Privacy Policy-driven Mashups

Soon Ae Chun*
School of Business
CUNY College of Staten Island
New York, NY, USA
soon.chun@csi.cuny.edu
*Corresponding author

Janice Warner
School of Business
Georgian Court University
Lakewood, NJ, USA
warnerj@georgian.edu

Angelos D. Keromytis
Computer Science Department
Columbia University,
1214 Amsterdam Avenue,
New York, NY 10027
angelos@cs.columbia.edu

Abstract

Mashups are novel content created by extracting and combining data and services from diverse data sources, in an automated manner, using Web services. The Web 2.0 technologies make it easier for individuals to create contents in third party service sites or clouds, and make easier for other third party mashup organizations to access and combine individuals' data and content through mashups. A big unaddressed challenge is how to adequately protect the privacy of individuals when information about them in the data sources are to be accessed and joined by mashup providers, which is different from two-party interaction (i.e. an individual and Web sites) as in many online environment. . In this paper, we present a *Privacy Specification and Enforcement Model for Mashups* that considers privacy preferences expressed in three different logical networks: personal privacy policies (PPP), data source organization's privacy policies (SPP) and mashup organization's privacy policies (MPP). We present a privacy policy model using a multi-dimensional *privacy protection space* which includes parameters to specify private content, provider type, mashup-specific operators, and mashup purposes, using the open Semantic resources on the Web such as a Friend of a Friend (FOAF), service industry classification codes and UN product and service classification codes. The prototype architecture for the proposed mashup privacy protection engine is presented that interacts with distributed privacy policy networks to determine privacy-preserving data sharing and integration for mashup services.

1. Introduction

Mashups have extended the utility of electronic data available from web sites, allowing flexible content creation by integrating a combining multiple data sources including texts, images, videos and other media. Previously, data produced and posted by a Web application was "locked" in a data owner's site and users were passive "information sinks"[20]. In order to get access to the Web data or content in a form that could be manipulated, data consumers (users) needed screen scrapers, special purpose programs to process the contents. Alternatively, they could perform manual cut-and-paste operations, which would not allow for dynamic content creation. With the emergence of mashup editors and Web services, multiple data sources and services can be mixed and matched in creative ways to suit countless purposes. The relative ease of creating mashups is illustrated by the fact that dozens of mashups are created daily as can be seen on the Mashup Dashboard [27], The major categories of mashups are identified in [10], such as mapping mashups, video and photo mashups, search and shopping mashups, and news mashups among others. The mashup technologies include AJAX that is a bundle of technologies focused around the asynchronous loading and presentation of content, SOAP and REST Web protocols that facilitate communicating with the remote servers through Service Oriented Architecture, Screen Scraping, the Semantic Web RDF as well as RSS and Atom.

The relative ease of creation and amount of data available, however, has a negative side – unintended consequences in terms of lost privacy when data from multiple sources are combined with unexpected results. The case cited in the literature [35] that illustrates this point is related to a mashup of Amazon wishlist entries to profile a person as a subversive. A person might want to allow access to wishlist data from multiple companies to be mashed up together to create a global gift registry. However, they would not want the data on the wishlists to be used against them by mashup providers seeking only to defame.

When we consider the privacy of data used in mashups, we are concerned primarily with data types that identify someone or provide attributes that an individual may not want widely known. These attributes have to do with who they are, what they have, and what they have done. Clearly, the types of data an individual may want to keep private may depend upon who that individual is. Therefore, we propose a model for dealing with mashup privacy that is distributed and allows individuals to define their own preferences. In that way, the model is similar to one that uses P3P, the Platform for Privacy Preferences. Direct application of P3P, however, is not possible for mashups. P3P comes into play when an individual accesses a web site and is asked to provide information or accept cookies. The individual can make a decision as to whether or not to provide private information or use the services of that web site based on the privacy policies of the web site. In the case of mashups, the individual is not in direct communication with the web service provider and must trust that the data sources (i.e. data provider organizations) will protect data appropriately. A simple solution would be for data sources to allow users to opt-out of their data being used by any third party, including a mashup service provider.

This would be a fine solution for most mashup services - in general individuals do not want their private information generally available across the Internet. However, as the "programmable web" becomes more possible [27], there may be mashups which provide services to users, pushing information and support to them without them even asking for it or mashups which use private information to determine trends and statistics of benefit to the general public. In such cases, a user may wish their private information to be accessible but used in an acceptable manner.

Suppose we have a mashup service that draws from multiple resume collection web services and mashes data from the resumes with job postings data and map data in order to give perspective employers a set of relevant resumes that belong to people within mileage limits specified by the employers. Clearly someone with a resume hosted on one of the data source web site might be interested in putting themselves in front of perspective employers in this way. How could they allow this service (and other services deemed useful to them) to be given access to their data while preventing other mashup providers access to their address and mobile phone number?

We propose a fine grained mashup privacy protection system in a distributed network of data sources (data service providers or data providers), data owners (i.e. individuals whose data is being stored at the data sources) and mashup providers (data consumers). This approach allows users to specify what specific data types might be used in a mashup service and under what conditions the data might be released. These conditions depend upon the mashup provider, the mashup service's purpose and the operations to be performed. A similar situation exists for corporations participating in environments such as in an industry exchange. Member business might be happy to share data for certain purposes but not for others.

To implement our solution, we propose *privacy protection spaces* through which *privacy protection engines* determine whether data is allowed to be released, a *personal privacy policy network* through which an individual's preferences might be consulted by data sources as well as mashup providers. We define mechanisms for data release decision making as well as enforcement of an individual's preferences. We describe the components of our model individually and then combine them per an architecture and implementation description followed by a concrete example.

In section 2, we present an overall framework or a schematic component architecture to facilitate our overall approach. In section 3, a generalized mashup Privacy Protection Model is presented which is extended from our earlier work in government domain [12][13]. The model elements necessary for privacy specifications are introduced. Section 4 describes how to evaluate each policy and combined policies for mashup purposes. In Section 4, the prototype system design and implementation issues are discussed and explain the steps involved in the privacy policy enforcement. Section 5 describes some relevant literature followed by the conclusion section.

2. Overview of Privacy Protection for Mashups

In this section, we briefly summarize our approach with the overall architecture and its components as shown in Figure 1. The details are described in sections 3, 4 and 5. The mashup related policies involve a three party collaboration for protecting the private data sharing and mashup: (1) content (data) providers' privacy policy, (2) personal policy of an individual whose data is stored in the data source organization as well as (3) the mashup provider's data privacy. Each of these three policies form a logical network which is marked in dotted lines in the diagram, hence, the PPP network for personal privacy policies, SPP network for different content source organizations' privacy policies and MPP network for different mashup service providers' privacy policies. What we propose is to provide an infrastructure or a portable (e.g. app) platform, called Privacy Policy Discovery and Evaluation Engine for Mashup services, to discover three types of policies distributed at each network when a mashup service is requested. It evaluates and executes the composed policies that honor personal privacy policy, source organization's privacy policy and the mashup provider's privacy policy. In addition, the privacy policies distributed in different networks can be registered to this infrastructure through the registry service to facilitate discovering and locating the policies to be evaluated for mashups.

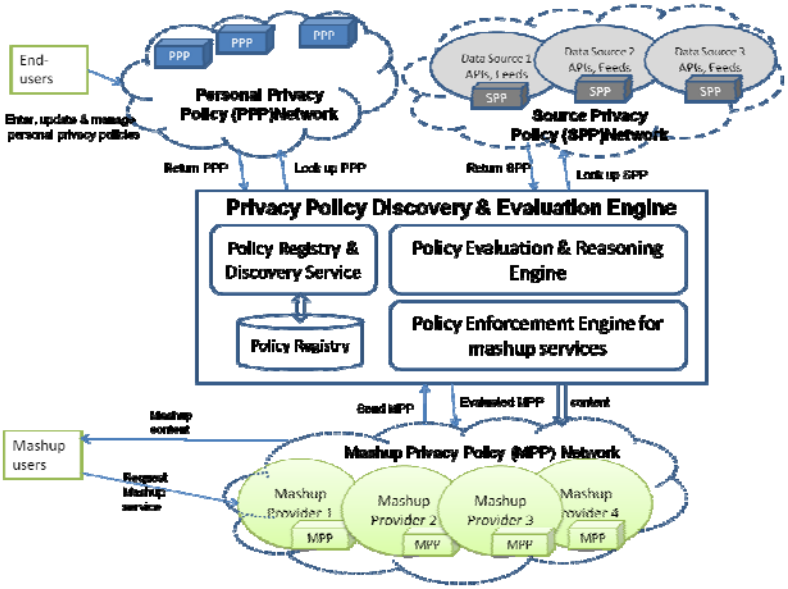

Figure 1 Overall Mashup Policy Discovery, Evaluation and Enforcement Engine

Each privacy policy network has tools for the end users or organizations to specify and manage their own privacy policies for mashups. A policy repository in each network can be accessed by our policy evaluation engine to retrieve policies as needed. The end users who are not experts in policy specification can use a modeling or editing tool which provides the users with semantic ontologies or hierarchical metadata classification systems (e.g. Standard providers' codes) to express the fine-grained privacy policies.

3. Mashup Privacy Policy Model

Three sets of privacy policies apply to mashups. First is the privacy policy of the web service providing the data, the *Source Privacy Policy* (SPP). Mashup providers are also expected to have published privacy policies, their *Mashup Privacy Policy* (MPP). Content providers can consult these policies before releasing data. As will be described later, we propose that these policies contain an indication of the mashup purpose. Data content providers could choose not to release data for use in mashup services when the purpose is not acceptable to the individuals whose privacy is being protected.

To complement the privacy policies of the web services, we propose to add fine grained control by individuals represented by the content. Our privacy protection model allows individuals to specify and publish privacy preferences, called *Personal Privacy Policy* (PPP), which are published on repositories accessible using Uniform Resource Identifiers (URIs). Of concern are data content providers who have information deemed to be of a private nature. This would include name, identification number, financial information and/or any other information that an individual may want to keep private from the general public. Data content providers would consult the URIs on behalf of those represented by their content when making a decision concerning release of data to a mashup provider. In addition, the use of PPP will allow for a notification service for individuals of privacy violations due to mashups. Finally, the PPP can

be used to make the mashup provider aware of the personal privacy policies that impact the implementation of its service with an indication of why a specific data access is denied.

