Interest-based Negotiation for Asset Sharing Policies

Christos Parizas[1], Geeth De Mel[2], Alun D. Preece[1], Murat Sensoy[3]
Seraphin B. Calo[2] and Tien Pham[4]

[1] School of Computer Science and Informatics, Cardiff University, UK
{C.Parizas, A.D.Preece}@cs.cardiff.ac.uk
[2] IBM T.J. Watson Research Center, Yorktown Heights, NY, USA
{grdemel, scalo}@us.ibm.com
[3] Department of Computer Science, Ozyegin University, Istanbul, Turkey
murat.sensoy@ozyegin.edu.tr
[4] US Army Research Lab, Sensors & Electron Devices Directorate,
Adelphi MD, USA
tien.pham1.civ@mail.mil

Abstract. Resource sharing is an important but complex problem to be solved. The problem is exacerbated in a coalition context due to policy constraints placed on the resources. Thus, to effectively share resources, members of a coalition need to negotiate on policies and at times refine them to meet the needs of the operating environment. Towards achieving this goal, in this work we propose a novel policy negotiation mechanism based on the interest-based negotiation paradigm. Interest-based negotiation promotes collaboration when compared with more traditional negotiation approaches such as position-based negotiations.

1 Introduction

Negotiation is a form of interaction usually expressed as a dialogue between two or more parties with conflicting interests that try to achieve mutual agreement about the exchange of scarce resources, resolve points of difference and craft outcomes that satisfy various interests. In order to cooperate and search for mutual agreements, the involved parties make proposals, trade options and offer concessions. The automation of the negotiation process and its integration with autonomic, multi-agent environments has been well-researched over the last few decades [1, 2].

The approaches for automating negotiation can be classified into three major categories : (1) game theoretic (2) heuristic, and (3) argumentation based [2]. The first two approaches represent traditional bilateral negotiation protocols wherein each negotiation party exchanges offers aiming to satisfy their own interests. These are called position-based negotiations (PBN). In these approaches the participants attack the opposing parties' offers and try to convince them for the suitability of their own offers. Typically these approaches are formalized as search problems in the space of possible deals by focusing on negotiation objectives. Argumentation-based negotiation (ABN) has been introduced as a means to enhance automated negotiation by exchanging richer information between negotiators. Interest-based negotiation (IBN) is a type of ABN where the agents exchange information about the goals that motivate their negotiation [3,4]. IBN unlike PBN tackles the problem of negotiation by focusing on why negotiate for rather than on what to negotiate for and aims to lead negotiating parties to win-win solutions.

Multi-party teams are formed to support collective endeavors which otherwise would be difficult, if not impossible, to achieve by a single party. In order to support such activities, resources belonging to collaborating partners are shared among the team members; mechanisms to share resources in this context are actively and broadly explored in the research community. This is due to the impact that different sharing modifications (what to share, with who, when and under what conditions) can bring into the collaboration, with respect to domains such as security, privacy and performance to name only a few. Consider the following: a) crisis management situations where responders affiliated with different national and organizational groups form coalitions and share resources (e.g. sensors, network connectivity and data storage) in an ad-hoc manner, in order to provide humanitarian assistance; b) resource sharing in corporate environments such as the recent *MobileFirst*[5] partnership between IBM and Apple where cloud and other services are shared in a daily basis; or c) a short-lived opportunistic mobile network comprised of a few peer members, established for message routing or data sharing. In all these cases access control mechanisms that specify resource sharing need to be implemented. A suitable mechanism for managing access control on resources of such systems is the Policy-based Management System (PBMS).

This work presents a framework for enabling authorization policy negotiation in multi-party, cooperative and dynamic environments. This framework is aimed at policy makers who are not necessarily experts in IT or negotiation techniques, responsible for modifying policies responding to situational changes. To the best of our knowledge there is no mature work done on policy negotiation, while the vast majority of negotiation work in multi-agent environments: a) utilizes PBN approaches and b) invariably ignores the special characteristics of multi-party, collaborative environments. In this work we propose a novel, interest-based policy negotiation framework. It is our belief that by understanding the negotiating parties interests and crafting options that can meet their requirements, IBN could provide a negotiation mechanism which promotes good collaboration unlike PBN, which creates an adversarial negotiation atmosphere. Moreover PBN with its fixed, opposing positions is a cumbersome negotiation method to cope with dynamic environments [2]. The proposed negotiation framework can operate in parallel to a PBMS. It considers an approach that proposes modification of strict policies, in order to maximize overall usability of collab-

