Policy-Governed Information Exchange in a U.S. Army Operational Scenario

Larry Bunch
Florida IHMC
lbunch@ihmc.us

Jeffrey M. Bradshaw
Florida IHMC
jbradshaw@ihmc.us

Clifford O. Young
CSUSB
cyoung@csusb.edu

Abstract

The authors are investigating how emerging policy and semantic web technologies can be used to help provide the best set of available tactical information to the soldier in the field. In this initial effort, we have developed a system that demonstrates the potential of these technologies in a small-scale U.S. Army mockup scenario. The system represents and reasons about domain-specific policies to help recognize what documents the end solder is allowed to receive given the current mission context. The system also relies on policies to help recognize when appropriate human approval can be obtained or a specific transformation of the information can be performed to allow the information to be sent. Semantic web technologies are further used to describe the properties and features of each document and relate these features to mission contexts in which the information is likely to be appropriate. The result is a compelling demonstration of the role that policies and semantic web technologies can play in promoting the Army's need to share information while remaining vigilant of the requirements to protect methods and sources.

1. Introduction

Researchers from the U.S. Army Research Labs (ARL)[1] and the Florida Institute for Human and Machine Cognition (IHMC) are collaborating to demonstrate technologies that can ultimately help provide the best set of available tactical information to solders in the field. To this end we have developed a small Army information exchange scenario and a demonstration system that helps a user search a set of tactical documents for those potentially allowed to be sent to a particular solder, identify documents appropriate to the soldier's current mission context, and to recognize and fulfill any policy requirements regulating the information exchange.

Our approach employs semantic web technologies including the Web Ontology Language (OWL) [1] and Resource Description Framework attributes (RDFa) [2] to model the actors, documents, missions, and other elements of the scenario. This defines machine-accessible attributes and relationships among the scenario entities, such as the features of a particular document and the types of document features most desirable for a certain mission type. The model is then augmented with formal policies (also in the OWL language) using the KAoS policy services framework [3] to represent the regulations governing user access to documents in the scenario. Finally, the policy services are integrated with the Simple Protocol and RDF Query Language (SPARQL) [4] to enable dynamically filtering query results to ensure policy-compliance and to also query for policy information such as a requirement to obtain specific human approval for a document to be released.

2. Example Army Tactical Scenario

This scenario is intended to provide a backdrop against which we can show the capabilities of selected technologies within a mission setting. The battlefield situations described represent current tactics, techniques, and procedures that are enhanced with near-term future assets. The scenario will serve to promote discussion about future capabilities and equipment that might be required to operate on the battlefields of tomorrow. The following is a very brief description of the scenario.

The 3rd Brigade Combat Team (BCT) staff receives operations orders for Delta Company which is one of the companies that comprise the brigade. A BCT staff member then prepares an information package to accompany the mission orders. This package may contain any number of documents from an existing collection of text, images, video, and map layers. The goal is to help this BCT user assemble the most complete and mission-appropriate set of information in

[1] This work was supported by Army Research Labs cooperative agreement number W911NF-07-2-0088.

a timely manner. This requires the BCT staff member to recognize all of the documents that Delta Company could potentially be allowed to receive and select from among these the most appropriate documents for the current mission. Finally, this user must ensure that any policies governing the release of each document are followed. Our demonstration system helps the user accomplish this task by first parsing incoming operations orders for information about the unit, mission, and priority intelligence requirements. This information is then used to query a model about the set of available documents as well the policies governing release of these documents.

The second part of the scenario involves enforcing the need to share information where "[c]ritical information that the warfighter didn't know existed, and the owner of the information didn't know was important, must be made available within a global information environment easily accessible to commanders at all levels" [5]. The BCT staff receives a new intelligence report and must determine which, if any, units should be notified about this new information. Our system similarly assists with this task by parsing the incoming report and querying for missions with matching priority intelligence requirements and associated policies concerning notification requirements.

3. Ontology and Policy Application to Military Mission Intelligence

We have identified the following beneficial capabilities that ontology and policy technologies enable in this military mission intelligence scenario.

- An ontology-based policy services framework able to interpret mission intelligence requirements coupled with mission metadata enables an automated solution for the identification of appropriate intelligence products.
- A mission ontology coupled with intelligence product metadata enables an automated policy framework to filter the various intelligent products regarding classification level and appropriateness for delivery to mission units.
- An automated policy framework may obligate software agents to redact intelligence products, human analysts or commanders to modify and authorize product delivery, and/or directly pass on intelligence products in accordance with in-place security procedures.
- A system that represents a unit commander's and/or staff's specific or unique intelligence requirements within the automated procedures maintains a vital linkage to reflect "ground truth" which is necessary for successful mission execution and completion.

Figure 1 depicts the policy and ontology dimensions that relate users to intelligence products in this domain. By modeling these dimensions we can begin to define the 'best set of available information' as that which is authorized or can be authorized through human approval and/or data filtering, and that is mission-related and/or a local requirement for the unit.

