Roles Considered Harmful in Policy-based Management for Dynamic Organisations

Kevin Feeney, David Lewis, Vincent Wade
Centre for Telecommunications Value Chain Research and the Knowledge and Data Engineering Group
Trinity College Dublin
Dublin, Ireland
{Kevin.Feeney | Dave.Lewis | Vincent.Wade}@cs.tcd.ie

Abstract—Using roles for modeling organizations has become common in commercial policy based access control systems and widely accepted in policy-based management research for the grouping of policies. In this paper we argue that the role abstraction is inflexible in the face of many forms of organizational change and thus only an appropriate abstraction for mostly static organizational structures. We describe a novel policy grouping abstraction based upon communities. We ground the community-based approach through an application to dynamic spectrum access.

Keywords: Policy-based Management, roles, Communities, Spectrum Management

I. INTRODUCTION

In this paper, we start with the goal that Policy-Based Management (PBM) systems should be able to incorporate changes in the real world organizations which use them without requiring the re-programming of the system. Changes in the requirements of organizations should be accommodated by changes to policy rules rather than changes to software. Organizations are increasingly dynamic entities with changing requirements and PBM systems should be able to model these changes and enforce any of the consequent requirements by updating rules rather than modifying the functionality of components. By accommodating as much of the variability of requirements as changes to rules rather than changes to the software, the cost of managing and integrating changes into the system - implementing organizational change - should be minimized.

In reflecting organizational requirements, PBM must integrate information systems into human organizations so as to accommodate the methods, processes and organizational structures of the human organization rather than vice versa. PBM systems should embody the accumulated expertise of the organization, i.e. the shared understanding of the strategic and operational behavior of the organization. In assessing the ability of PBM systems to meet these goals we must evaluate:

- the comprehensiveness with which a PBM system can reflect the organizational structure and behavioral rules of the organization it serves;
- the ease with which this reflected model can change as the organization experiences change

- the ability of the policy-authoring process to itself be managed in accordance with organizational needs.

In this paper we analyze *roles* as the state of the art in grouping abstractions used in PBM systems to models of organizational structure and associated operational behavior. We identify problems with the role abstraction when dealing with dynamic organizations and present details of the *community* abstraction which is specifically aimed at modeling dynamic organizations. We then provide a brief example of how our implementation is being used to allow dynamic policy-based management of spectrum resources for wireless networks.

II. FROM MANAGEMENT GOALS TO ORGANIZATIONAL GOALS

It can be assumed that most Policy Based Management Systems will be deployed within organizations in order to fulfill organizational goals rather than being deployed to achieve the goals of the individuals who manage them. There may be many individuals with various different roles within the organization who act as human managers of information systems in some respect and their management requirements may be contradictory. Furthermore, the organizational goals at the highest level may contradict the goals of individual managers, who operate within different areas of an organization and work primarily to achieve their local goals which may not align with the global goals of the organization. Although this area has been acknowledged as problematic [1], most research into the problem of mapping management goals to concrete policies has implicitly assumed that management goals are themselves non-contradictory and that it is possible to capture a set of management goals which can be refined into a set of non-conflicting policies. Research in organizational theory [2] strongly suggests that this is not the case. The challenge is therefore to align the policy engineering process to the flow of authority and degrees of autonomy in an organization so that policies more naturally reflect organizational goals and can respond to changes to organizational structure. Management requirements will vary across the organization and the goals of some organizational units will conflict. In such cases, it is only by referencing the higher-level goals of the organization that a resolution can be reached.

Another major problem for policy based management systems stems from the complexity of deploying them. Most current approaches depend upon a difficult and intensive phase of requirements analysis prior to deployment, using methodologies borrowed from requirements engineering. In [3], an attempt is made to develop a role-modeling framework which captures much of the complexity of real world organizations, by modeling roles under a number of different criteria. However, there is no evidence that such approaches, even where complex models are used, works in practice. The complexity of this task is not surprising, however, when the approach is looked at through the lens of the organizational theory. The specification of policies is carried out by a group of experts who attempt to translate the various management goals of the organization into an exhaustive set of concrete policies. This policy set is thereafter maintained by expert administrators who can modify it, but significant modifications will themselves require significant analysis before they can be deployed. This is a decidedly Taylorist [2] approach to organizational modeling and it is thus no surprise that such systems are difficult to integrate into real world organizations. In one of the few published articles that examined the problems of deploying PBM solutions in practice, Michael Jude noted that:

