A Policy Oriented Architecture for the Web: New Infrastructure and New Opportunities

Renato Iannella
Principal Scientist
National ICT Australia (NICTA)
<renato@nicta.com.au>

Karen Henricksen
Research Scientist
National ICT Australia (NICTA)
<karen.henricksen@nicta.com.au>

Ricky Robinson
Research Scientist
National ICT Australia (NICTA)
<ricky.robinson@nicta.com.au>

1. Introduction

ICT architectures have evolved over the past decades from standalone mainframe systems (70s) to desktop systems (80s) to component-based systems (early 90s) to web-based systems (late 90s) to service-oriented architectures today. Slowly, but surely, this business-led evolution is improving the capabilities of ICT systems as new requirements mature. So, what is next?

All organisations need to collaborate and all organisations have different and dynamic rules and guidelines that govern these relationships. This varies from a consumer buying music for their iPod to a medical officer needing to analyse a number of reports and provide a recommendation to the Minister for a new health directive. In all cases, information and services are provided from autonomous organisations, some open, some closed, with different rules for access, different obligations, different expectations of service levels, and different consequences for non-conformance. In essence, the interactions are driven by *dynamic policies*.

Today's current infrastructure, in particular, the service-oriented architectures, are not positioned to address the business and technical issues dealing with the "policy oriented architecture". Specifically, they deal at the lower layers of infrastructure, not the semantically richer information layers. For example, they can deal with a quality-of-service policy that needs to guarantee a high bit-rate transfer, but baulk at a Privacy policy that needs the names removed from medical records before the summary information can be distributed to medical researchers who in turn can only publish the results in a open access journal, which must not allow commercial advertising next to the paper. This is not only a case of policy integration and harmonisation, but of *policy integrity*.

The next steps in the evolution are now clearer [WEI06]:

> *"...we will argue that a new generation of Policy-Aware Web technology can hold the key for providing open, distributed and scaleable information access on the World Wide Web."*

A Policy Oriented Architecture is a new direction that will address the need to manage multiple and conflicting policies in the future distributed service-oriented world. This will increase connectivity across disparate systems as they can achieve a new level of automated interoperability guided by declarative policies that can adapt to different contexts and environments. For example, a policy oriented architecture could lead to increased trust (eg fraud detection and secure transactions) as independent cooperating services exchange, understand, and enforce policies on identity, access control, privilege management, and user profiles.

2.   Policies, Policies, Policies

Currently, there are numerous policies being used or under development for the Web. These include, but are not limited to; Privacy, Rights Management, Context Awareness, Identity, and Access Control. Each of these policy languages represents a particular business and technical view of information policies based on the specific needs of the community that they currently address. An example snap shot would include:

- Privacy: W3C Platform for Privacy Preferences [W3C02]
- Rights: Open Digital Rights Language [ODR02]
- Identity: Liberty Framework [LIB06]
- Access Control: Extensible Access Control Markup Language [OAS05]
- Context: Context-based Adaption Rules and Preferences [HEN06]

These five classes of policies are important in that their overlap in semantic and technical requirements is significant, but all are still approached as vertical new technologies. The key to this group of policies is that they all deal with three core entities: People, Content, Permissions.

Even with an abundance of current policy languages, new groups still develop new languages. For example, the "Common Policy: A Document Format for Expressing Privacy Preferences" [SCH06] - an IETF Internet Draft - proposes a new language for authorisation policies controlling access to application specific data. Although not exactly compatible with the 5 policy languages listed above, their proposed model is based on rules with conditions, actions, and transformations. All common in other policy languages.

What is needed is a Policy Framework to capture the common semantic and architectural components of these verticals to enable consistent interaction with all policies. This abstract framework will embody the core attributes of all policies and support specific attributes for different policy capabilities.

3.   Having Your Cake And Eating It Too

The Semantic Web - and the infamous Layer Cake [MIL02] - is well understood in the technical community. The challenge now is determining where the Policy Framework layer fits. This is critical as if the layer is not at the most "accessible" point, then there is a risk of low adoption.

The key decisions need to be based on how many of the lower layers are currently being used by policy languages, and how many vertical layers a Policy Framework should try to address. This must be based on an extensive review of the communities that have developed and successfully deployed policy languages in the web and related sectors (eg the mobile environment).

There needs to be a balance between policy languages based on Semantic Web technologies (such as Rein [KAG06]) and those based on Structured Web technologies. They can both coexist and be interoperable if an inclusive approach is taken. It is important to note that the Structured Web also manages "semantics" and that it is not exclusive to the (capital-S) Semantic Web.
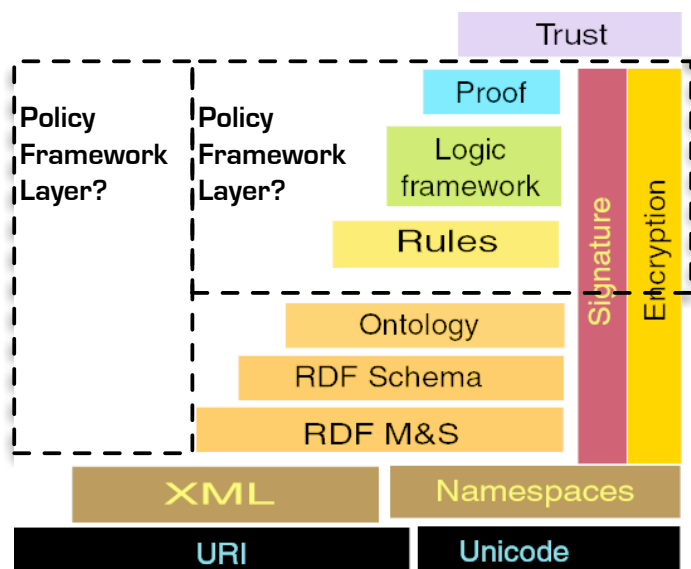
4.   The Policy Oriented Architecture

The three key research and standardisation challenges posed by a Policy Oriented Architecture for the Web are shown in the Figure below. These are:

- How to Model the policy framework,
- How to Represent the Model, and
- How to Architect web-based solutions.