3.1 Mashup Privacy Protection Spaces

Individual privacy protection may be based on five aspects of the data and mashup service:

- **Data Types (DT)** - the set of personal data used in the mashup service;
- **Linking Parameters (LP)** - the set of personal data used to link data sources together;
- **Operations (OP)** - the set of operations that will be performed on the data sources;
- **Provider Type (PT)** - the type of mashup service provider;
- **Purpose (MP)** - the purpose of the mashup service.

A *privacy protection space* is formed by combining specific values specified for these five parameters. Upon protection space instances, constraints are placed authorizing or restricting release of data when the parameters are met. Individuals may have multiple protection spaces thus allowing private data to be shared under certain protection spaces and not in others. Each value for each parameter can be accompanied by a positive (+) or negative (-) sign, indicating whether the space is inclusive or exclusive of the parameter values.

3.1.1 Data Types

The data types, DT, describe the individual personal data items that should be protected. They are one or more uniform resource identifiers (URIs) which reference resource descriptions written in XML using the resource description framework (RDF). For example, we use The Friend of a Friend (FOAF) documents on the Web written in FOAF XML vocabulary that describe people and things related to the people, such as images, forming decentralized linked information system. The FOAF documents contain not only the basic information about people, organizations, or groups, such as names, addresses, but also how they are related to things, such as ”attend meetings,” or ”is depicted with photos,” etc.

Thus, FOAF vocabulary consists of tags mixed with RDF and OWL statements to describe this personal information and its linking relationships to other objects or concepts (other URIs). The FOAF documents in RDF/OWL formats contain publicly available personal information that can be referenced from other FOAF elements. For example, the Friend of a Friend (FOAF) schema (RDF document) describes many personal data types such as “name” (defined as foaf:name), “membership” (defined as foaf: member) and “img” (defined as foaf:img), as well as other personal information, such as interests and links to friends or groups (defined as foaf:knows).

```
<foaf:Person rdf:about="\#js" xmlns:foaf="http://xmlns.com/foaf/0.1/">
<foaf:name>John Smith</foaf:name>
<foaf:homepage rdf:resource="http://homeserver.org/~smith" />
<foaf:img rdf:resource="http://server1.org/smith.jpg" />
<foaf:knows rdf:nodeID="Jane"/>
</foaf:Person>
```

This FOAF document contains the name of John Smith, and links to his home page and image URIs described in RDF resources, and states that John knows Jane (assume Jane’s foaf RDF is defined). FOAF can contain OWL statements that link to other ontology class resources and relationships, such as John’s membership in a group, which is a subclass of an organization.

A DT to be protected for privacy is specified with a tuple *(E, URI, Sign)* where E represents the protected element, *URI* denotes the optional resource URIs where E can be located, and *Sign* to show the inclusion in the protection space. For example, the protected information is the name of the person specified in the RDF schema as in the URI with 'js' in the specified Web site: *DT= (foaf:name [rdf:resource="http://exam.com/test.rdf#js] sign="+")*. When a DT is specified, all equivalent DTs and all subclasses of the DT are also included in the protection space. When a DT is specified with a "+" sign, it means that only those data types listed are covered by the protection space. When a DT is specified with a "–" sign, it means that all data types other than those listed are covered by the protection space.

3.1.2 Linking Parameters

The linking parameters, LP, are data types which may or may not be used in mashup services for combining data from multiple sources. As data types, they are also described by URIs using publicly available schemas or ontologies. The main purpose of including LP in the protection space is to ensure the integrity of the combination of data linked from different sources. For example, if a person has a very common name, they might be concerned if their name were used to link data from different sources. The results might include data that is not really associated with them. Therefore, they might want to prevent any combination that linked based on name. When a LP is specified with a "+" sign, it means that only those linking parameters listed are covered by the protection space. When an LP is specified with a "-" sign, it means that all linking parameters other than those listed are covered by the protection space.

3.1.3 Operations

The following shows a list of mashup operations, OP that the mashup providers may perform on a set of data types linked from multiple sources.

- ***Combine:*** The *combine* operation takes data from multiple sources and displays it together. For example, an individual's name and address might be linked with data from a campaign donations database and displayed together. We represent this operation in our privacy protection space as DT = (name; address; obamaForPresident:donation) where name, address and donation would actually be URIs.
- ***Represent:*** The *represent* operation takes data from multiple sources and displays a representation of it. For example, an average income for individuals in a particular zip code might be calculated from income tax data by a mashup. The *represent* operation has many suboperations, all of which are mathematical operations on a particular data type such as total, count, max, min, and median. In general, the represent operation may be used to characterize a set of data records and would in itself not violate the privacy of any particular individual.
- ***Overlay:*** The *overlay* operation links data and displays it in multiple ways. A typical instantiation is mapping - taking an address from one database and displaying it on a map. Alternatively, the data points could be graphed on a scatter plot or linked with photographs.
- ***Sequence:*** The *sequence* operation links data from different sources in a temporal sequence. For example, addresses might be displayed over time.

- ***Personalize:*** The *personalize* operation links data with content in which a person is expected to be interested based on an analysis of the data values. The mashups use the data to categorize an individual and customize the service for her/him.

In the absence of particular purpose, these operations can be used to constrain the data release. Thus, individuals may choose to restrict mashup providers who *combine* a set of data but not those who *represent* the same data in an aggregate form. When an OP is specified with a "+" sign, it means that only those operations listed are covered by the protection space. When a DT is specified with a "-" sign, it means that all operations other than those listed are covered by the protection space.

3.1.4 Provider Type

Provider Type (PT) is a tricky aspect to define for protection spaces. It is included as a proxy for "trust" in the provider to use the data only for stated purposes. This is similar to how trust is implemented in P3P where some web sites are allowed to leave and access cookies and others are not.

In the same manner as P3P, PT can be a list of web sites that are part of the protection space. Both positive authorization and negative authorization could be given. Negative authorization might provide very little privacy protection, however. This is because it is envisioned that mashups will be created by anyone across the Web. It would be nearly impossible to block every potential mashup provider by explicitly listing them. Negative authorization provides better protection but would clearly limit the utility of any new mashup service that relied on private data.

Alternatively, PT can be specified via industry codes such as those standardized in the North American Industry Classification System (NAICS) [21]. The NAICS codes are six digit codes where the first two digits indicate a broad industry such as agriculture, health or finance. Each additional digit indicates a more specific industry within a larger industry. Figure 2 show a piece of the NAICS classification scheme for the Finance and Insurance Industry. NAICS codes are set up hierarchically. Thus values for PT can be more or less specific. For example, if someone were particularly sensitive to use of data by the financial industry, they might set up a privacy protection space that included PT = 520000 to constrain release of private data to mashup providers in that entire industry. Alternatively, if the person was concerned only with credit card companies, they might specify a PT value of 52221.

An example of mashup service where NAIC classification system is used to provide the same semantics for data providers is the Longitudinal Employer-Household Dynamics (LEHD) program at the U.S. Census Bureau [38]. The LEHD program is a data mashup service to produces a summary reports that combine federal, state and Census Bureau data on employers and employees, according to the NAIC's sector, industry and subcategory classification [37].

When a PT is specified with a "+" sign, it means that only those provider types listed are covered by the protection space. When a PT is specified with a "-" sign, it means that all provider types other than those listed are covered by the protection space.

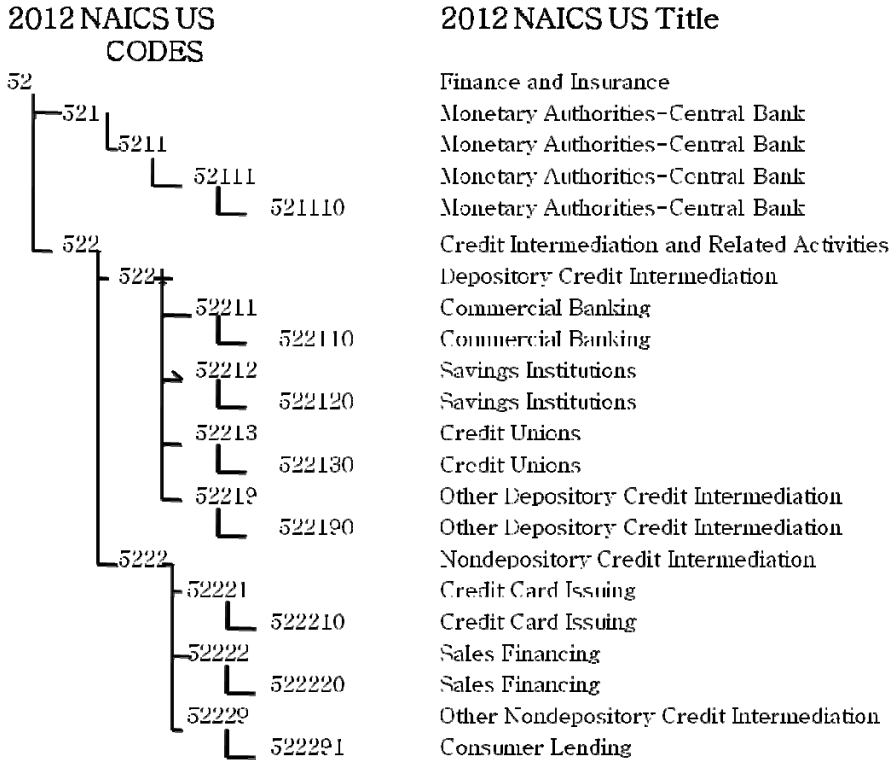

Figure 2 Excerpt from the North American Industry Classification System (NAICS) used to specify Mashup Provider Type

We continue to consider other ways that trust of a mashup service could be represented and calculated. Calculating and evaluation trust in web services is a topic that has been covered by several researchers [39][32][41]. They point out that trust may depend upon many different factors, is context specific (web services might be trusted for some things and not others) and varies over time. Liu [39] motivates the need for multi-dimensional representation of trust depending upon multiple factors. This is clearly true for a web service consumer who would like to choose services based on various factors. This is not the case for mashups because the individual(s) concerned is/are not the consumer(s) of the service. Concerning mashups, there is only one dimension - how well the results of the Mashup match the stated purpose of the mashup. Over time, a third party might collect and calculate reputation scores. If this happens, we would augment our protection space with a trust parameter which would specify the requirements for the reputation scores.

3.1.5 Mashup Purpose

Mashup purpose, MP, is categorized based on intended usage of the mashup. Since mashups are services, we represent MP by identifying the service provided by the mashup. OWL-S, the semantic web ontology, used to include service category as part of the service profile. Now it is in a separate file to allow for easier modularization. Service categories make use of an ontology of services that may be on offer. High level services could include classification on the bases of industry taxonomies such as North American Product Classification System (NAPCS) [22] or the United Nation Standard Product and Service Code (UNSPSC) [36].

