, on the basis of the partially ordered structures obtained by classifying SubjectConcepts and ObjectConcepts. Authorizations of opposite mode that apply to the same subject/object pair can then be identified. Fig. 4 lists the conflicts introduced by the authorizations in Fig. 3: they are identified by the authorizations involved $(+,-)$ and the subject/object pair generating the conflict by inheriting opposite authorizations. The last two conflicts also apply to concepts subsumed by ExternalResource, but it is sufficient to consider their common super-class to resolve them all. Summarizing, we can identify two possible sources of modality conflict:

1. *Terminological conflicts*, when a concept inherits conflicting conditions from super-classes.
2. *Extensional conflicts*, as individuals may belong to multiple classes, inheriting conflicting conditions.

In Fig. 4, all but the last one are conflicts of the first kind and can be spotted at policy design-time (either when context information is defined, or else when authorizations are being written). The last conflict belongs to the second type and can be spotted at compile-time only if entities involved (primarily subjects) are statically typed. In this case, authorizations can also be evaluated in the closed-world assumption, as explained in Sec. 3.2.

4.1 Terminological conflicts

In other words, conflicts of the first type can be spotted and, in some cases, resolved by considering the concept hierarchy used for propagating authorizations. The *specificity principle* being applied is based on the notion of *domain nesting precedence* [19, 18] whenever this relationship exists. Otherwise, authorizations inherited from incomparable entities can be compared with a *semantic similarity measure* $s(.,.)$ [7, 23]. Four different cases can be identified:

(+,-)	Subject	Object
(A1,A4)	MasterStudent	TechnicalReport
(A3,A2)	MasterStudent	TechnicalReport
(A1,A2)	MasterStudent	TechnicalReport
(A5,A4)	OutSourcedDeveloper	TechnicalReport
(A3,A4)	ExternalResource	TechnicalReport
(A3,A6)	ExternalResource	Note123

Figure 4: Conflicts generated by the definitions in Fig. 3.

1. The conflict can be trivially reconciled by looking at the classified ontology, as in the first two tuples: condition pairs (A1,A4) and (A3,A2) (Fig. 5a and 5b), apply to MasterStudent with opposite sign, but conditions A1 and A2, respectively, apply to a more specific concept and should then take precedence.

2. The conflict cannot be resolved by looking for a more specific concept, as in the third tuple where the condition pair (A1,A2), in Fig. 5c, directly apply with opposite mode to MasterStudent.

3. The conflict involves incomparable entities within the categorization of subjects or objects, such as authorizations (A5,A4) conflicting w.r.t. OutSourcedDeveloper and being inherited from the unrelated concepts TechnicalReferent and AuthSubjA5, see Fig. 5d. In this case, a similarity measure can be used to compare the following measures and decide which concept is more specific to OutSourcedDeveloper:

$$s(\text{OutSourcedDeveloper}, \text{TechnicalReferent}) \quad (8)$$

$$s(\text{OutSourcedDeveloper}, \text{AuthSubjA5}) \quad (9)$$

4. The conflict involves two conditions whose subjects and objects are respectively more and less specific than each other. This conflict is exemplified by conditions A3 and A4 in the fifth tuple, being more and less specific according to subjects and objects respectively, see Fig. 5e and 5f. Here we can only compute, for each branch, the least of the similarities between the concepts involved (in this case, the other similarities are always equal to 1):

$$s(\text{ExternalResource}, \text{TechnicalReferent}) \quad (10)$$

$$s(\text{TechnicalReport}, \text{Document}) \quad (11)$$

Anyway, interpreting results is difficult because comparing distinct branches of context information may not reflect the mutual specificity between concepts.