A unified model of the various policy requirements will capture the core concepts and structures common to all policies. This abstract level model will then be expressed in numerous representations. Representations will capture greater details and be grounded in new, existing, and improved languages for the expression of policies. These representations will then be manifested in architectures. Architectures are concrete implementations that are built to specific environmental, infrastructure, or community requirements.
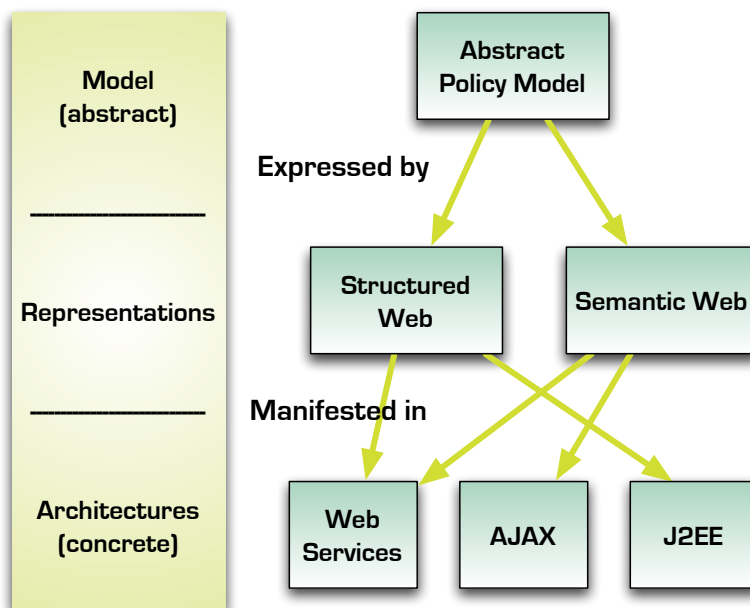
These challenges provide the framework for addressing even deeper policy-specific challenges, such as the evaluation, enforcement, and reasoning of policies, and how to deal with inconsistencies across policies.

5.   The Ladder Of Opportunity

As [SHA06] discusses, the Web has evolved from the ground up following a "ladder of authority" - a sequence of specifications that incrementally build greater levels of infrastructure for the Web. The greatest challenge the Web faces today is to capitalise on the efforts that have gone on previously in the development of specific policy languages - and to build the next layers of infrastructure to support a policy-aware web. This will transform the web from an information delivery system into an information management system that will meet the emerging needs of the web community. In particular, it will address communities that have based their professional and societal life on the functions and structure of the web - and are now looking for greater infrastructure support for common services. Just like HTML, the web now needs reliable structures for policy management.

References

[HEN05] Karen Henricksen and Jadwiga Indulska. *Developing context-aware pervasive computing applications: Models and approach*. Journal of Pervasive and Mobile Computing, volume 2(1), pages 37-64. Elsevier, February 2006

[KAG06] Lalana Kagal, Tim Berners-Lee, Dan Connolly, & Daniel Weitzner, *Using Semantic Web Technologies for Open Policy Management on the Web*, 21st National Conference on Artificial Intelligence (AAAI) 2006

[LIB06] *Liberty Identity Federation Framework*. The Liberty Alliance, 2006. <http://www.projectliberty.org/>

[MIL02] Eric Miller. *The Semantic Web*. W3C Track at the WWW2002 Conference. <http://www.w3.org/2002/Talks/www2002-w3ct-swintro-em/Overview.html>

[OAS05] *OASIS eXtensible Access Control Markup Language*, Version 2.0. OASIS Standard. <http://docs.oasis-open.org/xacml/2.0/XACML-2.0-OS-NORMATIVE.zip>

[ODR02] *Open Digital Rights Language*, Version 1.1 Specification. ODRL Initiative, 19 September 2002 <http://odrl.net/1.1/ODRL-11.pdf> and <http://www.w3.org/TR/odrl/

[SCH06] H. Schulzrinne (et al). *Common Policy: A Document Format for Expressing Privacy Preferences*. IETF Internet Draft, August 10, 2006 <http://www.ietf.org/internet-drafts/draft-ietf-geopriv-common-policy-11.txt>

[SHA06] Nigel Shadbolt, Wendy Hall, & Tim Berners-Lee. *The Semantic Web Revisited*. IEEE Intelligent Systems. May/June 2006

[WEI06] Daniel Weitzner, Jim Hendler, Tim Berners-Lee, & Dan Connolly. *Creating a policy-aware web: Discretionary, rule-based access for the world wide web*. In Elena Ferrari and Bhavani Thuraisingham, editors, Web and Information Security. IRM Press, 2006

[W3C02] *The Platform for Privacy Preferences*, Version 1.0 Specification. W3C Recommendation 16 April 2002 <http://www.w3.org/TR/P3P/>

Acknowledgments