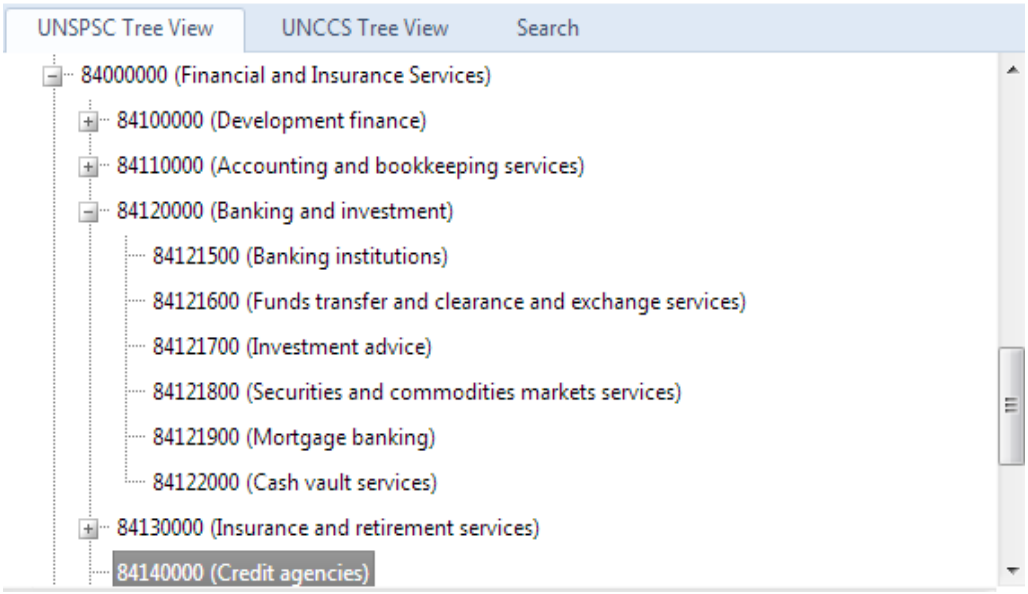

Figure 3 Excerpt from the UN Standard Product and Service Code (UNSPSC) used to specify Mashup Purpose

In those categorization schemes, all mashup services might be categorized as information services. While true, we are more interested in the purpose of presenting the information in the form provided. For example, if a mashup service provides a combined view of mortgage history for an individual from disparate organizations, the mashup purpose might be *mortgage banking* (UNSPSC code 84121900). A small excerpt of the UNSPSC codes in Financial and Insurance services domain can be seen in Figure 3. Similarly, the implementation of a mashup purpose (MP) can use the North American Product Classification System (NAPCS). Figure 4 shows the example of product/services provided by NAIC service providers 52 which is the financial and insurance providers and these can be used in the MP specification in the privacy policy. When a MP is specified with a "+" sign, it means that only those purposes listed are covered by the protection space. When a DT is specified with a "-" sign, it means that all purposes other than those listed are covered by the protection space.

In a data warehouse where data from different sources need to be integrated, some data is classified into the same UNSPCS according to their function, purpose or task before the data is being integrated as the same record in the data warehouse [4]. For our job search example, we may use the UNSPCS code 93141800 for employment which has subclass codes ranging from 93141801to 93141814, including 93141810 for Career development services.

Industry Subject Area	Working Group Code	United States		
		Title	Definition	NAICS Industries Producing the Product
52	1.1	Financing services	Providing services that result in the provision of money and granting of credit to businesses, consumers, and governments.	522110 522120 522130 522190 522210 522220 522291 522292 522293 522298
52	1.1.1	Loan services	Providing direct lending of funds under legal contract, either unsecured or secured by the assets being financed or by other assets, but without the exchange or the use of securities as collateral. Includes: • interest and origination and other fees received from sales of loans.	522110 522120 522130 522190 522220 522291 522292 522293 522298
52	1.1.1.1	Loans to financial businesses	Making loans to financial businesses. Includes interest received, origination and other fees received, and revenue from sales of loans. Includes: • interest received and origination and other fees received from sales of loans. • loans made to banks, trust companies, investment dealers and brokerages, and insurance companies, etc. Excludes: • providing financing using purchase-repurchase agreements is in product 1.3.2, Repurchase agreements.	521110 522110 522190
52	1.1.1.1.1	Loans to depository financial institutions	Making loans to depository financial institutions, such as banks. Excludes: • providing financing using purchase-repurchase agreements is in product 1.3.2, Repurchase agreements.	521110 522110 522190
52	1.1.1.1.2	Broker's call loans	Making loans to security and commodity contract brokerages, used to finance underwriting costs and margin lending, usually short-term and secured by securities. Excludes: • providing financing using purchase-repurchase agreements is in product 1.3.2, Repurchase agreements.	522110
52	1.1.1.1.9	Loans to financial businesses, nec.	Making loans to financial businesses, not elsewhere classified. Excludes: • making loans to depository financial institutions is in product 1.1.1.1.1, Loans to depository financial institutions. • making call-loans to security and commodity contract brokerages is in product 1.1.1.1.2, Broker's call loans. • providing financing using purchase-repurchase agreements is in product 1.3.2, Repurchase agreements.	522110 522190

Figure 4 Excerpt from NAPCS Product List for NAICS 52: Finance and Insurance

3.2 Personal Privacy Policy Network

Individuals may cite their preferences concerning release of their private information to mashup service providers by registering their preferences in a repository. A repository is needed because requests for their private information are independent of interaction with them. Repositories that house privacy preferences are assumed to be widely known and accessible to web services through URIs. Web services that provide data to others can, in fact, ask individuals for their preferences when they collect data and submit preferences to the repository on their behalf or refer them to a repository. This was the case in the example provided in our last section.

Collections of privacy preference repositories can form a distributed network, such as in different clouds. We refer to this as a *personal privacy policy network (PPP network)*. The PPP network serves as the publish and pull infrastructure for personal privacy policies, augmenting legal and organizational specific privacy policies to provide more fine-grained privacy control.

Personal privacy preferences in a repository can be maintained and updated by individuals. If a web service enters a privacy preference on behalf of an individual, the individual should be issued with a certificate or credential such that they can access the privacy preferences repository and update the policies as needed. The credential could be electronic or a smart card or both. The smart card would serve both as an identity card and a store of information that gives the user authority to access information about themselves and change policy governing that information. It can also include links to identification IDs associated with various services.

The PPP network will allow individuals to have more control over their own private data, through direction participation in protecting the data. This participatory privacy protection accommodates a high degree of individual differences in privacy and may foster greater levels of trust in web services that collect data.

3.3 Specifying Personal Privacy Policies

Mashup privacy protection is related to but different from privacy protection provided via P3P. P3P governs interactions with a web site. Users may consider a web site's privacy polices before submitting data and may opt-in and opt-out of certain uses. P3P policies are described using XML statements which include the following elements: purpose, recipient, retention, data-group. Purpose describes use of the data by the web site. Some of the defined purposes include:

- **current:** completion and support of activity for which data was provided
- **develop:** information may be used to enhance, evaluate, or otherwise review the site, service, product or market.
- **tailoring:** information may be used to tailor or modify content or design of the site where the information is used only for a single visit to the site and not used for any kind of future customization
- **pseudo-decision:** information may be used to create or build a record (a profile) of a particular individual or computer that is tied to a pseudonymous identifier, without tying identifying data to the record.
- **individual-decision:** information may be used to determine the habits, interests, or other characteristics of individuals and combine it with identified data to make a decision that directly affects the individual.

- **contact:** information may be used to contact the individual for the promotion of a product or service.

Users could protect their data by specifying only those purposes acceptable to them. That would implicitly exclude mashups. Alternatively, mashups could be added as another purpose to the P3P purpose list. In either case, mashups would be included or excluded. Decisions would not be fine grained. Therefore, we see a need to have privacy policies that are associated specifically with mashups.

Three sets of privacy policies need to be assessed when considering a request for content from a mashup provider that concerns personal information – individual's personal privacy policies (PPP), source content provider policies (SPP) and mashup service provider policies (MPP). The first set is the personal privacy policies of any individuals for whom personal information is requested. Personal privacy policies (PPP) use protection spaces as the basis of specifying an authorized or an unauthorized use of private data by a mashup service provider. For our privacy policy specification in support of mashups, authorization or non-authorization is specified by just two commands – ALLOW and DENY which can be applied to any particular privacy protection space.

Let us consider the example introduced in Section 1. Suppose, up to now, Jane Smith has no personal privacy policies because she does not know of any instance of her personal data appearing on the Web. However, she submits her resume electronically to a resume listing service "JobMatch.com". At this point because she is submitting personal data to a web service that has notified her that they will share her personal data, she decides she should publish a PPP. Although she could have implemented a PPP independent of interacting with "JobMatch.com", "JobMatch.com" is pro-active and asks whether Jane has a PPP registered when she signs up for their service. When she says she does not, they offer to register a PPP for her with private data types included in the "JobMatch.com" service already populated. While there is a lot of personal information in her resume, let us say that Jane is only sensitive about protecting her mobile phone number which she tends to keep private. Therefore, she selects *v:mobileT*el where *v:* indicates the vcard ontology [3] which provides more specific definitions of telephone number than available from the foaf ontology.

For linking parameters, given she has a very common name, she is concerned that she will be confused with other Jane Smiths. Therefore, she requires that linking be done on name (foaf: fullName) and address (foaf: address). She has no particular restrictions concerning the operations performed on her data.

For mashup purposes, she specifies UNSPSC 93141810, the code associated with career development services. For provider type, she allows any mashup provider who can help her find a good job is welcome to see the personal information associated with her resume. For that reason also, she provides no further specification for provider type. , .