[5] http://www.ibm.com/mobilefirst/us/en/

orating assets while remaining faithful to existing authorization policies. The main contributions of this work are as follows:

- definition of an interest-based authorization policy negotiation model
- specification of an architecture for its integration with PBMS
- demonstration of its application on a user friendly policy representation
- presentation of a walkthrough for its execution utilizing a policy negotiation scenario

The remainder of the paper is organized as follows: in Section 2 we discuss previous literature on policy negotiation approaches and in Section 3 we present a walkthrough of a policy negotiation scenario. Section 4 describes the policy negotiation framework, the policy language, and its interface to PBMS by means of an architectural overview. Section 5 presents the algorithmic steps for IBN achievement through policy refinement. We conclude the document in Section 6 by summarizing our contribution and outlining future research directions.

2 Related Work

The first computer applications for supporting bilateral negotiations were developed in late 1960s [5]. The reason for their emergence was to assist human negotiators to overcome weaknesses related to negotiation process such as cognitive biases, emotional risks, and their inability to manage complex negotiation environments. Although there is rich literature on negotiation protocols in autonomous, multi-agent environments, there is very limited and no mature work done on policy negotiation. We see the role of policies in managing large, complex and dynamic systems as of a high importance and the existence of sophisticated ways to do so imperative. We believe that the integration of an effective negotiation mechanism on a PBMS works towards this direction. Moreover, no work had previously attempted to bring the IBN paradigm into policy negotiation.

The authors of [6] present requirements of policy languages which deal with trust negotiation and focuses on the technical aspects and properties of trust models to effectively evaluate access requests. It does not depend on any aspects of policy negotiation and the scenarios it deals with are less dynamic compared to our problem domain. [7] proposes an architecture that combines a policy-based management mechanism for evaluating privacy policy rules with a policy negotiation roadmap. The work is very generic and does not provide clear evidence of any effectiveness of the proposed approach, while lacking any evaluation. [8] is one of the first works that looks into policy negotiation and covers the area in depth. It also looks into collaborating environments and introduces the notion of ABN in policy negotiation. However it focuses on a very specific application domain in which it deals with writing insurance policies while maintaining a common and collaborative knowledge base.

The work discussed on [9] has several similarities to our work; it deals with cooperating environments and a PBMS is employed in support of service composition in a distributed setting. The authors have used a negotiation framework to effectively compose services. Its main difference with the work proposed herein is that the objective of the negotiation performed in [9] is the services that are managed by policies, not the policies themselves. We believe that in order to decrease the management overhead the objective of negotiation should be the policies. This is because policies are the core of PBMS and the logical component where the systems management resides. Finally, [10] proposes a policy negotiation approach and presents its architecture. It lacks any effectiveness evaluation while it does not consider either multi-partner, dynamic environments or ABN and IBN paradigms.

3 Interest-based Policy Negotiation Scenario

Below we provide illustrative scenarios to motivate the use of IBN in policy negotiation in resource sharing situations. In Subsection 3.1 we revisit the classic orange scenario discussed in best-selling *Getting to YES* [11] and then expand it to a mobile resource sharing scenario in Subsection 3.2.

3.1 The Chefs-Orange Scenario

Two chefs who work in the same kitchen both want to use orange for their recipes. Unfortunately there is only one orange left in the kitchen. Instead of starting negotiating on who is going to get the orange (as in a PBN, zero-sum approach), the two chefs opt to follow the IBN approach. Thus, they ask each other why they need the orange. In other words they try to better understand their underlying goals of using the orange. Answering the why question it turns out that one chef needs only the oranges flesh (to execute a sauce recipe) while the other needs only its peel (for executing a dessert recipe) and so they share the orange accordingly achieving a win-win negotiation outcome.

3.2 Authorization Policy Negotiation

An individual P2 wants to access a smartphone device SMD owned by an individual P1. However, P1 has a set of restrictions which are captured by policy set R on how to share SMD with other people. These restrictions may reflect privacy concerns (e.g. by accessing their smartphone one can have access to their photos), security, and so forth. For the sake of clarity, in this example, we assume that the set R contains the following policy constraint R1: do not share the device SMD with anyone else but its owner P1. When P2 asks for permission to use the physical device SMD, R1 prohibits this action. Ostensibly there is little room for negotiation here, if one follows a PBN approach with the current set of policies.

