Towards A Conceptual Model For Privacy Policies

Marco Casassa Mont, Siani Pearson, Sadie Creese, Michael Goldsmith, Nick Papanikolaou

HP Laboratories
HPL-2010-82

Keyword(s):

privacy policies, policy hierarchy, policy refinement, conceptual model

Abstract:

This paper presents a conceptual model for privacy policies that takes into account privacy requirements arising from different stakeholders, with legal, business and technical backgrounds. Current approaches to privacy management are either high-level, enforcing privacy of personal data using legal compliance, risk and impact assessments, or low-level, focusing on the technical implementation of access controls to personal data held by an enterprise. High-level approaches tend to address privacy as an afterthought in ordinary business practice, and involve ad hoc enforcement practices; low-level approaches often leave out important legal and business considerations focusing solely on technical management of privacy policies. Hence, neither is a panacea and the low level approaches are often not adopted in real environments. Our conceptual model provides a means to express privacy policy requirements as well as users' privacy preferences. It enables structured reasoning regarding containment and implementation between various policies at the high level, and enables easy traceability into the low-level policy implementations. Thus it offers a means to reason about correctness that links low-level privacy management mechanisms to stakeholder requirements, thereby encouraging exploitation of the low-level methods.

External Posting Date: July 21, 2010 [Fulltext] Approved for External Publication
Internal Posting Date: July 21, 2010 [Fulltext]

Towards A Conceptual Model For Privacy Policies

Marco Casassa Mont, Siani Pearson
Systems Security Lab
HP Labs
Bristol, UK
e-mail: {marco.casassa-mont, siani.pearson}@hp.com

Sadie Creese, Michael Goldsmith,
Nick Papanikolaou
International Digital Laboratory
University of Warwick
Coventry, UK
e-mail: {S.Creese, M.H.Goldsmith,
N.Papanikolaou}@warwick.ac.uk

***Abstract*— This paper presents a conceptual model for privacy policies that takes into account privacy requirements arising from different stakeholders, with legal, business and technical backgrounds. Current approaches to privacy management are either high-level, enforcing privacy of personal data using legal compliance, risk and impact assessments, or low-level, focusing on the technical implementation of access controls to personal data held by an enterprise. High-level approaches tend to address privacy as an afterthought in ordinary business practice, and involve *ad hoc* enforcement practices; low-level approaches often leave out important legal and business considerations focusing solely on technical management of privacy policies. Hence, neither is a panacea and the low level approaches are often not adopted in real environments. Our conceptual model provides a means to express privacy policy requirements as well as users' privacy preferences. It enables structured reasoning regarding containment and implementation between various policies at the high level, and enables easy traceability into the low-level policy implementations. Thus it offers a means to reason about correctness that links low-level privacy management mechanisms to stakeholder requirements, thereby encouraging exploitation of the low-level methods.**

Keywords - privacy policies, policy hierarchy, policy refinement, conceptual model

I. INTRODUCTION

Enterprises manage and administer huge sets of personal data which are collected as part of normal business practice. This process is complex and involves meeting a wide range of requirements, including the need to satisfy data protection laws and privacy, as well as any service requirements made by the enterprise or the consumer. Often such requirements are captured in the form of a policy or policies. However, there is not yet a unified view of the different approaches to policies existing in an enterprise. This makes it hard to guarantee that the combination of the various implementations does indeed meet all the requirements being made of the enterprise and is aligned with legal requirements. Furthermore, the process of assessing this alignment is subject to human error.

In general there are two extreme approaches to management and enforcement of privacy policies. There is firstly a pragmatic approach: driven mainly by risk assessment and risk management and tailored to current business practices. It involves identifying suitable high level policies and points to act on, but then typically requires the deployment of pragmatic control points, which are very dependent on the specific scenario/environment. The control points enforcing policies are often hardcoded within applications and services in an *ad hoc* way, and so cannot easily be reused in different scenarios and organisational contexts. However, this seems to be the norm in business practice today.

