

Leveraging Controlled English at the Network Edge for Policy-Based Management of ISR Services

Christos Parizas*, Alun Preece*, S. Yousaf Shah[†], Boleslaw Szymanski[†], Petros Zerfos[‡], Dominic Harries[§],
Christopher Gibson[§]

*School of Computer Science and Informatics Cardiff University, UK, Email: {C.Parizas, A.D.Preece}@cs.cardiff.ac.uk

[†]Rensselaer Polytechnic Institute, Department of Computer Science, Troy, NY, USA, Email: {shahs9, szymab}@rpi.edu

[‡]IBM T.J. Watson Research Center, Yorktown Heights, NY, USA, Email: {pzerfos@us.ibm.com}

[§]IBM Hursley, UK, Email: {christopher.gibson, dharries}@uk.ibm.com

Abstract— Service-oriented Architectures (SOAs) for Wireless Sensor Networks (WSNs) are an active research topic. Yet, autonomous configuration of services for real life constraints (spatio-temporal, input/output interoperability, policies, security etc.) is still a challenging problem. In domains such as emergency response and military multi-partner coalition operations, constraints applied to auto-configured services are typically regulated through policies. Traditionally these policies are created at the center of the operational network by high-level decision makers and are expressed in low-level policy languages (e.g. Common Information Model Simplified Policy Language) by technical personnel. This makes them difficult to understand and work with by non-technical users operating at the edge of the network. In this paper we investigate the use of Controlled English (CE) as a means to define a policy representation that is both human-friendly and machine processable. We present a policy-based SOA management approach by developing a CE domain model that allows CE-expressed policy rules, which are evaluated directly by the service composition process to configure compliant services. The use of a CE policy model is intended to benefit coalition networks by bridging the gap between technical and non-technical users in terms of policy creation and negotiation, while at the same time being directly processable by a policy-checking system without transformation to any other technical representation.

I. INTRODUCTION

From conventional military to disaster relief operations accurate, reliable and actionable intelligence is needed in order for the operations to be effective. The significant proportion of this intelligence is increasingly produced by wireless, ad-hoc Intelligence, Surveillance and Reconnaissance (ISR) systems, which provide key capabilities to the command authorities for intelligence collection, exploitation and battle management. Service configuration in such systems is a challenging problem as the requirements of the applications hosted on ISR assets change over time and these changes must be reflected in the system configuration. As events (e.g. a node fails, a service becomes unavailable on a node, etc.) happen over time the configuration mechanism should dynamically reconfigure ISR assets according to the new requirements. An efficient configuration mechanism should be able to configure services in a way that their inputs and outputs are interoperable to perform a complex task.

Suppose ISR assets are deployed as a support system for a disaster relief effort. A monitoring system configured in

such a scenario might use audio and video feeds produced by other services to provide surveillance of the area to the effort coordinators. The service configuration in such a scenario should not only consider input/output portability [1] but also other factors, such as energy cost and spatial relevancy of services to the area of interest. In such a scenario, services that are more relevant (e.g. have a larger sensing range in the mission area) are more useful than services that provide the same outputs but with lower relevancy.

In aforementioned environments ad-hoc Communities of Interest (CoIs) act together to achieve common objectives, forming coalitions. A coalition is a set of organisations that work together usually in peer-to-peer environments where through collaboration they are able to jointly perform tasks that they would not be able to perform or perform poorly otherwise [2]. Typically, coalition partners own different sets of ISR assets and have their own inherent constraints, which are stated as a set of policies (including security, privacy and legal policies) on how to share their infrastructures with the others in accordance with operational procedures. These constraints also configure ISR services taking into account for example spatio-temporal, cost of services and input/output interoperability parameters.

Moreover, the aforementioned environments are highly dynamic, thus we need to consider a policy development and enforcement model which is able to quickly, easily and distributedly form, reform and negotiate policies according to environmental changes. A key step towards the development of such a model is to push policy development – and through it the decision making center – close to the source of situational changes in order to reduce the reaction time. The users who must first cope with unexpected operational changes are those at or near the edge of the network, so we believe it would be very beneficial if they were able to form, reform and negotiate policies themselves without waiting for a central authority to approve. The Network Edge approach to designing command and control (C2) concepts, organizations and systems to meet requirements of complex endeavors has become popular in recent years as it involves the empowerment of individuals at the edge of the organization [3]. An emerging issue related to pushing decision making through policy formation at or near the edge, is the technical gap between existing low-

level policy languages and non-technical users that operate in these areas. The vast majority of personnel at the network and organizational edges are not IT experts and so lack technical skills in terms of formal policy languages.