4.2 Extensional conflicts

The second category of conflicts is due to multiple inheritance from unrelated concepts. To exemplify this, consider

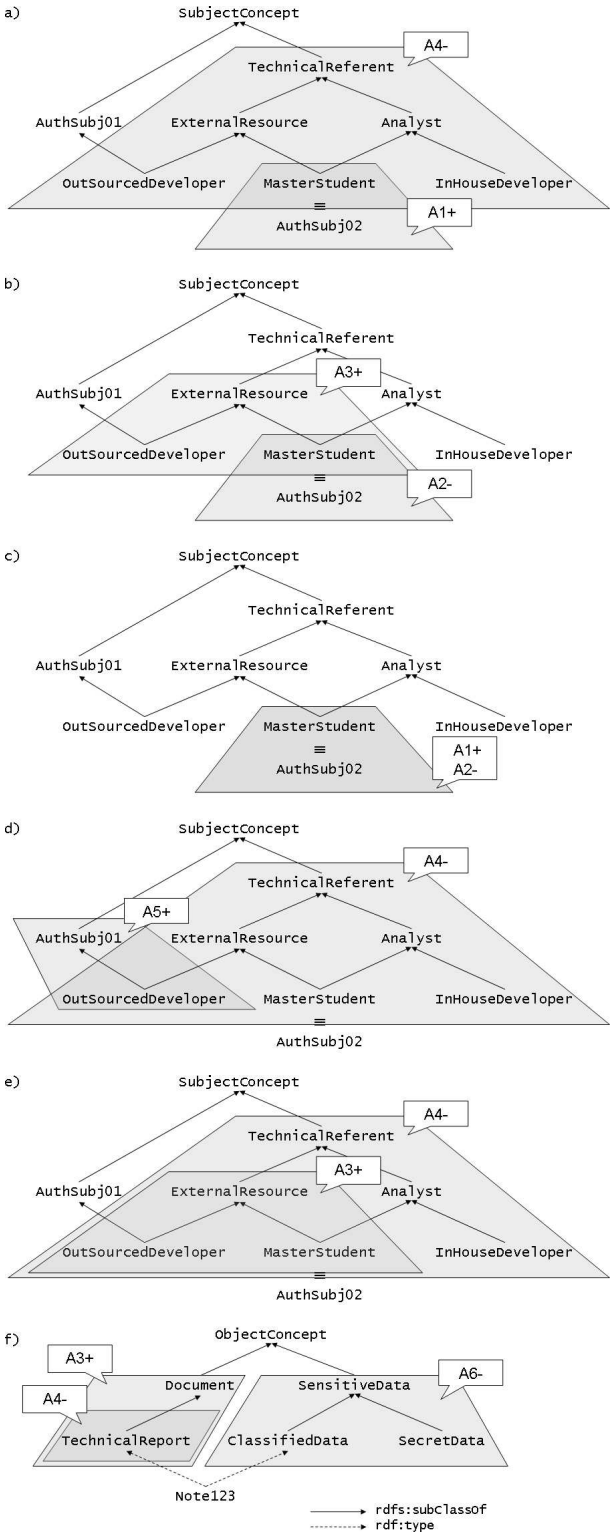

Figure 5: Propagation of authorizations in the hierarchy of subjects and objects.

the last conflict in Fig. 4: by looking at the classified structure (Fig. 5f), there is no evidence that A6 could collide with another authorization, since no positive authorization applies to SensitiveData or one of the concepts subsumed by it. Still ClassifiedData is only made disjoint with SecretData (because clearance levels are mutually exclusive) and then nothing prevents the coexistence of the following assertions, see Fig. 2a:

$$\mathsf{ClassifiedData}(\mathsf{Note123})$$

$$\mathsf{TechnicalReport}(\mathsf{Note123})$$

Two more equivalent conflicts are related to the concepts subsumed by ExternalResource and can be as usual ignored. Reconciling this conflict requires the notion of *most specific concept* ($msc(.)$) of an instance i in the KB, which is defined as the concept C such that:

1. $KB \models C(i)$
2. $D \sqsubseteq C \ \forall \ D \ s.t. \ KB \models D(i)$

where $\models$ is the standard semantic deduction. In DLs allowing for very expressive constructs (e.g., cyclic definitions and existential restrictions) the msc can only be approximated to a given depth [15, 14]. However, according to the very simple terminology modeling context information in our example, it is straightforward to compute the most specific concept w.r.t. Note123:

$$msc(\mathsf{Note123}) = \mathsf{TechinicalReport} \sqcap \mathsf{ClassifiedData}$$