However by applying IBN and trying to understand the underlying interests of the involving parties, we believe the situation could be handled in a satisfactory manner for both parties. For example, asking the why question it turns out that P2 needs a data service (as opposed to the physical device) in order to execute the task *Email submission* and P1 does not mind sharing a data connection as a hotspot with a trusted party; if P1 could get to know why P2 needs the device, the situation could be solved to the satisfaction of both parties. All an IBN mechanism needs to do in this case is to introduce another policy – actually a refinement of the existing policy – to R1 to say that data service can be shared among trusted parties. Thus, we argue that in such cases by understanding the situation and broadening the space of possible negotiation deals, one can reach a win-win solution.

The intuition behind ABN is that the negotiating parties can improve the way they negotiate by exchanging explicit information about their intentions. This information exchange reveals unknown, non-shared, incomplete, and imprecise information about the underlying attitudes of the parties involved in the negotiation [12]. As stated earlier, IBN is a type of ABN where the negotiating parties exchange information about their negotiation goals, which then guide the negotiation process. Thus, the why party of the intention is of major importance when compared with the what part. Finally, we would say that the IBN is more of a negotiation shortcut method rather than a typical negotiation process. By attacking the problem of negotiation, IBN skips the proposals making, the options trading and the need for negotiating parties to offer concession as in PBN cases. In the next section, we shall introduce our IBN-based policy framework and provide our intuition behind the approach.

4 Interest-based Policy Negotiation Framework

The design and development of frameworks for establishing negotiation needs to achieve some desirable outcomes that are secured by meeting a set of systematic properties: guaranteed success (i.e., negotiation protocol that guarantees agreement), simplicity (i.e., easy for the optimal decision to be determined by participants), maximizing social welfare (i.e., maximization of the utilities sum of negotiation participants) to name a few [13]. The main objective of the negotiation framework we propose is to maximize social welfare.

In environments that often suffer from asset scarcity (demand exceeds supply), and many tasks may be competing for the same resource like the ones described in Section 1, paragraph 3, the formation of coalitions offers alleviation by bringing more resources to the table. The relationships between coalition parties in those scenarios are mostly peer-to-peer (P2P). However, we do not assume fully cooperative scenarios. Partners often pursue cooperation but they do not want to share sensitive intelligence that can deliver greater value to the opponents [14]. In the literature this kind of relationship model, where parties have cooperative and competitive attitudes from time to time, is called coopetition [15]. The PBMS and its sets of policies is in charge here, playing a regulative role in order to keep balance between asset sharing and asset "protection".

The more strict the partners' policies are, the higher the barriers towards collaboration are set. This is where the IBN mechanism comes in, trying to lower these barriers in order to establish better collaboration trough asset sharing (i.e., increase overall the number of executed tasks and thus increase the social welfare) while maintaining the compromise from the asset owners point of view at the same levels.

The framework presented herein allows negotiation on policies with minimal human intervention. In traditional system management, policies associated with PBMS are static (or rarely change); these systems, however, fail miserably in dynamic environments where policies need to adopt according to situational changes. We note that it is not prudent to assume human operators in these environments can effectively be on top of every change to manage PBMS(s) effectively; they require automated assistance.

Summarizing its contribution, the IBN negotiation framework considers a cooperative negotiation approach which modifies strict policies aiming to a) maximize social welfare by increasing the overall usability of collaborating assets while b) remaining faithful to existing authorization policies, maintaining their core trends. Utilizing such a tool, a multilateral policy transformation can be achieved considering multi-party input and criteria for the benefit of the coalition. Each negotiation session considers sets of two negotiators (bilateral negotiation approach). The issue that needs to be settled during any negotiation process is the granting (or not) of access to non-sharable assets. From that perspective the framework deals with single-attribute negotiations.