In summary, her first PPP is PPP_1 = *ALLOW {DT = [-v : mobileTel], LP = [foaf : fullName; foaf : address], OP =∅, MP = unspsc : 93141810, PT =∅}*. This policy is stored in a PPP repository on a PPP network accessible to be read by any web service. It allows access to any mashup provider who provides career development services to all data except mobile telephone number but restricts them from linking data to other sites unless there is a match between Jane's full name and address. All other data releases are not acceptable to Jane and the implication is that they are denied. While this PPP was defined in

response to input of data at a web site, it would apply to any web service that is capable of releasing information. PPP_1 can be represented in a pseudo XML-based policy language, as shown below:

```
<POLICIES>
<POLICY id="jane.smith.ppp1" uri="http://myppp.com/jsmith547>
  <RULE directive=ALLOW>
    <DATATYPE name="v:mobileTel" sign=negative>
    <LINK name="foaf: fullName" sign=positive>
    <LINK name="foaf: address" sign= positive>
    <PROVIDERTYPE />
    <OPERATION />
    <PURPOSE code=unspsc: 93141810 sign= positive>
  </RULE>
</POLICY>
</POLICIES>
```

Over time, Jane could add additional preferences by defining additional protection spaces. She could also relax or constrain PPP_1 by adding or removing parameters from the protection space.

Source content providers also have privacy policies - an instance of which is called a *source privacy policy* (SPP). It is constructed exactly the same way as a PPP. When an SPP is modified to comply with one or more PPPs, the revised policy is called an SPP*.

Mashup providers are requested to submit the mashup privacy policy (MPP) associated with the service for which the request is being made. While consisting of the same parameters as the other policies, the values of the parameters are more specific. The data types provided should be those data types that are requested. The LP is any requested parameters that will be linked across content sources. The operation will consist of operations that will be performed on the requested data. Two provider types will be provided - the specific identity of the provider as well as an industry code. Finally, the purpose will be provided using service codes.

4. Mashup Privacy Policy Evaluation and Enforcement

Today some web service providers will only allow access to their data to pre-certified mashup providers to whom they have given a credential, key or access login. This paradigm of data release is limited in a large scale environment such as the Web. Our proposed model opens up the process by allowing mashup providers to gain access to data by indicating the business they are in (via provider type), the data types that they will use and the purpose of their mashup. If these are acceptable to the data provider and individuals associated with the data, data release can occur automatically.

Specifically, an interactive dialogue occurs between the mashup provider and the content provider(s) whenever the mashup provider requests data from the content provider(s). The mashup provider indicates the parameters associated with themselves and their service. Specifically, they provide their mashup privacy policy (MPP) complete with all the components of the privacy protection space they provide. It can be coded and transferred via a privacy preference language such as APPEL [19] , XPref [30] [1] which improves APPEL based on XPATH , P3P [34] which allows companies to specify their online privacy practices that can be checked against a user's, or EPAL [24] that ensures that information is protected and used in accordance with the responsible organization's privacy policies in B2B information exchanges. A comparative study on the privacy policy specification in P3P and EPAL reveals the capability or shortcomings of these policy languages [40]. In this paper, we do not advocate

for a particular specification language to implement PPP or an organization's privacy policies, but we focus on the concepts associated with privacy in the mashup environment and propose a feasible architecture for privacy policy specifications, retrieval and evaluation for the stakeholders of a mashup process. The content provider compares the indicated parameters in the MPP with their own policies (SPP) and those of any individual whose private data might be released through the PPP network in order to make a decision as to whether the data should be released.

One example to illustrate the privacy policy enforcement is a governmental service mashups, where the citizen's data in government agencies are used in creating a mashup content. As a primary source of constituent data, the government has the obligation not only to make public data available for citizen access as stated in the Freedom of Information Act, but also to protect the privacy of individual citizen's records as stated in the Privacy Act. Some personal data that source web service providers might have may be in the public domain. This is particularly true of Governmental data sources where in general they must enforce the *no-disclosure without consent* rule of the Privacy Act with the following exceptions [34]:

- Intra-agency need-to-know exemption authorizes the intra-agency disclosure of a record for necessary official purposes.
- Required Freedom of Information Act (FOIA) disclosure exemption states that the data should be released to FOIA requests unless it is exempted from the FOIA rules.
- Routine use exceptions states that data release or sharing is allowed if they are used in obvious routine tasks, e.g. federal tax payer information collected by the federal tax administration is disclosed to state tax officials for state tax administration.
- A record can be disclosed to the Bureau of Census for the purposes of planning or carrying out a census or survey.
- A record can be disclosed to a recipient who assures that the record will be used solely as a statistical research or reporting record and the record is to be transferred in a form that is not individually identifiable.
- A record can be disclosed to the National Archives and Administrations for the purpose of archiving or the evaluation of archiving.
- A record can be released to support law enforcement activities.
- A record can be disclosed to a person who intends to show the compelling circumstances affecting the health and safety of an individual.
- A disclosure of records to House of Congress or any under its jurisdiction.
- Disclosure to the Comptroller General, or any of his authorized representatives is permitted for the performance of the duties of the General Accounting Office
- A data record should be released for the normal course of court proceedings, including court ordered discovery.
- The Debt Collection Act authorizes agencies to disclose bad debt information to credit bureaus.

Regulatory policies that apply to a source content provider's data should be incorporated in their SPP. There are three different cases to consider in the enforcement of PPP policies in conjunction with SPP policies:

Case 1: When SPP and PPP are in agreement, apply either policy to make a disclosure decision.

Case 2: When SPP permits disclosure but PPP does not, either follow PPP or notify individual of disclosure if it does not conflict with regulatory policies.

Case 3: When PPP permits disclosure but SPP does not, the source provider has a choice.

The result of comparing SPP to PPP as described in the three cases above is the modified SPP (SPP*) for the specific mashup request. The source content provider makes a decision as to whether to provide the requested data by comparing the mashup service provider's MPP to SPP*. What does it mean to compare the preferences in PPPs, SPPs, SPP*s and MPPs? What we are doing is determining if the policy which should apply has a protection space that encloses the protection space that is in the other privacy policy. When comparing two protection spaces PS_1 and PS_2, $PS_1 \Theta PS_2$ where Θ symbolized *encloses* iff:

- $\forall dt_i \in DTofPS_1 \; \exists dt_j \in DTofPS_2 \mid eq(dt_i, dt_j) \vee dt_j = subClass(dt_i)$
- $\forall lp_i \in LPofPS_1 \; \exists lp_j \in LPofPS_2 \mid eq(lp_i, lp_j)$
- $\forall op_i \in OPofPS_1 \; \exists op_j \in OPofPS_2 \mid eq(op_i, op_j)$
- $\forall pt_i \in PTofPS_1 \; \exists pt_j \in PTofPS_2 \mid eq(pt_i, pt_j) \vee pt_j = subClass(pt_i)$
- $\forall mp_i \in MPofPS_1 \; \exists mp_j \in MPofPS_2 \mid eq(mp_i, mp_j) \vee mp_j = subClass(mp_i)$

If PS_1 *encloses* PS_2, then the policy associated with PS_2 may safely be applied. Otherwise, there is a conflict.

For example, when PPP states the resume can be disclosed along with the age and gender information for the purpose of career development services, but SPP may state that they can share the resume, along with the gender but no age. In this case, the PPP encloses SPP. Thus SPP is more stringent policy statement. Thus the SPP* is equal to SPP. On the other hand, if PPP does not allow the release of address to be shared, while SPP may allow the release of the address information. In this case, the PPP is more stringent than SPP, thus SPP* will be rewritten to replace the clause of address sharing to no address sharing. The mashup company who want to plot the job applicants pool by their location (i.e. overlay the addresses of applicants on a map), this mashup conflicts with the SPP* thus, it may modify its mashup data request for the county or town of applicants, but not their addresses.

5. Mashup privacy protection system architecture

The mashup privacy protection system architecture and interaction flows are shown in Figure 5. It implements the proposed privacy model for mashup Web applications. A *personal privacy policy network* is a distributed architecture where individuals can publish privacy policies that they wish to have applied to the use of their private data. The privacy protection engine discovers and consults the Personal Privacy Policies (or preferences), data Service providers Privacy Policies (SPP) and the Mashup service provider Privacy Policies (MPP), and evaluate and enforce the privacy policy-driven data compositions. The PPP, MPP and SPP privacy policies are specified using links to the Semantic Web resources such as FOAF resources, NAIC classification ontology, and UNSPSC classification codes. The specified privacy policies are registered to the privacy protection server to be discovered to retrieve the details of the policies in the respective policy repositories for evaluations and enforcement in the mashup services.

The architecture needs to support several basic functions. The first component is the privacy protection engine that provides an interactive data access through Web services, and ensures private data is not released in conflict with personal privacy policies (PPP), source privacy policies (SPP) and mashup policies (MPP). It performs reasoning to calculate the policy enclosures and identifying policy conflicts. The privacy-preserving mashup service architecture also includes a *notification service* to alert individuals of potential privacy loss. It has an *audit service* which monitors data dissemination and compares it to the data usage policies published by the data provider as well as the personal privacy policies of those individuals represented by the data. Another component is an off-line management process that helps to ensure that data access is provided as efficiently as possible.

