QTIP: Multi-Agent NLP and Privacy Architecture for Information Retrieval in Usable Web Privacy Software

Vlado Keselj
Faculty of Computer Science, Dalhousie U.
Halifax, NS, Canada
vlado@cs.dal.ca

Dawn Jutla
Sobey School of Business, Saint Mary's U.
Halifax, NS, Canada
dawn.jutla@smu.ca

Abstract

We present a generic natural language processing (NLP) architecture, acronym QTIL, based on a system of co-operating multiple agents (Q/A, T, I, and L agents) which can be used in any information system incorporating Internet Information Retrieval. We then introduce a hybrid multi-agent system (MAS) architecture, acronym QTIP, for the privacy domain through integrating the PeCAN (Personal Context Agent Networking) and QTIL MAS architectures. There are two areas where NLP is used: in the user-MAS interaction and in the process of resource indexing and matching. These two areas map to the Q/A-agent and to the I-agents. We propose using a lightweight Head-driven Phrase Structure Grammar (HPSG) natural language method for the Q architectural layers and qualitatively justify its applicability. We provide an example of employing the HPSG formalism for Information Retrieval using natural language capability via Privacy Web Services in one instantiation of the QTIP architecture. Independent preliminary results for HPSG on the Q level show that our approaches for enhancing the usability of PET tools are promising.

1. Introduction

State-of-the-art privacy agents, such as the P3P-based agent PrivacyBird [25], are currently limited by fixed-format form interfaces. User preferences are restricted to specifying what a single form accommodates, which may be mainly user rules for handling of his/her emotional data such as health, financial, and physical data. A good example of such a form is the Privacy Preference Settings form that AT&T Bird uses (see [25]). These settings are done at a large grain level which is user-friendly in itself but lacks flexibility for personalizing privacy software according to a wide range of subjective user preferences and weightings of these preferences.

To succeed in achieving the one-to-one personalization goal of tomorrow's business information systems, including privacy information systems, we expect natural language processing (NLP) will play a major role. These business information systems will incorporate external feeds from reliable networked sources for improved business intelligence and for gaining competitive advantage. Improved NLP capability will obviously make information systems more usable and more customizable. Indeed, the Internet Information space is huge. With a collection of that size, use of NLP is needed even more, because purely keyword-based retrieval methods tend to retrieve too many documents. The Internet collection is also very dynamic even in specialized domains, such as the privacy domain.

Initial tools for privacy, such as P3P agents, and for example AT&T Bird, are fairly informational to the user. Currently these P3P agents first retrieve P3P-formatted privacy policy statements from web sites, then perform a 2-way match between business privacy practices and user privacy preferences, and finally produce user summary statements for the user. Examples can be found in [3], [25], or through use of the Bird P3P agent.

However, many potentially useful Internet resources remain untapped and unusable. As a sector example, sites such as www.canlli.org, www.austlii.org, and www.law.cornell.edu all share a similar mandate to make legal information available and freely accessible to ordinary citizens. The current keyword-based search on privacy law on these sites is not useful or easily decipherable to most of us. The potential for tapping these Internet resources and making privacy Web services more usable – in terms of perceived usefulness and ease of use – is great. Future privacy Web services based on Web information retrieval with NLP can include the user being able to easily seek out useful knowledge about other countries' privacy laws, and assess a country's privacy culture.

Hence as a first step, we are motivated to investigate a high-level conceptual multi-agent architecture (MAA) which integrates natural language (NL) capability and is used for the Internet Information retrieval (InIR) task. Secondly, we develop such an architecture, acronym QTIL, and propose this architecture's integration with existing agent-based privacy web architectures for user privacy software. Thirdly, we instantiate the resulting integrated NLP architecture for the privacy domain, QTIP, implementing Head-driven Phrase Structure Grammar (HPSG) NL method at the question-answering interface. We demonstrate that this instantiation of the QTIP architecture can provide for more usable privacy

software and thus can make privacy enhancing tools (PET) more usable.