4.1 Policies Under Negotiation

Several policy-based management systems that utilize different policy languages have been proposed in the literature. KAoS is a management tool for governing software agent behavior in grid computing using an ontological representation encoded in OWL [16]. Ponder is an object-oriented policy language used for managing systems and networks [17] while XACML is the OASIS standard access control policy language for web services [18]. The proposed policy negotiation framework is applied on authorization policies expressed in the Controlled English (CE) policy language [19]. CE policy language is an ontological approach that uses a Controlled Natural Language (CNL) for defining a policy representation that is both human-friendly (CNL representation) and unambiguous for computers (using a CE reasoner) [20]. CE is used to define domain models that describe the system to be managed. The domain models take the form of concept definitions and comprise objects, their properties, and the relationships among them. These domain model components are the building blocks of the attribute-based CE policy language.

Each policy rule follows the if-condition(s)-then-action form and consists of four basic grammatical blocks as shown below:

- **Subject:** specifies the entities (human/machine) which interpret obligation policies or can access resources in authorization policies
- **Action:** what must be performed for obligations and what is permitted for authorization

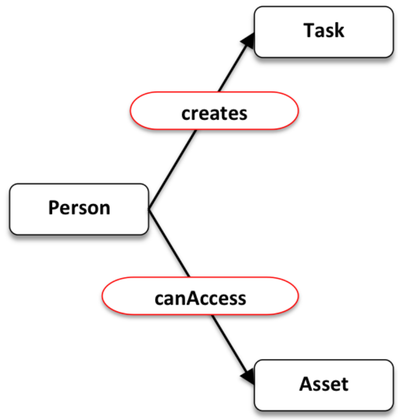

Fig. 1. Authorization Policy Negotiation Scenario: Domain Model

– **Target:** objects on which actions are to be performed
– **Constraints:** boolean conditions

The utilization of CE here is two-folde. It does not only is the user friendly formal representation of the system to be managed but also helps decision makers who lack technical expertise to understand in a more transparent way the complexities associated with policy negotiation. Figure 1 provides a graphical depiction of the CE-based domain model, which describes the smartphone access scenario of Section 3.2 , while the CE representation of policy R1 is shown below.

```
Policy R1
If
( there is an asset A named SMD ) and
( there is a person P named P1 )
then
( the person P canAccess the asset A )
.
```

4.2 The IBN in Asset Sharing process

The role of policies in managing a system is to guide its actions, towards behaviors that would secure optimal systems outcomes. Authorization policies manage actions of both, hard (sensing devices, distributed databases, smartphone devices) and soft resources (human-in-the-loop asset owners/requestors). Different users have different rights, relationships and interests in regards to deployed resources. Non-owner users want to gain access to the resources in order to serve their tasks needs, while owners want to protect their resources from unauthorized use. There is a monopolistic asset usage case. The proposed approach considers both concerns in a single mechanism providing a framework that pursues a win-win negotiation outcome for any sets of negotiators. In other words, it tries through negotiation to redefine what is a suboptimal system outcome given: a) the currently-deployed resources and b) the tasks needs of the system that is managed.

