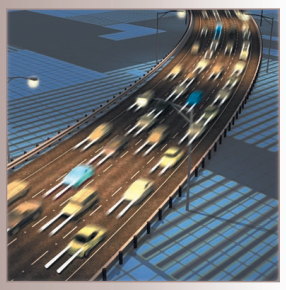

Editors: K.J. Lin • *klin@uci.edu*
Yan Wang • *yanwang@ics.mq.edu.au*

Privacy Ontology Support for E-Commerce

Privacy is becoming increasingly important due to the advent of e-commerce. E-commerce applications frequently require customers to divulge many personal details about themselves that must be protected carefully in accordance with privacy principles and regulations. Here, the authors define a privacy ontology to support the provision of privacy and help derive the level of privacy associated with e-commerce transactions and applications. The privacy ontology provides a framework against which e-commerce sites can benchmark their privacy policies and implementations.

Michael Hecker, Tharam S. Dillon, and Elizabeth Chang
Curtin University of Technology

Privacy in e-commerce and computing has attracted more and more attention over the past decade, with several incidents highlighting its importance. For example, between 2002 and 2003, JetBlue Airways released millions of customer records to a private US Department of Defense (DoD) contractor, directly violating JetBlue's privacy policy; the DoD then merged that data with data from a different source to identify possible terrorist suspects.[1] In a different case, DSW Shoe Warehouses revealed that it had lost roughly 100,000 data records containing personal information, including credit-card numbers and other personal data, which has been used for fraudulent activity.[2] Customer concerns about privacy protection are a growing inhibitor to the transition from traditional commerce to its electronic counterpart. A firm involved in privacy violations can experience a severe financial impact due to the costs associated with impact determination, notification, and recovery[3] as well as a loss of market value and capitalization.[4] Because privacy violations negatively affect all parties involved (from customers to clients to companies), organizations have a substantial stake in protecting consumers' privacy as thoroughly as possible.

Privacy on the Web faces massive problems due to two major factors: first, "the inherently open, nondeterministic nature of the Web"; second the "complex,

1089-7801/08/$25.00 © 2008 IEEE

leakage-prone information flow of many Web-based transactions that involve the transfer of sensitive, personal information."[5] The current standard for privacy preservation on the Web is the Platform for Privacy Preferences (P3P)[6] in conjunction with A P3P Preference Exchange Language (APPEL).[6] P3P defines a platform through which service providers can describe their privacy policies electronically. With APPEL's help, consumers can match their own preferences to the provider's policy and decide whether to use a given service. Although P3P automates the decision process for matching user preferences, it doesn't let users control their own information once they've entered it. Another concern comes from P3P's inability to link concepts from different domains together and match user preferences accordingly.

Privacy on the Semantic Web has even more severe implications, given that by its very nature, it annotates and links sources together. Security researchers have made several proposals for creating a highly abstract *security ontology* (based on security patterns, for example[7]). Similarly, a *privacy ontology* could use such security principles and mechanisms to support privacy methodologies and keep information under the control of the data's subject.

We propose a privacy ontology that will enable agents to understand content on the Semantic Web. Principally, if agents understand the concepts in their own domains as well as privacy concepts and their impact on privacy levels, those agents could exchange information and services while preserving security and privacy requirements.[8]

Developing the Privacy Ontology

We can view an ontology as "a shared conceptualization"[9] of a domain on which all parties agree. We could integrate privacy concepts into an application-specific domain ontology, but to do so is limiting — it ties concepts of privacy to the application-specific domain and doesn't permit reusing such concepts in other domains. We can represent a domain of interest (privacy) via both generic and specific ontologies.[9] Specific ontologies are also known as *ontology commitments*[10] — they commit to using all the higher-level ontology concepts and specifications.

A privacy ontology shows different concepts and the associations between those concepts, enabling interoperability and letting us determine the impact or privacy level a given (digital or nondigital) transaction has on a data subject when he or she agrees to enter it. This benefits other transaction participants as well because they can essentially use the ontology to model their (privacy) policies and procedures to comply with regulations within their domains. The ontology can also guide system developers who need to implement privacy functionality or mechanisms by mapping the concepts and making it clear what privacy actually refers to, without requiring those developers to be experts in the privacy domain.