2. QTIL: Natural Language Processing Architecture for Internet Information Retrieval

Our generic QTIL MAA for InIR is shown in Fig. 1.1. QTIL consists of 4 main architectural layers or levels. The Q-level contains Q/A-agents (Question-Answering agents) which transform the NL-based user query into an appropriate internal format and passes it to the top-level planning agents, the T-agents. A T-agent develops a high-level query plan and communicates more specific tasks to the intermediate agents, (I-agents, IA). The I-agents are equipped with NLP capabilities as well. The motivation is not to have all of the I-agents to be able to process any kind of text across many domains, but to have each agent specializing in a certain kind of text for a specific domain. I-agents effectively communicate, exchange, and reuse knowledge gathered by various agents working on different tasks and for different users. An advantages of this approach is that we can have many specialized I-agents that are limited to certain domains. This approach of using NLP in a modular domain-specific way is sometimes denoted as *distributed NLP*.

Finally, several low-level retrieval tasks are sent to the bottom level of the QTIL hierarchy—to low-level retrieval agents, L-agents. Each of the L-agents—also called wrappers—is capable of making a specific type of connection, or perhaps a connection to just one specific Internet resource. For example, an L-agent can be capable of connecting to a specific search engine. It opens a connection, forms an appropriate query, gets the results and passes them up the agent hierarchy. The role of the I-agents and the T-agent in this bottom-up direction is to filter and to combine information. Results finally reach the Q/A-agent, which presents them to the user and the interaction continues.

I-agents handle two sets of tasks: (1) transforming and passing the query in a top-down direction, and (2) transforming and passing the results in a bottom-up direction.

A natural question arises at this point. Why do we not assign these tasks to two different types of agents? One important expected feature of the I-agents is their ability to reuse information in the fashion of a cache memory. After filtering and passing up the results, the I-agents can keep the results (selectively or not) in their persistent knowledge base and use them if a similar query comes up. They would not be able to do that if they never saw the results. There can be specialized I-agents that are not part of this two-direction information flow, which are recruited by other I-agents to do a specific task that might be only remotely related to information gathering. We will leave this option open but it is not the main concern of our discussion at this point.

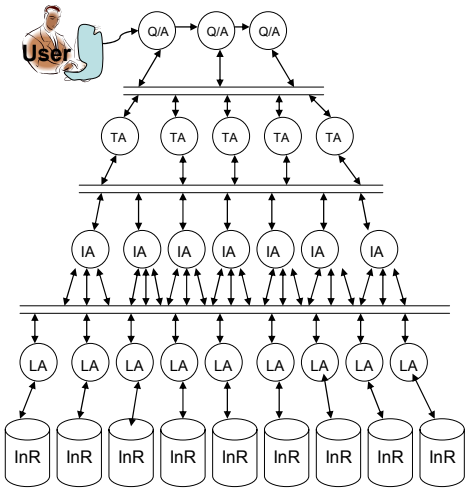

Figure 1.1 QTIL Multi-Agent Internet Retrieval Architecture

In summary, there are two areas where NLP is used: in the user-MAS interaction and in the process of resource indexing and matching. These two areas map to the Q/A-agent and to the I-agents. The Q/A-agent translates the user's NL query into an internal form and, later on, it can use NL generation to produce results. The I-agent activities involve the use of NLP in resource indexing, matching, terminology extraction, semantic indexing, document clustering and profiling, and even automatic reasoning. The semantic information gathered is based on resources content as well as on the resource meta-data such as README files, search engine front pages, or manually created descriptions, such as RDF descriptions for Semantic Web.

WebQA [17] can be thought of as an instantiation of the QTIL architecture, even though software modules are described as opposed to software agents. However agents can be used to implement the modules for mapping purposes. WebQA uses a Q-level query parser with NL capability, a T-level Resource/Locator Decomposer module, an I-level Summary Retriever, and L-level use of metasearch engines and other Web data sources. The Mulder QA Architecture [16], can also be formulated as a specific instantiation of the QTIL architecture. What differentiates Mulder from the WebQA instantiation is that it is a tightly coupled instance where there are no I-agents, but rather the Q/A level is heavyweight with NLP parsing and question classification in user-to-data source direction, and with answer question and selection via voting in the other direction towards the user. Mulder's Query formulation module can be implemented by a T-agent. Mulder takes the parse tree generated from the Q-level output and translates the question directly into a

series of search engine queries. Parihk and Murty [22] propose an integrated NLP and IR architecture that instantiates a Q-level module for question analysis, an I-level query generator, and the L-level use of the Google search engines.