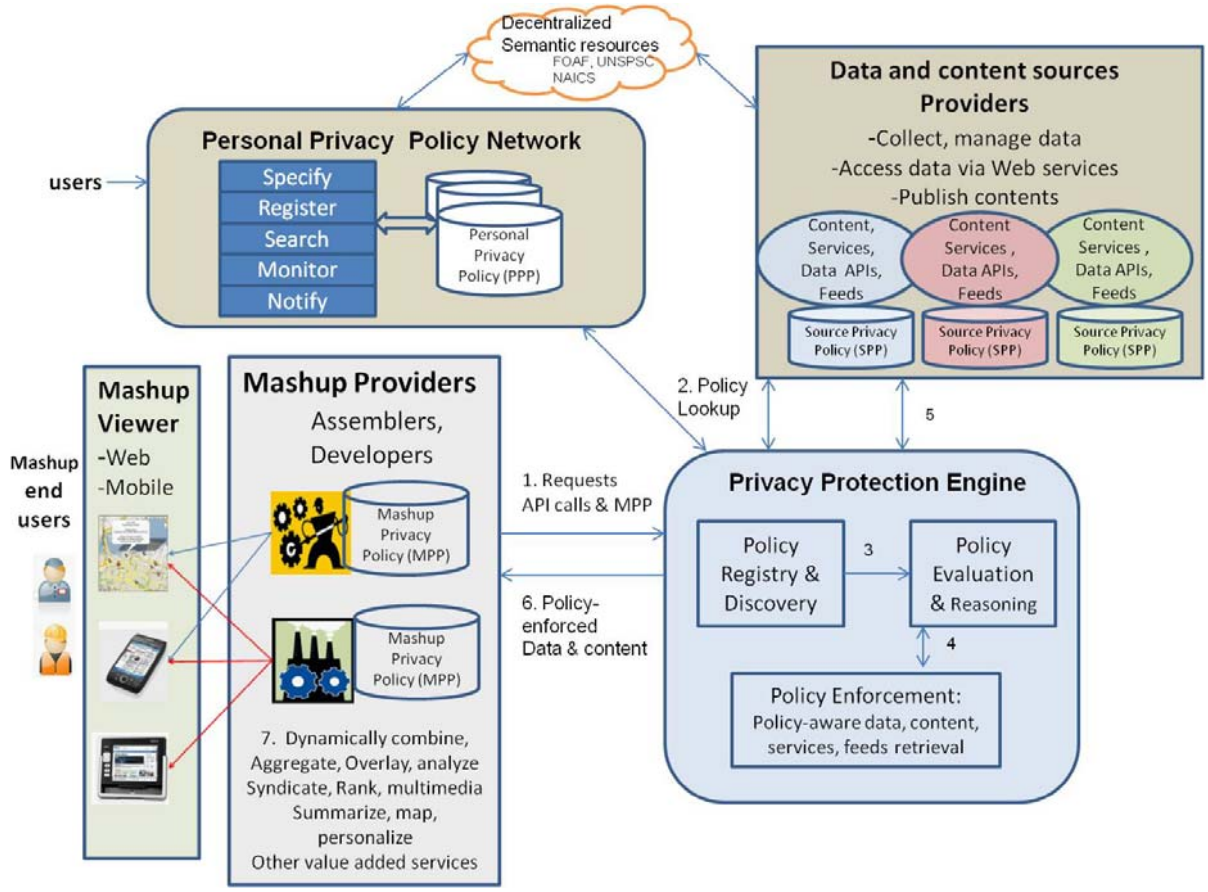

Figure 5 Mashup Privacy Protection System Architecture

5.1 Interactive Privacy Preserving Data Mashup Process

In step 1, the mashup provider (a mashup developer) requests the data or contents using APIs and Web services and send its Mashup provider's privacy policy for the particular data or service mashup. This step starts with the mashup provider discovering suitable content provider(s), such as a government agency (e.g. IRS)[1] and making a request for access to the data. The request includes the resource location of the mashup provider's own privacy policy statements (MPP), such as the purpose and requested data

[1] The discovery of the content provider is out of scope of this paper.

items, intended target audience, etc. Once the request arrives, in step 2, the privacy protection engine looks up the registry of policies and fetch relevant source policies (SPP) as well as the data owner's personal privacy policy (PPP) published in a personal privacy policy network, that can be served through a personal Web server or by a third party Web server repository. In step 3, the policy evaluation and reasoning component is invoked to evaluate the three policies through comparisons and reasoning. When the requested mashup data usage (purpose) in MPP policy agrees with the source policies SPP as well as the PPP, data owner's personal policies, the data is retrieved and returned.

When there are conflicts or disagreements among the MPP, SPP, and PPP, the policy evaluation and reasoning component transforms the mashup data request to a privacy-aware request that applies the most stringent privacy policies. Step 4 calls the policy enforcement component, where the transformed SPP* is sent to the source provider to fetch the right data or feeds from the source providers. In step 6, the request result is sent over to the mashup providers. In addition, the data disclosure notification is sent to the data subject, in case the SPP mandates data disclosure against the personal privacy preferences, as in the case of Freedom of Information Act inquiries. In step 7, the mashup provider assembles the data to deliver to the mashup requesters or publish the new data or service product, according to the evaluated policies.

5.2 Advantages of Privacy Policy-driven Mashups

Recently, social networks (e.g. Facebook) have also adopted the opt-in privacy specification [45]. Although the users do not have all the options they need, only the ones that the service provider offers, this is an attempt to give users control of their data and to make the user privacy policy more explicit than in the opt-out method. In the opt-out method, any data that is not excluded by the user will be considered owned and thus resalable by the service provider. Thus, our approach to give the users the full power to specify their privacy policies for any data service provider is one step towards greatly improved privacy protection.

In addition, as in the case of Facebook, the user privacy policy is specified by the service provider side only (e.g., for specific applications provided by one organization) that may not be reusable by other service organizations. Our architecture, where the privacy specification and enforcement are performed by independent components, separated from the data provider and from the mashup service providers as well as from the users, has several advantages:

1) The users can specify their privacy policy once in their privacy repository and these policies will be reusable, i.e. any data service providers (e.g., Electronic commerce sites, social network providers, health organizations, etc.) can refer to the policy.
2) The tie-in between the service provider and the user policy network can benefit the users to enable them to control and manage their own privacy policies in a more consistent fashion. Any changes of their policy will be available to all service providers simultaneously, instead of requiring the user to visit each organization's site separately to make comparable changes to the policies.
3) The policy evaluation and enforcement engine can be developed as a downloadable plug-in component for all data providers, rather than forcing each organization to develop its own data privacy evaluation and enforcement engine. Alternatively, the policy evaluation engine can be implemented as a cloud service that is operated by an independent organization.
4) The source organization's privacy policies are also separately specified and evaluated by the same engine for release or sharing. Thus, the corporate information system does not have to build an ad-hoc

privacy policy enforcement engine. Thus, the information assurance and auditing of information systems is greatly simplified in terms of privacy protection.

5) The mashup service providers also can benefit since their output of the data mashups (usually a composite of several data sources) can be easily validated by checking personal privacy policies a second time. This will ensure that there are no further privacy violations. Their mashup privacy policy can be published and shared to make them accountable for their composite services.

5.3 Implementation Issues

Managing Privacy Policy Specifications: The architecture also provides the capability of specifying the privacy policies MPP, SPP and PPP, by respective parties, i.e., mashup provider, data source provider, and the data owner (an individual), using the semantic resources. However, the specification of purposes or provider types can vary widely, that may cause errors or interoperability issues when comparing, evaluating policies across different privacy networks. To prevent that, the system provides an entry form to select a NAIC classification code for service provider types (PT), to select the UNSPSC codes for mashup purposes (MP), and to specify the privacy data type DT with FOAF URIs that can show the foaf elements. The list of mashup operations can be also presented for a selection list. Figure 6 is a simple example of the personal privacy policy entry form where the Data Type elements can be selected and entered by navigating with visualized FOAF resource site, instead of manually entering. Similarly, the mashup purpose can be entered through the UNSPSC classification hierarchy in a pop-up window, and allow users to navigate and select a product function or service code. This visual interface will enhance the usability of specifying the fine-grained privacy policies as well as reducing errors associated with the specifications.

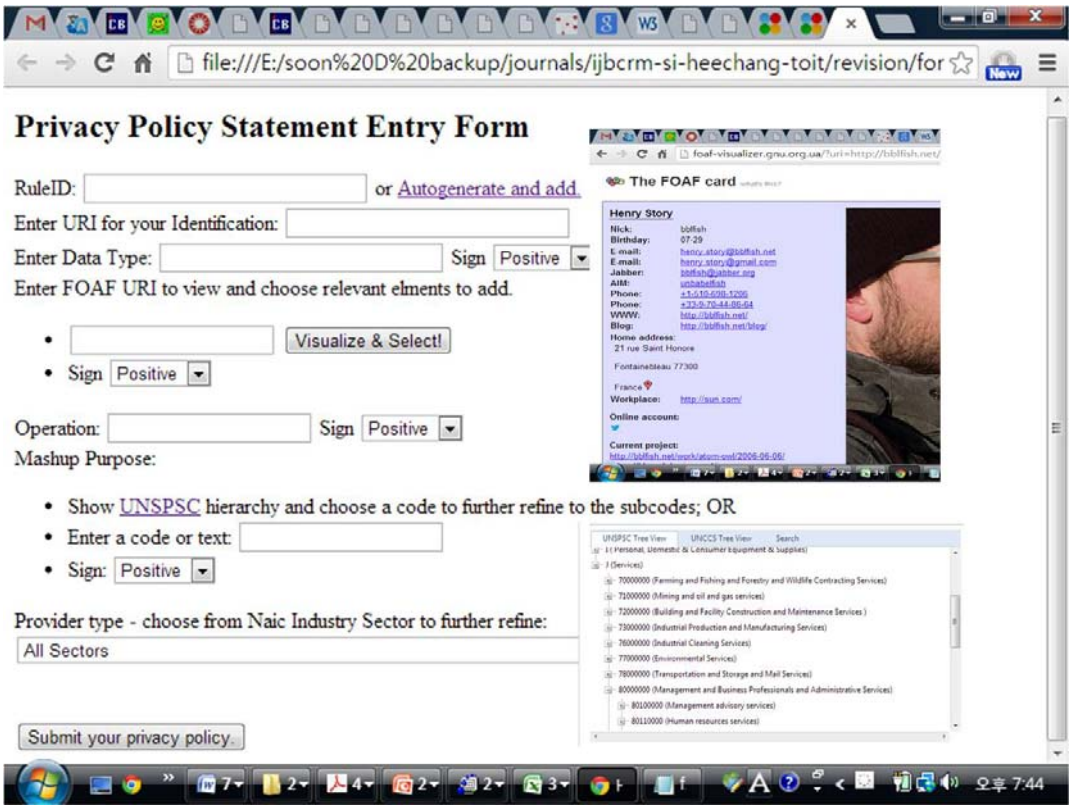

Figure 6 Personal Privacy Policy Entry Form with the visual navigation options

Managing Policy Updates and Discovery: Another issue is that the specification and management of policy parameters is not easy even if the data is static but that is certainly not the case in real-life situations. The data is constantly changing with new data being constantly added and with user preferences on existing data changing as well. Therefore, a system is needed to derive applicable privacy protection spaces whenever a change in individual privacy preferences occurs. The privacy policy evaluation engine that are implemented at data providers is responsible for knowing where individual privacy preferences are housed and ensuring that they are linked to the relevant repositories. In other words, the PPP registry should be updated timely to reflect the changes. Since real-time access to these repositories for hundreds or even thousands of individuals would be prohibitive, privacy protection spaces can be pre-calculated and pre-defined. These spaces are then updated only when data source providers are notified that a privacy preference change has been made. In case the engine is implemented in a cloud service, the central registry of policies can be updated and propagated easily.

Complexity Studies: The complexity issues of the proposed architecture can occur in different areas. For example, implementation complexity includes issues such as how developers integrate the data according to many different policies from different individuals. Some may allow an overlay of detail information about them over a map, while others may not want it. These individual differences make the mashup complexity high for the developers. Performance overhead includes how much the privacy evaluation and enforcement would slow down a given mashup due to the policy discovery and evaluation steps. A network overhead issue includes how much more network traffic the policy evaluation generates. These are some of the practical issues we are addressing when implementing the proposed architecture.