The contribution of this paper lies on two pillars. The first is proposal of a model for configuring ISR services, which considers spatial constraints on service selection, and configures the services by choosing low-cost and spatially relevant services improving the spatial relevancy of the overall system. Second, the use of a Controlled Natural Language (CNL) named Controlled English[4] (CE) as a means for defining a policy representation that is both human-friendly and unambiguous for computers. More specifically we present:

- A novel self-recovering and fault tolerant approach for the configuration of services with spatial and other policy constraints.
- A generic cost mechanism for services.
- Whether CE is expressive enough to capture a variety of high-level, attribute-based ISR service management policies.

The remainder of the paper is organised as follows: In Section II we present the service configuration approach with spatial relevance, while in section III we discuss our decentralized policy development model. In section IV we present the benefits of using CE as a policy representation format. In sections V and VI we conceptualize an ISR service management system and we present a set of policy management rules using CE respectively. Finally VII concludes the paper and discusses future work.

II. SERVICE CONFIGURATION WITH SPATIAL RELEVANCY

Service Configuration with Spatial Relevancy of a service to the area of interest plays a major role in the configuration of ISR services. Therefore, we aim to configure services that are both low-cost and highly spatially relevant to the required composite service. However both of these problems are NP-hard ([5], [1]). Although one is a minimization problem and the other is a maximization problem, we model both as a single minimization problem and apply a Set Cover heuristic to find the minimum cost service composition. We incorporate the relevancy aspect in the configuration of a service via a generic cost function. In a service configuration, any use of a particular service incurs some cost to the hosting node. This cost can be flat, such as energy consumed, or a combination of factors such as edge delay, battery consumption and processing time costs; we refer to such a cost as the BaseCost. Every sensor service has a BaseCost associated with it, which is incurred when the service is used. In case the user is not interested in the spatial relevancy of the service (which can be specified by the user in the request) our system will configure the system aiming at minimizing the BaseCost. We introduce a generic cost function that incorporates both the BaseCost and the RelevancyCost. The latter cost represents the irrelevancy of a sensor service to the users requested area of interest.

$$AggrCost = \alpha \times BaseCost + \beta \times (1 - Relevancy(\Gamma_x)) \quad (1)$$

$$\Gamma(x) = \frac{|C_R \cap c_{r,x}|}{|C_R|} \quad (2)$$

where C_R denotes area of interest with radius R and $c_{r,x}$ denotes the area of coverage of service x with radius r .

The equation 1 shows the aggregated cost (AggrCost) incurred when a sensor service is used where α and β are the user defined weights balancing the impact of the *BaseCost* and *RelevancyCost*. Both the relevancy cost and base cost are normalized to the same range before the aggregated cost is calculated. We then use heuristics of the Set Cover problem to find the minimum cost composition. The cost function based on the specified weights maximizes the covered area while minimizing the base cost. In this work, we consider relevancy as the overlap between the area of interest and the area covered by the measurement of a sensor. The measurement from sensors can be any kind of information collected from the environment. For simplicity, we model the area covered by a sensor as disk of radius r around the sensor location, where the location of the sensor is defined by latitude, longitude and altitude. We also assume that the user specifies area of interest is disk of radius R defined in the request for configuration. Our system is not restricted to a disk coverage model; we have developed an extensive library for various coverage models (hexagonal, polygon etc.) that are easily pluggable into the system according to the formulation of coverage areas and overlapping regions.