To create our privacy ontology, we must first develop a glossary of terms, which requires us to collect information about what privacy mechanisms and privacy principles are currently available. Generally, legislative documents provide a solid foundation for those concepts and are usually covered by individual nations' privacy regulations. We used the privacy notions and concepts from the European Parliament Directive 95/46/EC[11] because privacy legislation in the European Union is more protective than in many other countries. A more comprehensive and concise guide of these rules is available elsewhere.[12,13]

When developing a privacy ontology, we must model static concepts, which are very general and apply to privacy and access to information in general — for instance, resources and entities, and the relations between them. Later on, we add safeguards to protect resources and the processes that apply to them.

Privacy Concepts

Given that privacy deals generally with data subjects' ability to control information related to them according to their own interests, it seems intrinsically linked to security and whether to grant or revoke other entities the privilege of accessing that data. Hence, to start creating an ontology for privacy, we use the model of classical authentication and authorization.

Figure 1 shows a diagram of the different components required to authenticate an *individual* and authorize him or her to access a certain *resource* in a classical enterprise system. Each individual can have multiple *identities* (for different purposes, for instance). Every identity has exactly one `Credential` — a `Name-Passphrase`, `Certificate`, or even `No-Credential`, which provides anonymity for the

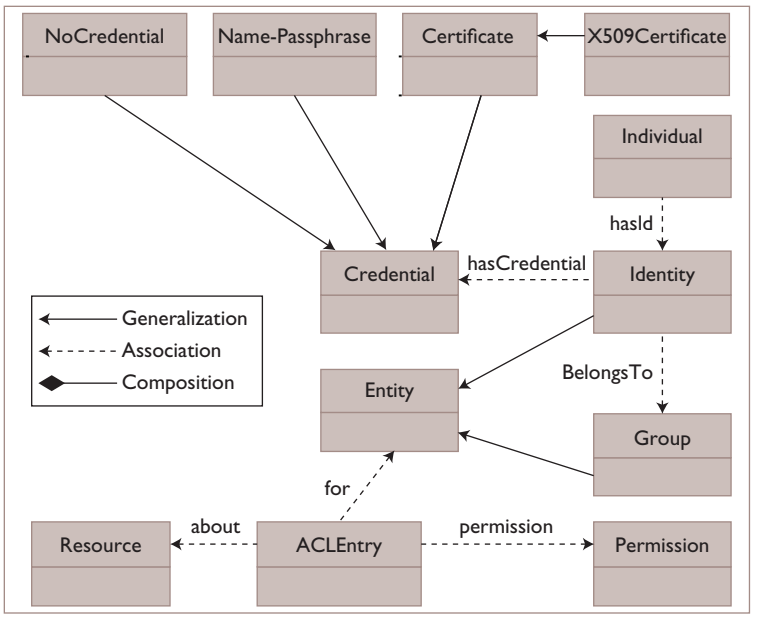

Figure 1. Classical authentication and authorization. Here, we can see the concepts involved in classical enterprise authentication and how they relate to each other.

individual behind it (for example, anonymous FTP or guest accounts). To grant (or deny) an identity access to a certain resource, an appropriate `ACLEntry` must exist. It defines which `Entity` (group or identity) can receive which permissions to which resources. However, such a system lacks a very important property — namely, the data subject that the information is about.

From a classical perspective, privacy is different from security. Security systems are usually designed in a way that protects the data from unauthorized intruders, while someone with total control over the system, and hence all the data, sets up the rules. A privacy system, on the other hand, must empower a client to control its own data and limit access to it, even from the system itself.

Accordingly, such a system requires a data subject, resources (about the data subject), and entities that will access those resources; Figure 2a shows this basic relationship, which is the first step toward a privacy ontology. This very basic ontology makes just one privacy-related statement — the expression that a resource is about a specific entity, the `DataSubject`.

However, a data subject might not necessarily want to be known by its resource, and the accessing entity might not reveal its identity to access just general information; thus, we need to introduce the concept of `Identities`. Both sides — the data subject and the entity accessing the resources — will want to control how they're identified, so the logical action is to associate a set of identities with every entity that will also apply to the data subject (after all, a data subject is just a more specific entity). Furthermore, an entity with multiple identities can act differently depending on which one it's using and can even choose to remain anonymous if necessary. We can categorize identities themselves as either able to identify the entity they belong to or not. We can also distinguish further among a nonidentifying `Identity`, a completely anonymous one, and a pseudo-anonymous one. The major difference between the latter two is that a pseudo-anonymous `Identity` can be reused multiple times, identifying just that pseudonym; an anonymous one is usually used just once and thus leaves no traces between different (trans)actions.