Security and Provenance of privacy policies: The policy repositories can be subject to attack. Additionally, often the privacy policies are themselves sensitive by their nature. In the architecture, the personal privacy policies are sent over a network to be composed with other policies to determine the data mashups. By including a policy stating that one's health data on a particular disease should not be shared, he/she reveals how sensitive this data is. The architecture thus requires a component for securing privacy policies, policy composition, and policy provenance data (i.e., history of privacy policies that participated in the mashups). In addition, the mashup data needs to retain the provenance information for digital forensics and audits, if needed, to track whether the data is being used according to the purpose(s) stated in the privacy policies. The secure policy composition as in [48] and the provenance techniques such as [49] are being considered.

6. Related Work

The existing P3P (Platform for Privacy Preferences) protocol allows organizations to express their privacy policies on the information they collect from users through Web browsing. It declares their intended use of information and gives users more control of their personal information by setting their privacy levels and opt-in and opt-out options. P3P-compliant web sites and the end-user tools, such as Privacy Bird or Privacy Finder, match the policies of web sites with a user's privacy preferences and advise the user when there is a potential privacy issue. In web browsing, privacy protection can be performed while the individual is on-line and engaged with a web site.

While the mashup environment is also a Web environment, P3P is not directly applicable. Clearly, the individuals whose privacy may need to be protected may not engage directly with the mashup provider. It is difficult to specify privacy preferences for services created from diverse sources of information by mashup providers. It is also difficult to track and audit the dissemination of the information to the mashup providers in conformance with personal privacy preferences because there is no direct involvement of the individual whose privacy might be compromised in the interaction.

Various privacy protection languages such as EPAL [40] and XACML [23] exist that can be the basis of mashup protection. Access control of private information should be at the field level, such that it will be possible to give different users different views of the data collected. No language currently exists to address specification of purpose or linking parameters. We are looking into an RDF-based Linked Data model for specifying the privacy policies that has expressive power for specifying the relationships over Web resources (URIs) with sufficient semantics (e.g. ontologies or classifications as we discussed). The Linked Data repositories can be queried using SPARQL.

Hippocratic DBs [28] transform queries such that they access only policy-compliant information. Ager et al. [33] adapts the query transformations to apply to access and disclosure of classified and other sensitive information maintained by government agencies. Specifically they define how information can be filtered based on policies. Their techniques could be used in implementing our model where filtering and access control depends not only on who is requesting the information but also for what purpose.

A difficult problem is how to manage privacy implementations that allow an individual to review policy, specify their own requirements and determine enforcement of policy. Brodie et al. [6] take steps toward addressing the enforcement issue and describes a workbench that allows users to interactively define their personal policies. Specific work on Web service security has been done by Lesk [17] and Hatala et al. [16]. This approach allows only certified users access to Web services. This is the approach taken by some existing API providers. In [29] a permit based access delegation model is presented to allow users to grant access rights ("permits") delegation to mashups, thus, the authorization servers and back-end applications do not have to maintain elaborate state information which helps to be scalable. However, it does not directly address the privacy of user data in mashups. In database integration domain, a privacy preserving data integration is studied [7] where the privacy framework takes an administrative centric approach where the administrator defines privacy views (what is considered private), the privacy policies and the purpose of applications. Our approach, on the other hand, considers the privacy policies that can be stated and managed by different parties in mashing up data or services.

Chris Hanson et al. [8] have created a data purpose algebra for specifying constraints on data usages. For every data set, the algebra defines allowed sources, data categories that may be collected and the purposes for which collected data may be used. The algebra is used to define changes in data and policy that apply once data is transferred and modified using certain operations. This work focuses on the data source provider's policies on how to constrain the data usage by others, but do not consider the decentralized specification of personal and providers' policies and reasoning. Thus, in our approach, an individual has finer control, disallowing certain types of mashup providers from using data, while allowing other mashup providers to use the data. Braun et al. [46] and Lu et al. [47] discuss the importance of securing the provenance data for digital forensics. It is difficult to control the usage of mashups at the end user side. The mashup usage control requires the usage provenance data to ensure that the purported usages are

in fact valid. In addition, the provenance of the privacy policies that were relevant to the mashups needs to be secured and preserved.

Often the information assurance includes risk assessment and data governance policies for business continuity in case of software attacks, natural disasters, etc. [44][42][43] The proposed architecture where the privacy policy specifications and enforcement components are separate from their core services will allow the businesses to easily audit their services in accordance with their own privacy policies and with the users' privacy policies. In case of violations, it is easier to address the privacy risks and manage them, allowing them to have better business continuity and sustainability.

7. Conclusions and Future Work

Mashups introduce new privacy challenges for individuals, because data might be combined and released by web services, such as mashups, with which the user has no interaction. We introduced a model for addressing the privacy challenges. It allows a user to describe their privacy preferences in terms of data types, provider types and mashup purposes. It makes extensive use of existing ontologies to allow users to define data types in consistent ways and to allow policies to be compared and integrated.

Our process allows users to register their preferences concerning private data in well-known repositories. These preferences are called personal privacy policies (PPP). Content sources who have content associated with individuals have their own privacy policies associated with release of personal data called source privacy policies (SPP). We have proposed a *privacy policy evaluation and enforcement engine* that can be implemented and downlable by any content source provider. Alternatively, it can be implemented as a cloud service for policy evaluation. The privacy policy evaluation and enforcement engine can consult PPP and SPP as well as the mashup providers' privacy policy (MPP) before releasing personal data to a third party such as a mashup provider. The SPP policies are compared with the PPP and MPP to generate SPP*. The resulting SPP* are combinations of the SPP in accordance with individual preferences (PPP) and in accordance with the requesting mashup providers policies (MPP). All three policy sets consider data types, linking parameters, operations, provider type and mashup purpose. Specific combinations of these five parameters are considered to be a *privacy protection space* over which access can be allowed or denied.

Our contribution is to make specification of privacy policy fine-grained so that data is usable for as wide an audience as possible, for situations where data sharing is for the public good, while keeping data private from the general public. We also make use of the Semantic Web resources in specifying these privacy policies that are linking decentralized data on the Web. Our prototype architecture is presented for showing the components and discussing implementation issues, especially the usability of fine-grained privacy policy specification.

Future works include analysis of the policy specification system requirements through user studies, developing a working prototype system to test the feasibility of the proposed distributed privacy policy networks to ensure the data integration and sharing is privacy preserving. The policy discovery and evaluation engine needs to implement the different policy comparisons and reasoning to resolve the potentially conflicting policies from different networks. The complexity-related evaluation study (e.g. performance overhead or network overhead) is also a remaining future task to measure whether the proposed approach is practical.

Acknowledgement

This material is based upon work by Keromytis supported by (while serving at) the National Science Foundation, and work by Chun partially funded by NSF DUE-1241687. Any opinion, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] A. Berglund, S. Boag, D. Chamberlin, M.F. Fernandez, M. Kay, J. Robie, and editors J. Simeon. Xml path language (XPATH) 2.0. Technical report, W3C Recommendation, January 2007.

[2] A. elKalam and Y. Deswarte. Privacy requirements implemented with a javacard. In *21st Annual Computer Security Applications Conference (ACSAC'05)*. IEEE, 2005.

[3] An ontology for vcards. www.w3.org/2006/vcard/ns, accessed in 2008.

[4] Anantha Ramakrishnan, Leveraging power of UNSPSC for Business Intelligence, Satyam White Paper, http://www.unspsc.org/Portals/3/Documents/Leveraging%20the%20Power%20of%20UNSPSC.pdf, accessed in 2013.

[5] B. Medhahed, A. Rezgui, A Bouguettaya, and M. Ouzzani. Infrastructure for e-government web services. *IEEE Internet Computing*, Jan/Feb 2003.

[6] Carolyn Brodie, Clare-Marie Karat, John Karat, and Jinjuan Feng. Usable security and privacy: A case study of developing privacy management tools. In *Symposium on Usable Privacy and Security (SOUPS)*, July 2005.

[7] Chris Clifton, Murat Kantarcioglu, AnHai Doan, Gunther Schadow, Jaideep Vaidya, Ahmed K. Elmagarmid, Dan Suciu: Privacy-preserving data integration and sharing. DMKD 2004: 19-26.

[8] Chris Hanson, Tim Berners-Lee, Lalana Kagal, Gerald J. Sussman, and Daniel J. Weitzner. Data-purpose algebra: Modeling data usage policies. In *POLICY*, pages 173–177, 2007.

[9] Christin Moore. The growing trend of government involvement in it security. In *InfoSecCD Conference*. ACM, October 2004.

[10] Duane Merrill, Mashups: The new breed of Web app: An introduction to mashups, 2009, ibm.com/developerWorks

[11] http://lehd.ces.census.gov/, accessed in 2013. http://www.census.gov/eos/www/napcs/, accessed in 2013.

[12] Janice Warner and Soon Ae Chun, A Citizen Privacy Protection Model for E-Government Mashup Services, Proceedings of the 9th International Conference on Digital Government Research, Montreal, Canada, 2008: 188-196.

[13] Janice Warner and Soon Ae Chun, Privacy Protection for Government Mashups, Information Polity: Volume 14, Editions 1 & 2, 2009: pp 75-90.

[14] L. Korb. Privacy in distributed electronic commerce. In *35th Annual Hawaii International Conference on System Sciences (HICSS-35'02)*, 2002.

[15] L. Peyton and M. Nozin. Tracking privacy compliance in b2b networks. In *Sixth International Conference on Electronic Commerce (ICEC'04)*, 2004.

[16] M. Hatala, T. Eap, and A. Shah. Federated security: Lightweight security infrastructure for object repositories and web services. *Proceedings of International Conference on Next Generation Web Service Practices*, 2005.

[17] M. Lesk, M. Stytz, and R. Trope. Providing web service security in a federated environment. *IEEE Security and Privacy*, Jan/Feb 2007.

[18] Map of offenders: Locate offenders in your area. http://familybeacon.com/, accessed in 2007.

[19] Marc Langheinrich. A p3p preference exchange language 1.0 (appel1.0). Technical report, W3C Working Draft, April 2002.