"Policy-based network management (PBNM) turned out to be difficult to put into practice. Early adopters have found that developing and deploying policies is not simple, cheap or quick. Instead, PBNM has been a time-intensive, complex, expensive process. Additionally, it has demanded that the enterprise organization mutate to match the technology-rather than the technology meeting the enterprise's management needs." [4].
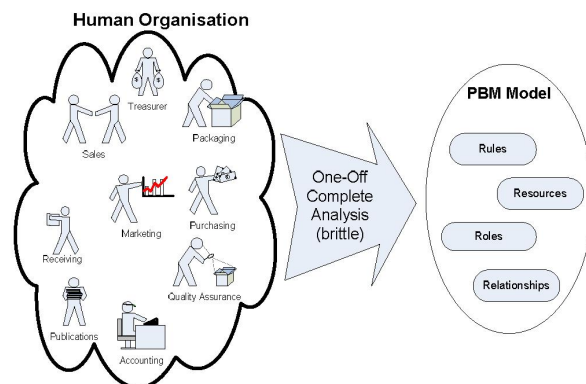

Figure 1 The Brittle Nature of One-Off Requirements Engineering for PBM

PBM systems should be able to incorporate changes in the real world organizations which use them without requiring the re-programming of the system. Changes to the requirements of the organization should be accommodated by changes to policy rules rather than changes to software. If goals, membership, relationships with other organizations, business rules or any other aspect of the organization that is modeled by the system changes, this should be incorporated as choices in the behavior of the system and should not require changing the functionality of the system. Organizations are dynamic entities with changing requirements and policy based management systems should be able to model these changes and enforce any of the consequent requirements

through updating rules rather than modifying the functionality of components. However, using current approaches, the cost of making extensive changes to the policy system can be great, since it may require extensive re-analysis and the derivation of a new exhaustive set of policy rules to capture the new situation.
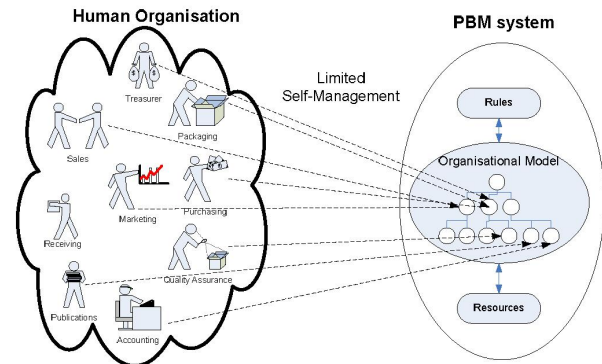

Figure 2 Using a dynamic organizational model to enable self-management

By incorporating a model of the organization into the PBM system, many of these problems can be addressed, as shown in figure 2. Organizational goals can be broken down into lower-level management goals, covering the goals of units within the organization and these goals can be broken down further into goals of sub-units. By linking goals to their organizational context, a clear policy refinement path can be identified, linking higher-level goals of the entire organization to lower-level management goals of specific units. Policy conflicts which result from conflicts between organizational units can be addressed by reference to higher level policies. The complexity of requirements engineering and change management can be reduced by decentralizing the modeling process and making groups within the organization responsible for managing their own model within the system – mirroring the situation in real world organizations where there is a growing trend in agile organization to higher levels of autonomy and decentralized decision making [5,6]. By incorporating a model of the organization into the PBM system, and linking policy specifications and resources to specific organizational contexts, the system can reason about policies (which can be considered to be decisions) based on factors such as the following.

- Who made the decision?
- With whose authority?
- What is the scope of their authority within the organization?

These are important considerations for people in real world organization when they decide how to act in response to decisions, particularly where those decisions conflict.

PBM systems have generally focused on the role construct, derived from the RBAC model [7], as the fundamental grouping abstraction of subjects in the system. Although the role is an extremely convenient abstraction, it is a simple construct – a set of users mapped to a set of permissions – and as such it is used to model various different elements of real-world organizations – from structural elements to functional

742

groupings to supervisory relationships. Due to this over-loading of the construct, it is difficult to map the set of roles and relationships within the model to a model of the real world organization. More recent models, such as TBAC [8] and OrBAC [9] add grouping constructs in order to model functional units and separate domains of authority within organizations, but these additions do not amount to an integrated organizational model. Furthermore, the fact that these models rely upon separate administrative hierarchies [10] further fragments the organizational model. On the other hand, more complex PBM systems such as Ponder [11] provide a wealth of grouping abstractions and relationships in addition to the role. However, this flexibility does not in itself solve the problem of modeling the organization, it merely facilitates it. Furthermore, the flexibility comes at a cost in terms of the complexity of the specification language and the difficulty in analyzing and comprehending state changes. This means the view of the organization at the time of roles were identified is so embedded into the policy set that is ossified against subsequent organizational change.