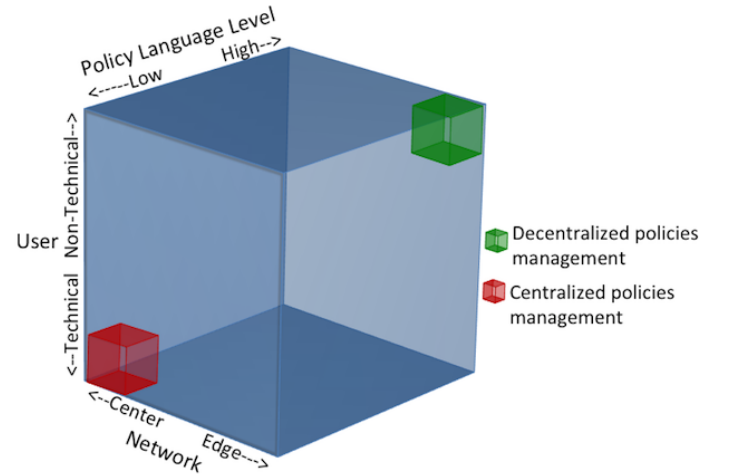


Fig. 1. Policy-based management: Centralized and Decentralized approaches

III. PUSHING THE DECISION MAKING AT THE NETWORK'S EDGE

In Figure 1 we present the centralized and decentralized approaches. The three axes of the blue cube represent the three main features related to ISR policy-based management in coalition environments. The user types that operate in such environments in terms of their IT expertise, which varies from Technical and Non-Technical, the place in the organization network where users operate which varies from Center (e.g.

a military base) to Edge (e.g. warfare theater) and the user-friendliness level of the applied policy language which might be either Low (e.g. a technical language understood only by IT experts) or High (e.g. a language close to natural language easy to be understood by non-experts).

The red cube represents the current state of policy development and enforcement models, which can be developed by technical users operating near or at the center of the organisation, using low-level policy representation languages and are cumbersome in a highly dynamic military environment. The green cube represents the problem space in which our contribution is situated. A policy model, which is able to cope with highly dynamic environments by enabling the policies development by non-technical users, who operate near or at the edge using high-level interface policy languages. In other words, the red cube represents a centralised directive policy-based management system based on the Industrial Age model, while the green represent a decentralised, emergent policy-based management system based on the Information Age model [6]. It is worth noting that the user and network variables in Figure 1 are not binary. This means that the users can span from IT experts to users who lack any technical skills including those with different technical knowledge levels. We claim that a high-level interface policy model can empower non-technical users (e.g. military planners and intelligence analysts) while at the same time cause no loss of technical users' expressiveness power (e.g. power provided via usage of low-level policy languages). As far as the organisation network is concerned we focus on users that operate at any place in between the military headquarters and the head of a battlefield operation.

IV. ACHIEVING EDGE C2 USING CE

As mentioned previously, the *edge* model empowers the users who operate at the edge of the network while in addition allowing for intra-edge communication without requiring permission from a central authority. Crucial preconditions for a successful application of *edge* model, apart from the need for enhanced peer-to-peer horizontal interaction among the users on the field, is the moving of senior personnel into roles operating at the edge [3]. Thus, users operating at the edge become more responsible and take further substantial initiatives, such as sharing and allocation of resources and establishment of engagement rules in a highly dynamic manner as a response to operational changes. Establishing a broader and deeper degree of shared awareness and understanding as well as a higher adaptability of the collective C2 process, the *edge* seems to be a promising and more effective approach than others[7]. Given the aforementioned characteristics of *edge* C2 approach and its inherent agility, it seems to be an ideal approach for contemporary multi-partner, complex and dynamic coalition operations.

Controlled Natural Languages (CNL) were first introduced for bridging the gap between formal representation languages

(e.g. OWL¹)[8] and natural languages (e.g. English) and introduce a user-friendlier knowledge representation form than the common formal languages [9]. Moreover, CNLs being a subset of natural languages (NL) are less complex and ambiguous, so they present improved interpretation for machines compared to NLs. In this work we use Controlled English[4] (CE), a type of CNL designed to be readable by a native English speaker whilst representing information in a structured and unambiguous way. The structure of CE is simple but fully defined by a syntax, which makes the language parsable by computer systems. CE aspires to provide a human-friendly representation format that is directly targeted non-technical, domain-specialist users to encourage a richer integration between human and machine reasoning capabilities [4].

Since we have not experimentally tested CE's understandability with the understandability of other lower level well-known policy languages such as CIM-SPL we cannot safely claim that CE is a user friendlier representation than its predecessors. However, there are in literature several works[10], [11], [12] which conducted experiments to test and compare the friendliness to humans of CNLs versus formal languages such as OWL. The results of the experiments in all cases led to the fact that CNLs like CE can do better in terms of understandability than formal languages; in addition they can achieve better results in situations where users have little or no technical training.