[20] Mary Ann Davidson and Elad Yoran. Enterprise security for web 2.0. *IEEE Computer*, 40(11), November 2007.
[21] North American industry classification system (naics). http://www.census.gov/epcd/www/naics.html, accessed in 2008.
[22] North American product classification system.
[23] OASIS Open, eXtensible Access Control Markup Language, (XACML) Version 3.0, January 2013. Accessed in http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.pdf
[24] P. Ashley, S. Hada, G. Karjoth, C. Powers and M. Schunter. Enterprise Privacy Authorization Language (EPAL 1.1) Specification. IBM Research Report. http://www.zurich.ibm.com/security/enterprise-privacy/epal. 2003.
[25] P. Ashley, S. Hada, G. Karjoth, C. Powers, and M. Schunter. Enterprise privacy architecture language (EPAL). Technical report, W3C, 2003.
[26] Privacy policy profile of xacml. Technical report, OASIS, 2004.
[27] Programmable Web: Mashup Dashboard, http://www.programmableweb.com/mashups , accessed in 2013.
[28] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu. Hippocratic databases. In *28th Very Large Database Conference (VLDB)*, 2002.
[29] Ragib Hasan, Marianne Winslett, Richard M. Conlan, Brian Slesinsky, Nandakumar Ramani, Please Permit Me: Stateless Delegated Authorization in Mashups. ACSAC 2008: 173-182.
[30] Rakesh Agrawal, Jerry Kiernan, Ramakrishnan Srikant, and Yirong Xu. An xpath-based preference language for p3p. In *Twelfth International World Wide Web Conference (WWW2003)*, pages 629–639. ACM Press, May 2003.
[31] Robert Gerber. Mixing it up on the web: Legal issues arising from internet "mashups". *Intellectual Property and Technology Law Journal*, 18(8), August 2006.
[32] Stefania Galizia. WSTO: A classification-based ontology for managing trust in semantic web services. In *ESWC*, pages 697–711, 2006.
[33] T. Ager, C. Johnson, and J. Kernan. Policy-based management and sharing of sensitive information among government agencies. Technical report, IBM Almaden Research Center, 2006.
[34] The Platform for Privacy Preferences 1.1 (P3P1.1) Specification: W3C Working Group Note 13, 2006. http://www.w3.org/TR/P3P11/, accessed in 2007.
[35] Tom Owad. Data mining 101: Finding subversives with amazon wishlists. http://www.applefritter.com/bannedbooks, accessed in 2008, January 2006.
[36] United Nations standard products and services code. https://www.ungm.org/info/Unspsc.aspx, accessed in 2013.
[37] United States Census Bureau, Industry Focus Application, http://lehd.did.census.gov/applications/industry_focus.html, accessed in 2013.
[38] United States Census Bureau, Longitudinal Employer-Household Dynamics
[39] Wei Liu. Trustworthy service selection and composition reducing the entropy of service oriented web. In *3rd International Conference on Industrial Informatics*, 2005.
[40] William H. Stufflebeam, Annie I. Antón, Qingfeng He, Neha Jain: Specifying privacy policies with P3P and EPAL: lessons learned. Proceedings of the 2004 ACM workshop on Privacy in the electronic society, 35, 2004.
[41] Yao Wang and Julita Vassileva. A review on trust and reputation for web service selection. In *ICDCS Workshops*, 2007.
[42] Holmes E. Miller, Integrating sustainability into business continuity planning, International Journal of Business Continuity and Risk Management, Vol 2(3): 219-232, 2011.
[43] Chen-Huei Chou; Fatemeh Mariam Zahedi, When natural disasters strike: managing individual and organisational needs with web-based systems, International Journal of Business Continuity and Risk Management, Vol 4(1):75-91, 2013.

[44] Maurice Eugene Dawson Jr.; Miguel Crespo; Stephen Brewster, DoD cyber technology policies to secure automated information systems, International Journal of Business Continuity and Risk Management, Vol 4(1):1-22, 2013.

[45] Information Assurance: Then, Now and Moving Forward, Newsletter of the Information Assurance Professionals, Vol 15(1) Winter, 2012.

[46] Uri Braun, Avraham Shinnar and Margo Seltzer, Securing provenance, *Proceedings of the 3rd conference on Hot topics in security*, HOTSEC'08 Article No. 4, 2008.

[47] Rongxing Lu, Xiaodong Lin, Xiaohui Liang, and Xuemin (Sherman) Shen, Secure Provenance: The Essential of Bread and Butter of Data Forensics in Cloud Computing, *Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security*, Pages 282-292, 2010.

[48] Basit Shafiq, Soon Ae Chun, Jaideep Vaidya, Nazia Badar and Nabil Adam, Secure Composition of Cascaded Web Services, *Proceedings of the 8th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing* (CollaborateCom '12), 2012

[49] Eleni Gessiou, Vasilis Pappas, Elias Athanasopoulos, Angelos D. Keromytis, Sotiris Ioannidis: Towards a Universal Data Provenance Framework Using Dynamic Instrumentation, *Proceedings of International Conference ICT Systems Security and Privacy Protection* (SEC 2012): 103-114, 2012.

Appendix 1: Attached Figures

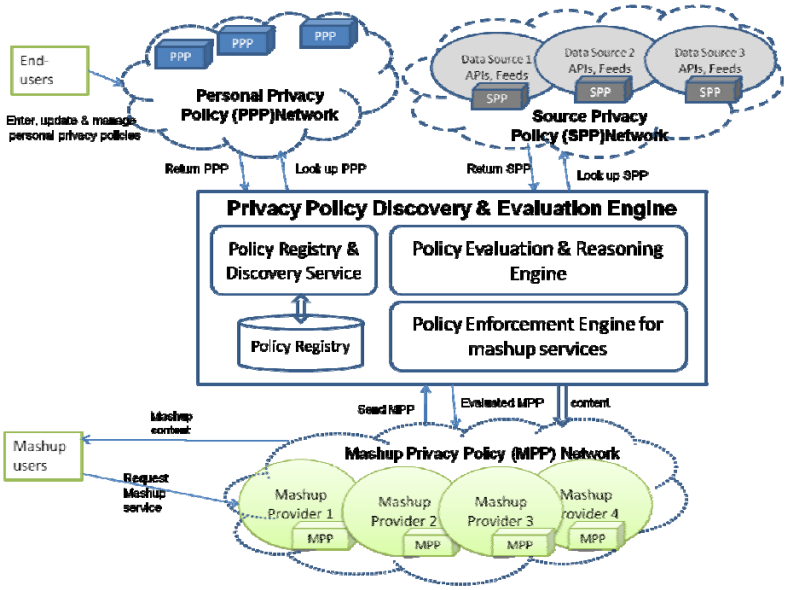

Figure 1 Overall Mashup Policy Discovery, Evaluation and Enforcement Engine

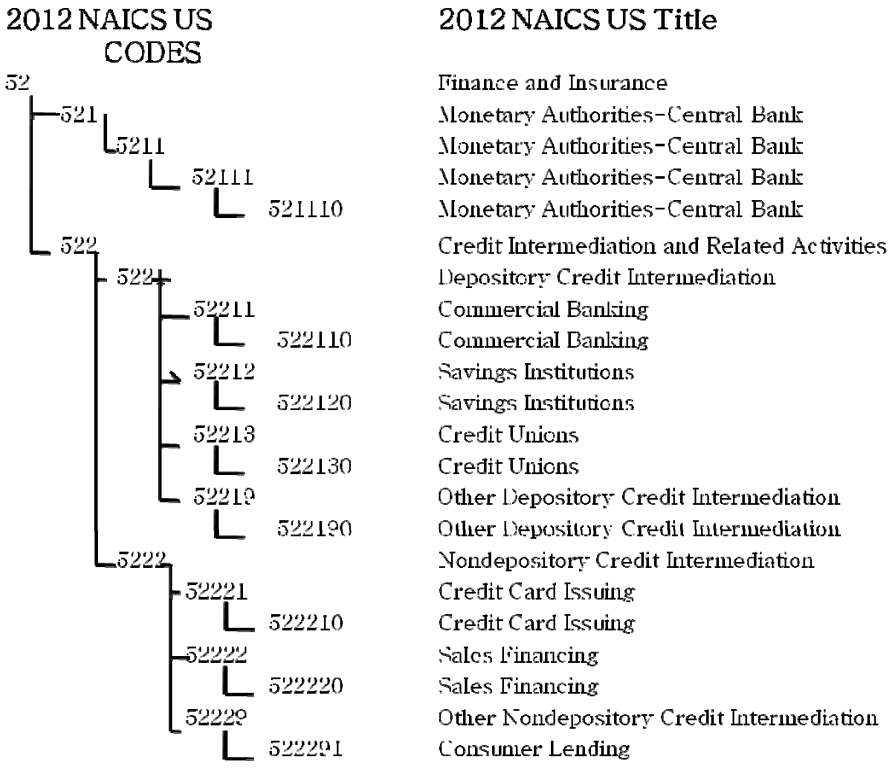

Figure 2 Excerpt from the North American Industry Classification System (NAICS) used to specify Mashup Provider Type

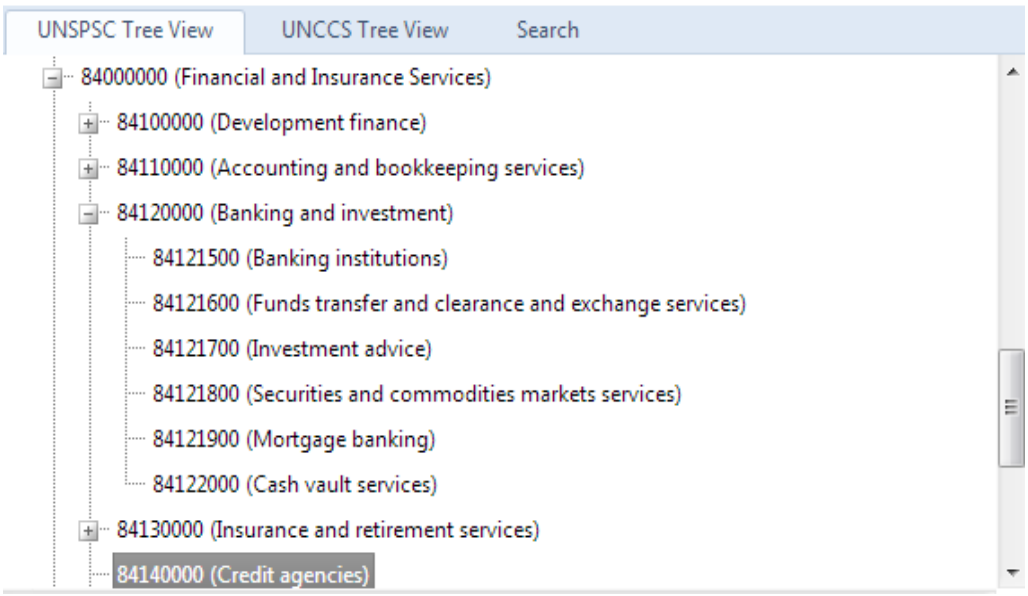

Figure 3 Excerpt from the UN Standard Product and Service Code (UNSPSC) used to specify Mashup Purpose