III. COMMUNITIES VERSUS ROLES

The Community Based Policy Management System (CBPMS) [12] uses a community construct in place of the common role construct. Communities facilitate a top-down functional decomposition of an organization into a hierarchy of organizational units, each of which can be managed with limited autonomy. The community at the root of the hierarchy, representing the entire organization, is progressively broken down into sub-units through invocation of the CBPMS primitives, shown below.

| **Community Record Management Service Primitives**	
The CBPMS Community Record Management Service (CRMS) supports 12 management primitives which allow an organization to define its policies through a dynamic evolutionary process. *Resource authorities* are delegated down the community hierarchy to provide an authority scope for each community – specifying what events the community can author policies for.	
Genesis/expel	Create / Destroy a root community
spawn/cull	Create / Destroy a sub-community
Delegate/recall	Delegate authority to a community
policy/revoke	Define a community policy
Federate/withdraw	Join or leave a federation
Grant	Assign ownership to a community
gatekeeper	Define a community membership rule

Authority is distributed to the various organizational sub-units by means of delegation or *resource-authorities* which define the scope of the authority of each unit. The community hierarchy is itself modeled as a resource-authority tree, as shown below, allowing for distribution of authority for the modeling process itself.

Resource Authorities

All managed resources are modeled as authority trees, each of which can be divided into an action tree and a target tree. A resource authority is a triple [R,T,A] where R is the resource model, T is a node on the resource's target tree and A is a node on the resource's action tree. Every delegation and policy is associated with a particular resource authority, which defines the scope of the delegation or policy. The authority trees for a community resource is shown below
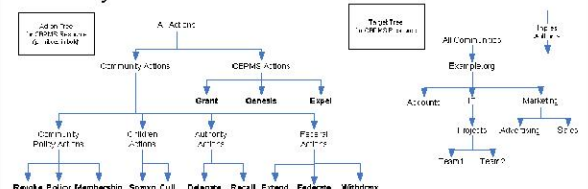
IV. DYNAMIC SPECTRUM ACCESS APPLICATION

A full web-based implementation of the CBPMS has been developed based on PHP and applied to several application area including online open source software development [13] and access control for instant messaging and for location-based presence data [14]. By way of example we discuss briefly the more recent application of CBPMS to Dynamic Spectrum Access (DSA) [15]. The application of the CBPMS to the problem of managing policy for DSA has been carried out under the auspices of the Centre for Telecommunications Value Chain Research (CTVR) at Trinity College Dublin. The CTVR has been granted a license by the Irish Communications Regulator (COMREG) to utilize a specified band of spectrum in order to experiment with DSA technology. In order to apply the CBPMS to the problem of managing policy in this field, the first requirement was a policy language and context model which could capture the various significant parameters of the domain. The DARPA-XG [16] policy platform was selected as a base from which to build a policy language. Figure 4. shows the policy condition definition form for the domain.

The community and resource model shown in figure 5 below were adopted for this experiment. The basic resource is the spectrum itself, as defined by frequency parameters. Chunks of frequency can be delegated to the CTVR and from there to the various groups that constitute it (e.g. TCD, DSG). Once a chunk of spectrum has been delegated to a group, that group can define policies regarding who can access that frequency. The CBPMS conflict resolution model will ensure that any policies defined higher up the community tree will have precedence over those defined further down – which is precisely what is required in this domain where regulators are concerned that the liberalization of the market should not prevent them from establishing system-wide policies which will be respected by all users.

Figure 4 CBPMS-DSA Policy Definition Screen

Figure 5 CBPMS-DSA Community and Resource Model

One final aspect of this experiment that is worth mentioning is the decentralized nature of the deployed CBPMS. Each of the communities is hosted on its own server and each of the communities is an independent resource. This means that each of the autonomous bodies can host their own community server, without granting any access to the other communities in the system. The spectrum resource can be delegated between the communities and policies can be applied anywhere in the community model.