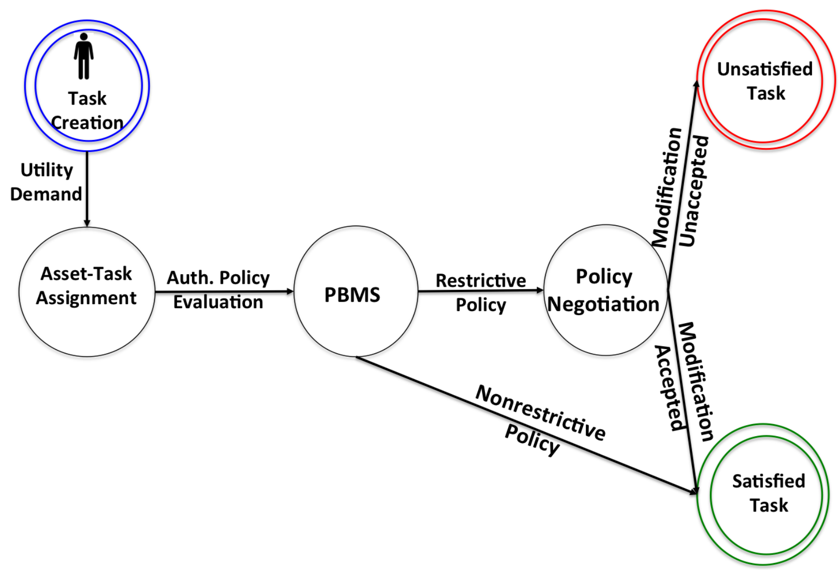

Fig. 2. Interest-based policy negotiation and task implementation

The finite state diagram of Figure 2 provides a depiction of the role the policy negotiation framework plays in the tasks implementation of collective endeavors. The human task creator, wanting to serve their appetite for information, creates tasks, which require a utility demand. The asset-task assignment component is in charge of optimizing the task utility by allocating the appropriate resources (information-providing assets) to each task. The PBMS component is responsible then for evaluating and enforcing authorization policies developed by multi-party collaborators. In the case of a non-restrictive authorization policy the task creator gets their task served. If the policy rule is restrictive, the policy negotiation component takes over. It modifies the policy rule accordingly, and passes it to the asset owner for confirmation. Depending on the asset owners decision the task is either satisfied or unsatisfied.

4.3 IBN Enabled PBMS

The policy negotiation framework can be integrated into a PBMS as a plug-in, enabling negotiation in policy enforcement process. A PBMS, as defined by standards organizations such as IETF and DMTF, consists of four basic components as shown in Figure 3: a) the policy management tool, b) the policy repository, c) the policy enforcement point (PEP), and d) the policy decision point (PDP) [21]. The policy management tool is the entry point through which policy makers define authorization policies to be enforced by the system. The policy repository is the component where the policies generated by the management tool are stored (step A1). PEP is the logical component that can take actions on enforcing the policies. Given the access request conditions, the PEP contacts PDP (step A2), which is then responsible for fetching the necessary polices from the policy repository (step A3, A4), evaluates them and decides which of them need to be enforced on PEP (step A5).

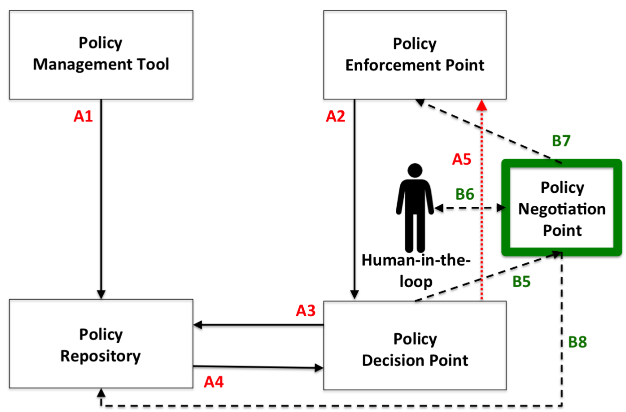

Fig. 3. IBN extended PBMS

In addition to the four basic PBMS elements, Figure 3 also includes a human-in-the-loop element, representing the roles played by the asset requestor and owner in the negotiation process. The additional component where the IBN framework resides is called the Policy Negotiation Point (PNP) and lies between the PEP and PDP, interfacing also with the human-in-the-loop element. As mentioned before, the PNP is triggered to attempt to modify authorization policies when a user creates a task that cannot be served due to restrictive policies. The dashed lines show optional communication between the components which is only established when a policy negotiation incident occurs. The red numbered parts of the figure (flow paths which are prefixed by As) describe the PBMS operational flow, while the green parts (flow paths which are prefixed by Bs) replace step A5 (red, dotted line) with the policy negotiation extension. Note that the separation between the components can be only logical when they reside in the same physical device. When PNP detects a restrictive policy (step B5) it modifies it following the steps described at the following section and passes it to the asset owner for confirmation (step B6). If the asset owner confirms the replacement, the proposed policy is then enforced on the PEP (step B7) and it is also stored in the policy repository replacing its predecessor (step B8). Otherwise step A5 is executed as before.

5 Achieving IBN through Policy Refinement

In general, negotiation protocols contain the set of rules that manage the interaction between negotiating parties [2]. These rules define who is permitted to participate in the negotiation process and under what conditions (i.e. negotiating and any non-negotiating third parties). The rules also manage the participants actions throughout the process. In addition they define the decision of the negotiators towards the proposals.

The negotiating parties in our scenario as mentioned before are essentially decision makers who generally lack negotiation expertise. Thus the IBN mechanism tries to take, as much as possible, the negotiation weight off their shoulders rather than providing them the means for making proposals and trade options themselves. However, it does not exclude them completely from the negotiation process as in fully automated models. To achieve such behavior it simply applies the IBN principles described in Chefs-Orange scenario of Section 3.1, exploiting the domain models semantics, the semantics of the polices and the seamless relation between them as they both share the same CE representation.