We introduce and further explain the CE structure and syntax in section III where we define the coalition assets sharing ontology. Here, we present a simple authorisation policy rule expressed in both: CIM-SPL in Table I and CE in Table II representations to show the different levels of human-friendliness of the two approaches. Suppose the simple scenario in a coalition operation context where an authorisation policy, which allows a user to access an asset if the user and the asset are both affiliated with the same partner.

Subject: user

Object: asset

Condition: users partner == assets partner

Decision: allow

As the examples below show, CE is a more user-oriented representation compared to CIM-SPL, while CIM-SPL seems a more concise one compared to CE. It is straightforward to non-technical users to read and understand the policy rule implemented in CE even if they have little knowledge of the domain model. CE representation is not far away from the policy's plain-text explanation above. On the contrary in order for a user to understand the policy rule implemented in CIM-SPL, some technical-programming skills are needed. However, some training is also needed for a user in order to develop policy rules in CE. Although, provided that CE is defined by

¹<http://cies.hhu.edu.cn/pweb/~zhuoming/teachings/MOD/N4/Readings/5-3-B1.pdf>

```

Condition
{
  subject.affiliation() == object.affiliation()
}
Decision
{
  canAccess.allow()
}

```

TABLE I
CIM-SPL REPRESENTATION

```

if
( the asset A is affiliated to partner P ) and
( the user U is affiliated with the partner P )
then
( the user U canAccess the asset A )
.

```

TABLE II
CE REPRESENTATION

syntax and grammar rules inherently closer to NL, we believe that also via particular user interfaces (such as Conversational UIs²): the training time for a user to learn composing policies is significantly shorter than the time needed for a user to learn how to develop policy rules in representations such as CIM-SPL. Due to the high level of user-friendliness, the use of CE in operations where *edge* C2 is applied seems to have a wide range of applicability for non-technical users operating at the edge.

V. CE CONCEPTUALISATION OF ISR SERVICE MANAGEMENT

Being a type of CNL, CE can be used to define domain models, which take the form of concepts definitions. Obeying to first-order logic these concepts comprise objects, their properties and the relationships between them. CE language supports multiple inheritance and can build hierarchies of concepts. Once the model is built, it then can be instantiated accordingly, via defining facts, based on the concepts and relationships defined in the model. CE allows any instance to be asserted as any number of concurrent concepts [13] (e.g. “the user U1 is a private and is an intelligence analysts and is a/an...”). It can also be used for developing rules, which follow the “if - condition – action” form, which can be executed on the model. Both the rules and the results of the rules execution are expressed in CE sentences without needing any other formal notations. The rationale behind the rule’s evaluation is automatically created by the system and is pushed to the user as well. Rationale is a set of reasoning steps, each one of which is defined as a “because” relation between multiple conditions in different rules and each single decision conclusion. The reasoning steps follow a backward-chaining interpreter in order to calculate and develop the rationale.

The IBM Controlled Natural Language Processing Environment³ (CE Store) is a web application which provides an information-processing environment within which human and machine agents (i.e. Java coded entities) can develop and interact with existing CE-based conceptual models. Within the CE Store different types of agents can develop logical

inference rules (i.e. policy rules) and execute them on a pre-developed conceptual model. In section VI we exploit the rules creation ability of CE in order to define a variety of attribute-based ISR service management policies.

In order to develop CE-based policy rules, we first need to define the ontology, which captures the objects that we want to manage, their properties and the relationships between them. With the following sample CE definitions we cover part of the ontology in Figure 2 while we present the basic capabilities and structure of CE as a domain concept developer.

To create a new object in the ontology you simple conceptualise it as follows:

```
conceptualise the User U.
```

To define concepts’ properties there are two forms, which are semantically identical but allow the subsequent facts to be expressed in slightly different ways:

Verb singular form:

```
conceptualise the team M
  ~ is led by ~ the teamLeader D.
```

Functional noun:

```
conceptualise a ~ coalition ~ C that
  has the partner P as ~ member ~.
```

CE can define any number of properties for a concept in a single sentence but it currently cannot mix *verb singular* and *functional noun* properties in the same sentence. Moreover, one can write as many sentences as they like for a single concept. CE Store will amalgamate all sentences for that concept into the model when it loads the sentences.