The QTIL multi-agent approach is proposed to be a comprehensive general solution to the InIR problem since:

1. The multi-agent model is flexible. If we want to adopt a new Internet communication form, then we only have to create a new low-level retrieval agent (L-agent). It will speak the new Internet ``language" on one side and the inter-agent communication language on the other.
2. It results in less expensive total indexing. There is no need for frequent updates. The agents will attempt to find the answer dynamically, in the allotted search time.
3. It results in less expensive bandwidth cost since the agents memorize and exchange useful information among themselves. They do not attempt to collect information in advance but in a lazy fashion.
4. It addresses the deep-Web retrieval issue. Using the semantic level of NLP, the I-agents can match a user query to its generalizations, such as a description of an archive where the answer to that query can be found.
5. It addresses the keyword barrier by using NLP and conceptual matching. These more sophisticated matching criteria result in increased precision and recall compared to keyword-based methods.

3. Privacy Requirements

We intend to apply the generic QTIL architecture to the privacy domain. In order to do so, we need to specify relevant privacy requirements. We provide the following seven examples of user questions that could incorporate Internet IR assuming that Internet resources are available from which answers can be synthesized.

1. Do privacy laws and authorities exist in Canada to enforce the intentions stated within the privacy policies on organizations' Web sites?
2. Which privacy law is applicable to the context of my current online transaction with this organization's Web site?
3. Which private data protection law has precedence for my current online transaction at this organization's Web site?
4. Could my personally identifiable information be shared with a third party business partner of this organization that is in a country with poor privacy laws?
5. Does this company share customer data with a third party partner originating from a country with human rights abuses?
6. Do my privacy preferences match the privacy practices of each of this organization's third party business partners?
7. Do CheatersInc or UnGreenCompany engage in unethical or environmentally-unfriendly business practices?

Internet IR is possible in each of these cases as demonstrated in the Privacy Web services [9] that can implement the low-level answering of these questions. Regulatory privacy Web ontologies [6,7,8] are the Internet resources that address the regulatory type privacy questions that are exemplified in questions 1 thru 3. Automated answering of questions 4 thru 7 require multiple P3P [3] and other cooperating agents as found in those works maturing online privacy such as PeCAN [7,8] and the Social Contract Core [11].

Electronic privacy research [2,9,7,28,19] and various implementations (e.g. AT&T's Bird, Microsoft's IE6) encourage organizations to provide explanations to their customers for why and what purposes data is being collected and with whom the collected data can be shared. The rationale is that comprehension on the users' part will prevent misunderstandings, increase the perception of user control, and hence increase e-commerce trust. Nickel and Schamburg [21] provide empirical results to support that interfaces conveying a high level of privacy significantly increased user trust. In this sense, it is intuitive to hypothesize that natural language capability in such online user interfaces will facilitate organizations to provide customized privacy information for each user and hence further increase trust creation.

The next section briefly overviews the PeCAN architecture which we intend to integrate with the QTIL architecture proposal presented in section one of this paper. The integration of the two architectures allows for more usable privacy information system or privacy enhancing tools. The user's perception of the usefulness of the privacy-enhancing tools and systems increases and also the "ease of use" of the privacy software increases. The resulting unified architecture adds this necessary natural language capability to privacy software. The clean separation of the QTIL and PeCAN architectures is also an advantage as one can merge PeCAN with other NLP architectures and vice versa.

4. The PeCAN Privacy Architecture

The Personal Context Agent Networking architecture (PeCAN) e-privacy architecture supports the following initial list of privacy-related requirements/transactions [6]:

1. Store and maintain user privacy preferences/beliefs, regulatory beliefs, transaction-related beliefs, organization beliefs, sector beliefs, stakeholder beliefs, user private data and personae, profiles, roles, service-site data, audit trails, historical data, and contracts.