This concept description can be inserted in the subsumption hierarchy of Fig. 2; the result is shown in Fig. 6. In order to reconcile the conflict (A3,A6), the more specific authorization w.r.t. msc(Note123) can be determined by calculating the following similarities:

$$s(msc(\mathsf{Note123}), \mathsf{SensitiveData}) \tag{12}$$

$$s(msc(\mathsf{Note123}), \mathsf{Document}) \tag{13}$$

4.3 Semantic Similarity Measures

Whenever colliding policy conditions cannot be resolved by means of domain nesting, it is still possible to compare the domain scope expression referenced by authorizations with the concepts the opposing conditions apply to, as suggested in the previous section with the similarities (8) to (13). Semantic similarity $s(.,.)$ is a positive definite, symmetric function based on the taxonomical structure of an ontology. The function applies to the cartesian product of the set of all concepts defined in the ontology and has its maximum value on the diagonal. The notion of *relatedness* takes into consideration the whole set of semantic links (not only is-a) between concepts. Together with the notion of *distance*, these measures have been widely investigated by Computational Linguistics and Artificial Intelligence (see [7] for some examples). Generally speaking, there is no *best* measure to accomplish conflict resolution, as distinct branches of context information (e.g., SubjectConcepts and ObjectConcepts) as well as different categorizations for each of them (e.g., ObjectConcepts indexed by both document typology and clearance level) can have a different measure best fitting its semantics. The most widely used technique for computing semantic similarity is based on *path distance*

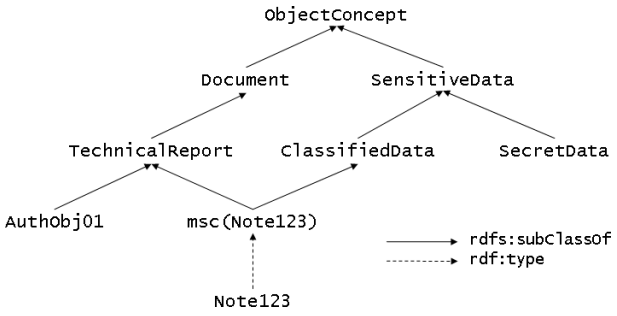

Figure 6: The extensional conflict addressing Note123 turned into a terminological one by computing the most specific concept.

between concepts in the subsumption hierarchy; the disadvantage of this technique is that, unless weighted, it assumes that the distance of each *semantic hop* is the same in each part of context description. In *Leacock and Chodrow's similarity* [16], only the shortest path length between the concepts being compared and the overall hierarchy depth determines the measure. *Wu and Palmer's similarity* [28] also takes into account the *least* (i.e., most specific) *common subsumer* (*lcs*) of the concepts to be compared but discards the weighting factor constituted by the hierarchy depth. Considering the conflict (A5,A4), both measures would prove AuthSubjA5 to be more specific than TechnicalReferent w.r.t. OutSourcedDeveloper. Path distance becomes less significant when comparing similarity measures calculated in unrelated contexts, such as for the conflict (A3,A4) where both SubjectConcepts and ObjectConcepts are taken into consideration. Moreover, path distance is not effective to reconcile extensional conflicts because multiple type relationships generally indicate different (and hence not easily comparable) categorizations of individuals. Moreover, computing the *msc* associated with an individual does not help reconciling the extensional conflicts, such as (A3,A6), as it would uniformly augment all the path distances being compared.