To define a property with a *textual value* rather than a relationship to another instance CE uses the word “value” as below:

```
conceptualise an ~ asset ~ A that
  has the value I as ~ inputType ~ and
  has the value O as ~ outputType ~ and
  has the value ME as ~ mgrs easting ~ and
  has at most one value MN as ~ mgrs northing ~ and
  has at most one value AL as ~ altitude ~ and
  has the value R as ~ coverageRadius and
  has the value C as ~ cost ~.
```

Once the conceptual model is defined, the next step is the instantiation of the model through fact sentences. Below we present instantiation examples for some of the objects, properties and relationships represented in ontology of Figure 2.

²As recently observed in Wired by Dr. Ron Kaplan from Nuance Communications’ NLU R&D Lab – <http://www.wired.com/opinion/2013/03/conversational-user-interface/> – checked 27th March 2013.

³<https://www.ibm.com/developerworks/mydeveloperworks/groups/service/html/communityview?communityUuid=558d55b6-78b6-43e6-9c14-0792481e4532>

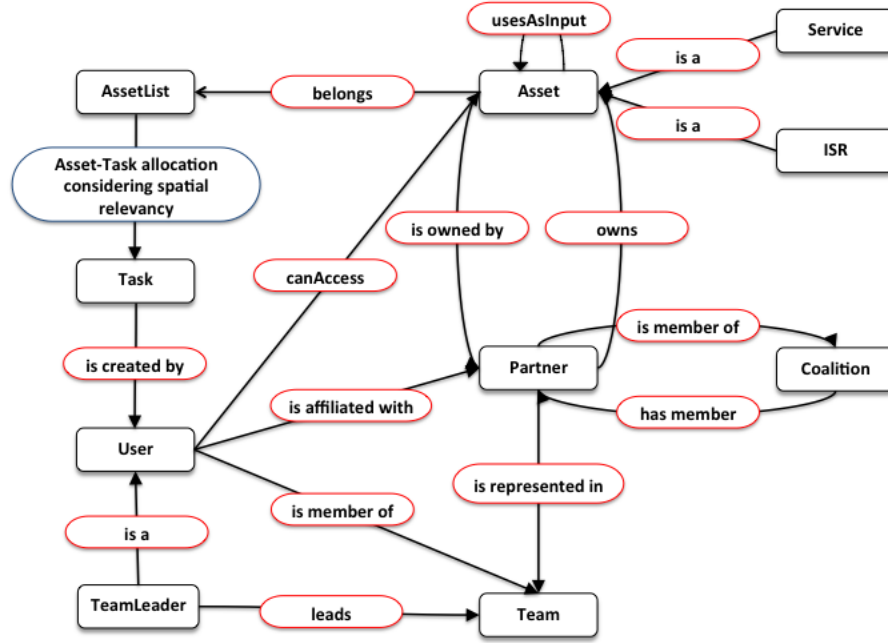


Fig. 2. Coalition assets management ontology.

there is a partner named UK.

the partner UK
 is represented in the team t1 and
 owns the asset a1 and
 is member of the coalition US-UK.

there is a user named u1 that
 has 'uid1' as userId and
 has 'intell1' as expertise.

there is an asset named a1 that
 has 'Acoustic' as inputType and
 has 'Acoustic' as outputType and
 has '100' as coverageRadius.

For simplicity we hide all the complex coverage relevancy and asset to task allocation processes into the blue relationship bubble in Figure 2.

VI. CE POLICY RULES

In this section we develop high-level, attribute-based policies expressed in CE. Our goal is to verify whether CE is expressive enough to capture a variety of ISR service management rules.

Research and development in policy technologies within International Technology Alliance⁴ (ITA) project has led to the development of the Policy Management Toolkit [14]. This toolkit was developed to perform a variety of management functions on sets of policies applicable to sensors, sensor platforms, and networks [15]. The developed policies regulate

aspects including platform control, sensor and system control, sensor information access control and information flow protection. We present policies that cover only some of the aspects above, but we have extensively tested CE's expressiveness in policy rules covering all the aforementioned spectrum. We present here a variety of authorization, spatio-temporal and input/output interoperability policy rules applied in the context of ISR coalition operations. The developed high-level policy rules use as building blocks different attributes of the concepts defined in the ontology of Figure 2. Attributes are sets of properties that are used in the ontology to describe concepts. Each rule consists of three grammatical blocks: a Subject (i.e. user, or asset), an Object and an Action that the Subject wants to perform on the Object (i.e. *canAccess*).