2. Maintain privacy-aware user contexts (presently constrained to electronic commerce tasks).
3. Match the site's P3P-enabled privacy policy to the user's preferences and possibly follow the guidance from the user
4. Change user preferences dynamically as the system "learns" .
5. Interact with third party agents (e.g. service-site agents) or invoke informational privacy Web Services, for example to provide the user with information about the applicable privacy legislation around the transaction, or to invoke a P3P agent to find out a jurisdiction of an electronic commerce transaction.
6. Monitor user behavior on the client-side to align user actions with stated privacy preferences/beliefs and thus maintain consistent profiles.
7. Download and use boilerplate user P3P preferences for dealing with particular organizations e.g. social norm preferences from association site for a role when these become available.
8. Support privacy negotiation in future in certain electronic commerce transactions (e.g. buy).
9. Provide flexible querying, summary reporting, historical records from monitor logs.
10. Filter external feeds and add to knowledge base around privacy regulations.
11. Upload contextualized data (e.g. updated private data, preference data, and profile data relevant to a requesting entity) to user-approved list of external entities.
12. Give relevance feedback to the system.

For overview and to illustrate PeCAN's "big picture", in Fig. 2, we show an architecture of cooperating client-side and Web agents that supports all the requirements (1 thru 12) in an e-privacy model. Fig.2 shows interactions among agents and access to repositories. There are currently four key client-side agents in the PeCAN architecture – the personal context manager agent [8], the regulatory agent, the arbitrator agent and the monitor agent.

In this paper we focus on the regulatory agent as it relies heavily on Internet Information Retrieval for privacy guidelines, rules, and any user-pertinent privacy governance information. This agent maintains an up-to-date knowledge base by invoking appropriate privacy Web services [6] and by accepting and filtering external feeds such as from Web watchdog associations such as epic.org, BBBOnline, hil-watch.com, and privcom.gc.ca.

The regulatory agent also interacts with external agents to effect trust intervention mechanism by the government, community, association, and business stakeholders. Three representative external agents are shown, iCritics, Social Core [11] and Web services agents. It should be noted that in (Ackerman et al, 1999) the term iCritic agent refers to either internal or external agent that supports the user privacy. We use the term in a more restrictive sense to refer only to external agents that provide information on service sites that can be used by user privacy mechanisms. An example is an iCritic agent monitoring reputable associations, such as the BBBOnline, providing information on a site in terms of complaints by customers. Social Core agents assist the user with setting-up privacy preferences by providing special-group preference recommendations for various activities. For instance, an agent representing PTAs may provide recommended privacy settings to be used by children. Web Service agents, or Web services, provide a variety of information to the PeCAN system. One example is information about regulations that apply in different privacy regions/countries so that the user agent(s) could adjust privacy preferences accordingly, and the user could take appropriate actions in terms of managing her private data collected by service sites.

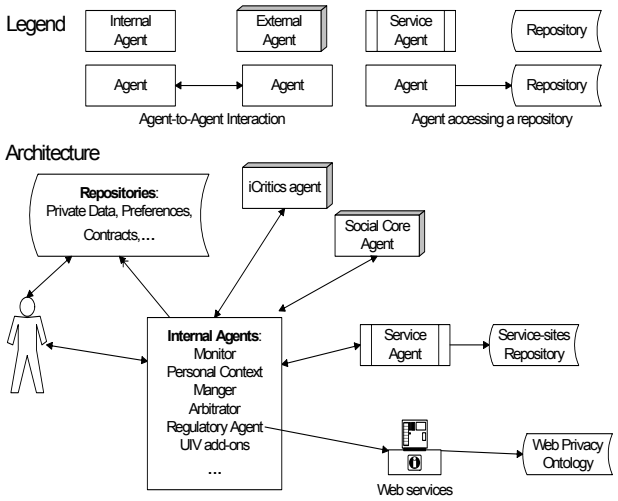

Figure 2. The PeCAN MultiAgent Architecture.

5. QTIP Architecture

Integrating the QTIL and PeCAN architectures involves a number of intuitive mappings. PeCAN's client-side agents: regulatory, context manager, arbitrator, and monitor are I-agents. PeCAN's external agents or privacy Web services are L-agents. We map these L-agents to the P-layer of the resulting QTIP architecture. The external agents and privacy Web services on the P-layer all use web standards i.e. XML-based languages, including SOAP, UDDI, WSDL, RDF and OWL, and XML-based data models such as P3P for the privacy domain. The Q and the T-layers are directly adopted from the QTIL architecture.