The objective of the negotiation is the restrictive policies themselves. Asking the why question like in Chefs-Orange scenario to the requestor side, the PNP gets as a reply the reason why they need the asset for. Asking the why question to the asset owners/policy authors side, it gets the reasons why they do not want to grant access to their assets respectively. The prerequisite for the PNP here is to have full and accurate knowledge of the managed system. This is achieved by having unlimited and unconditional access to both domain model and policies. Unlike the majority of the proposed PBN approaches, the human-in-the-loop negotiators in our case are ignorant of the preferences of their opponents, while their knowledge in terms of the domain model reaches only the ground of their own expertise.

Utilizing CE for the formal representation of the environment to be managed, and as the language for expressing policies, the IBN, human-machine communication (i.e. communication between PNP and non-IT expert negotiators) for exchanging information regarding the negotiation is a transparently achievable task. The CE human-machine communication has been described in previous work [22]. However, trying to automate as much as possible the negotiation process, the why question is rather rhetorical here. In the requestors case the answer to the why question is quite simple and straightforward and the PNP is aware of it just by looking at the domain model. The asset requestor clearly wants to access the asset in order to execute their task. Hence, a desired negotiation outcome as far as the requestor is concerned, is the derivation of a policy that has them included in the set of *Subject* policy block with positive access (i.e., canAccess) *Action* to a *Target* set that includes the prohibited asset capable of serving their task's needs.

Inferring the answer to the why question from the asset owners side for understanding their interests and broadening the negotiation space is a more challenging task. In general any application of authorization systems aims to specify access rights to resources. Thus, a simple answer would be including the reasons why they want to decline access rights to their own resources. Looking carefully at the policy, these reasons are basically described from the policys *Constraints* block. The policy R1 of Section 4.1 is rather a simple one referring deliberately to a simple scenario and this might not be easily inferred. Considering other more complex policy rules with several conditions describing constraints such as the age of the requestor or their expertise this is easier inferred.

However this is not exactly the answer to the why question we are looking for here. Considering the policies as the means for guiding systems actions towards behaviors to achieve optimal outcomes, the *Constraints* policy block refers to the actions level of the policy. Our focus here is on the higher level, this of the systems behavior. Focusing on a higher level, gives us the agility to find different policies as far as the actions is concerned, that provides the same functionality in terms of behavior; and the different policies we are looking for are those which serve the needs of the asset requestors as well. Achieving this goal we achieve a win-win negotiation outcome like the one described in Chefs-Orange scenario. The next four steps describe the process to reach such an outcome.

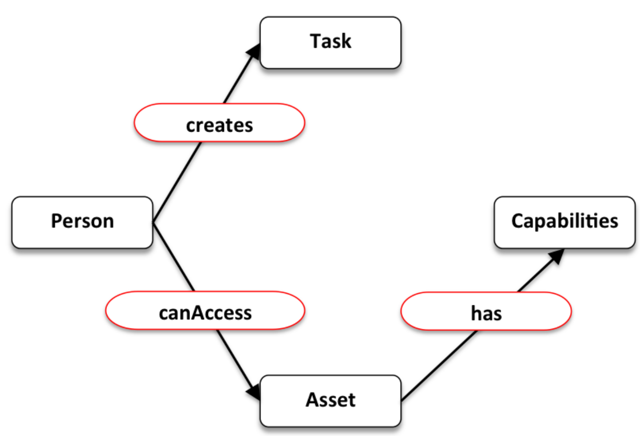

Fig. 4. Ontology Modification: Step one

Step 1: The simplistic domain model of Figure 1 presents only the concepts involved in the smartphone scenario of Section 3.2 and their relationships. It hides however their properties. Assume that the concept Asset has a property named *Provided capability* and that the Assset instance named SMD has the

Provided capability property named *Tethering*. Thus, the policy R1 by denying access to SMD, it denies access to any of SMDs provided capability as well. The IBN process starts taking as input the policy's *Target* block first. Trying to broaden the negotiation space in order to find alternative policies that satisfy both negotiators it separates the SMD from its Provided capability property and updates accordingly the ontology as shown in Figure 4 generating the respective CE sentences. The concept Capability and the respective relationship between Asset and Capability is now created.