Industry Subject Area	Working Group Code	United States		
		Title	Definition	NAICS Industries Producing the Product
52	1.1	Financing services	Providing services that result in the provision of money and granting of credit to businesses, consumers, and governments.	522110 522120 522130 522190 522210 522220 522291 522292 522293 522298
52	1.1.1	Loan services	Providing direct lending of funds under legal contract, either unsecured or secured by the assets being financed or by other assets, but without the exchange or the use of securities as collateral. Includes: • interest and origination and other fees received from sales of loans.	522110 522120 522130 522190 522220 522291 522292 522293 522298
52	1.1.1.1	Loans to financial businesses	Making loans to financial businesses. Includes interest received, origination and other fees received, and revenue from sales of loans. Includes: • interest received and origination and other fees received from sales of loans. • loans made to banks, trust companies, investment dealers and brokerages, and insurance companies, etc. Excludes: • providing financing using purchase-repurchase agreements is in product 1.3.2, Repurchase agreements.	521110 522110 522190
52	1.1.1.1.1	Loans to depository financial institutions	Making loans to depository financial institutions, such as banks. Excludes: • providing financing using purchase-repurchase agreements is in product 1.3.2, Repurchase agreements.	521110 522110 522190
52	1.1.1.1.2	Broker's call loans	Making loans to security and commodity contract brokerages, used to finance underwriting costs and margin lending, usually short-term and secured by securities. Excludes: • providing financing using purchase-repurchase agreements is in product 1.3.2, Repurchase agreements.	522110
52	1.1.1.1.9	Loans to financial businesses, nec.	Making loans to financial businesses, not elsewhere classified. Excludes: • making loans to depository financial institutions is in product 1.1.1.1.1, Loans to depository financial institutions. • making call-loans to security and commodity contract brokerages is in product 1.1.1.1.2, Broker's call loans. • providing financing using purchase-repurchase agreements is in product 1.3.2, Repurchase agreements.	522110 522190

Figure 4 Excerpt from NAPCS Product List for NAICS 52: Finance and Insurance

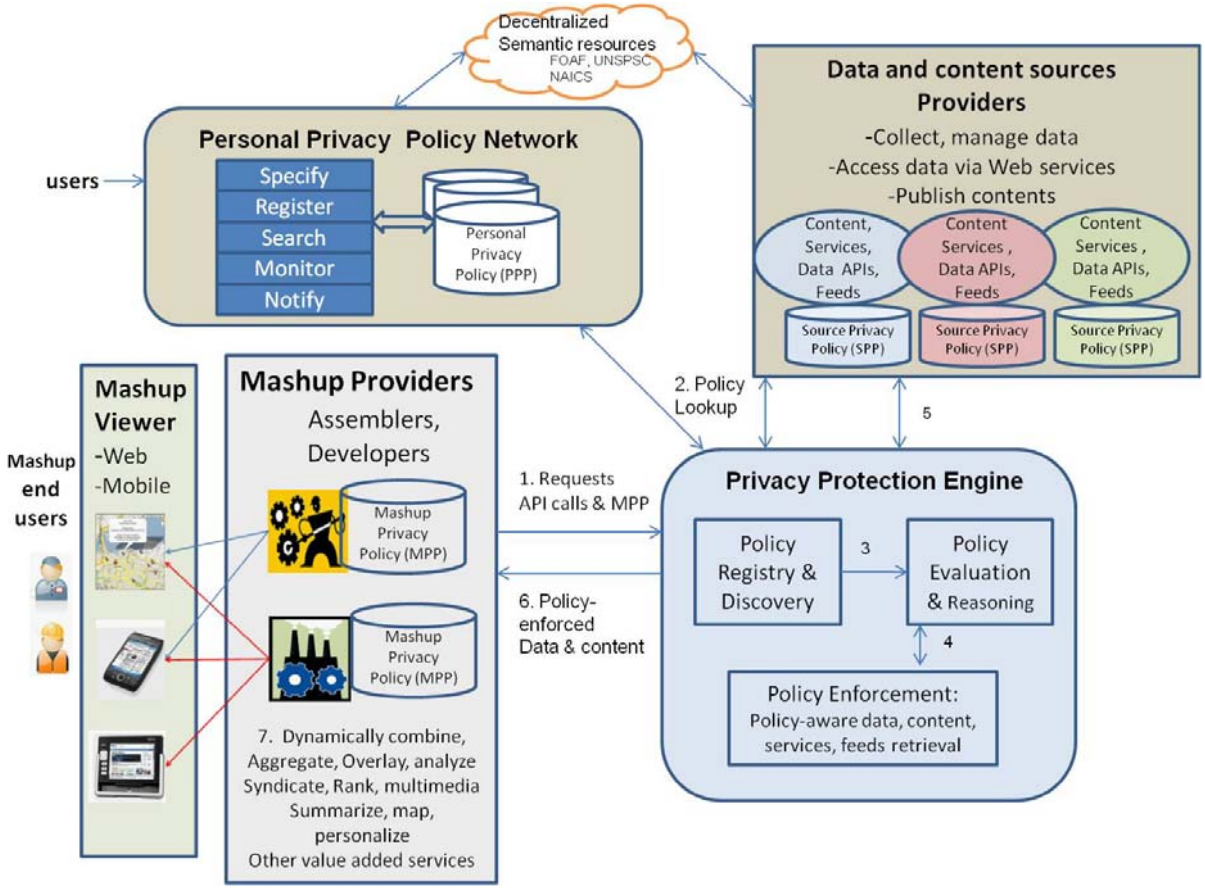

Figure 5 Mashup Privacy Protection System Architecture

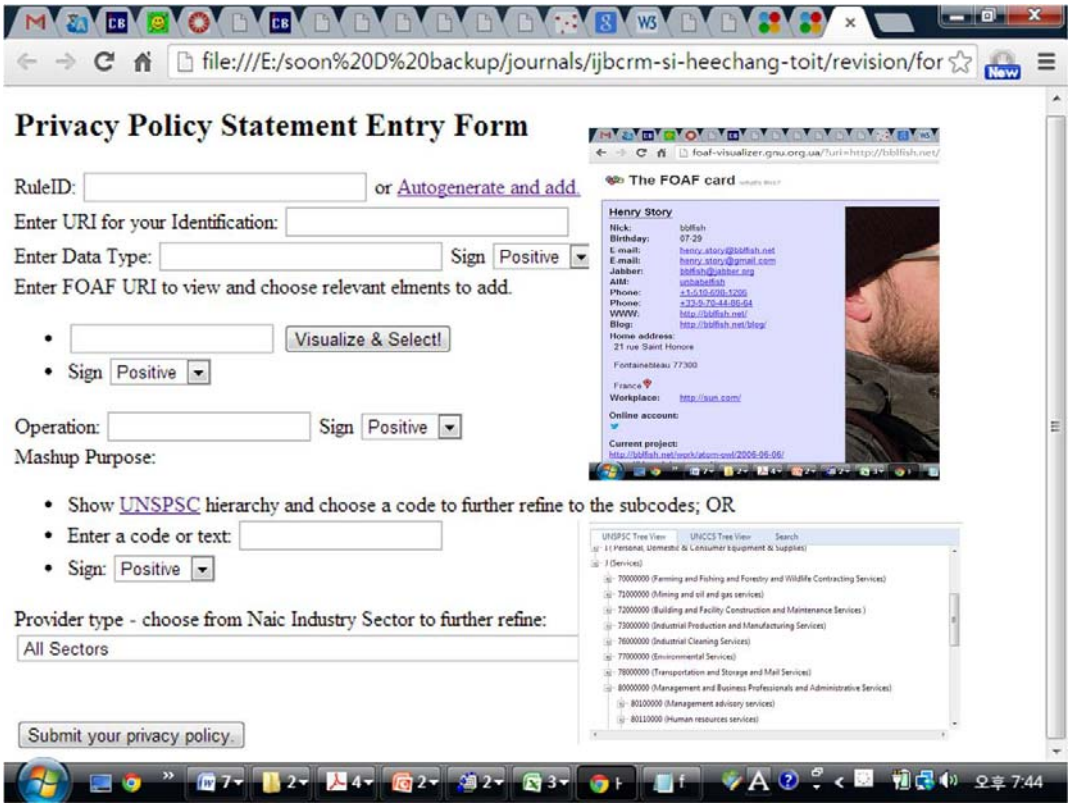

Figure 6 Personal Privacy Policy Entry Form with the visual navigation options

Appendix 2: Author Biographies

Soon Ae Chun is an Associate Professor of Information Systems and a Doctoral Faculty of Computer Science at the City University of New York – College of Staten Island and the Graduate Center. She is a director of Information Security Research and Education (iSecure) Lab. Her research areas include Security and Privacy, Semantic Web, Data Integration, Social Data Analytics and Workflow. Her research work has been applied to the digital government, health informatics, environmental science and e-learning domains. The current research projects include the cyber security ontology development and semantic integration of data sources. She is a Board Member of the Digital Government Society, and a member of IEEE, ACM and AIS.

Janice Warner is Dean at the Georgian Court University School of Business in Lakewood NJ. Previously an engineer in the data networking industry, she is interested in the strategic use of technology in management. Current research involves business analytics in addition to knowledge management, security and privacy. She also continues to teach in the business programs due to her love of teaching where she strives to inspire her students to think creatively and critically. Janice is a member of the ACM and is a Board Member for the Monmouth Ocean Development Council and Lakewood Chamber of Commerce.

Angelos D. Keromytis is an associate professor of Computer Science at Columbia University in the City of New York, and the director of the Network Security Lab. He is currently serving as a Program Director at the NSF CISE Division of Computer and Network Systems (CNS). His general research interests are in systems and network security, and cryptography. His current interests revolve around software hardening, system self-healing, network denial of service, information accountability, and privacy. He is an ACM Distinguished Scientist since 2012, and a founder of Allure Security Technology Inc.