On the other hand, frequently research in this space tends to focus instead on a purely technical approach and narrowly propose yet another language or formal model for security, access control or obligation policies without taking into account legal, business and operational requirements. Hence, related policy languages might be too generic or detached from real requirements; often these languages and models are of interest to the research community but seldom widely adopted in real environments. We believe that there is a major gap between the two approaches and that there is a unique opportunity to combine aspects of each and provide mechanisms to bridge the two.

Our approach is to develop a conceptual model rich enough to describe high-level policies typically expressed in natural language, and structured to support their refinement and mapping into low-level technical policies for practical enforcement in an information system. In the EnCoRe (Ensuring Consent and Revocation [1]) project, we are exploring this approach while specifically focusing on an important aspect of privacy: the management of data subjects' (users') preferences with regard to the handling of their personal data. In EnCoRe such preferences actually equate to expressions of consent and revocation relating to rights to handle and process personal data.

This paper is structured as follows: we discuss related work in Section II. In Section III we analyse the advantages and disadvantages of current approaches to privacy management and proposes a hybrid approach to the development and enforcement of privacy policies, which takes into account:

- legal, security and business requirements,

[1] See http://www.encore-project.info.

- the outcomes of risk and privacy impact assessment,
- what is feasible technologically.

Section IV presents a practical example of policy enforcement for consent and revocation. Section V presents our initial conceptual model and VI concludes our discussion.

II. Related Work

There are some structured databases of privacy laws that are being developed. In particular, the Governance, Risk Management and Compliance Global Rules Information Database Initiative (GRC-GRID) [21] has begun to create a DB of international "rules, regulations, standards, and government guidance documents". Similarly, Archer Compliance Management solution [22] allows creation of a personalized workspace to set up standard reports.

IBM and Sun have done some research on privacy policy management such as EPAL [9] and XACML [5] which are low level privacy policy languages and not well suited for human user understanding. Other privacy policy languages are …

Translation of legislation/regulation to machine readable policies has proven very difficult, although there are some examples of how translations of principles into machine readable policies can be done: Privacy Incorporated Software Agent (PISA) project [14] (where privacy principles derived from [15] were modelled and used as a backbone in conversations between agents [16]), P3P [17,18] (where user privacy preferences were matched against web site privacy statements) and PRIME [19] (involving the definition and usage of various types of user and service side privacy policies). It is an open problem how to interpret and model arbitrary laws: it is a much easier problem to map company policies to lower level implementable policies, or human-readable output. HP Privacy Advisor [23] represents company policies in a machine readable format and analyses these to provide human-readable customized output relating to specific circumstances. In the Sparcle project [20], IBM Research built an editor to support transforming natural based policies into XML code that can be utilized by enforcement engines. This makes it easier for non-experts to input rules into the system, but the output format itself is not user friendly and is targeted towards machine execution. The REALM project [11] from IBM Research has also worked on translating high level policy and compliance constraints into machine readable formats. In addition, Breaux and Antón [13] have carried out some work on how to extract privacy rules and regulations from natural language text.

III. Policy Layers and Dependencies in Organisations

Organisations need to cope with a variety of policies and constraints that emerge from many different sources, including legislation (national and international), societal expectations, business requirements and (where appropraite) individual preferences expressed by users and customers. We concern ourselves here specifically with those policies relating to the handling of personal data and privacy.

Whilst privacy requirements are in general context dependent we believe that there are a core set of privacy concepts which are common and underpin the various controls designed to deliver privacy against this varying set of requirements. They are, in effect, a tool box which can be utilised depending upon the unique requirements of the situation. But, due to the heterogeneity of the policies and of the languages in which they are expressed, it may not be always obvious what the core requirements are. However, if these are clearly identified, we will be able to better formalise and classify privacy-related policies, laws and technical solutions enabling a simplification and easier re-use of the technologies and methodologies designed to implement such policies. Further, the extraction of such core requirements might make it easier to compare privacy legislation with the technical implementation of privacy constraints in a product.