Another technique, *feature matching*, derives similarity on the basis of shared properties [27] but it is not applicable in this paper because our model does not make use, so far, of property relationships between concepts. *Information Content* (IC) [23, 13] is a third category of similarity measure based on Information Theory, measuring the variation of information between description levels in the concept hierarchy. Intuitively, the more individuals are ascribed to a concept, the less the informative value associated with the concept itself. For a measure of this kind to be applied in our model, the occurrence probability of concepts in a corpus of texts grounding the IC notion has to be translated into the exact ratio of concept instances. *Resnik's similarity* [22] is based on the IC associated with the *lcs* of the concepts to be compared. *Lin's similarity* [17] also considers the IC associated with the concepts being compared. This category of measures is more feasible to compare similarities evaluated in unrelated contexts, such as for the conflict (A3,A4), since the IC takes into account the local population of individuals. However, computing the *msc* does not change the IC associated with the *lcs* of the concepts whose similarity is being evaluated.

Finally, a recently proposed notion of similarity [9] explicitly takes into account the extensional component of a KB (i.e., individuals and their type relationships with concepts) because the extensions of completely unrelated concept descriptions may overlap (i.e., unless disjoint, they can share individuals) and this should be taken into account when computing similarity. This measure takes into consideration the number of individuals from the concepts being compared and the number of individuals they share in common (i.e., are instances of their intersection). The taxonomical structure of concepts is not explicitly taken into account, anyway it determines the type relationships that can be inferred from known facts and eventually the number of individuals associated with a given concept. In terminological conflicts, concepts to be compared are connected along the subsumption hierarchy, otherwise opposing authorizations would not clash with each other. In extensional conflicts this is not the case, but determining the *msc* can translate the conflict into a terminological one, which also takes into account multiple type relationships of the same kind.

5. CONCLUSIONS AND OUTLOOK

In this paper, we presented a simplified model for ontology-based context definitions in the specification and enforcement of AC authorizations. Anyway, the model can be easily extended to encompass any aspect of context description, as each of these can be processed separatedly. We then apply policy propagation and detect modality conflict that can be solved by applying either the specificity principle or a semantic similarity measure. We are already comparing different measures to be applied to specific branches of context description.

The first extension will be modeling part-of relationships between heterogeneous concepts (e.g., defining a Workgroup as composed by a ProjectManager, an ArtDirector, and one or more Developers) and distinguishing between functional and non-functional relationships of this kind (e.g., whether an individual can be part of more than one entity). By introducing custom property definitions for modeling role assignments w.r.t. a given instance, it is also possible to express a more finely-grained notion of separation of duties: as an example, by introducing properties develops and validates linking programmers to software modules, it can be stated that a programmer cannot validate a library she is directly working on. By adopting a very expressive DL it is also possible to consider cardinality constraints and quantifiers when writing policies (e.g., apply a condition to programmers that are working on at least n modules). When considering arbitrary property definitions, similarity measures based on the taxonomical structure of context descriptions may fail to portray the actual relationships between concepts or individuals and therefore more generic measures should be considered [8].

The small set of operators we used for context descriptions resulted in an expressive power far inferior to the (tractable) expressiveness achievable with DL; on the other hand, it kept the complexity of inference procedures under control. Actually, only the union operator leads to the non-deterministic expansion of the structures processed by tableaux algorithms and constraint systems [6]. Our future work will clarify the complexity issues associated with the inference procedures being used, indicating the *computational cliffs* marking the possible extensions to the basic model presented in this section.

6. REFERENCES

[1] Pellet OWL reasoner. http://www.mindswap.org/2003/pellet.

[2] Privacy and Identity Management for Europe. European RTD Integrated Project, FP6/IST Programme - www.prime-project.eu.org.

[3] Resource Description Framework (RDF). W3C Specifications - http://www.w3.org/RDF.

[4] Web Ontology Language (OWL). W3C Specifications - http://www.w3.org/2004/OWL.

[5] C. A. Ardagna, E. Damiani, S. D. C. di Vimercati, C. Fugazza, and P. Samarati. Offline expansion of xacml policies based on p3p metadata. In *Proceedings of International Conference on Web Engineering (ICWE)*, pages 363–374, 2005.

[6] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003.

[7] E. Blanchard, M. Harzallah, H. Briand, and P. Kuntz. A typology of ontology-based semantic measures. In *Open Interop Workshop on Enterprise Modelling and Ontologies for Interoperability*, 2005.