The different categories of identities also relate to the data subject's different types of resources. A resource can potentially identify a data subject directly or, even better, can identify one of the data subject's identities, which might be congruent with the data subject him- or herself.

On the other hand, a resource might not identify the data subject but might just indentify a pseudo-identity. Evaluating which identity category a resource belongs in isn't a precise process and must usually be classified by the data subject. For example, an email address can be a pseudo-anonymous resource as well as an identifying one, and, in general, the data subject can choose whether to use an email address with identifying characteristics.

Nevertheless, identities and differently classified resources (see Figure 2b) are required concepts for a privacy ontology because they support anonymity and let participating entities decide how they'll appear in transactions or how others will see them in general.

Next, we require concepts and mechanisms to actually grant entities access to the multiple resource types. To specify different levels of access, we specialize the entity concept by introducing subsets of concepts. Figure 2b shows their hierarchical organization and that they support various levels of access to the information in question.

A `ResourceAuthorizer` that acts on the data subject's behalf (and with its permission) is a

rather common and important component. For example, in a medical scenario, parents usually make decisions for their children if those children are still minors.[14]

Data subjects usually have total control of their own resources because they're both the `ResourceModifier` and, implicitly, a `ResourceAuthorizer`. However, data subjects can only make active changes to their own policies if they're alive, so we must further classify a data subject as either alive or not. This becomes very important when we consider certain rules and regulations in various countries because living entities receive different treatment as regards privacy than those who have died. Figure 2b shows the relevant relationships between the different entity types but, for the sake of clarity, we omit any associations between other components. The relationship between the `ResourceReader`, `ResourceAdder`, and `ResourceDeleter` and the `ResourceModifier` is a simple union because it combines those three resource concepts. Figure 2c shows the entire entity hierarchy.

Today, privacy concerns frequently arise when data travels across national borders and entities collect and process data in a different country than the one in which the data subject resides. However, this is more a legal problem because every country has its own (if any) laws protecting personal information. Hence, we must add support for different legal territories to our ontology, a discussion that's beyond this article's scope.