We consider policies to fit within a layer model which in itself represents a hierarchy of policies. In this model, high-level policies express general requirements and rights that individuals have with regards to their privacy, as embodied typically in the law, business and regulatory requirements as they contain general constraints on business practice with regards to personal data. At the highest level of the classification, there is a set of requirements which are set out by international agreements and directives, such as the European Data Protection Directive or the EU Safe Harbour agreement. Further, many countries have national data protection legislation, such as the Data Protection Act 1998 in the UK, or HIPAA, GLBA, SB 1386, COPPA and various State Breach laws in US. With regards to regulation in particular, there are export and transborder flow restrictions on personal data that need to be enforced. Privacy laws and regulations constitute the topmost layers of policy hierarchy regarding personal data with which an enterprise must comply. Such policies are often expressed in natural language as is typically the case with related data subjects' preferences.

At this high level of abstraction, security requirements may include adherence to the Sarbanes-Oxley Act (SOX) for financial reporting, or the PCI Data Security Standard (DSS). These may be refined to a set of policies at a lower level. Similarly, business requirements include contractual obligations, information lifecycle policies and the enterprise's own internal guidelines. All of the above influence how personal data is collected, stored and administered.

Low-level policies are those which describe how privacy requirements are implemented in a particular piece of hardware, or in software that handles personal data. Such policies comprise detailed conditions on how particular data may be handled within a system: often these are just statements prohibiting particular accesses of the data, in which case they are referred to as access control policies.

At lower levels there are various operational and technical policies that are machine readable and enforceable by policy management frameworks, e.g. [1,4,6,19]. Among

these there will be policies expressing how a particular class of data is to be treated, and these are only specific to the data, not to the system implementing the policies. Even more low level will be policies that are system-specific, and cannot be ported directly to other privacy-preserving platforms. For instance, policies specific to a particular health information system may contain specialized fields that do not exist in other similar systems. Figure 1 below is a diagrammatic representation of the different layers within which privacy policies are implemented. High level are policies relating to the layers from "Application/Service Layer" and above, while the layers below consist of low level policies. The preferences of a data subject can be considered as belonging between the business and legal layers.

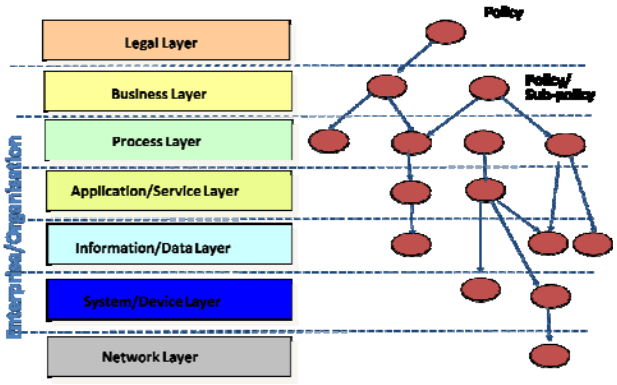

Figure 1. The different layers in which privacy policies are implemented.

What is clear from the above analysis is that the origins of privacy requirements which an enterprise has to meet are very diverse, and they arise at many different levels of abstraction. In an ideal world, lower level policies should always be the result of refinements, or special cases, of the higher level ones. In the real world high-level requirements change over time. Data subjects and data controllers (service providers) will all exercise choice relating to their preferences and risk appetite. This will make it impossible for a system to always be a correct refinement of requirements, as it will take time for choices to be implemented. It will be for the data subjects to decide whether they are being offered appropriate service levels regarding the response to their choices, and for service providers to determine what level of guarantee is appropriate for their business model. Law and regulation will also evolve over time, although much more slowly and in a manner which should give enterprises sufficient time to ensure that they are addressing (or at elast attempting to address) changes to related policy requirements. Privacy requirements are so heterogeneous that is not always possible to treat them consistently, and yet it is necessary to ensure that all these assorted requirements are simultaneously met for the correct functioning of society.

A key assumption in our definition of a hierarchy, as opposed to a loose grouping of policies by theme and/or level of detail, is that there is a relation of *containment* between the different levels described. It should often be the case that higher-level policies express requirements that should be made more explicit (*refined*) in lower-level ones. In that sense, higher level policies contain requirements expressed at the lower levels, albeit in a more abstract or generic form. This justifies their placement at the upper level of the hierarchy. The more formally a policy can be expressed, the more chance we have of creating automatic enforcement mechanisms reusable technology. However, there will always be policies which are, by design, open to interpretation and requiring human intervention.