Suppose *U* is a user, *A* is an asset, *P* is a coalition partner, *T* is a team and additionally the properties *B* as an asset capability, *E* which is a user expertise and the asset location *C* we have the predicates:

canAccess(*U*, *A*) == true **if** user *U* can access asset *A*
isAffiliated(*U*, *P*) == true **if** user *U* is affiliated with partner *P*
isRepresented(*P*, *T*) == true **if** partner *P* is represented in team *T*
owns(*P*, *A*) == true **if** partner *P* owns asset *A*
member(*U*, *T*) == true **if** user *U* is member of team *T*
coverageRadius(*A*, *SM*) == true **if** asset *A* has coverageArea

⁴<https://www.usukitacs.com/>

= SM

expertise(U, E) == true **if** user U has E as expertise
usesAsInput(A, A') == true **if** asset A' uses as input asset A
cost(A, C) == true **if** asset A uses has C as execution cost
taskType(T, E) == true **if** task T has E as type

In order to help the understanding of the developed policy rules we express each of them (“Rule 1” - “Rule 4”) in three different ways: a plain text description, a formal definition rule using the predicates above and a CE-based representation.

Rule 1

If User U is member of team T they can access asset A if it is owned by partners P and partner P is represented in team T (i.e. team-based asset sharing).

```
canAccess(U, A) if  
  member(U, T)  $\wedge$  isRepresented(P, T)  $\wedge$  owns(P, A)  
if  
  (the user U is member of the team T)and  
  (the partner P is represented in the team T)and  
  (the partner P owns the asset A)  
then  
  (the user U canAccess the asset A)
```

Rule 2

If asset A has coverageRadius larger than zero and the User U is affiliated with partner P then user U canAccess asset A.

```
canAccess(U, A) if  
  coverageRadius(A, SM) > 0  $\wedge$  isAffiliated(U, P)  
if  
  (the user U is affiliated with the partner P)and  
  (the asset A has the value CR as coverageArea)and  
  (the value CR > '0')  
then  
  (the user U canAccess the asset A)
```

Rule 3

If user U has intell as expertise and asset A has energy cost less than 10 then user U canAccess asset A.

```
canAccess(U, A) if  
  expertise(U, E)  $\wedge$  cost(A, C) < 10  
if  
  (the user U has the value EX as expertise)and  
  (the value EX = 'intell')and  
  (the asset A has the value C as cost)and  
  (the value C < '10')  
then  
  (the user U canAccess the asset A)
```

Rule 4

If Asset A has coverage radius greater than 100m and tasks T has taskType acoustic then asset A' uses asset's A output as input

```
usesAsInput(A, A') if  
  coverageRadius(A, SM)  $\wedge$  taskType(T, E)
```

```
if  
  (the asset A has the value I as outputType and  
  (the value I = 'Acoustic')and  
  (the asset A' has the value I' as inputType and  
  (the value I' = 'Acoustic')and  
  (the asset A has the value CR as coverageArea)and  
  (the value CR > '100'))  
then  
  (the asset A' usesAsInput asset A)
```

The policy examples above show that CE is able to express policy rules based on different sets of conditions derived from concepts' properties and relationships. In particular, we show that CE can express policies that regulate access control of different subsets of users to different subsets of assets considering each time different sets of attributes derived from the conceptual model; or it can express different kind of policies considering spatial coverage and input/output interoperability issues. In essence the properties of the ontology's concepts are the factors that set the limits of CE's expressiveness as a policy language. The more interconnected the entities are (i.e. in terms of relationships between them) and the more properties they have, the more complex and more strict rules one can develop by combining them.

In section V we mentioned that when a CE rule is executed, a rationale which is a set of reasoning steps, each of which is defined as a “because” relation between conditions and the rules decision is automatically created. Below we present the created rationale after executing a simple rule on our ontology.