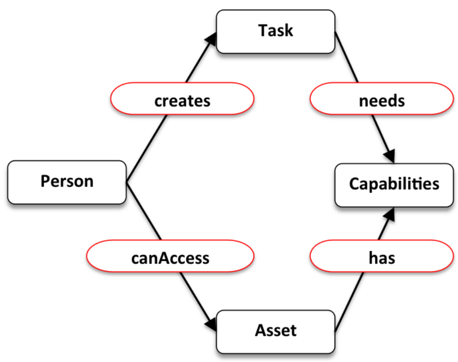

Fig. 5. Ontology Modification: Step two

Step 2: Each Task of Figure 1 requires a set of capabilities in order to be served. The concept Task has a property named *Required capability* and the Task instance *Email submission* has a number of required capabilities including that of *Tethering*. The second step of IBN process gets as input the Task and separates it from its Required capability property and updates accordingly the ontology as shown in Figure 5 generating the respective CE sentences as in Step 1.

Step 3: Often the tasks' capability needs might span outside the capabilities offered by one particular asset (e.g., a task might need to utilize capabilities provided by a number of assets). The IBN process, taking input from the previous two steps makes the matching between Asset's provided capabilities and Task's required capabilities. It matches this way the subset of the prohibited SMD's properties that are needed for the implementation of the desired Task and updates accordingly the ontology as shown in Figure 6 generating the respective CE sentences as in Step 1. The asset requestor now can access a subset/subsystem of asset that of its provided capability.

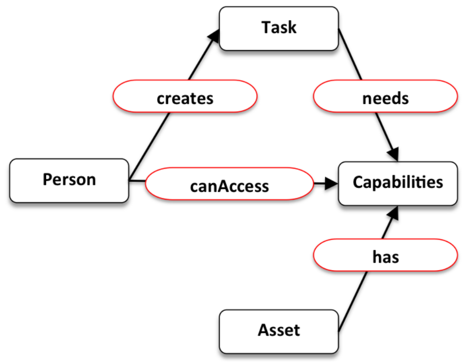

Fig. 6. Ontology Modification: Step four

Step 4: This step performs the *policy refinement*[6]. The asset requestor (i.e., P2) is the *Subject* block of the refined policy, which has as *Action* block a positive authorization action (i.e., canAccess) and its *Target* block contains, the provided by the prohibited Asset and required by the desired Task Capability (i.e., Tethering). The CE refined policy *R1-Refined* below is passed then to the asset owner for approval.

```
Policy R1-Refined
if
  ( there is a capability C named Tethering ) and
  ( there is a person P named P2 )
then
  ( the person P canAccess the capability C ).
```

The asset owner P1 is in charge of confirming or not the replacement of policy R1 from the proposed policy R1-Refined. In the case of confirmation the refined policy is then enforced on SMD providing access to SMD's tethering capability, and is also stored in the policy repository replacing its predecessor. The successful completion of IBN leads the negotiating parties to a win-win negotiation, with the asset requestor getting the task of *Email submission* served and the asset owner prohibiting any physical access to SMD.

6 Conclusion & Future Work

In summary the proposed IBN framework provides an effective policy negotiation mechanism for revising asset sharing policies in dynamic, multi-party environments. The framework is seamlessly interfaced with standardized PBMS and it

[6] Note that the term policy refinement herein refers to a different process than the policy refinement in [23], which describes the process of interpreting more general, business layer policies to more specific, system layer ones.

provides means to directly negotiate with policies. Our belief is that this is an important feature to have as PBMS is where the core components of the systems management logic resides. Moreover the IBN approach fits in multi-party environments where collaboration is promoted to achieve mutually satisfactory negotiation outcomes. Finally, utilizing CE-based policies in the framework eases the burden of the non-technical user in managing the PBMS and negotiate on them. As for the future research, there are plans for extending the IBN steps with regards to broadening the negotiation space considering components such as the users and the tasks of the system to be managed. In addition we plan to evaluate the proposed policy negotiation framework a) by conducting human-lead experiments, and b) by running simulations and comparing the results with respect to PBN approaches, especially in collaborative setting.