One might classify current research in privacy policy description, management and enforcement using Figure 2. The vertical axis represents the varying levels which policies are expressed at ranging from high-level (legal, regulatory) to low level (security/access control policies and user preferences). The horizontal axis characterises the degree to which policies are formalised, ranging from natural language to machine readable formats.

A significant amount of research falls into quadrant III. This is no surprise as the development of policy languages goes hand in hand with the development of machine readable descriptions of low-level technical policies.

It is evident from the figure that there are many other viewpoints and levels of abstraction that are of concern in policy management, and that there is scope for much work in the areas labelled as quadrants I, II and IV.

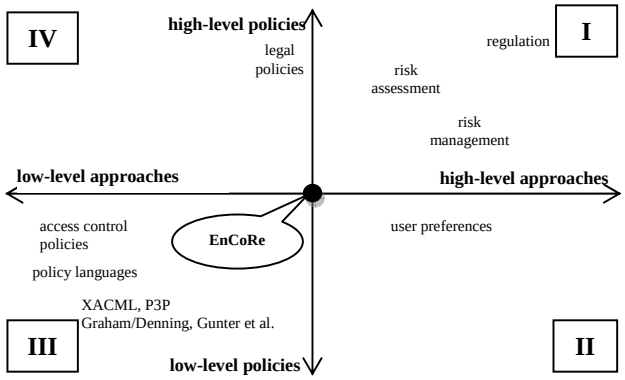

Figure 2. Policy layers versus description and enforcement approaches.

Quadrants II and III pertain to the low-level policy implementations and the degree to which they directly implement requiremetns expressed in natural language versus machine readable formats. Specifically, we see that there is a research opportunity providing a link between natural language requirements and policy implementations (II), whereas those requirements expressed directly into machine readable formats are a good fit (III). In reality we expect most requirements to begin life in II and that system implementations find ways of pulling these into III. Our conceptual model is designed to provide a formal framework within which tracability between requirements expressed in II can be linked to corresponding requirements in III.

Quadrants I and IV pertain to mapping of high-level policies directly into machine readable formats. Here the research question is to understand just how much of this kind of policy is ambigious and requires context-dependent human intervention. We focus on ensuring that our conceptual model is rich enough to describe all of these high-level policies, so that where core privacy requirements can be identified we can directly map into formalised a machine readable formats to support technology controls.

The specific area of management of privacy policies, security constraints, consent and revocation [2] is of particular interest because it is at the intersection of legislation, user requirements and management of privacy and security technical policies within and across organisations. What is particularly desirable is to devise an intermediate representation of policies that embodies high-level requirements whilst being directly translatable to (potentially existing) low-level policies or access control languages such as XACML [5], EPAL, P3P [3], P-RBAC [6], and the like. Such a representation should not be tied to a particular implementation language.

It may be argued that our definition of a hierarchy for privacy policies is arbitrary, as the level of detail contained in privacy-related documents, from international legislation down to business and regulatory policies, varies substantially by domain of application. It is the case that how the hierarchy is defined is heavily *context-dependent.* Our classification is based on research within EnCoRe, taking into consideration privacy requirements coming from a variety of sources (including legal, social and tehcnical ones) related to the following scenarios:

- Employee data held within an enterprise
- Biobanks
- Assisted living

We expect that these case studies will guide our intuition regarding a core, common set of privacy requirements, and hence suggest the evolution of our conceptual model.

IV. Examples of Policy Rules in Different Layers

Access control and privacy policies related to the protection of personal data typically contain stipulations about:

- for which purposes a data processor may collect personal data
- which types of personal data are considered sensitive, and hence are subject to additional restrictions
- for how long collected personal data may be held
- whether and how personal data may be shared with third parties
- which actions a data processor must take in case of a privacy breach

These reflect privacy principles that are common to different levels of policies in the hierarchy presented in Section II. What is desirable is to have a uniform conceptual representation of the policies defined in the different layers. We consider here some of the distinctive features of these different types of policies, and for some we identify a general format; this is precisely what is needed for a conceptual representation. In future work we hope to find the structures that are common to all the different types of policies and to characterise them in a formal manner.