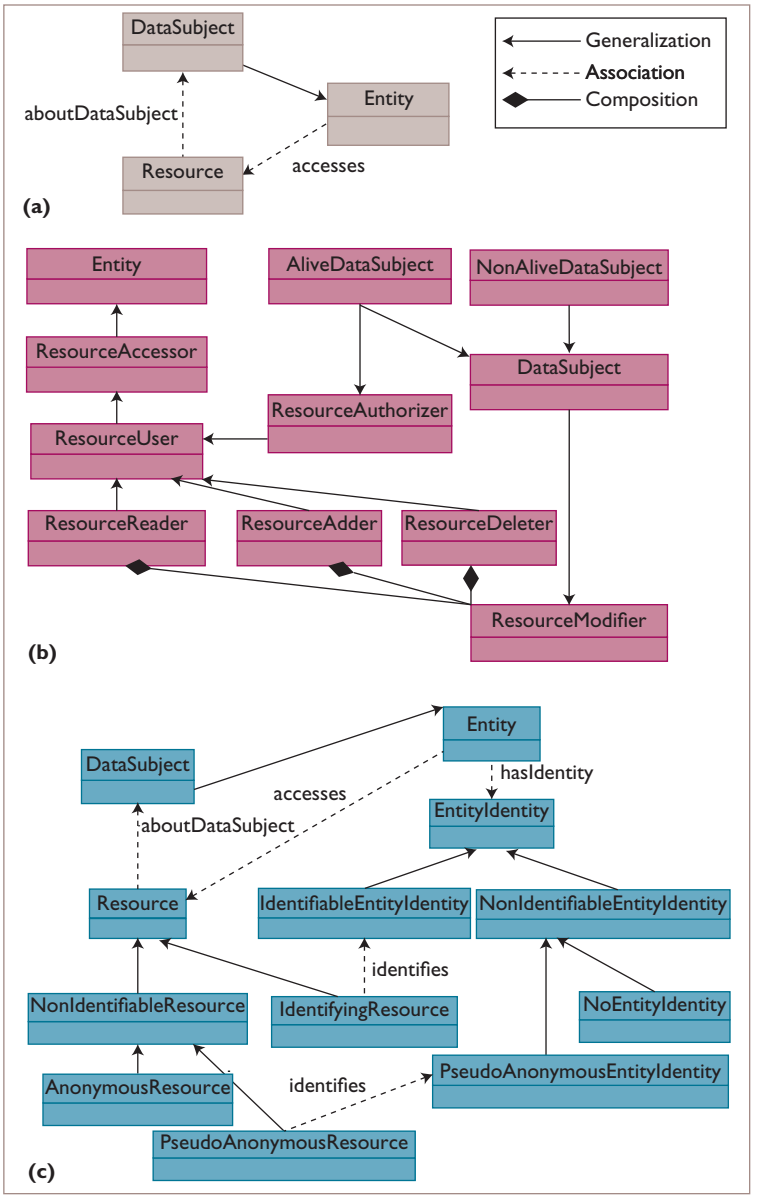

Figure 2. Privacy ontology development. At its most basic, we can see the (a) resources and data subject. We then add (b) the entity, and ultimately have (c) an entity hierarchy.

Policies

A policy refers to a concept that specifies a definite course or method of action and guides and determines present and future decisions. Hence, a policy is also a resource — more specifically, a resource about a resource. The concept of "consent" is part of every privacy policy. Usually, a `ResourceAuthorizer` — that is, the data subject in most cases — issues consent. We categorize consent in multiple ways: `Explicit`, `Implicit`, `NoConsent`, and `UndefinedConsent`, the latter of which applies when we don't know whether consent was given. Implicit consent usually occurs when we can assume that the data subject is either vitally interested in giving it or legally obligated to do so (for example, recording IP addresses might not be in a data subject's interests but might be required by law to prevent fraud, meaning telecommunication providers have implicit consent to collect and store such data). A life-threatening situation would also necessitate implicit consent for accessing medical information because we assume that saving the data subject's life is more important than privacy protection.[14] Additionally, consent has certain conditions and purposes, and might be valid only for a certain period.

Safeguards

Security is a vital part of supporting privacy.

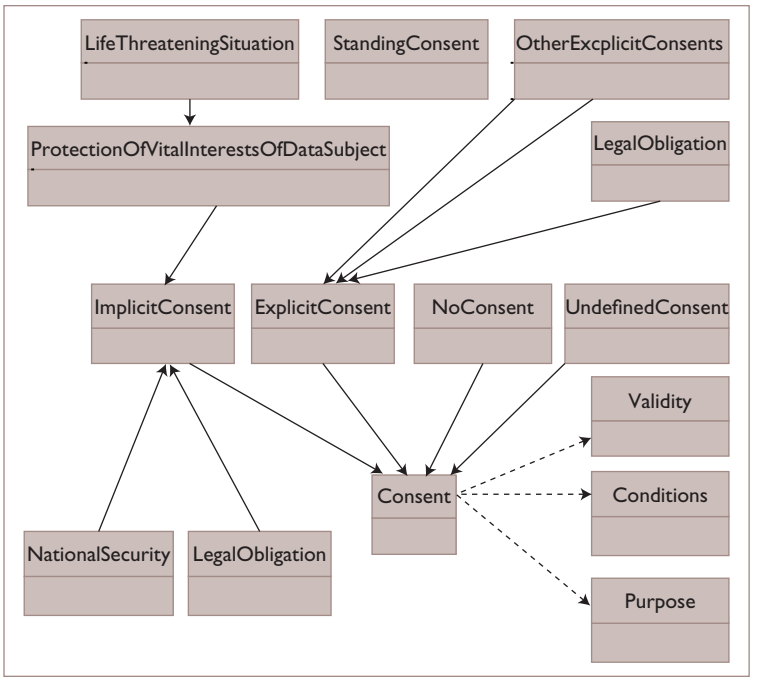

Figure 3. Consent. The hierarchy describes different types of consent with related concepts and links.

Security methods offer appropriate protection from privacy loss in many circumstances. Fortunately, security concepts and patterns as well as ontologies based on them already exist,[7] so we can easily incorporate them into our work. The term `Safeguard` refers generally to any method that protects a certain asset from attackers. However, the term *asset* is too broad for our requirements for a privacy ontology; rather, we assume that the classification `Resource` is a subset of such assets. Threat protection applies in two main ways, depending on a resource's state, which can be in transit or not (that is, in storage). However, because a resource's state isn't its own property but rather the property of the transaction it might be in, we can't apply additional protective measures to the resource itself when in transit. It's more useful to apply such safeguards to transactions, or what we call processes and activities. Hence, any process can have certain safeguards that protect valuable information in that process, but that safeguard must match the process's needs. For example, a process transmitting data from one entity to another can't use a safeguard that covers only data not in transit. On the other hand, we must clearly implement such a safeguard if we want to protect resources themselves when they're stored in any way: we could apply normal encryption methodologies to any instance of a resource to secure it from attacks and hence potential privacy loss.