We use an instantiation of the QTIP architecture, using the regulatory I-agent, to describe the architectural flow. To recap, a Q/A-agent gets the query from the user,

translates it into an inter-agent format and sends it to a T-agent. The T-agent can develop a high-level plan for solving the query, it can break it into sub-queries, and, generally, since it knows a lot about other agents, it can decide whom to send the query or sub-queries to. In our example, the sub-queries then reach the regulatory I-agent. The regulatory I-agent incorporates a knowledgebase about the privacy regulations domain and it can make the final plan and translate it into the low-level actions.

Some of these actions might be done internally and others are realized using the P-agents. The P-agents are the ``fingers'' of the system. They get in touch with Internet resources, and they are the interfaces that translate the inter-agent language to the resource-specific language and vice versa.

Then, information flows in the opposite direction—from P-agents to regulatory I-agent, which can do some intermediate processing, information extraction, updating of beliefs in the PeCAN repository, or caching. The regulatory I-agent sends the answers to the T-agent, which can do the fusion of several answers if obtained from several agents. The final results reach the Q/A-agent, which decides how to present them to the user.

5.1 Q/A and I NLP Layers

There are two key areas where NLP is used in the QTIP architecture: in the user-MAS interaction (Q/A level) and in the process of resource indexing and matching (I agent level). The Q/A-agent translates the user's NL query into an internal form and, later on, it can use NL generation to produce results. Let us also consider the Regulatory I-agent. This agent is an expert agent on privacy regulations. It invokes Privacy Web services and other external agent services in order to maintain a belief system for the user around privacy regulations in multiple contexts such as across organizations and countries.

Q/A and I-agents use different kinds of NLP. More precise NLP is used in a Q/A-agent, even though it may consume more running time. Typically only one sentence is processed pre query so this overhead is not significant. On the other hand, it is preferable that the user is properly understood. The regulatory I-agent has the opposite requirements. It processes a large number of documents; hence, it is important that it is as efficient as possible. Parsing correctness is not as vital – if parser cannot parse the whole sentence, the sub-sentence phrases contain still useful information and indexing nuggets.

5.2 An implementation of the QTIP Architecture

We demonstrate the QTIP architecture through instantiation of a single Q/A agent, zero T-agents, a regulatory I-agent, and multiple P agents. The syntactic and semantic formalism used to parse and capture query meaning is HPSG—a state of the art unification-based, NLP formalism for processing natural language at syntactic and semantic level [27,23,26]. It has been shown that the formalism can be successfully used to capture meaning of the open-domain factual questions [13], and it has been used in question answering. The formalism is amenable to modular design [12,14], which is a desired feature in adapting a general grammar for specialized domains, such as the privacy domain.

The regulatory agent invokes the relevant P-agents using Web services standards, specifically SOAP RPC queries. The regulatory I-agent discovers individual Web services from a public UDDI directory. The regulatory agent creates a proxy by using the WSDL URL and invokes the individual Web service. From the results which the P-agent returns, the regulatory agent composes a response to the Q/A agent. In our implementation, the P-agents are implemented as Web services. The following example illustrates processing of two user queries.

Example. The user sends two queries to her Q/A agent as follows:

1. *What are the privacy laws applying to business in Canada?*
2. *According to PIPEDA, in what situation user's data can be collected without the user's consent*?

After parsing the questions in the unification-based grammar adapted for RQL representation we obtain the following semantic representations:

$$\begin{bmatrix} \text{SELECT:} & \langle \boxed{X}, \boxed{Y} \rangle \\ \text{FROM:} & \langle \boxed{X} \left[\text{ns3:Use\_law:} \ \boxed{Y} \right] \rangle \\ \text{WHERE:} & \begin{bmatrix} like \\ 1: & \boxed{X} \\ 2: & \text{http://newOnto.org/f86432832ab\#Canada} \end{bmatrix} \\ \text{USING:} & \langle \left[\text{ns3: http://newOnto.org/f8643283ab} \right] \rangle \end{bmatrix}$$