Acknowledgment: This research was sponsored by the US Army Research Laboratory and the UK Ministry of Defense and was accomplished under Agreement Number W911NF-06-3-0001. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the US Army Research Laboratory, the US Government, the UK Ministry of Defense or the UK Government. The US and UK Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

References

1. D. Walton and E. Krabbe, "Commitment in dialogue: Basic concepts of interpersonal reasoning." *Albany, New York: SUNY Press*, 1995.
2. N. Jennings *et al.*, "Automated negotiation: Prospects, methods and challenges," *Group Decision and Negotiation*, vol. 10, no. 2, pp. 199–215, 2001.
3. R. Fisher and W. Ury, "Getting to yes: Negotiating agreement without giving in," *New York: Penguin Books*, 1983.
4. P. Pasquier *et al.*, "An empirical study of interest-based negotiation," *Autonomous Agents and Multi-Agent Systems*, vol. 22, no. 2, pp. 249–288, 2011.
5. M. Delaney, A. Foroughi, and W. Perkins, "An empirical study of the efficacy of a computerized negotiation support system (nss)," *Decision Support Systems*, vol. 20, no. 3, pp. 185–197, 1997.
6. K. Seamons *et al.*, "Requirements for policy languages for trust negotiation," *3rd International Workshop on Policies for Distributed Systems and Networks, POLICY 2002*, pp. 68–79, 2002.
7. I. Jang, W. Shi, and H. Yoo, "Policy negotiation system architecture for privacy protection," *Proceedings - 4th International Conference on Networked Computing and Advanced Information Management, NCM 2008*, vol. 2, pp. 592–597, 2008.
8. M. Comuzzi and C. Francalanci, "Agent-based negotiation in cooperative processes: Automatic support to underwriting insurance policies," *ACM International Conference Proceeding Series*, vol. 60, pp. 121–129, 2004.
9. J.-F. Tang and X.-L. Xu, "An adaptive model of service composition based on policy driven and multi-agent negotiation," *Proceedings of the 2006 International Conference on Machine Learning and Cybernetics*, vol. 2006, pp. 113–118, 2006.

10. V. Cheng, P. Hung, and D. Chiu, "Enabling web services policy negotiation with privacy preserved using xacml," *Proceedings of the Annual Hawaii International Conference on System Sciences*, 2007.
11. R. Fisher *et al.*, *Getting to yes: Negotiating an agreement without giving in; 2nd repr. ed.* London: Random House Business Books, 1999.
12. I. Rahwan *et al.*, "Argumentation-based negotiation," *Knowledge Engineering Review*, vol. 18, no. 4, pp. 343–375, 2003.
13. T. Sandholm, "Distributed rational decision making," *Multiagent Systems (ed. G. Weiss) MIT Press*, pp. 201–258, 1999.
14. C. Loebecke, P. C. Van Fenema, and P. Powell, "Co-opetition and knowledge transfer," *SIGMIS Database*, vol. 30, no. 2, pp. 14–25, 1999.
15. A. Brandenburger and A. Nalebuff, "Co-opetition," *New York: Doubleday*, 1997.
16. A. Uszok *et al.*, "Kaos policy management for semantic web services," *IEEE Intelligent Systems*, vol. 19, no. 4, pp. 32–41, 2004.
17. N. Damianou, N. Dulay, E. Lupu, and M. Sloman, "The ponder policy specification language," *In proceedings of Workshop on Policies for Distributed Systems and Networks*, 2001.
18. OASIS. (2013) extensible access control markup language (xacml) version 3.0. [Online]. Available: http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.pdf
19. C. Parizas, D. Pizzocaro, A. Preece, and P. Zerfos, "Managing isr sharing policies at the network edge using controlled english," *In proceedings of Ground/Air Multisensor Interoperability, Integration, and Networking for Persistent ISR IV*, 2013.
20. D. Mott, "Summary of controlled english," *ITACS*, 2010.
21. R. Yavatkar, K. Pendarakis, and R. Guerin. (2000) A framework for policy-based admission control. [Online]. Available: http://www.ietf.org/rfc/rfc2753.txt
22. A. Preece, D. Braines, D. Pizzocaro, and C. Parizas, "Human-machine conversations to support mission-oriented information provision," *Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM*, pp. 43–49, 2013.
23. A. Bandara, E. Lupu, J. Moffett, and A. Russo, "A goal-based approach to policy refinement," *Proceedings - Fifth IEEE International Workshop on Policies for Distributed Systems and Networks, POLICY 2004*, pp. 229–239, 2004.