An example of a high-level policy is the set of data protection principles set out in the Data Protection Act. These principles (paraphrased versions of which follow) require that personal data shall:

- Be processed fairly and lawfully and shall not be processed unless certain conditions are met;
- Be obtained for a specified and lawful purpose and shall not be processed in any manner incompatible with that purpose;
- Be adequate, relevant and not excessive for those purposes;
- Be accurate and, where necessary, kept up to date;
- Not be kept for longer than is necessary for that purpose;
- Be processed in accordance with the data subject's rights;
- Be kept secure from unauthorised or unlawful processing and protected against accidental loss, destruction or damage by using the appropriate technical and organisational measures;
- Not be transferred to a country or territory outside the European Economic Area, unless that country or territory ensures an adequate level of protection for the rights and freedoms of data subjects in relation to the processing of personal data.

Such legislation may be translated into policies, but for some of these, refinement and/or interpretation will be necessary in order to translate these into operational (technical) policies, and this is easier to do with some policies than others. For example, it is straightforward to do this with notification requirements and some forms of transborder data flow, but not for transparency and adequacy requirements.

An example of a simple related privacy-aware access control policies could be conceptually expressed as an **`if...then...else`** rule:

```
Target: Personal Data D
    if (Data Requestor wants to access
    personal data D for Purpose P)
    and (data subject has given consent
    for this data)
    then Allow Access
    else Deny Access
```

Similarly, for transborder data flow, rules may also be represented in the same form, such as:

```
    if (all source countries are members
    of EEA and all target countries are
    members of EEA)
    then (no problems with transborder
    data flow)
```

This type of rule is not an access control policy or an obligation policy, but is a different type of policy – a 'compliance policy'.

Notice and notifications require checking for "triggering" conditions and the context. Again, an **`if...then`** rule could be used to capture these concepts . For example:

```
if (<country legal entity resides
in> is member of [Belgium,
Portugal])
then (provide notification)
```

This is more like an obligation policy, but note that it is not triggered by access control [1]. Another example would be that if there were a data breach then it would be necessary to notify the legal authorities and end users. This is an obligation policy, of a type that is triggered by an event.

The key point here is that it is possible to identify some common patterns and concepts across these types of policies along with intermediate representations (e.g. rules) that are independent of underlying technical policies but which may nevertheless be fairly directly mapped onto these.

A similar analysis of policies can be made from a business and security perspective. Business policies, for example, relate to the treatment of information throughout its lifecycle, and that are also relevant to consider as background. These include: availability and recovery time policies, change control policies, binding contractual arrangements with third parties, service level agreements (SLAs) and IT governance policies. Also in this category are internal guidelines (that can map onto access control policies, obligation policies and/or compliance policies), and contractual obligations, which could relate to clauses included in contracts with clients, or to information contained within SLAs, etc.

Security requirements and related policies often originate in information security standards dictating methodologies and common security practices. These include: PCI DSS, Standard of Good Practice for Information Security, OCTAVE and CORAS (these two are risk management methodologies), ISO 27001/2 (an international standard outlining best practices), BS 10012:2009 (a British standard outlining best practices); DoD MIL-STD-1629A (a US Department of Defense risk management methodology). Examples of requirements from PCI-DSS are: restrict access to cardholder data by business need to know; track and monitor all access to network resources and cardholder data

Usually these security requirements dictate constraints on who can do what on which protected resource, given a specific context. Conceptually this can be expressed in terms of access control policies:

```
Target: Resource X
      if (Data Requestor is User U/Role
      R in Context C)
      then (Allow access to X)
      else (Deny access)
```

At a conceptual level we notice similarities about how to represent these constraints across different domains.