V. CONCLUSIONS

In this paper we have argued that existing role-based policy engineering approaches are insufficiently flexible for modelling dynamic organizations. Already, it is observed that deployed systems, such as IBM Workplace product, that do benefit from the use of roles, also require a team of consultants to identify those roles, a task that can take several months for a large organization with large numbers of roles [17]. As organizational structure changes these role definitions can quickly become redundant, especially when the understanding of organizational function is incomplete. CBPM offer a means for the organization to react to change and to organically encode business rules as policies as they become sufficiently understood. This will enable organizations to become increasingly agile, especially as they managed the fine-grained sharing of authority over resources in federated business scenarios such as DSA.

ACKNOWLEDGMENT

REFERENCES

[1] N. Damianou, A. Bandara, M. Sloman and E. Lupu, A Survey of Policy Specification Approaches, Department of Computing, Imperial College of Science Technology and Medicine, London, 2002.

[2] P. Thompson and D. McHugh, Work Organisations, Palgrave, New York, 2002.

[3] R. Crook, D. Ince, and B. Nuseibeh, "Towards an Analytical Role Modelling Framework for Security Requirements", Proceedings of 8th International Workshop on Requirements Engineering: Foundation for Software Quality (REFSQ-02), Essen, Germany, 9-10 September 2002.

[4] M. Jude, "Policy-based Management: Beyond the Hype." In Business Communications Review, available from http://www.bcr.com/bcrmag/2001/03/p52.asp, March 2001. pp 52-56

[5] R. Barrett, "People and Policies: Transforming the Human-Computer Partnership", 5th IEEE International Workshop on Policies and Distributed Systems and Networks, IEEE, 2004

[6] J. Surowiecki, The Wisdom of Crowds: Why the Many Are Smarter Than the Few and How Collective Wisdom Shapes Business, Economies, Societies and Nations Little, Brown, 2004

[7] Sandhu, R., D. Ferraiolo and R. Kuhn. The NIST Model for Role-Based Access Control: Towards A Unified Standard. In Proceedings of the 5th ACM Workshop on Role-Based Access Control, Berlin, Germany, pp. 47-61, 26-28 July 2000

[8] Georgiadis, C., Mavridis, I., Pangalos, G, Thomas, K, "Flexible Team-based Access Control Using Contexts", Proceedings of the 6 ACM Symposium on Access control models and technologies, May 2001, pages 21-27

[9] A. El-Kalam, S., Benferhat, A. Miege, R. El-Baida, F. Cuppens, C. Saurel, P. Balbiani, Y.Deswarte, and G. Trouessin. "Organization based access control". In Proc. of the 4th IEEE International Workshop on Policies for Distributed Systems and Networks, page 120, Washington, DC, USA, 2003.

[10] Cuppens, F., Miege, A. "AdOrBAC: an administration model for Or-BAC" International Journal of Computer Systems Science & Engineering. Vol. 19, no. 3, pp. 151-162. May 2004

[11] N. Damianou, A. Bandara, M. Sloman, E. Lupu, "A Survey of Policy Specification Approaches", Department of Computing, Imperial College of Science Technology and Medicine, London, 2002.

[12] Feeney, K., Lewis, D., Wade, V. "Policy Based Management for Internet Communities", in Proc of 5th IEEE International Workshop on Policies for Distributed Systems and Networks (Policy 2004), June 8th-9th 2004, IBM Thomas J Watson Research Center, New York, USA, pp 23-34

[13] J. Keeney, K. Carey, D. Lewis, D. O'Sullivan, V. Wade "Ontology-based Semantics for Composable of Autonomic Elements", in proc of Workshop on AI in Autonomic Communications at 19th International Joint Conference on AI, Edinburgh, Scotland, July 5th 2005

[14] K. Quinn, A. Kenny, K. Feeney, D. Lewis, D. O'Sullivan, V. Wade, "A Framework for the Decentralisation and Management of Collaborative Applications in Ubiquitous Computing Environments",in Proc. IFIP/IEEE Network Operations and Management Systems, Vancouver, April 2006

[15] D. Lewis, K. Feeney, K. Foley, L. Doyle, T. Forde, P. Argyroudis, J. Keeney, D. O'Sullivan "Managing Policies for Dynamic Spectrum Access", in Proc. of IFIP TC6 1st Autonomic Networking conference, Paris France, September 2006

[16] DARPA XG Working Group(2004). XG Policy Language Framework Request For Comments Version 1.0. Available at: http://www.ir.bbn.com/projects/xmac/rfc/rfc-policylang-1.0.pdf. Accessed 5th August 2006

[17] Driver, E., IBM Delivers on its vision of Role-based workplaces, Forrester, 12 Arpil 2006