Rule 5

```
if  
  (the asset A is owned by the partner 'US') and  
  (the user U has the value EX as expertise) and  
  (the value EX = 'intell')  
then  
  (the user U canAccess the asset A)
```

Rationale

```
the user 'u1'  
canAccess the asset 'a2'  
because  
the constant 'intell' = 'intell' and  
the user 'u1' has 'intell' as expertise and  
the asset 'a2' is owned by the partner 'US'
```

It is worth noting that the policy grammar that we apply here for the development of policy rules is a very simple one. It can be easily extended according to the ontology and/or the system parameters that the CE-based policy language needs to control.

VII. CONCLUSION & FUTURE WORK

In this paper we proposed a model for configuring ISR services which considers spatial constraints on service selection, and configures the services by choosing low-cost and spatially relevant services improving the spatial relevancy of the overall system. Moreover, we used CE as a means to define a high-level policy representation, which is both human-friendly and directly processable by policy-checking

systems. We claimed that a high-level policy representation would help non-technical users operating near or at the edge of an organisation to create new and negotiate existing policies. We first defined an ontology to capture the entities of a multi-partner ISR coalition operation and then we developed and executed on this ontology a variety of attribute-based policy rules. We showed that the limits of expressiveness and flexibility of CE as a policy language are defined by the ontology's entities, properties and the relationships between them.

A major aspect of a policy-based system management is the policy negotiation/relaxation when services are not implementable given the set of policies currently in force. A necessary prerequisite for a policy negotiation system is the creation of a mechanism for policy conflict detection. In terms of future work we plan to develop a mechanism for policy conflicts detection among policies expressed in CE. Moreover, we plan to integrate the CE-based policy language with other well-known policy enforcement models such as WPML measuring its effectiveness and efficiency in terms of usability and system performance.

ACKNOWLEDGMENT

We thank Dave Braines from IBM UK for his input and support in developing the CE-based policy representation and Jon Bakdash of the US Army Research Laboratory for valuable discussions on the context for this work. This research was sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defence and was accomplished under Agreement Number W911NF-06-3-0001. The views and conclusions contained in this document are those of the author(s) and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

REFERENCES

- [1] S. Geyik, B. Szymanski, and P. Zerfos, "Robust dynamic service composition in sensor networks," *Services Computing, IEEE Transactions on*, vol. PP, no. 99, p. 1, 2012.
- [2] S. Kraus and O. Shehory, "Coalition formation with uncertain heterogeneous information," 2003.
- [3] R. Hayes and D. Alberts, "Power to the edge: Command and control in the information age," *CCRP*, 2003.
- [4] D. Mott, "Summary of controlled english," *ITACS*, 2010.
- [5] G. Tychogiorgos and C. Bisdikian, "Selecting relevant sensor providers for meeting your quality information needs," in *Mobile Data Management, 2011 12th IEEE Int'l Conf. on*, 2011.
- [6] A. Simon and M. James, "The agile organization," *CCRP*, 2005.
- [7] D. Alberts, R. Huber, and J. Moffat, "Nato nec c2 maturity model," *CCRP*, 2010.
- [8] R. Schwitter, "Controlled natural language as interface language to the semantic web," 2005, pp. 1699–1718. [Online]. Available: <http://www.scopus.com/inward/record.url?eid=2-s2.0-38349020074&partnerID=40&md5=9c9cf8a3cdb4575d1838dc574a214048>
- [9] M. Tilbrook and R. Schwitter, "Controlled natural language meets the semantic web," in *Australasian Language Technology Workshop 2004*, vol. 2, 2004.
- [10] A. Bernstein and E. Kaufmann, "Gino - a guided input natural language ontology editor," 2006.
- [11] G. Hart, M. Johnson, and C. Dolbear, "Rabbit: Developing a control natural language for authoring ontologies," 2008.
- [12] T. Kuhn, "An evaluation framework for controlled natural languages," 2010.
- [13] P. Xue, D. Mott, D. Braines, S. Poteet, A. Kao, and C. Giammanco, "Information extraction using controlled english to support knowledge-sharing and decision-making," 2012.
- [14] T. Pham, G. Pearson, F. Bergamaschi, and S. Calo, "The ita sensor fabric and policy management toolkit," 2011.
- [15] G. Pham, T. Cirincione, "Sensor data and information sharing for coalition operations," 2012.