In the specific case of management of personal data, privacy and security concepts can be conceptually bundled in a uniform representation. For example, both privacy and security constraints could be represented in the same **`if...then...else`** rule model:

```
Target: Personal Data X
   If (Data Requestor is User U/Role R
   in Context C)
   and (Data Requestor wants to access
   personal data D for Purpose P)
   and (data subject has given consent
   for this data)
   then (Allow access to X)
   else (Deny access)
```

The above examples are meant to show the value of being able to explicitly and uniformly represent the concepts and constraints involved in different types of policies as a way to reason about them. We believe that a conceptual model should provide a way to consistently represent all these concepts across different domains without the constraints induced by any specific "technical" language.

To ensure continuity of the mapping between different layers, these requirements and policies need eventually to be mapped into enforceable technical policies, for example in languages such as XACML. This is where most of the conceptual gaps can be identified as well as the limitations of current technical approaches to policy languages. In the case of technical policies, we need to take into account a variety of details, for example where personal data and data subjects' preferences are stored, how to express constraints in a way that can be automatically enforced, how to deal with consent and revocation, etc. An example is given in the following diagram, described in full details in [4].

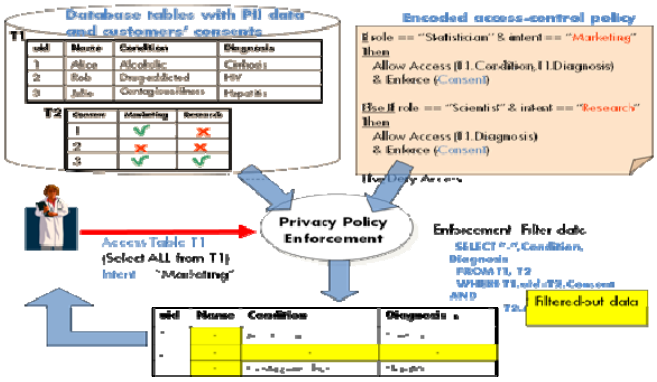

Figure 3. Privacy-aware Access Control.

In this example, a basic privacy-aware access control policy (in a "pseudo" conceptual representation) could look like the following:

```
Target: <Database:DB1, Table:T1>
    if (DataRequestor.role is "employee"
    and DataRequestor.intent is
    "Marketing")
    then ((Allow access to T1.Condition,
    T1.Diagnosis)
    & Enforce (Consent))
    else if (DataRequestor.intent is
    "Research")
    then
    (Allow access to T1.Diagnosis) &
    Enforce (Consent))
    else (Deny access)
```

This policy could be potentially mapped in technical policies such as XACML. However, an accurate analysis of the example policy above [4] highlights that the management of "*conditional YES*" is required i.e. postponing the check of consent at the Policy Enforcement Point (PEP). In this context, the Policy Decision Point (PDP) can only highlight the set of constraits/conditions that need to be checked and enforced at the enforcement time.

This cannot be easily achieved with the current XACML representation. As a result, in the EnCoRe project we had to "twist" the language and framework to achieve the desired outcomes.

A conceptual representation of policies would have enabled reasoning about them as well as the identification of constraints to be satisfied by the underlying levels.

What is desirable is to have a uniform conceptual representation of the policies across the different layers. In this section we have briefly discussed some of the distinctive features of the different types of policies, and for some we have identified a potential conceptual format; this is a first, initial step towards a full conceptual representation. In future work we hope to find the structures that are common to all the different types of policies and to characterise them in a formal manner. We have already developed elements of a formal access control model for consent and revocation that can be directly leveraged in this work.

V. Formalising the Conceptual Model

The examples of policy rules we have given so far demonstrate several different forms that privacy requirements take in real business applications. It is desirable to be able to automatically enforce as many policy rules as possible; for this, a machine-readable representation of the different forms of requirements is necessary. However, the purpose of a *conceptual* model is to provide a representation that enables human systematic reasoning about policies while at the same time being convertible into machine-readable code.