We can use many other techniques as safeguards for protecting or hiding information from unwanted assessors, but we limit ourselves to only a few while keeping the ontology open to extend them if necessary.

In addition to protecting information directly, we might also wish to be able to assess the quality of the entities interacting with the information. One way to do this is through *trust and reputation*, for which ontologies have already been developed.[9]

Standardizing Terms and Concepts

You might know quite a few of the terms we've introduced by the different names they have in the literature. To give terms common names, we use resources such as standard higher-level ontologies[15] as well as current Web privacy movements (for example, P3P) to streamline and clarify the ontology. However, because multiple names for equal concepts do exist, we provide synonyms we know to be more common. For example, the term `Entity` as we use it here is very broad, but the standard higher-level ontology, for example, might use the term `CognitiveAgent`, which is essentially "the legal notion of a person."[15]

Specialized vs. Generic Ontologies

The ontology we've presented so far contains some very generic concepts that support privacy or derive the level of privacy associated with a particular transaction or application. However, such an abstraction level should cover almost any area, both digital and nondigital.

To apply the privacy ontology to the e-commerce domain specifically, we need to derive a more specialized ontology, mapping between these core privacy concepts to e-commerce ones. As we explained earlier, the specialized ontology commits to using all the upper ontology concepts and specifications

An example of such a commitment is the generic "transaction" concept, which we define as a relationship between two parties exchanging resources. A commitment of such a transaction could be a "purchase," in which two parties exchange money and goods.

As we saw from the JetBlue privacy policy violation example, it's important to check the transfer and processing of personal informa-

tion with regards to privacy. In this case, we assume that the transfer of personal information — the *passenger name records* (PNRs) to be precise — from JetBlue to a government department was accidental, and no manual or semiautomated processes were in place that could have checked for possible violations and prevented the transfer. Had an ontology-based checking mechanism been in place, it wouldn't have allowed the release of personal information, thus any release wouldn't have been accidental.

Let's look at the steps required to set up the ontology commitment, using JetBlue's privacy violation as an example. We assume that we have a domain expert who is creating the required parts of the e-commerce ontology for the first time. The expert starts by extracting a set of comprehensive domain concepts, either manually or through automated procedures (such as text mining) and then assigns them proper names and relationships.

For our purposes, we show only the most important concepts and relationships, although the application domain contains countless concepts. The first concept would be the PNR, which contains information such as the passenger's name, contact details for the travel agent or airline office, ticketing details, and itinerary listings. The PNR could be even more comprehensive due to government requirements and might also contain the passenger's passport details, including number, nationality, and expiration date as well as a birth date and place.[1] Because we view the PNR as a resource, it comprises other resources or resource elements, which we must classify next. Not all the PNR's elements are necessarily related to the same application domain. Thus, the same resource might have already been classified with respect to privacy in a different application domain. In this case, the expert would simply reuse those concepts here — for instance, the consumer's name most certainly isn't specific to the e-commerce domain alone.

Once the domain expert finds all the resource-element concepts, he or she must specify their identification power. The passport number clearly identifies the associated data subject and the expert must categorize it accordingly. In the next step, the expert needs to assign ordinal values to the privacy principles each concept influences. The generic privacy ontology is hereby guiding the expert because it already describes the privacy principles that type of concept influences. Clearly, the domain expert must ensure that he or she assigns the specific concepts for the ontology commitment appropriately — otherwise the ontology wouldn't be complete and appropriate, and the outcome of any later privacy evaluation might be incorrect. Let's consider the consumer's full name in the PNR again. We've already determined that this is an identifier (although not necessarily unique). This association thus implies an influence on the anonymity privacy principle. Once the domain expert has created all elements of the PNR resource, he or she must determine the whole resource's impact. After the PNR is released in its entirety, all combined resource elements together influence privacy principles during potential evaluations. We can see that a combination of someone's full name, passport number, and birthday, for example, is a much more detailed identifier then those individual elements alone. Discussing how we implemented the algorithm combining each individual element's influential values is beyond the scope of this discussion.