and

$$\begin{bmatrix} \text{SELECT:} & \langle \boxed{X}, \boxed{Y} \rangle \\ \text{FROM:} & \langle \boxed{X} \left[\text{ns3:Is:} \ \boxed{Y} \right], \boxed{X} \left[\text{rdf:type:} \ \boxed{Z} \right] \rangle \\ \text{WHERE:} & \begin{bmatrix} like \\ 1: & \boxed{Z} \\ 2: & \text{http://newOnto.org/f86432832ab\#} \\ & \text{Collection\_without\_knowledge\_or\_consent} \end{bmatrix} \\ \text{USING:} & \langle \text{ns3} = \text{http://newOnto.org/f8643283ab}, \\ & \text{rdf} = \text{http://www.w3.org/1999/02/22-rdf-syntax-ns\#} \rangle \end{bmatrix}$$

The representations are formed according to the standard AVM (Attribute-Value Matrix) format used in unification-based grammars. The grammar is adapted according to an existing implementation of a small prototype privacy ontology (Jutla and Xu, 2004) for the Canadian privacy act that applies to commercial enterprises, Personal Information Protection and Electronic Documents Act (PIPEDA). The prototype privacy ontology is stored in the Netherlands in a Sesame RDF database. The P-agent queries the PIPEDA ontology

via its query Web service implementation class program. The P-agent first establishes an HTTP connection with the Sesame database at "*http://www.openrdf.org/sesame/*". The user then gets authenticated. The RQL [10] queries that correspond to the above AVM matrices are:

```
select X, Y from {X} ns3:Use_law {Y}
where X like http://newOnto.org/f8643283ab#Canada
using namespace ns3 = http://newOnto.org/f8643283ab"
```

and

```
select X, Y from  {X} ns3:Is {Y}, {X} rdf:type {Z}
where Z like "http://newOnto.org/f8643283ab#
Collection_without_knowledge_or_consent"
using namespace ns3 = http://newOnto.org/f8643283ab ,
rdf = http://www.w3.org/1999/02/22-rdf-syntax-ns#"
```

The answer for the query "*What are the privacy laws applying to business in Canada?*" is: PIPEDA. For the second query "According to PIPEDA, in what situation user's data can be collected without the user's consent?" there are three categories returned as a single response. They are "Disclosure without consent", "User without consent" and "Collect without consent". The breakdown of some components of response times for answering both these queries are shown in Table 1. The total P-agent response time of 7.24 ms is not unacceptable for Web information retrieval of targeted and relevant privacy regulation information for the user. Recall that these results are superior to search engine results consisting of pages of hyperlinks which the user has to manually open and then search for relevant text.

Table 1. Response Time measurements for P-agent

	Response Time in ms
Total Time	7,244
UDDI Discovery Web serv.time	5,018
Invoke P-agent Web serv.time	1,562
Sesame First Query Time	204
Sesame 2nd Query Time	198

6. Related Work

NLP architectures with intelligent IR were being talked about nearly 20 years ago. Jacobs and Rau [5] introduce the SCISOR architecture for integrating NLP and IR. The intelligence in SCISOR was not agent-based and was introduced from standard NLP techniques implemented in modules and from contextual IR techniques. SCISOR was a partial pre-cursor to WebQA [17], and Mulder [16]. Mulder combines IR with statistical NLP [16]. Mulder and Web QA architectures map nicely to the proposed general QTIL architecture introduced in this paper.

Popescu et al.[24] propose a theoretical framework that is implemented in a PRECISE NL interface which maps "semantically tractable" NL questions to SQL queries in targeted domains. Example test domains were restaurants, jobs, and geography. Some MAA for NLP [4] focus on the specialization of agents for NLP tasks. One agent could be a specialist in syntax, while another is a specialist in semantics, or temporal reasoning, or anaphora resolution and so on. More recently, [1] propose TRIPS, (The Rochester Interactive Planning System), an agent-based architecture for the specific conversational systems or speech domain. TRIPS' architecture is an integration of knowledge specific domain agents, in this case speech domain agents, and user-interface Q/A agents for interpretation and planning response.