A conceptual model defined in a strict mathematical way would have the benefit of being completely unambiguous, but it would likely be too restrictive, especially if it is intended to capture privacy laws and regulations. A more flexible approach would be to describe privacy requirements in a semi-formal manner. One can be very systematic and formal about the *purpose* of different policy rules, and in terms of *syntax* it should be possible to identify the main patterns of usage that occur in privacy requirements.

To illustrate this last point: above we identified policy rules as typically having the structure **if** <some condition is met> **then** <$action_1$> **else** <$action_2$>. The syntax and semantics of the conditions and actions allowed in such rules are essentially informal. However, our analysis shows that there are at least the following core set of rule types:

- **notification rules**: such rules describe when and how data subjects should be notified regarding accesses, uses, and transfers of their personal data. Such rules appear in low-level policies – forcing an implemented system to send email or instant messages when a condition is triggered – as well as in high-level policies and legislation: the Data Protection Act, for example, specifies that data subjects can make *subject access requests* (SARs), forcing a data controller to notify them of any data held about them, for which purposes, and with whom this data has been shared.
- **access control rules:** such rules specify who can access data held by an enterprise; for instance, personal data about employees should only be available to the HR department and to the employees (on an individual basis).
- **update/creation rules:** these rules express who is permitted to modify personal data that is held, and under which conditions. The right to update data or even create new data is usually reserved for the data subject, and certainly such a right exists in legislation. Rules specifying who can perform such changes to data held typically take into account the *role* of the parties making them (cf. role-based access control).

- **protection rules:** there will be rules specifying protections on particular data, usually protections of a technical nature, such as encryption. These are most easily described in technical, low-level privacy policies, since the parameters and algorithm for encryption can be explicitly defined; however, requirements for encryption are increasingly found in privacy regulation and company privacy policies.

These rule types are the essence of our conceptual model, and provide a natural means of expressing both high-level policies and setting requirements for low-level implementations. They enable the expression of actions assocaited with granting and revoking consent for the use of personal data. The full conceptual model encompasses a formal syntax and semantics, which is presented elsewhere [24].

VI. CONCLUSIONS AND FUTURE WORK

We have discussed in this paper issues to do with the description, management and enforcement of policies in organisations. Specifically we highlighted the gap existing from a high-level approach to policies driven by risk and privacy impact assessment and low-level technical policies.

We strongly believe this gap needs to be filled to enable continuity of requirements and constraints across all these levels and enable proper enforcement of policies. To achieve this we proposed the adoption of a conceptual policy model, to enable reasoning and mapping of concepts at lower levels of abstraction.

In our analysis we illustrated: the range of privacy policy levels that exist (each level corresponds to a different layer of abstraction) and some of the related, distinctive requirements; elements of a conceptual model which characterises the properties of these different types of policy; first thoughts on an access control model which might be used to describe different types of policy rules in a uniform way, with an emphasis on user preferences.

Our future work will seek to validate and refine our conceptualisation of a policy hierarchy, specifically with a view to ensuring that our conceptual model for privacy policy is rich enough to cater for all needs. We will also investigate the utility of the conceptual model by application to case studies, initially within the EnCoRe project. We hope to be able to identify core privacy properties across the case studies, which can be easily mapped into reusable low-level control mechanisms. We also hoep that this will offer opportunities to simplfy the human interfaces, and reduce the amount of human intervention required making it simpler and more cost effective.

REFERENCES

[1] Marco Casassa Mont (2006). *On the Need to Explicitly Manage Privacy Obligation Policies as Part of Good Data Handling Practices.* Proceedings ofW3C Workshop on Languages for Privacy Policy Negotiation and Semantics-Driven Enforcement, 17-18 October 2006, Ispra, Italy.

[2] Marco Casassa Mont, Siani Pearson, Gina Kounga, Yun Shen, and Pete Bramhall (2009). *On the Management of Consent and Revocation in Enterprises: Setting the Context.*Technical Report HPL-2009-49, HP Labs, Bristol.

[3] L. Cranor, B. Dobbs, S. Egelman, G. Hogben, J. Humphrey, M. Langheinrich, M. Marchiori, M. Presler-Marshall, J. M. Reagle, M. Schunter, D. A. Stampley, and R. Wenning (2006). *The Platform for Privacy Preferences 1.1 (P3P1.1) Specification.* World Wide Web Consortium Note NOTEP3P11-20061113.