After the domain expert has determined the resource concepts, the next step is to identify the actors in the domain. Generally, e-commerce, with its emphasis on business-to-consumer interactions, has the consumer (the data subject) and one or more parties that collect and process information related to the data subject. From the generic privacy ontology perspective, we would refer to these actors as entities, with further classification requirements. In our example, we limit our discussion to an agent who's collected data from the data subject (a travel agent, for instance), JetBlue as an abstract party, and the DoD contractor.

In reality, many more parties might be involved that have access to the data, such as courier services or agents managing bookings. It doesn't matter whether those parties can or should access the data, as long as they're involved with it somehow — for example, we can't assume that a courier service wouldn't read private documents given the opportunity. This is also true for cases in which data is encrypted and then transmitted because the safeguard used is part of the evaluation process.

Because we've derived the three parties involved from the entity concept, we must also specify their jurisdictions, which can significantly affect the subsequent evaluation. The

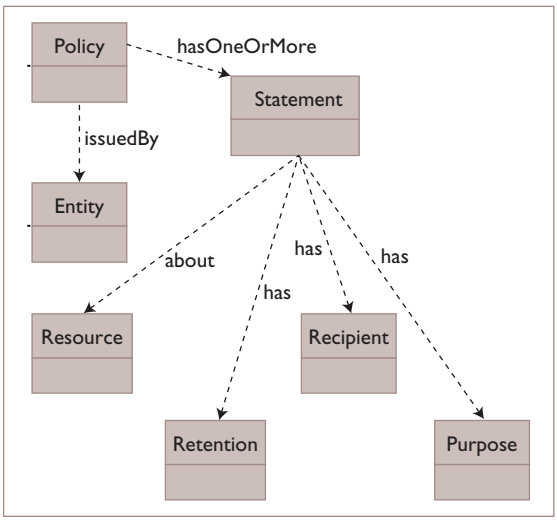

Figure 4. Privacy policy abstraction. This high-level abstract overview of a privacy policy shows the concepts involved and their interactions with each other.

laws and regulations that apply to an entity's jurisdiction, therefore, must be associated ontologically. This means that we must tap into an ontology for the judicative domain that accurately describes privacy laws. We won't attempt to create such an ontology here, but we assume we can create it with a reasonable amount of time and effort. Thus, we assume that laws and regulations are chosen by the domain expert to both protect privacy and keep other legal requirements in mind. With the JetBlue example, both New York state and federal laws and regulations (of which there are many[3]) apply. Without losing generality, we can also assume that all other parties involved fall within the same jurisdiction. Although it's not important for this small example, we would need to model many more concepts related to the storage and transfer of data between agents and JetBlue to determine how much protection the data subject can expect when entering data into the system.

In the next step, we must model the different concepts from JetBlue's privacy policy. Because many privacy policies are already available in a P3P format, it's relatively easy to generate the appropriate concepts from it and link them to the appropriate ones in the privacy ontology. In our case, the important statement in JetBlue's privacy policy is that it won't release any data to third parties that could potentially identify individuals. Figure 4 shows a high-level overview of such a policy.

As mentioned, we could model the relevant part such that our entity (JetBlue) has issued a policy stating that it keeps identifying resources for a certain amount of time (retention) and releases them only to JetBlue itself. Hence, we can derive that releasing identifying data to any other third party violates the policy. Also, given that our entity is bound by the jurisdiction we previously described, violating the privacy policy can incur harsh penalties. Note, finally, that it's impossible to actually have a perfect protection mechanism because once data's been collected, the collecting party could use it for any purpose if legislative mechanisms aren't in place to properly protect it.

In e-commerce, privacy is a significant factor in whether consumers adopt Web-based transactions. Here, we've highlighted some of the basic concepts for making a privacy ontology; in its current stage, ours is far more detailed and still developing. We hope that it will help form the foundation of privacy for e-commerce applications and Web sites. You can see some examples at http://privacy.debii.curtin.edu.au on how to use this ontology for e-commerce applications. The ontology itself is available in both an online diagram as well as a Web Ontology Language (OWL) version generated by Protégé, which we're using to enter the concepts and make inferences about constraints and relationships. In the future, we'll evaluate scenarios from different domains.