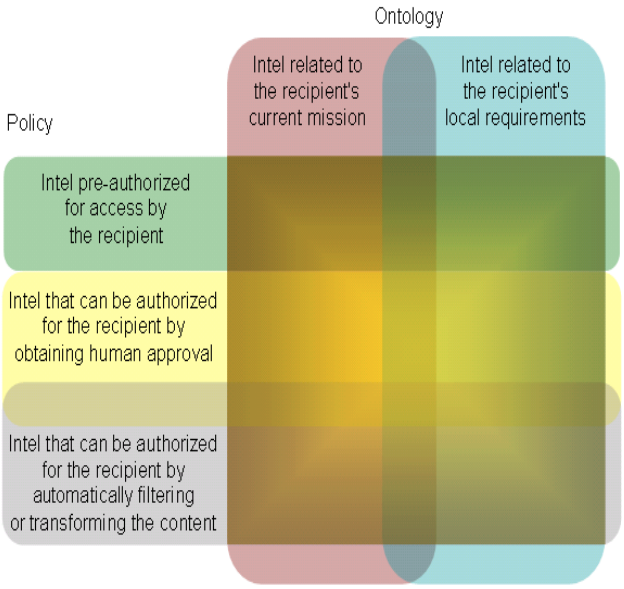

Figure 1. Policy and ontology dimensions relate users to intelligence products.

4. Scenario Ontology

An OWL ontology was created to define the actors, entities, and relationships in the scenario and this ontology forms the basis for further defining policies and queries for an automated system. In this scenario, we defined a class of Documents with properties such as Classification-Level and Perishability as well as a list of Document Features such as Topography, Routes, Enemy Activity, etc. We also define types of Military Missions such as Peacekeeping and Assault missions. Each type of Mission is then associated with a list of Document Features that are appropriate for the mission.

The ontology is also populated with information about each instance of a document, actor, and mission. For text documents represented in HTML, the RDFa syntax enables embedding the ontology information within the document. For other document types, the metadata is defined externally in an OWL file that

contains URL references to each document defined therein.

5. Policies in Force

The following policies governing access to the documents are represented formally in a KAoS model as a combination of authorization policies that allow user access to documents and obligation policies that require (or waive) approval or transformation on the data. The policy statements here paraphrase the formal KAoS policies which are represented in OWL and are too large to include here.

Authorization Policies

- BCT members authorized to access documents that have a classification level of SecretOrBelow (e.g. Secret, Sensitive, and Unclassified)
- Company members are authorized to access documents that have a classification level of Secret and are Perishable.
- Company members are authorized to access documents that have a classification level of SensitiveOrBelow.

Approval Policies

- BCT members are obligated to approve Company member access to documents that are SensitiveOrAbove.
- EXCEPT: BCT members are not obligated to Approve Company member access to HTML documents that are Sensitive and Perishable.

Redaction Policies

- BCT members are obligated to redact source-identifying text for Company member accesses to HTML documents that are SensitiveOrAbove.

Notification Policies

- BCT members are obligated to Notify the creator of any Orders containing priority intel requirements that match the features of a received document.

6. SPARQL Queries with Integrated Policy Checking

The RDF metadata about each document, the semantic mission-feature-document relationships defined in the ontology, and the policies are loaded into an ontology model and joined together through a query in the SPARQL language as shown in Figure 2. The SELECT clause identifies the variables to appear in the query results, and the WHERE clause provides the pattern that the data, in this case a document, must satisfy to be included in the results. In Figure 2, the variables 'classification', 'title', 'doc-id', and

```
PREFIX rdf:     <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:    <http://www.w3.org/2000/01/rdf-schema#>
PREFIX b3an:    <http://localhost/b3an/meta/b3an-intel-ontology.owl#>
PREFIX auth:    <java:b3an.filter.>
PREFIX kaos:    <urn:KAoS#>
SELECT DISTINCT ?approve ?classification ?title ?doc_id ?location ?transform
WHERE
{  ?doc_id b3an:title ?title .
   ?doc_id b3an:identifier ?location .
   ?doc_id b3an:classification-level ?classification .

1  ?doc_id b3an:p-feature ?doc_feature_proto .
   b3an:a_AttackMission b3an:missionPriorityInfo ?mission_priority_info_proto .
   ?mission_priority_info_proto auth:baseType ?mission_priority_info_type .
   ?doc_feature_proto rdf:type ?mission_priority_info_type .

2  OPTIONAL {(?doc_id kaos:DeltaCompany) auth:approvalRequired ?approve}
   OPTIONAL {(?doc_id kaos:DeltaCompany) auth:transformRequired ?transform}

3  FILTER auth:authorized(?doc_id, kaos:DeltaCompany)}

ORDER BY DESC(?approve) DESC(?transform) ?title
```

Figure 2. An example SPARQL query with integrated policy checking.

'location' are bound to properties defined explicitly for each document in the RDF metadata. The part of the WHERE clause labeled (1) matches the document features from the metadata with the *types* of document features related to the current Peacekeeping mission via the OWL ontology.