[4] Marco Casassa Mont, Robert Thyne, Privacy Policy Enforcement in Enterprises with Identity Management Solutions, PST 2006, 2006.

[5] OASIS, eXtensible Access Control Markup Language (XACML), http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml.

[6] Qun Ni, Alberto Trombetta, Elisa Bertino, and Jorge Lobo (2007). Privacy-aware role based access control. In *Proceedings of the 12th ACM Symposium on Access Control Models and Technologies* (Sophia Antipolis, France, June 20-22, 2007). ACM, New York, pp. 41-50.

[7] Rodolfo Ferrini, Elisa Bertino (2009). A Comprehensive Approach for Solving Policy Heterogeneity. In *ICEIS 2009 -Proceedings of the 11th International Conference on Enterprise Information Systems* (Milan, Italy, May 6-10, 2009), pp. 63-68.

[8] Ioannis Agrafiotis, Sadie Creese, Michael Goldsmith, and Nikolaos Papanikolaou (2009). Reaching for Informed Revocation: Shutting Off the Tap on Personal Data. *Proceedings of Fifth International Summer School on Privacy and Identity Management for Life* (Nice, France, *7th – 11th September 2009).*

[9] IBM, The Enterprise Privacy Authorization Language (EPAL), EPAL specification, v1.2, http://www.zurich.ibm.com/security/enterprise-privacy/epal/, 2004.

[10] K. Vaniea, C. Karat, J.B. Gross, J. Karat and C. Brodie, Evaluating assistance of natural language policy authoring, Proc. SOUPS '08, vol. 337. 2008.

[11] IBM, REALM project, http://www.zurich.ibm.com/security/publications/2006/REALM-at-IRIS2006-20060217.pdf

[12] OASIS, eContracts Specification v1.0, www.oasis-open.org/apps/org/workgroup/legalxml-econtracts, 2007.

[13] D. Travis, T. Breaux and A.Antón, Analyzing Regulatory Rules for Privacy and Security Requirements. IEEE Transactions on Software Engineering, 34(1), pp. 5-20, 2008. [13] W3C, The Platform for Privacy Preferences, v1.0, http://www.w3.org/TR/P3P/, 2002.

[14] S. Kenny and J. Borking, The Value of Privacy Engineering, Journal of Information, Law and Technology (JILT), 1.http://elj.warwick.ac.uk/jilt/02-/kenny.html, 2002.

[15] Organization for Economic Co-operation and Development (OECD), Guidelines Governing the Protection of Privacy and Transborder Flow of Personal Data, OECD, Geneva, 1980.

[16] J. Borking, Privacy Rules: A Steeple Chase for Systems Architects, http://www.w3.org/2006/07/privacy-ws/papers/04-borking-rules/, 2007.

[17] L. Cranor, Web Privacy with P3P, O'Reilly & Associates, 2002.

[18] N. Damianou, N. Dulay, E. Lupu, M. Sloman, The Ponder Policy Specification Language, http://www-dse.doc.ic.ac.uk/research/policies/index.shtml, 2001.

[19] PRIME, Privacy and Identity Management for Europe, http://www.prime-project.org.eu, 2008.

[20] IBM: Sparcle project, http://domino.research.ibm.com/comm/research_projects.nsf/pages/sparcle.index.html

[21] The GRC-GRID, The Governance, Risk Management and Compliance Global Rules Information Database, http://www.grcroundtable.org/grc-grid.htm

[22] Archer: Compliance Management solution, http://www.archer-tech.com

[23] Siani Pearson, Tomas Sander and Rajneesh Sharma, "A Privacy Management Tool for Global Outsourcing", Proc. DPM'09, LNCS, Springer, October 2009.

[24] Ioannis Agrafiotis, Sadie Creese, Michael Goldsmith, Nick Papanikolaou, "The Logic of Consent and Revocation", Submitted, 2010. See http://go.warwick.ac.uk/nikos/publications.