References

1. M. Hansen et al., *Overview of Existing Assurance Methods in the Area of Privacy and IT Security*, tech. report D5.1.a, PRIME Consortium, 2004; www.prime-project.eu/prime_products/reports/assur/.
2. "Another Week, Another Identity Theft Scandal: Recent Data Security Breaches Underscore Need for Stronger Identity Theft Protections," ConsumersUnion.org, 2005; www.consumersunion.org/creditmatters/creditmattersupdates/002244.html.
3. R. Herold, *The Privacy Management Toolkit Version 1*, D.J. Lineman, ed., Information Shield Publishing, 2005.
4. H. Cavusoglu, B. Mishra, and S. Raghunathan, "The Effect of Internet Security Breach Announcements on Market Value: Capital Market Reactions for Breached Firms and Internet Security Developers," *Int'l J. Electronic Commerce*, vol. 9, no. 1, 2004, pp. 69–104.
5. A.R.A. Bouguettaya and M.Y. Eltoweissy, "Privacy on the Web: Facts, Challenges, and Solutions," *IEEE*

Security & Privacy, vol. 1, no. 6, 2003, pp. 40–49.
6. L. Cranor et al., "The Platform for Privacy Preferences 1.0 (P3P1.0)," and "A P3P Preference Exchange Language 1.0 (APPEL1.0)," World Wide Web Consortium (W3C) specifications, 16 Apr. 2002; www.w3.org/P3P/ and www.w3.org/TR/P3P-preferences/.
7. M. Schumacher, "Security Engineering with Patterns: Toward a Security Core Ontology," LNCS 2754, Springer-Verlag, 2003, pp. 87–96.
8. L. Kagal et al., "Security and Privacy Challenges in Open and Dynamic Environments," *Computer,* vol. 39, no. 6, 2006, pp. 89–91.
9. E. Chang, T.S. Dillon, and F.K. Hussain, *Trust and Reputation for Service-Oriented Environments*, John Wiley & Sons, 2006.
10. P. Spyns, R. Meersman, and M. Jarrar, "Data Modelling versus Ontology Engineering," *Special Interest Group on Management of Data* (SIGMOD), vol. 31, 2002, pp. 12–17.
11. "Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the Protection of Individuals with Regard to the Processing of Personal Data and on the Free Movement of Such Data," *Official J. European Communities (OJEC),* vol. L, no. 281, 1995, pp. 31–50.
12. R. Leenes, J. Schallaböck, and M. Hansen, "PRIME White Paper v2," Privacy and Identity Management for Europe, June 2007, www.prime-project.eu/prime_products/whitepaper/.
13. J. Huizenga, *Handbook of Privacy and Privacy-Enhancing Technologies: The Case of Intelligent Software Agents*, College Bescherming Persoonsgegevens, 2003.
14. A.M. Hecker and T.S. Dillon, "Ontological Privacy Support for the Medical Domain," *Proc. eHPass National e-Health Privacy and Security Symposium*, Queensland Univ. of Technology, 2006.
15. I. Niles and A. Pease, "Towards a Standard Upper Ontology," *Proc. Int'l Conf. Formal Ontology in Information Systems*, 2001, pp. 2–9.

Michael Hecker is completing a PhD on privacy and is also a computer systems officer at the Digital Ecosystems and Business Intelligence Institute (DEBII), Curtin University of Technology, Perth, Australia. His research interests include privacy and security as well as computer and Web systems in general. Hecker has a masters (Diplom) from Free University of Berlin, Germany. He is student member of the IEEE, the ACM, and the Australian Computer Society. Contact him at michael.hecker@curtin.edu.au.

Tharam S. Dillon is a Distinguished Research Professor at the Digital Ecosystems and Business Intelligence Institute (DEBII), Curtin University of Technology, Perth, Australia. His research interests include ontologies, Web semantics, and grid computing, as well as trust and reputation, object component-based conceptual modeling and design, and XML modeling. Dillon has a PhD in computer systems from Monash University, Australia. He is a fellow of the IEEE, the Institute of Engineers, Australia, and the Australian Computer Society. Contact him at tharam.dillon@cbs.curtin.edu.au.

Elizabeth Chang is the director of the Digital Ecosystems and Business Intelligence Institute (DEBII), Curtin University of Technology, Perth, Australia. Her research interests include ontologies, software engineering, object/component based methodologies, e-commerce, trust management and security, Web services, user interface, Web engineering, and logistics information. She is a senior member of the IEEE and a member of the Australian Computer Society. Contact her at elizabeth.chang@cbs.curtin.edu.au.