[8] C. d'Amato, N. Fanizzi, and F. Esposito. A Dissimilarity Measure for the $\mathcal{ALC}$ Description Logic. In *Proceedings of SWAP 2005, the 2nd Italian Semantic Web Workshop*. CEUR Workshop Proceedings, 2005.

[9] C. d'Amato, N. Fanizzi, and F. Esposito. A semantic similarity measure for expressive description logics. In A. Pettorossi, editor, *Proceedings of Convegno Italiano di Logica Computazionale (CILC05)*, Rome, Italy, 2005.

[10] E.Damiani, S. David, S. D. C. di Vimercati, C.Fugazza, and P.Samarati. Open World Reasoning in Semantics-Aware Access Control: a Preliminary Study. In *Proceedings of SWAP 2005, the 2nd Italian Semantic Web Workshop*. CEUR Workshop Proceedings, 2005.

[11] I. Horrocks and U. Sattler. A tableaux decision procedure for $\mathcal{SHOIQ}$. In *Proc. of the 19th Int. Joint Conf. on Artificial Intelligence (IJCAI 2005)*, 2005.

[12] S. Jajodia, P. Samarati, M. Sapino, and V. Subrahmanian. Flexible support for multiple access control policies. *ACM Transactions on Database Systems*, 26(2):214–260, June 2001.

[13] J. J. Jiang and D. W. Conrath. Semantic symilarity based on corpus statistics and lexical taxonomy. In *Proceedings of International Conference Research on Computational Linguistic*.

[14] R. Kusters and R. Molitor. Computing most specific concepts in description logics with existential restrictions.

[15] R. Kusters and R. Molitor. Approximating most specific concepts in description logics with existential restrictions. In *KI/OGAI*, pages 33–47, 2001.

[16] C. Leacock and M. Chodorow. Combining local context and wordnet similarity for word sense identification. pages 265–283, 1998.

[17] D. Lin. An information-theoretic definition of similarity. In *Proc. 15th International Conf. on Machine Learning*, pages 296–304. Morgan Kaufmann, San Francisco, CA, 1998.

[18] E. C. Lupu and M. Sloman. Conflicts in policy-based distributed systems management. *IEEE Transactions on Software Engineering* , 25(6):852–869, November/December 1999.

[19] E. C. Lupu and M. S. Sloman. Conflict analysis for management policies. In *Proceedings of the 5th IFIP/IEEE International Symposium on Integrated Network management IM'97, San Diego, CA*, 1997.

[20] A. Majidian. A model for static conflict analysis of management policies. *BT Technology Journal*, 17(2):53–59, 1999.

[21] G. Neumann and M. Strembeck. Design and implementation of a flexible RBAC-service in an object-oriented scripting language. In *ACM Conference on Computer and Communications Security*, pages 58–67, 2001.

[22] P. Resnik. Using information content to evaluate semantic similarity in a taxonomy. In *IJCAI*, pages 448–453, 1995.

[23] P. Resnik. Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence Research*, 11:95–130, 1999.

[24] P. Samarati and S. D. C. di Vimercati. Access control: Policies, models, and mechanisms. In R. Focardi and R. Gorrieri, editors, *Foundations of Security Analysis and Design*, LNCS 2171. Springer-Verlag, 2001.

[25] R. Sandhu. Rationale for the RBAC96 family of access control models. In *RBAC '95: Proceedings of the first ACM Workshop on Role-based access control*, page 9, New York, NY, USA, 1996. ACM Press.

[26] R. Sandhu, D. Ferraiolo, and R. Kuhn. The NIST model for role-based access control: Towards a unified standard. pages 47–64.

[27] A. Tversky. Features of similarity. *Psychological Review*, 84(4):327–352, 1977.

[28] Z. Wu and M. Palmer. Verb semantics and lexical selection. In *32nd. Annual Meeting of the Association for Computational Linguistics*, pages 133 –138, New Mexico State University, Las Cruces, New Mexico, 1994.