The speech domain agents manage observing the user's speech utterances and actions, user's preferences and changes in the user's world state. These domain-specific agents would then map to the I and P levels of our QTIP architecture. TRIPS's strength like QTIP's is in their clean and clear separation of linguistic and task- and domain-specific information agents. Such architectures allow for better interleaving of agents' tasks, and extend system's capabilities for constant changes, whether incremental or not. The SPA architecture [20] for the email domain can similarly be mapped to an integration with the QTIL architecture where the SPA dialogue agent combines the functionality of T and I-level agents and the email manager maps to an L agent.

The *Platform for Privacy Preferences (P3P)* was published by W3C in 2002 and, regardless of some shortcomings, it is the only contender on which to base privacy mechanisms and architectures for web applications. The latest working draft of P3P version 1.1 was released in Apr.2004. IE6 and Netscape Navigator 7 Web browser provide basic P3P functionality. AT&T provides a P3P agent called Privacy Bird as an add-on to IE6 browser match using a traffic-light metaphor in its interface. A study of users mainly over 50 year olds reports that the Privacy Bird is a useful agent The user privacy agents simplify the task of examining the privacy policies posted by the Web-sites and determining whether or not the they are acceptable to the users/clients – a task that is cumbersome and disliked by users [3].

The Resource Centre on P3P of JRC (JRC architecture 2004) has a basic privacy architecture that does not include access to Web-services or cooperation with *Trusted Third Parties (TTP)* as yet. An ontology for data protection is in the planning stage. It is a substantial and long-term undertaking that involves education and participation of the various stake-holders in arriving at the standard ontology (JRC ontology 2004).

Kim [15] argues that privacy be built into the Semantic Web and stresses the need for privacy ontology. This is also one of the conclusions in (Rezgui 2003). We proposed a high level model for a privacy ontology in [6], and implemented an ontology fragment as proof-of-

concept (Xu, 2004). The Web services described in the example in this paper accesses this ontology stored on Sesame, an RDF database.

The QTIL architecture is the first attempt we have seen to suggest a generic NLP-integrated architecture for internet information retrieval. It is also the only purely agent-based approach we can find. The QTIP architecture is even more unique as it is an integration of architectures specifically to support the privacy domain and the development of usable and scalable privacy information software.

7. Summary and Conclusions

This paper provides a number of novel contributions to the MAS application and web retrieval literature. Firstly, a generic architecture for agent-based web retrieval systems incorporating NLP is presented. This architecture is validated to the extent that many popular and rigorous instances of this architecture exist such as WebQA and Mulder. Secondly another novel architecture for the privacy domain, called QTIP, is proposed to improve the accuracy, ease of use, and perceived usefulness of privacy software. The integrated QTIP architecture enhances its component PeCAN architecture by allowing for NL interaction with the user. NLP capability is facilitated through HPSG as shown in our example implementation of a QTIL instantiation. QTIP's web services (via PeCAN) allow for the relevant retrieval of targeted privacy regulation text. The Q/A agents in QTIP present these results to the user in a friendly user format in acceptable time frames. Our highly modular distributed agent architectures support the rapid capability upgrades needed in today's agent-based software systems.

8. References

[1] Allen, J., Ferguson, G., Stent, A., An Architecture for more Realistic Converstaional Systems, IUI'01, Orl., Fl., pp. 1-8.

[2] Bodorik and Jutla D.N., Architecture for user-controlled e-privacy, *Proc.of the 2003 ACM Sym.on App.computing,*, Elect. commerce techn. track, Melbourne, Fl., 609 – 616.

[3] Cranor F.L., P3P: Making Privacy Policies More Useful, IEEE Security & Privacy, Nov-Dec 2003, 50-55.

[4] Fum, D., Guida, G., Tasso, C., A Distributed Multi-agent Architecture for Natural Language Processing, 12th Int. Conf.on Comp.Ling., Budapest, 22-27, 1988, pp. 812-814.

[5] Jacobs, P.S., and Rau, L.F., Natural Language Techniques for Intelligent Info. Retrieval, Proc.of the 11th ACM SIGIR on R&D in Inf.Retr., 1988, Grenoble, France, pp. 85-99.