6.1. Querying Obligation Policies

The variables 'approve' and 'transform' are bound to KAoS policy information by leveraging SPARQL's extensibility to define new property functions as shown in section (2) of Figure 2. The 'auth:approvalRequired' function is a custom java class, java.b3an.filter.approvalRequired, with a method that takes a document id and a user id as parameters and returns the id of a user obligated by policies to approve user access to the specified document. The 'approvalRequired' function returns null if no approval is required and the OPTIONAL clause ensures these records are still included in the results. The set of Redaction policies are integrated with the query results in a very similar way, though the return value of the 'transformRequired' function is the identifier of yet another function. The KAoS policy for Redaction specifies the Java class to instantiate and invoke with the contents of the document to obtain the transformed result. In this scenario the transformation operates over an HTML document containing RDFa markup. Any blocks of HTML that are marked as containing source identifying information (in this case 'creator' or 'contributor' properties) are redacted from the resulting HTML. Figure 3 is a screenshot of the demonstration client with a table view of the query results. The results of the 'approvalRequired' policy check are displayed as a column of checkboxes indicating which documents require approval and capturing the BCT user's input concerning the approval or denial. The

results of the 'transformRequired' policy check are displayed in Figure 3 as a column of 'RDFa Redactor' buttons which the user can select to preview the transformed version of each document.

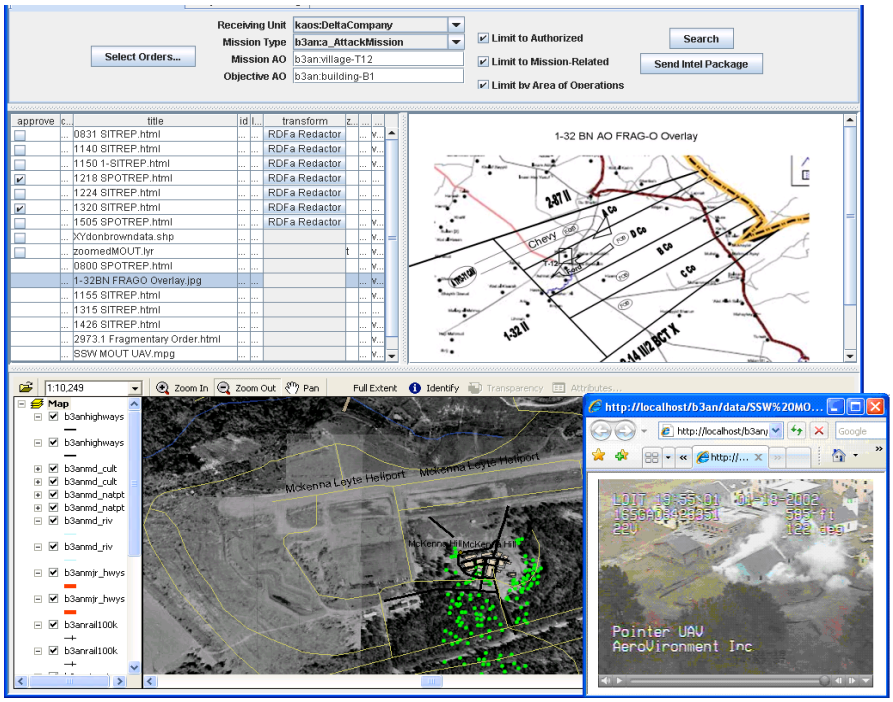

Figure 3. The demonstration client display for helping find authorized mission-related information.

6.2. Querying Authorization Policies

The part of the WHERE clause in Figure 2 labeled (3) also leverages the extensibility of SPARQL to integrate KAoS policies. In this case, 'authorized' is a filter function that takes the parameters of a document id and a user id and returns true if the user's access to the document is authorized by KAoS policies. Wrapping this policy-checking call in the FILTER statement eliminates any entries from the results that have a false authorization.

7. Enforcing the Need to Share Information

We further demonstrate enforcement of the need to share information through a similar process of parsing an input document for its RDFa description, then using that information to query ontological relationships as well as policy requirements. In this case, the SPARQL query is for documents in the model that are of the Orders type which contain one or more priority intelligence requirements matching a feature of the input document.

This combination of machine accessible metadata, a formal ontology relating this metadata to domain-specific constructs such as mission requirements, and the ability to interpret and enforce policies through these constructs is a powerful approach to overcoming some of the challenging barriers to automated Army information exchange.

8. References

[1] D. McGuinness, F. van Harmelen, OWL Web Ontology Language Overview. http://www.w3.org/TR/2004/REC-owl-features-20040210/.

[2] RDFa Syntax, A collection of attributes for layering RDF on XML languages. http://www.w3.org/2006/07/SWD/RDFa/syntax/.

[3] A. Uszok, J.M. Bradshaw, M. Johnson, R. Jeffers, A. Tate, J. Dalton, and S. Aitken. "KAoS policy management for semantic web services." IEEE Intelligent Systems 19, no. 4 (July/August 2004): 32-41.

[4] E. Prud'hommeaux and A. Seaborne. SPARQL Query Language for RDF. http://www.w3.org/TR/2007/PR-rdf-sparql-query-20071112/.

[5] General James E. Cartwright, USMC, Commander of the U.S. Strategic Defense Command. Statement before the Senate Armed Forces Committee. April 4, 2005.