[6] Jutla D.N, Bodorik P., Gao D., Management of Private Data: Web Services Addressing User Privacy and Economic, Social, and Ethical Concerns, in VLDB w/sh on Secure Data Management, Toronto, 2004, pp. 100-117

[7] Jutla D.N., and Bodorik P. Socio-Technical Architecture for Online Privacy. *IEEE Security and Privacy*, pp. 25-35, March/April 2005.

[8] Jutla D.N., Bodorik P., Zhang Y., PeCAN: An Architecture for Privacy-aware User Contexts for Electronic Commerce on the Semantic Web, *Information Systems*, in press, 2005.

[9] Jutla D.N., Kelloway, E.K., Saifi, S., Evaluation of the Impact of User Intervention Mechanisms for Privacy on Online SME Trust. *IEEE Conference on E-Commerce*, San Diego, 8 pages, July 2004.

[10] Karvounarakis G., Alexaki S., Christophides V., Plexousakis D.,Scholl, RQL: A Declarative Query language for RDF, Proc. of the eleventh international conference on WWW, Honolulu, Hawaii, pp. 592-603

[11] Kaufmann, J., H. and Powers, C. (2002), The social contract core. WWW 2002, May 7-11, Hawaii, 210-220.

[12] Keselj, V, Question Answering using Unification-based Grammar. In Advances in Artificial Intelligence, AI 2001, volume LNAI 2056 of LNCS, Springer, Ottawa, June 2001.

[13] Keselj, V, and Cercone, N. Just-in-time subgrammar extraction for HPSG. In Proceedings of PACLING'01, Kitakyushu, Japan, September, 2001a.

[14] Keselj, V. Modular HPSG. In Proceedings of the 2001 IEEE Sys/Man/Cybernetics Conference, Tucson, 2001b.

[15] Kim, A., Joffman, L.J., and Martin, C.D., Building Privacy into the Semantic Web: An Ontology Needed Now. *Semantic Web Workshop 2002,* Hawaii USA

[16] Kwok, C., Etzioni, O., and Weld, D.S., Scaling Question Answering to the Web, ACM Trans. on Information Systems, Vol. 19, No. 3, July 2001, pp. 242-262.

[17] Lam, S.K.S., and Ozsu M.T.,Querying Web Data – The WebQA Approach, Proc. of the 3rd International Conf-erence on Web Information Systems Engineering, 2002.

[18] Liu, C., Marchewka, J., Lu J., Yu, C. (2005) Beyond concern – a privacy-trust behavioural intention model of electronic commerce, Information and Management. Vol. 42, Issue 2, pp. 289-304.

[19] Patrick A., Kenny S., From Privacy Legislation to Interface Design: Implementing Information Privacy in Human-Computer Interfaces. In R. Dingledine (Ed.), *Proc.of PET2003*, Dresden, Germany, 2003. LNCS 2760.

[20] Nguyen, A., Woncke, W., IUI'05, San Diego, California, pp. 137-144.

[21] Nickel J, and Schaumburg H (2004), Electronic Privacy, Trust, and Self-Disclosure in e-Recruitment, CHI 2004, Lake Breaking Results Paper, April 24-29, Vienna, Austria

[22] Parikh J., Murthy M.N., Adapting Question Answering Techniques to the Web, Proc.of the Language Engineering Conference, 2002.

[23] Pollard, C and Sag, I. Head-Driven Phrase Structure Grammar, University of Chicago Press, Chicago, 1994.

[24] Popescu, A.M., Etzioni, O, Kautz, H., Towards a Theory of Natural Language Interfaces to Databases, IUI, 2003, Miami, Florida, pp.. 149-157.

[25] PrivacyBird, 2004, available at http://privacybird.com/, (access date: January 8, 2005)

[26] Sag, I, and Wasow, T. Syntactic Theory: A Formal Introduction, CSLI Publications, Stanford, 1999.

[27] Shieber, S. An Introduction to unification-Based Approaches to Grammar. CSLI Lec.Notes. Stanford, 1986.

[28] Teltzrow M., Kobsa A., Communication and Privacy of Personalization in e-Business, Proc.of WHOLES: A Multiple View of Individual Privacy in a Networked World, Stockholm, Sweden, 2004.